



ارائه شده توسط:

سایت ترجمه فا

مرجع جدیدترین مقالات ترجمه شده

از نشریات معتبر

بهینه سازی چند هدفی با استفاده از الگوریتم های ژنتیک: یک آموختار

چکیده

فرمولاسیون های چند شیئی (هدفی) مدل های واقعی مسائل بهینه سازی مهندسی پیچیده متعددی هستند. در بسیاری از مسائل زندگی واقعی، اشیای تحت نظر با یکدیگر تناقض پیدا می کنند، و بهینه سازی راه حلی خاص با توجه به یک شیء منفرد می تواند حاصل نتایج غیر قابل قبول از نظر دیگر اشیاء باشد. راه حل منطقی برای مسأله چند منظوره (شیئی) بررسی مجموعه ای از راه حل هاست، که هر یک از آن ها اهداف را در سطح قابل قبولی بدون اینکه دیگر راه حل ها بر آن تفوق یابند، برطرف می کنند. در این مقاله، بررسی اجمالی و آموختاری ارائه شده که الگوریتم های ژنتیک (GA) توسعه یافته را شرح می دهند، خصوصاً الگوریتم هایی که برای مسائل همراه با چند هدف طراحی شده اند. آن ها عمدتاً از GA مرسوم بواسطه استفاده از توابع تناسب تخصصی و معرفی روش هایی برای توسعه و پیشرفت تنوع راه حل ها متمایز شده اند.

1- مقدمه

هدف این مقاله ارائه بررسی اجمالی و آموزه ای از روش های بهینه سازی چند هدفی با استفاده از الگوریتم های ژنتیک است (GA). برای مسائل چند موضوعی اهداف معمولاً متناقضند، و از بهینه سازی همزمان هر هدف ممانعت بعمل می آورد. مسائل مهندسی حقیقی متعدد و حتی بیشتر آنها در حقیقت چندین هدف دارند، به عنوان نمونه، هزینه حداقل، عملکرد حداکثر، قابلیت اطمینان حداکثر و غیره. این مسائل مسائل مشکلی هستند اما واقعی اند. GA الگوریتم های فرا اکتشافی محبوبی هستند که خصوصاً مناسب این رده از مسائلند. GA های مرسوم برای هماهنگی با مسائل چند هدفی به دلخواه مشتری شده اند و این کار با استفاده از توابع تناسب تخصصی و ارائه روش هایی برای ترقی بدنه راه حل انجام شده است.

دو مشی کلی برای بهینه سازی چند هدفی وجود دارد. یکی از این مشی ها تلفیق توابع هدف منفرد در یک تابع ترکیبی منفرد یا انتقال تمام آنها به جز هدف موجود برای مجموعه محدودیت هاست. در مورد اول، تعیین یک هدف

منفرد با روش هایی نظیر نظریه سودمندی، روش مجموع وزن دهی شده و غیره ممکن است، اما مشکل ریشه در انتخاب صحیح وزن ها یا توابع سودمندی برای توصیف رجحان های تصمیم گیرنده دارد. در عمل، انتخاب دقیق و با دقت این وزن ها بسیار مشکل است، حتی برای افرادی که با دامنه مسأله آشنایی دارند. ترکیب این معایب آن مقیاس بندی است که در میان اهداف مورد نیاز است و اختلالات اندکی در وزن ها گاهی اوقات می تواند منجر به ایجاد راه حل هایی شود که اندکی متفاوتند. در مورد دوم، مسأله این است که برای حرکت اشیا در مجموعه محدودی، مقدار محدود کننده ای باید برای هر یک از این اهداف (اشیای) قبلی بنا نهاده شود. در این مورد می توان اندکی اختیار به کار برد. در هر دو مورد، روش بهینه سازی می تواند راه حل منفردی به جای مجموعه ای از راه حل هایی بازگرداند که می تواند برای مبادلات مورد آزمون قرار گیرد. به این دلیل، تصمیم گیرندگان مجموعه ای از راه حل های مناسب را با در نظر گرفتن چند هدف ترجیح می دهند.

دومین مشی عمومی تعیین راه حل بهینه Pareto کلی یا زیر مجموعه مشعر می باشد. مجموعه بهینه Pareto مجموعه ای از راه حل هاست که با توجه به یکدیگر برتری بر هم ندارند. در حین حرکت از یک راه حل Pareto به راه حل دیگر، همیشه مقدار معینی از قربانی در یک هدف (اهداف) برای رسیدن به مقدار مشخصی از سودمندی در دیگر اهداف وجود دارد. مجموعه راه حل های بهینه Pareto اغلب به راه حل های منفرد ترجیح داده می شوند چون می توانند هنگام در نظر گرفتن مسائل دنیای واقعی عملی باشند، چون راه حل نهایی تصمیم گیرنده همیشه یک تبادل است. مجموعه های بهینه Pareto می توانند اندازه های متفاوتی داشته باشند، اما اندازه مجموعه Pareto معمولاً با افزایش تعداد اهداف افزایش می یابد.

2- فرمولاسیون بهینه سازی چند منظوره (شیئی)

تصمیم گیرنده ای را در نظر بگیرید که می خواهد K هدف را به گونه ای بهینه سازی کند که اهداف تناسبی با هم ندارند و تصمیم گیرنده هیچ یک از اهداف را به دیگری ترجیح نمی دهد. بدون از دست دادن عمومیت، تمام اهداف از گونه کمینه شده هستند - یک هدف از نوع کمینه شده می تواند با ضرب در یک نوع منفی به یک نوع بیشینه شده تبدیل شود. مسأله تصمیم چند منظوره (شیئی) کمینه شده با K هدف به شرح زیر تعریف شده است:

با توجه به بردار متغیر تصمیم n بعدی $x = \{x_1, \dots, x_n\}$ در فضای راه حل X ، برداری x^* ای بیابید که مجموعه معینی از K تابع هدف $z(x^*) = \{z_1(x^*), \dots, z_K(x^*)\}$ را کمینه نماید. فضای راه حل X معمولاً با یک سری شرط ها محدود شده است، نظیر $g_j(x^*) = b_j$ for $j = 1, \dots, m$ ، و کران های اعمال شده روی متغیرهای تصمیم.

در بسیاری از مسائل دنیای واقعی، اهداف تحت نظر با یکدیگر تناقض پیدا می کنند. از اینرو، بهینه سازی X با توجه به هدف منفرد اغلب حاصل نتایج غیر قابل قبول با توجه به دیگر اهداف است. بنابراین، راه حل چند منظوره کاملی که بصورت همزمان هر تابع هدف را بهینه سازی کند، تقریباً غیر ممکن است. راه حل منطقی برای یک مسأله چند هدفی بررسی مجموعه ای از راه حل هاست، که هر یک از آنها اهداف را در سطح قابل قبولی برطرف می کنند، بدون اینکه راه حلی بر راه حل دیگر تفوق یابد.

اگر تمام توابع هدف برای کمینه سازی باشند، راه حل عملی X بر راه عملی y تفوق خواهد یافت $(x \succ y)$ ، اگر و تنها اگر $z_i(x) \leq z_i(y)$ برای مقادیر $i = 1, \dots, K$ صادق باشد و $z_j(x) < z_j(y)$ حداقل برای یک تابع هدف j صادق باشد. راه حل بهینه Pareto نمی تواند از نظر هر گونه هدفی بهبود یابد مگر اینکه حداقل یکی دیگر از اهداف بدتر شود. جموعه تمام راه حل های تفوق نیافته عملی در X با عنوان مجموعه بهینه Pareto نام برده شده است، و برای یک مجموعه بهینه Pareto، مقادیر تابع هدف متناظر در فضای هدف نمای Pareto نامیده شده اند. برای بسیاری از مسائل، تعداد راه حل های بهینه Pareto متعددند (شاید نامتناهی باشند).

هدف نهایی یک الگوریتم بهینه سازی شناسایی راه حل ها در مجموعه بهینه Pareto است. اما، شناسایی کل مجموعه بهینه Pareto برای بسیاری از مسائل چند هدفی از نظر عملی غیر ممکن است و این به دلیل اندازه آن است. به علاوه، برای بسیاری مسائل، خصوصاً برای مسائل بهینه سازی تلفیقی، اثبات بهینگی راه حل از نظر محاسباتی غیر ممکن است. بنابراین، مشی عملی برای بهینه سازی چند هدفی بررسی مجموعه ای از راه حل هاست (مجموعه Pareto ای که به عنوان بهترین مجموعه شناخته شده است). که نشان دهنده مجموعه بهینه Pareto و

همچنین مجموعه های محتمل است. با داشتن این مطالب در ذهن، باید مشی بهینه سازی چند هدفی به سه هدف متناقض زیر بردس:

1- بهترین نمای Pareto باید تا حد ممکن به نمای Pareto واقعی نزدیک باشد. بصورت ایده آل، بهترین مجموعه Pareto باید زیرمجموعه ای از مجموعه بهینه Pareto باشد.

2- راه حل های موحود در بهترین مجموعه Pareto باید توزیع یکپارچه داشته باشد و در مای Pareto متنوع باشد تا به تصمیم گیرنده تصویر درستی از تبادل ها ارائه دهد.

3- بهترین نمای Pareto باید کل طیف نمای Pareto را ثبت و ضبط نماید. این مستلزم بررسی راه حل ها در انتهای فضای تابع هدف است.

برای محدودیت زمان محاسباتی معین، اولین هدف به بهترین نحو با تمرکز جستجو روی ناحیه مشخصی از نمای Pareto خدمت رسانی شده است. از طرف مقابل، دومین هدف مستلزم تلاش تحقیقاتی برای توزیع یکپارچه در نمای Pareto است. سومین هدف سعی در بسط نمای Pareto در دو انتها دارد، و راه حل های فوق العاده و جدیدی کشف می نماید.

این مقاله نشان دهنده ی مشی های عمومی استفاده شده در GA چند هدفی برای نیل به این سه هدف متناقض است در حالی که مسأله بهینه سازی چند هدفی حل می شود.

3- الگوریتم های ژنتیک

مفهوم GA توسط Holland و همکارانش در دهه 1960 و 1970 توسعه یافت. GA از نظریه تکامل گرایی الهام گرفته که ریشه و اصل گونه ها را شرح می دهد. در طبیعت، گونه های ضعیف نامتناسب در محیطشان با مسأله انقراض روبرو هستند. گونه های قوی فرصت بیشتری برای انتقال ژن های خود به نسل های آینده از طریق تولید مثل دارند. در دراز مدت، ونه هایی که تلفیق مناسبی در ژن های خود حمل می کنند در جامعه خود غلبه می یابند. گاهی اوقات، در عرض فرایند تکامل، تغییرات تصادفی در ژن ها رخ می دهد. اگر این تغییرات مزایای علاوه بر

سازمانی در چالش برای بقا ارائه نمایند، گونه های جدید از گونه های قبلی تکامل می یابند. تغییرات ناموفق بواسطه انتخاب طبیعی حذف شده اند.

در واژه شناسی GA، بردار راه حل $x \in X$ یک فرد یا یک کروموزوم نامیده شده است. کروموزوم ها از واحدهای گسسته ای به نام ژن ها تشکیل شده اند. هر ژن یک یا چند ویژگی از کروموزوم را کنترل می کند. در پیاده سازی اصلی GA توسط Holland، ژن ها به عنوان ارقام دودویی در نظر گرفته شده اند. در پیاده سازی های بعدی، انواع ژن های مختلفی معرفی شده اند. بطور عادی، یک کروموزوم متناظر با راه حل منحصر بفرد X در فضای راه حل و کروموزوم هاست. این نگاشت کدگذاری نامیده شده است. در حقیقت کار GA روی کدگذاری یک مسأله است نه روی خود مسأله.

GA همراه با مجموعه ای از کروموزوم ها به نام یک گروه (جامعه) عمل می کند. این گروه بصورت عادی بطور تصادفی تخصیص دهی اولیه شده است. با تکمیل جستجو، گروه شامل راه حل های مناسب تر و متناسبتری می شود و در نهایت واگرا می شود، بدین معنا که راه حل منفرد بر آن تفوق می یابد. Holland علاوه بر ایت برهانی از واگرایی (قضیه شما) را برای بهینه کلی ارائه کرده که در آن کروموزوم ها بردارهای دودویی هستند.

GA از دو عملگر برای تولید راه حل های جدید از انواع موجود استفاده می کند: متقاطع و دگرگونی. عملگر متقاطع که معمولاً دو کروموزوم است والدین نامیده شده است و با یکدیگر تلفیق یافته اند تا کروموزوم های جدید را تشکیل دهند، یعنی فرزند (اولاد). والدین از میان کروموزوم های موجود در گروه همراه با رجحان در مورد تناسب انتخاب شده به گونه ای که انتظار می رود فرزند ژن های خوب را به ارث ببرد که والدین را متناسب تر می کند. با اعمال یکی در میان عملگر متقاطع، انتظار می رود که ژن های کروموزوم های مناسب بیشتر در گروه مشاهده شوند، و در نهایت منجر به واگرایی به یک راه حل ها مناسب کلی شود.

عملگر دگرگونی تغییرات تصادفی را وارد مشخصه های کروموزوم ها می کند. دگرگونی عموماً در سطح ژن بکار برده شده است. در پیاده سازی های GA معمول، نرخ دگرگونی (احتمال تغییر ویژگی های ژن) بسیار اندک است و وابسته به طول کروموزوم است. بنابراین، کروموزوم جدید تولید شده با دگرگونی خیلی متفاوت از کروموزوم اصلی

نخواهد بود. دگرگونی نقشی حیاتی در GA بازی می کند. همانطور که قبلاً بحث کردیم، برخورد منجر می شود که گروه با ایجاد کروموزم ها در گروه احتمالی واگرا شود. دگرگونی تنوع ژنتیکی را به گروه باز می گرداند و در فرار جستجو از بهینه محلی کمک می کند.

تولید نسل شامل انتخاب کروموزوم ها برای نسل بعدی است. در عمومی ترین مورد، تناسب یک فرد احتمال بقای او را برای نسل بعدی تعیین می کند. رویه های انتخاب متفاوتی در GA وجود دارند که بسته به این هستند که چگونه مقادیر تناسب استفاده شده اند. انتخاب نسبی، رتبه بندی، و انتخاب تورنامنت محبوب ترین رویه های انتخاب هستند. رویه یک GA ژنریک به شرح زیر داده شده است:

مرحله 1: $t=1$ قرار دهید. بصورت تصادفی N راه حل برای تشکیل گروه اولیه تولید کنید، یعنی P_1 . تناسب راه حل ها را در P_1 ارزیابی کنید.

مرحله 2: برخورد: گروه فرزند Q_t را به شرح زیر تولید کنید:

1-2- انتخاب دو راه حل x و y بسته به مقادیر تناسب.

2-2- استفاده از عملگر برخورد (مقاطع)، تولید مبدأ و اضافه کردن آن ها به Q_t .

مرحله 3: دگرگونی: هر راه حل $x \in Q_t$ را با نرخ دگرگونی از پیش تعریف شده ای دگرگون نمایید.

مرحله 4: تخصیص تناسب: مقدار تناسبی برای هر راه حل $x \in Q_t$ بر مبنای مقدار تابع هدف و نامحتمل بودن آن ارزیابی کرده و تخصیص دهید.

مرحله 5: انتخاب: N راه حل از Q_t بسته به تناسب آنها انتخاب کرده و آنها را در P_{t+1} مپی کنید.

مرحله 6: اگر شرایط توقف برقرار است، به جستجو خاتمه دهید و به گروه فعلی بازگردید، در غیر اینصورت $t=t+1$ قرار دهید و به مرحله 2 بروید.

4- GA چند هدفی

GA ها مشی مبتنی بر گروهی هستند که برای حل مسائل بهینه سازی چند منظوره بسیار مناسبند. GA تک هدفی ژنریک می تواند برای یافتن مجموعه ای از چندین راه حل تفوق نیافته در اجرای منفرد تغییر یابد. توانایی

GA در جستجوی همزمان نواحی مختلف یک فضای راه حل یافتن مجموعه متنوعی از راه حل ها را برای مسائل مشکل همراه با فضاهای راه حل غیر واگرا، ناپیوسته، و چند حالتی ممکن می کند. عملکرد برخورد GA می تواند از ساختارهای راه حل های مناسب با توجه به اهداف مختلف برای ایجاد راه حل های تفوق نیافته در بخش های کاوش نشده نمای Pareto بهره ببرد. به علاوه، اغلب GA های چند هدفی از کاربر نمی خواهند تا اهداف را اولویت بندی، مقیاس بندی یا وزن دهی کند. بنابراین GA ها به محبوب ترین روش های اکتشافی برای طراحی چند هدفی و مسائل بهینه سازی تبدیل شده اند. Jones و همکارانش گزارش داده اند که 90 درصد مشی رها برای بهینه سازی چند هدفی سعی بر تخمین نمای Pareto درست برای مسأله متضمن دارند. عمده این ها از تکنیک فرا اکتشافی استفاده کرده اند، و 70 درصد روش های فرا اکتشافی مبتنی بر روش های تکاملی بوده اند.

جدول 1



Algorithm	Fitness assignment	Diversity mechanism	Elitism	External population	Advantages	Disadvantages
VEGA [5]	Each subpopulation is evaluated with respect to a different objective	No	No	No	First MOGA Straightforward implementation	Tend converge to the extreme of each objective
MOGA [6]	Pareto ranking	Fitness sharing by niching	No	No	Simple extension of single objective GA	Usually slow convergence Problems related to niche size parameter
WBGA [8]	Weighted average of normalized objectives	Niching Predefined weights	No	No	Simple extension of single objective GA	Difficulties in nonconvex objective function space
NPGA [7]	No fitness assignment, tournament selection	Niche count as tie-breaker in tournament selection	No	No	Very simple selection process with tournament selection	Problems related to niche size parameter Extra parameter for tournament selection
RWGA [9]	Weighted average of normalized objectives	Randomly assigned weights	Yes	Yes	Efficient and easy implement	Difficulties in nonconvex objective function space
PESA [14]	No fitness assignment	Cell-based density	Pure elitist	Yes	Easy to implement Computationally efficient	Performance depends on cell sizes Prior information needed about objective space
PAES [29]	Pareto dominance is used to replace a parent if offspring dominates	Cell-based density as tie breaker between offspring and parent	Yes	Yes	Random mutation hill-climbing strategy Easy to implement Computationally efficient	Not a population based approach Performance depends on cell sizes
NSGA [10]	Ranking based on non-dominance sorting	Fitness sharing by niching	No	No	Fast convergence	Problems related to niche size parameter
NSGA-II [30]	Ranking based on non-dominance sorting	Crowding distance	Yes	No	Single parameter (N) Well tested Efficient	Crowding distance works in objective space only
SPEA [11]	Ranking based on the external archive of non-dominated solutions	Clustering to truncate external population	Yes	Yes	Well tested No parameter for clustering	Complex clustering algorithm
SPEA-2 [12]	Strength of dominators	Density based on the k -th nearest neighbor	Yes	Yes	Improved SPEA Make sure extreme points are preserved	Computationally expensive fitness and density calculation
RDGA [19]	The problem reduced to bi-objective problem with solution rank and density as objectives	Forbidden region cell-based density	Yes	Yes	Dynamic cell update Robust with respect to the number of objectives	More difficult to implement than others
DMOEA [20]	Cell-based ranking	Adaptive cell-based density	Yes (implicitly)	No	Includes efficient techniques to update cell densities Adaptive approaches to set GA parameters	More difficult to implement than others

اولین GA چندهدفی، که GA ارزیابی شده برداری نام دارد (یا VEGA)، توسط Schaffer ارائه شده است. پس از آن، چندین الگوریتم تکاملی چند هدفی توسعه یافتند شامل الگوریتم ژنتیک چند هدفی (MOGA)، الگوریتم ژنتیک Pareto دارای تورفتگی (NPGA)، الگوریتم ژنتیک مبتنی بر وزن (WBGA)، الگوریتم ژنتیک دارای وزن تصادفی (RWGA)، الگوریتم ژنتیک مرتب سازی تفوق نیافته (NSGA) الگوریتم تکاملی Pareto قوی (SPEA)، SPEA بهبود یافته (SPEA2)، استراتژی تکامل آرشیو Pareto (PESA)، انتخاب میتنی بر ناحیه در بهینه سازی چند هدفی تکاملی (PESA-II)، الگوریتم ژنتیک مرتب سازی تفوق نیافته سریع (NSGA-II)، الگوریتم تکاملی چند هدفی (MEA)، میکروGA، الگوریتم ژنتیک مبتنی بر چگالی رتبه (RDGA) و الگوریتم تکاملی چند هدفی پویا (DMOEA). توجه کنید که گرچه انواع مختلفی از GA چند موضعی در مقالات وجود دارند، اما این GA های ذکر شده به خوبی شناخته شده اند و الگوریتم های ارزشمندی هستند که در بسیاری از کاربردها استفاده شده اند و عملکرد آنها در چندین مطالعه مقایسه ای آزمایش شده است.

چند مقاله ارزیابی در مورد بهینه سازی چند منظوره تکاملی مطالبی منتشر کرده اند. Coello بیش از 2000 مرجع در وب سایت خود لیست می کند. معمولاً، GA چند منظوری بسته به رویه تخصیصشان یا روش های متنوع سازی متفاوتند. در جدول 1، هایلایت هایی از GA چند منظوره شناخته شده با مزایا و معایبشان ارائه شده اند. اغلب مقالات ارزیابی در مورد روش های تکاملی چند منظوره الگوریتم های مختلفی را معرفی و مقایسه می کند. این مقاله روش متفاوتی در پیش می گیرد و روی مسائل مهم در حین طراحی یک GA چند منظوره تمرکز می کند و تکنیک های معمول مورد استفاده در GA چند منظوره را برای نیل به سه هدف در بهینه سازی چند منظوره شرح می دهد. این روش در مقاله ارزیابی نوشته شده توسط Zitzler و همکارانش نیز استفاده شده است. اما هدف از بحث در این مقاله معرفی مولفه های GA چند هدفی برای محققان و متخصصانی است که پس زمینه ای از GA چند هدفی ندارند. همچنین شایان ذکر است که با وجود چندین الگوریتم امروزی که در بخش قبلی ذکر شده اند، بسیاری از متخصصانی که GA چند هدفی را برای مسائل خود بکار برده اند طراحی الگوریتم های دلخواه خودشان را با اتخاذ استراتژی هایی از GA های چند منظوره مختلف ترجیح می دهند. این مشاهده انگیزه دیگری برای

معرفی مولفه های GA چند هدفی به جای تمرکز روی چندین الگوریتم است. اما شبه برنامه ای برای چند GA چند منظوره شناخته شده نیز برای این ارائه شده که نشان داده شود چگونه این شبه برنامه ها در یک GA چند هدفی مشارکت داده شده اند.

5- موضوعات طراحی و مولفه های GA چند هدفی

5-1- توابع تناسب

5-1-1- مشی های مجموع وزن دهی شده

مشی کلاسیک برای حل یک مسأله بهینه سازی چند منظوره تخصیص وزن w_i به هر تابع هدف نرمال شده $z'_i(x)$ به گونه ای است که مسأله به مسأله هدف منفرد با تابع هدف عددی به شرح زیر تبدیل شود:

$$\min z = w_1 z'_1(x) + w_2 z'_2(x) + \dots + w_k z'_k(x), \quad (1)$$

که در آن $z'_i(x)$ تابع هدف نرمال شده $z_i(x)$ و $\sum w_i = 1$ است. این مشی مشی اولویت دار نامیده شده است چون از کاربر انتظار می رود که وزن هایی ارائه نماید. حل یک مسأله یا تابع هدف (1) برای بردار وزنی معین $w = \{w_1, w_2, \dots, w_k\}$ یک راه حل حاصل می نماید، و اگر چندین راه حل مطلوب باشند، مسأله باید چندین

مرتبه با تلفیق های وزنی مختلفی حل شود. مشکل اصلی با این روش انتخاب بردار وزنی برای هر اجراست. برای خودکار کردن این فرایند؛ Hajela و Lin و WBGA را برای بهینه سازی چند منظوره (WBGA-MO) در WBGA-MO پیشنهاد داده اند، که در آن هر راه حل x_i در گروه از محاسبه تابع هدف جمع بندی شده استفاده می کند. بردار وزنی w_i در کروموزوم راه حل x_i تعبیه شده است. بنابراین می توان چندین راه حل را بطور همزمان در یک اجرا مورد جستجو قرار داد. بعلاوه بردارهای وزنی می توانند برای ترقی تنوع گروه تعدیل گردند.

دیگر محققان یک MOGA مبتنی بر مجموع وزن دهی شده از چندین تابع هدف پیشنهاد کرده اند که در آن بردار وزنی نرمال سازی شده w_i بصورت تصادفی برای تمام راه حل های x_i در عرض فاز انتخاب در هر نسل تولید شده اند. هدف این روش تصریح چندین جهت جستجو در یک اجرا بدون استفاده از پارامترهای اضافی است. رویه معمول

RWGA با استفاده از وزن های تصادفی به شرح زیر داده شده است:

$E =$ آرشیو خارجی برای ذخیره راه حل های تفوق نیافته پیدا شده در عرض جیتجو؛

$N_E =$ تعداد راه حل های هوشمندانه انتقال یافته از E به P در هر نسل

مرحله 1: تولید گروه تصادفی

مرحله 2: تخصیص مقدار تناسبی به هر راه حل $x \in P_t$ با انجام مراحل زیر:

مرحله 1-2- تولید عدد تصادفی u_k در بازه $[0,1]$ بای هر هدف k , $k = 1, \dots, K$ و

مرحله 2-2- محاسبه وزن تصادفی هر هدف k به صورت $w_k = (1/u_k) \sum_{i=1}^K u_i$.

مرحله 3-2- محاسبه تناسب راه حل ه صورت $f(x) = \sum_{k=1}^K w_k z_k(x)$.

مرحله 3: محاسبه احتمال انتخاب هر راه حل $x \in P_t$ به شرح زیر:

$p(x) = (f(x) - f^{\min})^{-1} \sum_{y \in P_t} (f(y) - f^{\min})$ که در آن $f^{\min} = \min\{f(x) | x \in P_t\}$ است.

مرحله 4: انتخاب والدین با استفاده از احتمال های انتخاب محاسبه شده در مرحله 3. اعمال روش متقاطع به زوج

والد انتخابی برای ایجاد N منشأ (فرزند). تغییر و دگرگونی منشأ به P_{t+1} . در صورت لزوم بروز رسانی E .

مرحله 5: حذف تصادفی n_E از P_{t+1} و اضافه کردن تعداد راه حل های از E به P_{t+1} .

مرحله 6: اگر شرایط توقف برقرار نیست، $t=t+1$ قرار دهید و به مرحله 2 بروید. در غیر اینصورت، به E بازگردید.

مزیت اصلی مشی مجموع وزت دهی شده پیاده سازی رو به جلو است. چون یک هدف منفرد در تخصیص تناسب

استفاده شده است، GA هدف منفرد می تواند با تغییرات کمینه مورد استفاده قرار گیرد. به علاوه، این روش از نظر

محاسباتی کارآمد است. عیب اصلی این روش این است که تمام راه حل ها نمی توانند هنگام غیر همگرا بودن نمای

Pareto مورد بررسی قرار گیرند. بنابراین GA چند منظوره مبتنی بر مشی مجموع وزن دهی شده در یافتن راه

حل های توزیع شده روی سطح تبادل غیر همگرا به مشکل بر می خورد.

2-1-5- تغییر توابع هدف

همانطور که قبلاً اشاره شد، $VEGA$ اولین GA است که برای تخمین مجموعه بهینه pareto بواسطه راه حل های

تفوق نیافته استفاده شده است. در $VEGA$ گروه P_t بصورت تصادفی به زیر گروه با اندازه مساوی تقسیم شده

است؛ P_1, P_2, \dots, P_K . بنابراین، هر راه حل در زیر گروه P_i دارای مقدار تناسبی بر مبنای تابع هدف Z_i شده است. راه حل ها از برخورد و دگرگونی انتخاب شده اند. برخورد و دگرگونی در گروه جدید به همان شیوه ای که برای GA هدف منفرد بوده اند انجام شده اند.

رویه VEGA:

$$N_s = N/K \quad (N_s = N/K)$$

مرحله 1: با گروه اولیه تصادفی P_0 شروع کنید. $T=0$ قرار دهید.

مرحله 2: اگر شرایط توقف برقرار است، به P_t بازگردید.

مرحله 3: بصورت تصادفی گروه P_t را مرتب سازی کنید.

مرحله 4: برای هر هدف $k, k=1, \dots, K$ مراحل زیر را انجام دهید:

مرحله 4-1 برای $i = 1 + (k-1)N_s, \dots, kN_s$ ، مقدار تناسب $f(x_i) = z_k(x_i)$ را به i امین راه حل در گروه مرتب شده اختصاص دهید.

مرحله 4-2 بر مبنای مقادیر تناسب اختصاص یافته در مرحله 4-1، N_s راه حل میان راه حل های

$(1 + (k-1)N_s)$ ام و (kN_s) ام از گروه مرتب شده برای ایجاد P_{t+1} با اندازه N انتخاب کنید. $T=t+1$ قرار دهید، و به مرحله 2 بازگردید.

مشی مشابه با VEGA استفاده از تنها یک تابع هدف منفرد است که به صورت تصادفی هر باز در فاز انتخاب تعیین شده است. مزیت اصلی تغییر مشی اهداف پیاده سازی ساده آن و کارآمدی محاسباتی آن بخ عنوان GA تک هدفی است. در حقیقت، این روش بسط رو به جلوی یک GA هدف منفرد برای حل مسائل چند منظوره است. نقص عمده سوئیچینگ هدف این است که گروه گرایش به همگرایی به راه حل هایی دارد که در یک هدف مقدم هستند، اما نسبت به دیگر روش ها ضعیف هستند.

3-1-5- مشی های رتبه بندی Pareto

مشی های رتبه بندی Pareto صراحتاً از مفهوم غلبه Pareto در ارزیابی تناسب یا تخصیص احتمال انتخا به راه حل ها بهره می برد. گروه طبق قانون غلبه رتبه بندی شده است، و پس از آن هر راه حل دارای مقدار تناسبی بر مبنای رتبه آن در گروه شده است، و نه مقدار واقعی تابع هدف آن. توجه داشته باشید که در اینجا فرض بر این گذاشته شده که تمام اهداف کمینه سازی شده اند. بنابراین رتبه پایین تر متناظر با راه حل بهتر در بحث زیر است: اولین تکنیک رتبه بندی Pareto توسط Goldberg به شرح زیر ارائه شده است:

مرحله 1: $i=1$ قرار دهید و $TP=P$ قرار دهید.

مرحله 2: راه حل تفوق نیافته را در TP شناسایی کنید و آنها را به F_i اختصاص دهید.

مرحله 3: $TP=TPF_i$ قرار دهید. اگر $TP = \emptyset$ است به مرحله 4 بروید، در غیر این صورت $i=i+1$ قرار دهید و به مرحله 2 بروید.

مرحله 4: برای هر راه حل $x \in P$ در نسل t ، رتبه $r_1(x, t) = i$ if را در صورتی که $x \in F_i$ است تخصیص دهید. در رویه بالا، F_1, F_2, \dots نماهای تفوق نیافته نامیده شده اند، و F_1 نمای Pareto گروه P است. NSGA نیز گروه را در نماهای تفوق نیافته با استفاده از الگوریتم مشابه با الگوریتم ذکر شده در بالا طبقه بندی می کند. بنابراین مقدار تناسب ساختگی به هر نما با استفاده از تابع اشتراک تناسب تخصیص یافته به گونه ای که بدترین مقدار تناسب اختصاص یافته به F_i بهتر از بهترین مقدار تناسب اختصاص یافته به F_{i+1} است. NSGA-II که الگوریتم کارآمدتری است و الگوریتم مرتب شده تفوق نیافته سریع نامیده شده است برای تشکیل نماهای تفوق نیافته توسعه یافته است. Fonseca و Fleming از مشی تخصیص رتبه اندکی متفاوت به شرح زیر استفاده کرده اند:

$$r_2(x, t) = 1 + nq(x, t), \quad (2)$$

که در آن $nq(x, t)$ تعداد راه حل های تفوق یافته بر راه حل x در نسل t است. این روش رتبه بندی راه حل های قرار گرفته در نواحی فضای تابع هدف را جریمه می کند که بخش های محبوب متراکم نمای Pareto بر آنها تفوق

یافته اند. به عنوان مثال، در شکل 1 ب راه حل i تحت تسلط راه حل های c ، d و e است. بنابراین به آن رتبه 4 داده شده است، هر چند در همان نمای راه حل های f ، g و h است که تنها یک راه حل بر آنها تفوق یافته است. SPEA از رویه رتبه بندی برای تخصیص مقادیر تناسب بهتر به راه حل های تفوق نیافته در نواحی تحت نظر فضای هدف استفاده می کند. در SPEA، لیست خارجی E با اندازه ثابت راه حل های تفوق نیافته ای را ذخیره می کند که در عرض جستجو بررسی شده اند. برای هر راه حل $y \in E_n$ ، مقدار قدرت به شرح زیر تعریف شده است:

$$s(y, t) = \frac{np(y, t)}{N_p + 1},$$

که در آن $np(y, t)$ تعداد راه حل هایی است که y در P تفوق یافته است. رتبه $r(y, t)$ یک راه حل $y \in E$ به صورت $r_3(y, t) = s(y, t)$ اختصاص یافته و رتبه راه حل $x \in P$ به شرح زیر محاسبه شده است:

$$r_3(x, t) = 1 + \sum_{y \in E, y > x} s(y, t).$$

شکل 1 ج نمونه ای از روش رتبه بندی SPEA را شرح می دهد. در دو روش قبلی، تمام راه حل های تفوق نیافته رتبه 1 گرفته اند. اما این روش راه حل را در راه حل های تفوق نیافته پوشش می دهد چون کمترین تعداد راه حل ها را در فضای تابع هدف پوشش می دهد. بنابراین مجموعه گسترده با توزیع یکپارچه از راه حل های تفوق نیافته پیش برده است.

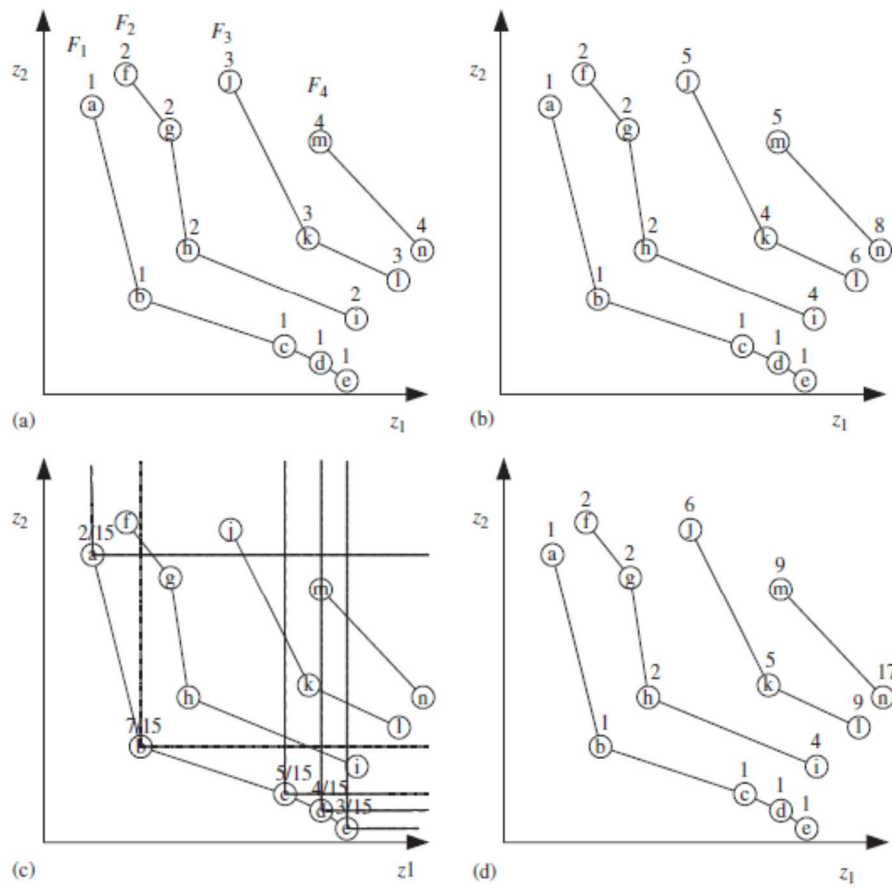
استراتژی چگالی رتبه بندی تجمیعی نیز سعی در اعمال جریمه افزونگی در گروه به دلیل نمود بیش از حد دارد. این روش رتبه بندی به شرح زیر داده است:

$$r_4(x, t) = 1 + \sum_{y \in P, y > x} r(y, t).$$

برای محاسبه رتبه راه حل x ، رتبه راه حل های تفوق یافته بر ایت راه حل باید ابتدا محاسبه گردد. شکل 1 د نمونه ای از این روش رتبه بندی را نشان می دهد (بر مبنای r_2). با استفاده از روش رتبه بندی r_4 ، راه حل های i ، l و n

بالتر از شرکای آنها در همان نمای تفوق نیافته رتبه بندی شده اند چون نسبت سطح تبادل پوش دهنده آن ها با سه راه حل نزدیک به هم c, d و e اشغال شده است.

گرچه برخی از روش های رتبه بندی تعریف شده در این بخش می توانند مستقیماً برای تخصیص مقادیر تناسب به راه حل های فردی مورد استفاده قرار گیرند، معمولاً با تکنیک های اشتراک تناسب متعددی تلفیق شده اند تا به هدف فرعی در بهینه سازی چند منظوره دست یابیم، و نمای Pareto متنوع و یکپارچه ای پیدا شود.



شکل 1 - روش های رتبه بندی استفاده شده در GA چند منظوره

2-5- تنوع: تخصیص تناسب، اشتراک گذاری تناسب و ایجاد تورفتگی

حفظ یک گروه متغیر در ملاحظه مهمی در GA چند موضوعی برای بدست آوردن راه حل هایی است که در نمای Pareto توزیع یکپارچه یافته اند. بدون در نظر گرفتن مقیاس های ممانعتی، گروه گرایش به شکل گیری

کلاسترهای اندکی در GA چند موضوعی دارد. این رخداد توده ژنتیکی نامیده شده است، و چندین روش برای جلوگیری از راندگی (توده) ژنتیکی به شرح زیر اختراع شده اند.

1-2-5- اشتراک تناسب

اشتراک تناسب جستجو در بخش های کاوش نیافته نمای Pareto را با کاهش مصنوعی تناسب راه حل ها در نواحی شلوغ تشویق می کند. برای رسیدن به این هد، نواحی متراکم شناسایی شده اند و روش جریمه ای برای تاوان دادن راه حل های قرار گرفته در چنین نواحی مورد استفاده قرار گرفته است.

ایده اشتراک تناسب اولین بار توسط Goldberg و Richardon در بررسی چندین بهینه محلی برای توابعه چند حالتی ارائه شد. Fonseca و Fleming از این ایده برای جریمه کردن راه حل های کلاستر بندی شده با همان رتبه به شرح زیر استفاده کرده اند:

مرحله 1: محاسبه فاصله اقلیدسی میان هر زوج راه حل x و y در فضای هدف نرمال سازی شده میان 0 و 1 به شرح زیر:

$$dz(x, y) = \sqrt{\sum_{k=1}^K \left(\frac{z_k(x) - z_k(y)}{z_k^{\max} - z_k^{\min}} \right)^2}, \quad (3)$$

که در آن z_k^{\max} و z_k^{\min} مقدار ماکسیمم و مینیمم تابع هدف $z_k(\cdot)$ هستند که در عرض جستجو مشاهده شده اند.

مرحله 2: بر مبنای این فواصل، محاسبه تعداد تورفتگی برای هر راه حل $x \in P$ به صورت زیر:

$$nc(x, t) = \sum_{y \in P, r(y, t) = r(x, t)} \max \left\{ \frac{\sigma_{\text{share}} - dz(x, y)}{\sigma_{\text{share}}}, 0 \right\}, \quad (4)$$

که در آن σ_{share} اندازه تورفتگی است.

مرحله 3: پس از محاسبه تعداد تورفتگی، تناسب هر راه حل به صورت زیر تعیین شده است:

$$f'(x, t) = \frac{f(x, t)}{nc(x, t)}.$$

در رویه بالا، σ_{share} مجاورت راه حل ها را در فضای هدف تعریف می کند (شکل 1 الف). راه حل ها در همان مجاورت بر تعداد تورفتگی یکدیگر تأثیر دارند. بنابراین راه حل در مجاورت شلوغ تعداد تورفتگی بالاتری خواهد داشت و احتمال انتخاب راه حل به عنوان والد را کاهش می دهد. در نتیجه، تورفتگینکثیر راه حل ها را در مجاورت خاص فضای تابع هدف محدود می کند.

جایگزین دیگر استفاده از فاصله در فضای متغیر تصمیم میان دو راه حل x و y است که به شرح زیر تعریف شده است:

$$dx(x, y) = \sqrt{\frac{1}{M} \sum_{i=1}^M (x_i - y_i)^2} \quad (5)$$

معادله (5) مقیاس تفاوت های ساختاری میان دو راه حل است. دو راه حل می توانند در فضای تابع هدف بسیار نزدیک هم باشند در حالیکه ویژگی های ساختاری بسیار متفاوتی دارند. بنابراین اشتراک تناسب مبتنی بر فضای تابع هدف می تواند تنوع در فضای متغیر تصمیم را کاهش دهد. اما Deb و Goldberg گزارش داده اند که اشتراک تناسب در فضای تابع هدف معمولاً کار بهتری نسبت به مورد مبتنی بر فضای متغیر تصمیم انجام می دهد.

یکی از معایب اشتراک تناسب مبتنی بر تعداد تورفتگی این است که کاربر باید پارامتر جدید σ_{share} را انتخاب کند. برای کنترل این مسأله Deb و Goldberg و Fonseca و Fleming مشی های سیستماتیک را برای تخمین و بروز رسانی پویای σ_{share} توسعه دادند. عیب دیگر ایجاد تورفتگی تلاش محاسباتی برای محاسبه تعدد تورفتگی هاست. اما مزایای اشتراک تناسب معمولاً از هزینه تشلا محاسباتی اضافی پیشی می گیرد. Miller و Shaw مشی اشتراک تورفتگی پویایی برای افزایش کارایی محاسبه تعداد تورفتگی را ارائه داده اند.

MOGA اولین GA چند منظوره بود که به صراحت و روشنی از تکنیک های رتبه بندی و تورفتگی مبتنی بر Pareto برای تشویق جستجو به سمت نمای Pareto واقعی در حین حفظ تنوع در گروه استفاده کرده است. بنابراین این مورد نمونه خوبی برای نمایش چگونگی یکپارچه شدن رتبه بندی مبتنی بر Pareto و اشتراک تناسب در یک GA چند منظوره می باشد. رویه MOGA به شرح زیر داده شده است:

رویه MOGA:

مرحله 1: با گروه اولیه تصادفی P_0 شروع کنید. $T=0$ قرار دهید.

مرحله 2: اگر شرایط توقف برقرار است، به P_t بازگردید.

مرحله 3: تناسب گروه را به شرح زیر ارزیابی کنید:

مرحله 1-3- تخصیص رتبه $r(x,t)$ به هر راه حل $x \in P_t$ با استفاده از شمای رتبه بندی ارائه شده در معادله (2).

مرحله 2-3- تخصیص مقادیر تناسب به هر راه حل بر مبنای رتبه راه حل به شرح زیر:

$$f(x, t) = N - \sum_{k=1}^{r(x,t)-1} n_k - .5 \times (n_{r(x,t)} - 1)$$

که در آن n_k تعداد راه حل ها با رتبه k است.

مرحله 3-3- محاسبه تعداد تورفتگی $nc(x,t)$ هر راه حل $x \in P_t$ با استفاده از معادله (4). مرحله 4-3- محاسبه

مقدار تناسب اشتراکی هر راه حل $x \in P_t$ به شرح زیر:

$$f'(x, t) = f(x, t) / nc(x, t).$$

مرحله 5-3- نرمال سازی مقادیر تناسب با استفاده از مقادیر تناسب اشتراکی

$$f''(x, t) = \frac{f'(x, t) n_{r(x,t)}}{\sum_{\substack{y \in P_t \\ r(y,t)=r(x,t)}} f'(x, t)} f(x, t).$$

مرحله 4: استفاده از روش انتخاب تصادفی مبتنی بر f'' برای انتخاب والدین برای مخزن تولید مثل تا گروه فرزند.

$$Q_t \text{ با اندازه } N \text{ پر شود. } P_{t+1} = Q_t.$$

مرحله 5: $t=t+1$ قرار دهید و به مرحله 2 بروید.

در SPEA2، مقیاس چگالی برای تمایز قایل شدن میان راه حل ها با یک رتبه مشابه استفاده شده است، که در آن

چگالی یک راه حل به عنوان معکوس فاصله با k امین نزدیکترین مجاور آن در فضای تابع هدف تعریف شده است.

چگالی یک راه حل مشابه با تعداد تو رفتگی آن است. اما انتخاب مقداری برای پارامتر k رو به جلوتر از انتخاب یک مقدار برای σ_{share} است.

2-2-5- فاصله پر ازدحام

مشی های فاصله شلوغ سعی در حصول گستره یکپارچه ای از راه حل ها در نمای Pareto شناخته شده بدون استفاده از پارامتر اشتراک تناسب دارند. به عنوان مثال، NSGA-II از روش فاصله شلوغ به شرح زیر استفاده می کند (شکل 2 ب):»

مرحله 1: رتبه بندی گروه و نماهای تفوق نایافته F_1, F_2, \dots, F_R ، برا هر نمای $j = 1, \dots, R$ مراحل 2 و 3 را تکرار کنید.

مرحله 2: برای هر تابع هدف k ، راه حل ها را در F_j به ترتیب نزولی مرتب کنید. $l = |F_j|$ قرار دهید و $x_{[i,k]}$ نشان دهنده l امین راه حل در لیست مرتب شده با توجه به تابع هدف k می باشد. $cd_k(x_{[1,k]}) = \infty$ و $cd_k(x_{[l,k]}) = \infty$.

$$cd_k(x_{[i,k]}) = \frac{z_k(x_{[i+1,k]}) - z_k(x_{[i-1,k]})}{z_k^{\max} - z_k^{\min}} \quad i = 2, \dots, l-1$$

را تخصیص دهید و برای $i = 2, \dots, l-1$ اختصاص دهید.

مرحله 3: برای یافتن فاصله شلوغ کلی $cd(x)$ راه حل x ، فواصل شلوغ راه حل را با توجه به هر هدف جمع کنید، یعنی $cd(x) = \sum_k cd_k(x)$.

مزیت اصلی مشی شلوغ شرح یافته در بالا این است که سنجش چگالی گروه در اطراف یک راه حل بدون نیاز به پارامتر تعریف شده کاربر در تکنیک انتخاب صورت می گیرد، نظیر σ_{share} یا k امین همسایه مجاور. در NSGA-II، این تکنیک شلوغ اپراتور انتخاب تورنامنت شلوغ نامیده شده است: بصورت تصادفی دو راه حل x و y انتخاب کنید؛ اگر راه حل ها در همان نمای تفوق نایافته قرار دارند، راه حل با فاصله شلوغ بالاتر برنده استدر غیر اینصورت، راه حل با کمترین رتبه انتخاب شده است.

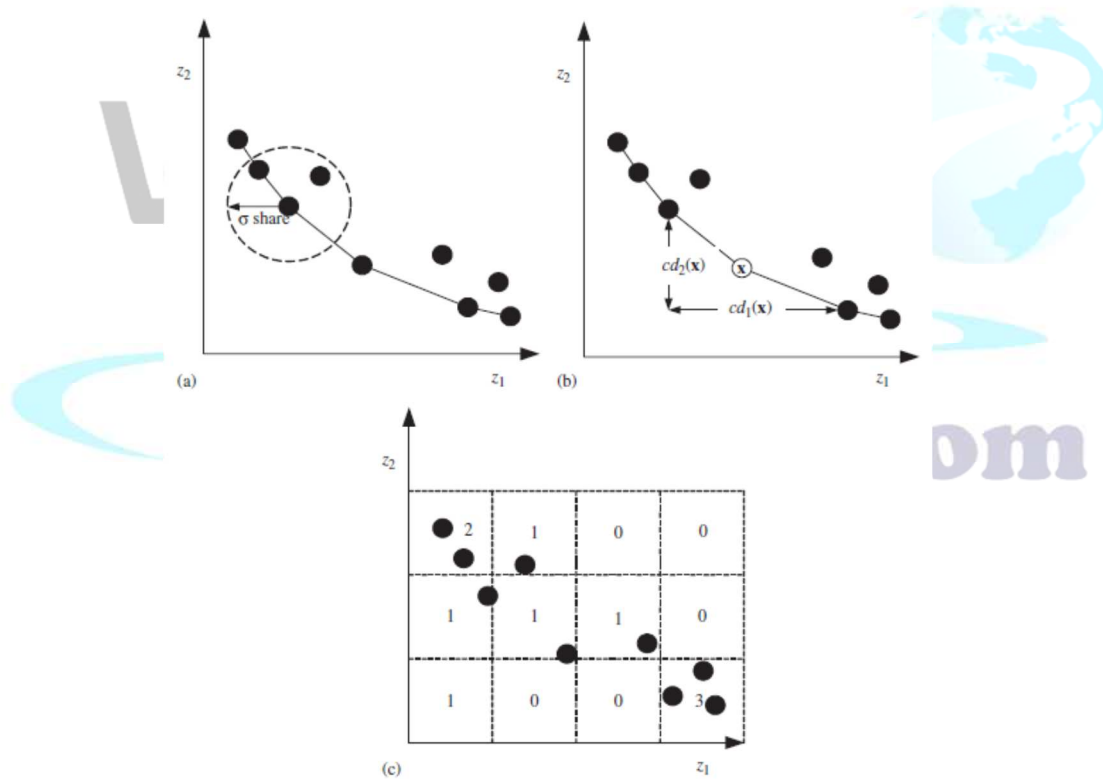
3-2-5- چگالی مبتنی بر سلول

در این روش، فضای هدف به سلول های k بعدی تقسیم شده است (به شکل 2 ج رجوع نمایید). تعداد راه حل ها در هر سلول به عنوان چگالی سلول تعریف شد است، و چگالی یک راه حل معادل با چگالی سلولی است که در آن راه حل قرار گرفته است. این اطلاعات چگالی برای دستیابی به چگالی مشابه با مشی اشتراک تناسب استفاده شده است. به عنوان مثال، در PESA، بین دو راه حل تفوق نیافته، راه حلی که چگالی کمتری دارد ترجیح داده می شود. رویه PESA به شرح زیر داده شده است:

رویه PESA:

$N_E =$ ماکسیمم اندازه آرشیو تفوق نیافته E ,

$N_p =$ اندازه گروه، $n =$ تعداد شبکه ها در هر محور تابع هدف.



شکل 2 - روش های متنوع استفاده شده در GA چند منظوره

مرحله 1: با گروه اولیه تصادفی P_0 شروع کنید و حاصل خارجی $E_0 = \emptyset, t = 0$ قرار دهید.

مرحله 2: فضای هدف نرمال سازی شده را به n^k مکعب تقسیم کنید که در آن n تعداد شبکه ها در یک محور هدف منفرد و K تعداد اهداف است.

مرحله 3: آرشیو E_t را یکی یکی به شرح زیر بروزرسانی کنید.

مورد 1: اگر راه حل جدیدی حداقل تحت نفوذ یک راه حل در E_t است، راه حل جدید را نادیده بگیرید.

مورد 2: اگر راه حل جدید بر برخی راه حل ها در E_t غلبه می یابد، آن راه حل های تفوق یافته را از E_t حذف کنید و به راه حل های موجود در E_t اضافه کنید. عضویت مکعب ها را بروز رسانی کنید.

مورد 3: اگر راه حل جدیدی تحت نفوذ هیچ راه حل دیگری نیست و بر هیچ راه حل دیگری در E_t نفوذ ندارد، این راه حل را به E_t اضافه کنید. اگر $|E_t| = N_E + 1$ است، بصورت تصادفی راه حلی از شلوغ ترین مکعب ها انتخاب کنید تا حذف شوند. عضویت مکعب ها را بروز رسانی کنید.

مرحله 4: اگر شرایط توقف برقرار است، متوقف شوید و به E_t بازگردید.

مرحله 5: $P_t = \emptyset$ قرار دهید و راه حل ها را از E_t برای برخورد و جهش بر مبنای اطلاعات چگالی مکعب ها انتخاب کنید. به عنوان مثال، اگر انتخاب تورنامنت باینری استفاده شده است، برنده راه حلی است که در مکعب هایی قرار گرفته که شلوغی آنها کمتر است. برخورد و جهش را برای تولید منشأ N_p اعمال کنید و آنها را در P_{t+1} کپی کنید.

مرحله 6: $t=t+1$ قرار دهید و به مرحله 3 بروید.

PESA-II از مشی متسقیم تری متابعت می کند، به نام راه حل مبتنی بر ناحیه، که در آن سلول ها و نه راه حل های منفرد در عرض فرایند انتخاب استفاده شده اند. در این روش، سلولی که خلوت تر است نسبت به سلول شلوغ شانس بالاتری برای انتخاب دارد. وقتی سلولی انتخاب شود، راه حل ها در آن سلول به صورت تصادفی برای مشارکت در برخورد و جهش انتخاب می شوند.

Lu و Yen و Lu مشی کارامندی برای شناسایی سلول راه حل در مورد ابعاد سلولی پویا ارائه داده اند. در این روش، عرض یک سلول همراه با k امین ابعاد هدف $(z_k^{\max} - z_k^{\min})/n_k$ است که در آن مقدار تابع هدف ماکسیمم و مینیمم کشف شده است. RDGA از مشی چگالی مبتنی بر سلول به شیوه جالبی برای تبدیل یک

مسأله k هدفی عمومی به مسأله بهینه سازی دو موضوعی با اهداف به حداقل رساندن مقدار رتبه فردی و چگالی گروه استفاده می کند. رویه این مشی به شرح زیر داده شده است:

رویه RDGA:

N_k = تعداد سلول ها در محوز تابع هدف k

مرحله 1: ایجاد گروه والد تصادفی P_0 با اندازه N ، $t=0$ است.

مرحله 2: تقسیم فضای هدف نرمال سازی شده به $n_1 \times n_2 \times \dots \times n_K$ فراسلول.

مرحله 3: بروزرسانی ابعاد سلول ها به صورت $d_k = (z_k^{\max} - z_k^{\min})/n_k$.

مرحله 4: شناسایی عضویت سلول هر راه حل $x \in P_t$.

مرحله 5: تخصیص مقدار چگالی به هر راه حل $x \in P_t$ به صورت $m(x,t)$ = تعداد راه حل های قرار گرفته در همان سلول با x .

مرحله 6: استفاده از روش منبع بندی برای رتبه بندی هر راه حل به صورت زیر:

$$r(x, t) = 1 + \sum_{y \in P_t, y > x} r(y, t).$$

مرحله 7: استفاده از رتبه و چگالی هر راه حل به عنوان اهداف مسأله بهینه سازی دو هدفی. استفاده از مشی

تخصیص تناسب VEGA برای به حداقل رساندن مقدار رتبه فردی و چگالی گروه در حین ایجاد مخزن زایش.

مرحله 8: اعمال برخورد و دگرگونی به مخزن زایش. والد منتخب برخورد را تنها با بهترین راه حل در سلول والد و

سلول های همسایه انجام می دهد. اجازه ندهید فرزندی در سلولی قرار گیرد که مورد نفوذ والدینش است. در صورتی

که والد مورد غلبه فرزند قرار گرفته است والد منتخب را جایگزین کنید. آرشیو راه حل های تفوق نیافته را

بروزرسانی کنید. $T=t+1$ قرار دهید و اگر شرایط توقف برقرار نیست به مرحله 3 بازگردید.

مزیت اصلی مشی چگالی مبتنی بر سلول این است که نگاهت چگالی کلی از فضای تابع هدف در نتیجه محاسبه

چگالی حاصل شده است. جستجو برای نواحی مسکونی تنک تابع هدف بر مبنای این نگاهت کلی مورد تشویق قرار

می گیرد تا راه حل ها خارج از نواحی شلوغ قرار گیرند. مزیت دیگر کارایی محاسباتی آن در مقایسه با تورفتگی یا تکنیک های چگالی مبتنی بر مجاورت است. Lu و Yen چندین ساختار داده ای و الگوریتم برای ذخیره کارآمد اطلاعات سلولی و تغییر چگالی سلول ها ارائه داده اند.

5.3 نابغه گرایی

نابغه گرایی زمینه GA تک هدفی است بدین معنا که بهترین راه حل یافته شده در عرض جستجو همواره تا نسل بعدی باقی می ماند. در این رابطه، تمام راه حل های تفوق نیافته ای که توسط GA چند هدفی کشف شده اند به عنوان راه حل های هوشمندانه در نظر گرفته شده اند. اما، ساده سازی نابغه گرایی در بهینه سازی چند منظوره رو به جلو نیست چون در بهینه سازی هدف منفرد عمده راه حل های هوشمندانه زیادی وجود دارند. اما، جدیدترین GA چند منظوره و تغییرات آنها از نابغه گرایی استفاده می کنند. مانطور که در بخش های 11، 36 و 37 بحث کردیم، GA چند منظوره با استفاده از استراتژی های نابغه گرایانه سعی در پیشی گرفتن از شرکای غیر هوشمندانه خود دارد. GA چند منظوره از دو استراتژی برای پیاده سازی نابغه گرایی استفاده می کند: (1) حفظ راه حل های هوشمندانه در گروه، و (2) مرتب سازی راه حل های هوشمندانه در لیست ثانویه خارجی و ارائه مجدد آن ها به گروه.

1-3-5- استراتژی هایی برای حفظ راه حل های هوشمندانه در گروه

انتخاب تصادفی اطمینان نمی دهد که یک راه حل تفوق نیافته تا نسل بعدی دوام بیاورد. پیاده سازی رو به جلوی نابغه گرایی در GA چند منظوره کپی کردن تمام راه حل های تفوق نیافته در گروه P_t در گروه P_{t+1} ، و سپس پر کردن باقی P_{t+1} با انتخاب از راه حل های تفوق یافته باقیمانده در P_t است. این روش وقتی که تعداد کل والد تفوق نیافته و راه حل های منشأ بزرگتر از N_p هستند کار نمی کند. برای کنترل این مسأله، چندین روش پیشنهاد شده اند.

Konak و Smith یک GA چند منظوره با اندازه گروه پویا و استراتژی نابغه گرایانه محض پیشنهاد کرده اند. در این GA چند منظوره، گروه تنها شامل راه حل های تفوق نیافته است. اگر اندازه گروه به کران بالای N_{max} برسد،

راه حل های $N_{\max} - N_{\min}$ از گروه با توجه به حفظ تنوع نمای تفوق نیافته فعلی حذف می شوند. برای نیل به این مورد، انتخاب تورنامنت غلبه Pareto به شرح زیر استفاده شده است:

دو راه حل بصورت تصادفی انتخاب شده اند و راه حل با تعداد تورفتگی بیشتر حذف شده است تا تمام راه حل ها تفوق نیافته باشند. GA چند منظوره نابغه گرایانه محض با اندازه گروه پویا نیز پیشنهاد شده است.

NSGA-II از اندازه گروه ثابت N استفاده می کند. در نسل t ، گروه منشأ Q_t با اندازه N از گروه والد P_t ایجاد شده است و نماهای تفوق نیافته F_1, F_2, \dots, F_R و در گروه تلفیقی $P_t \cup Q_t$ مشخص شده اند. گروه بعدی

با شروع از راه حل های موجود در F_1 ، و سپس F_2 پر شده اند. K را برابر شاخص نمای تفوق نیافته F_k قرار دهید که $|F_1 \cup F_2 \cup \dots \cup F_k| \leq N$ و $|F_1 \cup F_2 \cup \dots \cup F_k \cup F_{k+1}| > N$ است. ابتدا،

تمام راه حل ها در نمای F_1, F_2, \dots, F_k داخل P_{t+1} کپی شده است و سپس خلوت ترین راه حل $(N - |P_{t+1}|)$ به P_{t+1} اضافه شده است. این روش اطمینان می دهد که تمام راه حل های تفوق نیافته در

گروه بعدی شامل شده اند اگر $|F_1| \leq N$ ، باشد، و انتخاب صانویه مبتنی بر فضاله شلوغ تنوع را افزایش می

دهد. رویه کامل NSGA-II در زیر داده شده تا پیاده سازی نابغه گرایی بدون استفاده از گروه ارجی نسل دوم نشان داده شود.

TarjomeFa.Com

رویه NSGA-II:

مرحله 1: ایجاد گروه والد تصادفی P_0 با اندازه N . $t=0$ قرار دهید.

مرحله 2: اعمال برخورد و دگرگونی به P_0 برای ایجاد گروه فرزند Q_0 با اندازه N .

مرحله 3: اگر شرایط توقف برقرار است، متوقف شوید و به P_t بازگردید.

مرحله 4: $R_t = P_t \cup Q_t$ قرار دهید.

مرحله 5: استفاده از الگوریتم مرتب سازی تفوق نیافته سریع، شناسایی نماهای تفوق نیافته F_1, F_2, \dots, F_k در

R_t .

مرحله 6: برای $i = 1, \dots, k$ مراحل زیر را انجام دهید:

مرحله 6-1- محاسبه فاصله شلوع راه ها در F_i .

مرحله 6-2: ایجاد P_{t+1} به شرح زیر:

مورد 1: اگر $|P_{t+1}| + |F_i| \leq N$ است، بنابراین $P_{t+1} = P_{t+1} \cup F_i$ قرار دهید.

مورد 2: اگر $|P_{t+1}| + |F_i| > N$ ، است، خلوت ترین راه های $N - |P_{t+1}|$ را از F_i در P_{t+1} قرار

دهید.

مرحله 7: استفاده از انتخاب تونامنت مبتنی بر فاصله شلوع برای انتخاب والدین از P_{t+1} . اعمال برخورد و دگرگونی به

P_{t+1} برای ایجاد گروه فرزند Q_{t+1} با اندازه N .

مرحله 8: $t=t+1$ قرار دهید و به مرحله 3 بروید.

توجه داشته باشید که گروه فرزند و والد تلفیق شده شامل بیش از n راه حل تفوق نیافته می باشد، و NSGA-II

به GA نابغه گرایتنه محضی تبدیل می شود که در آن تنها راه حل های تفوق نیافته در برخورد و انتخاب شرکت

می کنند و مزیت اصلی حفظ راه حل های تفوق نیافته در گروه پیاده سازی رو به جلو است. در این استراتژی، اندازه

گروه پارامتر GA مهمی است چون هیچ آرشیو خارجی برای ذخیره راه حل های تفوق نیافته کشف شده استفاده

نشده است.

2-3-5- نابغه گرایی با گروه های خارجی

وقتی لیست خارجی برای ذخیره سازی راه حل های نابغه گرایانه استفاده شده است، چندین موضوع را باید در نظر

داشت. اولین مورد این است که کدام راه حل ها در لیست نابغه گرایی E ذخیره می شوند. اغلب GA های چند

منظوره راه حل های تفوق نیافته را در عرض جستجو ذخیره می کنند، و E هر بار با ایجاد یک راه حل جدید با

حذف راه حل های هوشمند گرایانه تفوق یافته توسط یک راه حل جدید با اضافه کردن راه حل جدید ذخیره می

کند. چندین ساختار داده ای برای ذخیره سازی، بروزرسانی، و جستجوی کارآمد در لیست E ارائه شده اند. موضوع

دیگر اندازه لیست E است. چون ممکن است تعداد بسیار زیادی راه حل بهینه Pareto برای یک مسأله وجود داشته

باشد، لیست نابغه ها می تواند رشد زیادی داشته باشد. بنابراین، تکنیک های اصلاح کردن برای کنترل اندازه E ارائه شده اند. به عنوان مثال، SPEA از روش کلاستر بندی پیوند میانگین برای کاهش اندازه E به حد بالاتر N استفاده می کند و این زمانی است که تعداد راه حل های تفوق نیافته از N به صورت زیر تجاوز می کند.

مرحله 1: در ابتدا، هر راه حل $x \in E$ را به C_i اختصاص دهید، یعنی $C = \{c_1, c_2, \dots, c_M\}$

مرحله 2: محاسبه فاصله میان تمام زوج های کلاسترهای C_i و C_j به صورت زیر:

$$d_{c_i, c_j} = \frac{1}{|c_i| \cdot |c_j|} \sum_{x \in c_i, y \in c_j} d(x, y).$$

در اینجا، فاصله $d(x, y)$ می تواند در فضای تابع هدف با استفاده از معادله 3 یا در فضای متغیر تصمیم با استفاده از معادله (5) محاسبه گردد.

مرحله 3: ادغام زوج کلاستر C_i و C_j با فاصله کمینه در میان تمام کلاسترها در یک کلاستر جدید.

مرحله 4: اگر $|C| \leq N$ است، به مرحله 5 بروید، در غیر این صورت به مرحله 2 بروید.

مرحله 5: برای هر کلاستر، راه حلی با مینیمم فاصله میانگین برای تمام راه حل های دیگر در همان کلاستر تعیین کنید. راه حل های اصلی را برای هر کلاستر حفظ کنید و دیگر راه حل ها را از E حذف کنید.

موضوع نهایی انتخاب راه حل های هوشمندانه از E است که باید به گروه مجدداً ارائه شوند. در (11، 19 و 20) راه

حل ها ب رای P_{t+1} انتخاب شده اند. برای پیاده سازی این استراتژی، گروه های P_t و E_t تلفیق شده اند، و مقدار

تناسبی به هر راه حل در گروه تلفیقی $P_t \cup E_t$ اختصاص داده شده است. و سپس N راه حل برای نسل بعدی P_{t+1}

بر مبنای مقادیر تناسب یافته انتخاب شده است. در این استراتژی، راه حل های $N-n$ از والدین انتخاب

شده اند و جدیداً فرزند جدید ایجاد شده و n راه حل از E_t انتخاب شده است.

SPEA و SPEA2 الگوریتم های کارامدی هستند که از لیست خارجی برای ذخیره راه حل های تفوق نیافته

کشف شده در جستجو استفاده می کنند. آنها نمونه های خوبی برای استفاده از گروه های خارجی هستند. رویه

SPEA2 به شرح زیر داده شده است:

N_E = ماکسیمم اندازه آرشیو تفوق نیافته E

N_P = اندازه گروه

K = پارامتر برای محاسبه چگالی $(k = \sqrt{N_E + N_P})$.

مرحله 1: تولید تصادفی راه حل اولیه P_0 و تعیین $E_0 = \emptyset$.

مرحله 2: محاسبه تناسب هر راه حل x در $P_t \cup E_t$ به شرح زیر:

مرحله 1-2 $r(x, t) = \sum_{y \in P_t \cup E_t, y > x} s(y, t)$ است که در آن $s(y, t)$ تعداد ره حل های موجود در

$P_t \cup E_t$ است که راه حل y بر آنها تفوق یافته است.

مرحله 2-2 محاسبه چگالی به صورت $m(x, t) = (\sigma_x^k + 1)^{-1}$ که در آن σ_x^k فاصله میان راه حل x و k آمین همسایه نزدیک آن است.

مرحله 2-3 تخصیص مقدار تناسب به صورت $f(x, t) = r(x, t) + \bar{m}(x, t)$.

مرحله 3: کپی کردن تمام راه حل های تفوق نیافته در $P_t \cup E_t$ درون E_{t+1} . دو مورد ممکن هستند:

مورد 1: اگر $|E_{t+1}| > N_E$ باشد، بنابراین راه حل های $|E_{t+1}| - N_E$ را با حذف تکراری راه حل ها با

ماکسیمم اصله σ^k هرس کنید. تمام گره ها را با آزمودن σ^l برای $l = k-1, \dots, 1$ بصورت پشت سر هم باز کنید.

مورد 2: اگر $|E_{t+1}| \leq N_E$ است، بهترین $N_E - |E_{t+1}|$ راه حل های تفوق یافته را طبق مقادیر

تناسب آنها از $P_t \cup E_t$ درون E_{t+1} کپی کنید.

مرحله 4: اگر شرایط توقف برقرار است، متوقف شوید و به راه حل های تفوق نیافته موجود در E_{t+1} بازگردید.

مرحله 5: انتخاب والدین از E_{t+1} با استفاده از انتخاب تورنامنت باینری با جایگزینی.

مرحله 6: اعمال اپراتورهای برخورد و جهش (دگرگونی) به والدین برای ایجاد راه حل های فرزند N_P . کپی کردن فرزند به $P_{t+1}, t = t + 1$ ، و سپس رفتن به مرحله 2.

نمونه های دیگر مشی های هوشمندانه با استفاده از گروه های خارجی PESA, RDGA, RWGA و DMOEA هستند.

4-5- کنترل محدود

اغلب مسائل بهینه سازی دنیای واقعی شامل محدودیت هایی هستند که باید برطرف شوند و بررسی مناسب تکنیک های کنترل محدودیت ها که در الگوریتم های تکاملی استفاده شده است توسط Colelo ارائه گردیده است. یک GA تک موضوعی می تواند از یکی از چهار استراتژی کنترل محدودیت مختلف استفاده کند: (1) حذف راه حل های غیر عملی (2) کاهش تناسب راه حل های غیر عملی با استفاده از تابع جریمه؛ (3) شکل دادن ماهرانه اپراتورهای ژنتیکی برای تولید راه حل های عملی و (4) تبدیل راه حل های غیر عملی به راه حل های عملی. کنترل محدودیت ها بطرز کافی برای GA چند منظوره مورد جستجو قرار نگرفته است. به عنوان نمونه، تمام GA های چند منظوره عمده فرض را بر این می گذارند که مسائل بدون محدودیت هستند. در حالیکه استراتژی های کنترل محدودیت (1)، (3) و (4) مستقیماً قابل کاربرد در مورد چند موضوعی هستند، و پیاده سازی استراتژی های تابع جریمه که به کرات در استراتژی کنترل محدودیت در GA تک منظوره استفاده شده است در GA چند منظوره رو به جلو نیست و این به دلیل این حقیقت است که تخصیص تناسب مبتنی بر رتبه تفوق نایافته یک راه حل است، نه مقادیر تابع هدف آن.

Jimenez و همکارانش استراتژی انتخاب دارای تورفتگی را برای کنترل غیر عملی بودن مسائل چند منظوره به شرح زیر ارائه داده اند:

مرحله 1: انتخاب تصادفی دو راه حل X و Y از گروه

مرحله 2: اگر یکی از راه حل ها عملی است و دیگری غیرعملی، برنده راه حل عملی است و توقف. در غیر اینصورت، اگر هر دو راه حل غیر عملی هستند به مرحله 3 بروید، در غیر اینصورت به مرحله 4 بروید.

مرحله 3: در این مورد، راه حل های X و Y هر دو غیر عملی اند. بنابراین از مجموعه مرجع تصادفی C در میان راه حل های غیر عملی موجود در گروه موردی انتخاب کنید. راه حل های X و Y را با راه حل های موجود در مجموعه مرجع C با توجه به درجه نامحتمل بودنشان مقایسه کنید. برای رسیدن به این مورد، مقیاس غیرعملی بودن را برای راه حل های X و Y محاسبه کنید و مواردی که در مجموعه C وجود دارند را در نظر بگیرید. اگر یکی از راه حل های X و Y بهتری است و دیگری بدتر از بهترین راه حل موجود در C است، با توجه به مقیاس نامحتمل بودن محاسبه شده، برنده راه حل است که نامحتمل بودن آن کمینه است.

اما، اگر یک گروه وجود دارد، یعنی، هر دو راه حل X و Y بهتر یا بدتر از بهترین راه حل موجود در C هستند، بنابراین تعداد تورفتگی آنها در فضای متغیر تصمیم برای انتخاب استفاده شده است. در این مورد، راه حل با تعداد تورفتگی کمتر برنده است.

مرحله 4: در موردی که راه حل های X و Y هر دو عملی اند، مجموعه مرجع تصادفی C را از میان راه حل های عملی موجود در گروه انتخاب کنید. راه حل های X و Y را با راه حل های موجود در مجموعه C مقایسه کنید. اگر یک از آنها در مجموعه C تفوق نیافته است و دیگری حداقل تحت نفوذ یک راه حل است، برنده مورد اول است. به عبارت دیگر، گروه ای میان راه حل های X و Y وجود دارد و تعداد تورفتگی راه حل ها در فضای متغیر تصمیم محاسبه شده است. راه حل با تعداد تورفتگی کمتر برنده انتخاب تورنامنت است.

رویه بالا مشی بسیط و جامعی برای برقراری ارتباط با غیرعملی بودن در حین حفظ تنوع و غلبه گروه است. معایب عمده این رویه پیچیدگی محاسباتی آن و اندازه تورفتگی آن است. اما، تغییرات نیز محتملند. در مرحله 4، به عنوان مثال، تعداد تورفتگی راه حل ها می تواند در فضای تابع هدف به جای فضای متغیر تصمیم محاسبه شود. در مرحله 3، راه حل با حداقل نامحتمل بودن می تواند به عنوان برنده اعلام شود بدون اینکه راه حل های X و Y با مجموعه مرجع C با توجه به غیر عملی بودن مقایسه شوند. چنین تغییراتی می توانند پیچیدگی محاسباتی رویه را کاهش دهند.

Deb و همکارانش مفهوم غلبه محدودیت و روش انتخاب تورنامنت مبتنی بر آن را ارائه داده اند که روش تورنامنت

محدود شده نامیده شده است. راه حل X راه حل غلبه محدودیت Y است اگر یکی از موارد زیر برقرار باشند:

مورد 1: راه حل X عملی و راه حل Y غیر عملی باشد.

مورد 2: راه حل های X و Y هر دو غیر عملی باشند؛ اما راه حل X محدودیت کمتری نسبت به Y داشته باشد.

مورد 3: راه حل های X و Y هر دو عملی باشند و راه حل X بر راه حل Y غلبه داشته باشد.

در روش تورنامنت محدود، ابتدا نماهای غلبه غیر محدود $F_1, F_2, F_3, \dots, F_R$ در شرایط غلبه محدودین به

جای مفهوم غلبه مداوم شناسایی شده اند. توجه داشته باشید که مجموعه F_1 متناظر با مجموعه راه حل های تفوق

نیافته در گروه هستند و نمای F_i به F_j ترجیح داده می شود. در انتخاب تورنامنت محدودیت دو راه حل X و Y

بصورت تصادفی از گروه انتخاب شده اند. بین X و Y برنده موردی است که اولویت بیشتری دارد. اگر راه حل های X

و Y هر دو در همان نما باشند، بنابراین برنده بر مبنای تعداد تورفتگی یا فاصله شلوفی راه حل انتخاب می شود.

مزیت اصلی روش تورنامنت محدود این است که این روش مستلزم پارامترهای کمتری است و می تواند به سادگی

به یک GA چند منظوره تبدیل شود. مشی مشابهی به نام انتخاب تورنامنت مبتنی بر غلبه، توسط Coello و

Montes برای حل مسائل تک هدفی با چندین محدودیت مشکل با استفاده از نسخه تغییر یافته NPGA استفاده

شده است. در این مشی انتخاب، غلبه با توجه به محدودیت های دو راه حل تعریف شده است. راه حل غیر عملی X

بر Y غلبه می یابد اگر محدودیت های کمتر یا معادل با آن داشته باشد. گره میان دو راه حل تفوق نیافته محدود با

تخطی از محدودیت های راه حل ها حل شده است.

5-5 GA چند منظوره موازی و پیوندی

تمام مطالعات قیاسی در مورد GA چند هدفی در مورد این توافق دارند که مکانیسم های حفظ هوشمندانه و

تنوع عملکرد را بهبود می دهند. اما، پیاده سازی نابغه گرایی و استراتژی های حفظ تنوع معمولاً مستلزم تلاش

محاسباتی اساسی و حافظه کامپیوتری است. به علاوه، ارزیابی توابع هدف می تواند زمان قابل توجهی در مسائل

دنیای حقیقی بگیرد. بنابراین محققان سرگرم کاشه زمان اجرا و الزامات منبع GA چند منظوره با استفاده از

ساختارهای داده ای پیشرفته شده اند. یکی از آخرین گرایش ها پردازش موازی و توزیع شده است. چندین مقاله جدید پیاده سازی موازی GA چند منظوره را در چندین پردازشگر ارائه کرده اند.

تلفیق کردن GA با الگوریتم های جستجوی محلی مکرراً در GA تک منظوره استفاده شده است. از این روش معمولاً با عنوان الگوریتم ممتیک یاد می شود. معمولاً یک الگوریتم جستجوی محلی به شرح زیر پیشی می گیرد:

مرحله 1: شروع با راه حل اولیه X

مرحله 2: تولید مجموعه ای از راه حل های همسایه در اطراف راه حل X با استفاده از قانون اختلال ساده.

مرحله 3: اگر بهترین راه حل در مجموعه مجاورت بهتر از X است، X را با این راه حل جایگزین کنید و به مرحله 2 بروید، در غیر اینصورت کار را متوقف کنید.

Paquete و Stutzle یک GA دو هدفی را شرح داده اند که در آن جستجوی محلی برای تولید راه حل های اولیه با بهینه سازی یک هدف استفاده شده است. Deb و Goel جستجوی محلی را برای راه حل های نهایی به کار برده اند. در مشی Ishinuchi و murata یک رویه جستجوی محلی به هر منشأ تولید شده با برخورد با استفاده از همان بردار وزنی والدین فرزند برای ارزیابی راه حل های مجاور اعمال شده است. مشابهاً، Ishibuchi و همکارانش نیز از مجموع وزن دهی شده توابع هدف برای ارزیابی راه حل ها در عرض جستجوی محلی استفاده کرده اند. اما جستجوی محلی بصورت انتخابی به راه حل های امید بخش اعمال شده است و وزن ها نیز بصورت تصادفی تولید شده اند و با استفاده از بردار وزن والدین تولید نشده اند. Knowels و Corne نسخه ای از PAES به نام MPAES ارائه کرده اند. PAES از مفهوم غلبه برای ارزیابی راه حل ها استفاده می کند و بنابراین در M-PAES مجموعه ای از راه حل های تفوق نیافته محلی در عرض جستجوی محلی بررسی شده اند. وقتی راه حل جدیدی در مجاورت ایجاد می شود، تنها با این مجموعه تفوق نیافته مقایسه می شود و بروزرسانی هایی لازمند. جستجوی محلی پس از اینکه ماکسیمم تعداد راه حل های محلی بررسی شدند یا ماکسیمم تعداد حرکات محلی بدون هیچ گونه بهبودی انجام شدند خاتمه می یابد. Tan و همکارانش اعمال یک رویه جستجوی محلی را به راه حل هایی پیشنهاد کرده که

جدای از دیگران قرار گرفته اند. به علاوه، اندازه مجاورت جستجوی محلی بسته به تنوع یا شلوغی راه حل هاست. با انتخابی بودن در اعمال جستجوی محلی، این استراتژی از نظر محاسباتی در حین حفظ تنوع کارآمد می شود.

6- GA چند منظوره برای بهینه سازی قابلیت اطمینان

بسیاری از مسائل مهندسی چندین هدف دارند شامل طرح سیستم مهندسی و بهینه سازی قابلیت اطمینان. چندین پیاده سازی جالب و موفق از GA چند منظوره برای این رده از مسائل وجود دارند. این موارد در پاراگراف های زیر شرح داده شده اند.

Marseguerra و همکارانش وقفه های تست نظارت بهینه را با استفاده از GA چند منظوره با هدف بهبود قابلیت اطمینان و دستسری پذیری تعیین کرده اند. تحقیق آنها GA چند منظوره را پیاده سازی کرده که به صراحت غیر قطعی بودن ها را در پارامترها در نظر می گیرد. اهداف در نظر گرفته شده عکس احتمال شکست سیستم مرود انتظار و عکس واریانس آن بوده اند. این موارد برای به جلو راندن جستجوی ژنتیک به سمت راه حل هایی استفاد ه شده که متضمن ارائه عملکرد بهینه با اطمینان بالا هستند. آنها بطرز موفقیت آمیزی رویه خود را در مورد سیستم پیچیده اعمال کرده اند.

Martorell و همکارانش انتخاب خصایص فنی و حفظ فعالیت ها را در کارخانه های انرژی هسته ای برای افزایش قابلیت اطمینان، دسترس پذیری و قابلیت نگهداری (RAM) تجهیزات مرتبط با ایمنی مطالعه کرده اند. اما برای بهبود RAM منابع محدود اضافی نیاز هستند. آنها نشان دهنده حیات پذیری و اهمیت روش پیشنهادی خود برای اطمینان دادن در مورد سطوح RAM بالا برای سیستم های مرتبط با ایمنی بودند. برای این مسائل، تصمیم گیرنده با مسأله بهینه سازی چند منظوره روبرو است که در آن پارامترهای طراحی، آزمایش و نگهداری متغیرهای تصمیم هستند. راه حل ها با استفاده از GA تک منظوره و GA چند منظوره بدست آمده اند.

Sasaki و Gen یک مسأله چند منظوره ارائه کرده اند که چندین تابع هدف فازی دارند و محدودیت های آن دارای کران بالای تعمیم بیافته و ساختاری هستند. آنها این مسأله را با استفاده از GA ترکیبی جدید حل کردند. برای نشان دادن کارایی روش شان، آنها مسأله طرح قابلیت اطمینان سیستم را مورد تجزیه و تحلیل قرار دادند.

تخصیص قابلیت اطمینان برای کمینه سازی هزینه های کارخانه مربوط به هدف ایمنی کارخانه توسط Yang و همکارانش پیشنهاد شد. برای مسأله آنها، بهینه سازی طرح برای بهبود طرح، عملیات و ایمنی کارخانه های انرژی هسته ای جدید نیاز است.

DJALLAH و Elegbede متدلوژی برای بهینه سازی دسترس پذیری و هزینه سیستم های سری موازی قابل تعمیر ارائه کردند. این بهینه سازی بهین سازی ای تلفیقی است و با متغیرها پیوسته و گسسته مدلسازی شده است. آنها مسأله را به مسأله تک هدفی تبدیل کردند.

DEB و همکارانش مسأله بهینه سازی دو منظوره کمینه سشازی طول سیم مجموع و کمینه سازی نرخ خطا در طرح گسترده مدار اعمال کردند.

KUMAR و همکارانش یک مشی GA چند منظوره برای طراحی شبکه های مخابراتی در حین کمینه سازی همزمان عملکرد شبکه و هزینه های طراحی تحت محدودیت قابلیت اطمینان ارائه کردند.

References

- [1] Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: empirical results. *Evol Comput* 2000;8(2):173-95.
- [2] Holland JH. *Adaptation in natural and artificial systems*. Ann Arbor: University of Michigan Press; 1975.
- [3] Goldberg DE. *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison-Wesley; 1989.
- [4] Jones DF, Mirrazavi SK, Tamiz M. Multiobjective meta-heuristics: an overview of the current state-of-the-art. *Eur J Oper Res* 2002;137(1):1-9.
- [5] Schaffer JD. Multiple objective optimization with vector evaluated genetic algorithms. In: *Proceedings of the international conference on genetic algorithm and their applications*, 1985.
- [6] Fonseca CM, Fleming PJ. Multiobjective genetic algorithms. In: *IEE colloquium on 'Genetic Algorithms for Control Systems Engineering'* (Digest No. 1993/130), 28 May 1993. London, UK: IEE; 1993.
- [7] Horn J, Nafpliotis N, Goldberg DE. A niched Pareto genetic algorithm for multiobjective optimization. In: *Proceedings of the first IEEE conference on evolutionary computation*. IEEE world congress on computational intelligence, 27-29 June, 1994. Orlando, FL, USA: IEEE; 1994.
- [8] Hajela P, lin C-y. Genetic search strategies in multicriterion optimal design. *Struct Optimization* 1992;4(2):99-107.
- [9] Murata T, Ishibuchi H. MOGA: multi-objective genetic algorithms. In: *Proceedings of the 1995 IEEE international conference on evolutionary computation*, 29 November-1 December, 1995. Perth, WA, Australia: IEEE; 1995.
- [10] Srinivas N, Deb K. Multiobjective optimization using nondominated sorting in genetic algorithms. *J Evol Comput* 1994;2(3):221-48.
- [11] Zitzler E, Thiele L. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE Trans Evol Comput* 1999;3(4):257-71.

eFa.Com

- [12] Zitzler E, Laumanns M, Thiele L. SPEA2: improving the strength Pareto evolutionary algorithm. Swiss Federal Institute Technology; Zurich, Switzerland; 2001.
- [13] Knowles JD, Corne DW. Approximating the nondominated front using the Pareto archived evolution strategy. *Evol Comput* 2000;8(2):149–72.
- [14] Corne DW, Knowles JD, Oates MJ. The Pareto envelope-based selection algorithm for multiobjective optimization. In: Proceedings of sixth international conference on parallel problem solving from Nature, 18–20 September, 2000. Paris, France: Springer; 2000.
- [15] Corne D, Jerram NR, Knowles J, Oates J. PESA-II: region-based selection in evolutionary multiobjective optimization. In: Proceedings of the genetic and evolutionary computation conference (GECCO-2001), San Francisco, CA, 2001.
- [16] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 2002;6(2):182–97.
- [17] Sarker R, Liang K-H, Newton C. A new multiobjective evolutionary algorithm. *Eur J Oper Res* 2002;140(1):12–23.
- [18] Coello CAC, Pulido GT. A micro-genetic algorithm for multi-objective optimization. In: Evolutionary multi-criterion optimization. First international conference, EMO 2001, 7–9 March, 2001. Zurich, Switzerland: Springer; 2001.
- [19] Lu H, Yen GG. Rank-density-based multiobjective genetic algorithm and benchmark test function study. *IEEE Trans Evol Comput* 2003;7(4):325–43.
- [20] Yen GG, Lu H. Dynamic multiobjective evolutionary algorithm: adaptive cell-based rank and density estimation. *IEEE Trans Evol Comput* 2003;7(3):253–74.
- [21] Coello CAC. A comprehensive survey of evolutionary-based multi-objective optimization techniques. *Knowl Inform Syst* 1999;1(3): 269–308.
- [22] Coello CAC. An updated survey of evolutionary multiobjective optimization techniques: state of the art and future trends. In: Proceedings of the 1999 congress on evolutionary computation-CEC99, 6–9 July 1999. Washington, DC, USA: IEEE.
- [23] Coello CAC. An updated survey of GA-based multiobjective optimization techniques. *ACM Comput Surv* 2000;32(2):109–43.
- [24] Fonseca CM, Fleming PJ. Genetic algorithms for multiobjective optimization: formulation, discussion and generalization. In: Proceedings of the ICGA-93: fifth international conference on genetic algorithms, 17–22 July 1993. Urbana-Champaign, IL, USA: Morgan Kaufmann; 1993.
- [25] Fonseca CM, Fleming PJ. Multiobjective optimization and multiple constraint handling with evolutionary algorithms. I. A unified formulation. *IEEE Trans Syst Man Cybern A* 1998;28(1): 26–37.
- [26] Jensen MT. Reducing the run-time complexity of multiobjective EAs: The NSGA-II and other algorithms. *IEEE Trans Evol Comput* 2003;7(5):503–15.
- [27] Xiujian L, Zhongke S. Overview of multi-objective optimization methods. *J Syst Eng Electron* 2004;15(2):142–6.
- [28] Coello CAC. 2005, <http://www.lania.mx/~ccoello/EMOO/EMOObib.html>
- [29] Knowles J, Corne D. The Pareto archived evolution strategy: a new baseline algorithm for Pareto multiobjective optimisation. In: Proceedings of the 1999 congress on evolutionary computation-CEC99, 6–9 July 1999. Washington, DC, USA: IEEE; 1999.
- [30] Deb K, Agrawal S, Pratap A, Meyarivan T. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Proceedings of sixth international conference on parallel problem solving from nature, 18–20 September, 2000. Paris, France: Springer; 2000.
- [31] Murata T, Ishibuchi H, Tanaka H. Multi-objective genetic algorithm and its applications to flowshop scheduling. *Comput Ind Eng* 1996;30(4):957–68.
- [32] Kursawe F. A variant of evolution strategies for vector optimization. In: Parallel problem solving from nature. First workshop, PPSN 1 proceedings, 1–3 October, 1990. Dortmund, West Germany: Springer; 1991.
- [33] Goldberg DE, Richardson J. Genetic algorithms with sharing for multimodal function optimization. In: Genetic algorithms and their applications; proceedings of the second international conference on genetic algorithms, 28–31 July, 1987. Cambridge, MA, USA: Lawrence Erlbaum Associates; 1987.
- [34] Deb K, Goldberg DE. An investigation of of niche an species fromation in genetic function optimization. In: Proceedings of the third international conference on genetic algorithms, George Mason University, 1989.
- [35] Miller BL, Shaw MJ. Genetic algorithms with dynamic niche sharing for multimodal function optimization. In: Proceedings of the 1996 IEEE international conference on evolutionary computation, ICEC'96, May 20–22, 1996, Nagoya, Japan. Piscataway, NJ, USA: IEEE; 1996.
- [36] Deb K. Multi-objective optimization using evolutionary algorithms. New York: Wiley; 2001.
- [37] Van Veldhuizen DA, Lamont GB. Multiobjective evolutionary algorithms: analyzing the state-of-the-art. *Evol Comput* 2000;8(2): 125–47.
- [38] Konak A, Smith AE. Multiobjective optimization of survivable networks considering reliability. In: Proceedings of the 10th international conference on telecommunication systems. Monterey, CA: Naval Postgraduate School; 2002.
- [39] Konak A, Smith AE. Capacitated network design considering survivability: an evolutionary approach. *J Eng Optim* 2004;36(2): 189–205.
- [40] Fieldsend JE, Everson RM, Singh S. Using unconstrained elite archives for multiobjective optimization. *IEEE Trans Evol Comput* 2003;7(3):305–23.
- [41] Mostaghim S, Teich J, Tyagi A. Comparison of data structures for storing Pareto-sets in MOEAs. In: Proceedings of the 2002 world congress on computational intelligence—WCCI'02, 12–17 May, 2002. Honolulu, HI, USA: IEEE; 2002.
- [42] Morse JN. Reducing the size of the nondominated set: pruning by clustering. *Comput Oper Res* 1980;7(1–2):55–66.
- [43] Ishibuchi H, Murata T. Multi-objective genetic local search algorithm. In: Proceedings of the IEEE international conference on evolutionary computation, 20–22 May, 1996. Nagoya, Japan: IEEE; 1996.
- [44] Lu H, Yen GG. Rank-density based multiobjective genetic algorithm. In: Proceedings of the 2002 world congress on computational intelligence—WCCI'02, 12–17 May, 2002. Honolulu, HI, USA: IEEE; 2002.
- [45] Coello CAC. A survey of constraint handling techniques used with evolutionary algorithms. Veracruz, Mexico: Laboratorio Nacional de Informtica Avanzada; 1999.
- [46] Jimenez F, Gomez-Skarmeta AF, Sanchez G, Deb K. An evolutionary algorithm for constrained multi-objective optimization. In: Proceedings of the 2002 world congress on computational intelligence—WCCI'02, 12–17 May, 2002. Honolulu, HI, USA: IEEE; 2002.
- [47] Jimenez F, Verdegay JL, Gomez-Skarmeta AF. Evolutionary techniques for constrained multiobjective optimization problems. In: Workshop on multi-criterion optimization using evolutionary methods GECCO-1999, 1999.
- [48] Coello CAC, Montes EM. Constraint-handling in genetic algorithms through the use of dominance-based tournament selection. *Adv Eng Inform* 2002;16(3):193–203.
- [49] de Toro F, Ortega J, Fernandez J, Diaz A. PSFGA: a parallel genetic algorithm for multiobjective optimization. In: Proceedings of the 10th Euromicro workshop on parallel, distributed and network-based processing, 9–11 January, 2002. Canary Islands, Spain: IEEE Computer Society.
- [50] Van Veldhuizen DA, Zydallis JB, Lamont GB. Considerations in engineering parallel multiobjective evolutionary algorithms. *IEEE Trans Evol Comput* 2003;7(2):144–73.

- [51] Wilson LA, Moore MD, Picarazzi JP, Miquel SDS. Parallel genetic algorithm for search and constrained multi-objective optimization. In: Proceedings of the 18th international parallel and distributed processing symposium, 26–30 April, 2004. Santa Fe, NM, USA: IEEE Computer Society; 2004.
- [52] Xiong S, Li F. Parallel strength Pareto multiobjective evolutionary algorithm. In: Proceedings of the fourth international conference on parallel and distributed computing, applications and technologies, 27–29 August, 2003. Chengdu, China: IEEE; 2003.
- [53] Knowles JD, Corne DW. M-PAES: a memetic algorithm for multiobjective optimization. In: Proceedings of the 2000 congress on evolutionary computation, 16–19 July, 2000. La Jolla, CA, USA: IEEE; 2000.
- [54] Paquete L, Stutzle T. A two-phase local search for the biobjective traveling salesman problem. In: Evolutionary multi-criterion optimization. Proceedings of the second international conference, EMO 2003, 8–11 April, 2003. Faro, Portugal: Springer; 2003.
- [55] Deb K, Goel T. A hybrid multi-objective evolutionary approach to engineering shape design. In: Evolutionary multi-criterion optimization. Proceedings of the first international conference, EMO 2001, 7–9 March, 2001. Zurich, Switzerland: Springer; 2001.
- [56] Ishibuchi H, Yoshida T, Murata T. Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling. *IEEE Trans Evol Comput* 2003;7(2): 204–23.
- [57] Tan KC, Lee TH, Khor EF. Evolutionary algorithms with dynamic population size and local exploration for multiobjective optimization. *IEEE Trans Evol Comput* 2001;5(6):565–88.
- [58] Marseguerra M, Zio E, Podofilini L. Optimal reliability/availability of uncertain systems via multi-objective genetic algorithms. *IEEE Trans Reliab* 2004;53(3):424–34.
- [59] Martorell S, Villanueva JF, Carlos S, Nebot Y, Sanchez A, Pitarch JL, et al. RAMS+C informed decision-making with application to multi-objective optimization of technical specifications and maintenance using genetic algorithms. *Reliab Eng Syst Safety* 2005;87(1): 65–75.
- [60] Martorell S, Sanchez A, Carlos S, Serradell V. Alternatives and challenges in optimizing industrial safety using genetic algorithms. *Reliab Eng Syst Safety* 2004;86(1):25–38.
- [61] Sasaki M, Gen M. A method of fuzzy multi-objective nonlinear programming with GUB structure by hybrid genetic algorithm. *Int J Smart Eng Syst Des* 2003;5(4):281–8.
- [62] Yang J-E, Hwang M-J, Sung T-Y, Jin Y. Application of genetic algorithm for reliability allocation in nuclear power plants. *Reliab Eng Syst Safety* 1999;65(3):229–38.
- [63] Elegbede C, Adjallah K. Availability allocation to repairable systems with genetic algorithms: a multi-objective formulation. *Reliab Eng Syst Safety* 2003;82(3):319–30.
- [64] Deb K, Jain P, Gupta NK, Maji HK. Multiobjective placement of electronic components using evolutionary algorithms. *IEEE Trans Components Packaging Technol* 2004;27(3):480–92.
- [65] Kumar R, Parida PP, Gupta M. Topological design of communication networks using multiobjective genetic optimization. In: Proceedings of the 2002 world congress on computational intelligence—WCCT'02, 12–17 May, 2002. Honolulu, HI, USA: IEEE; 2002.



برای خرید فرمت ورد این ترجمه، بدون واتر مارک، اینجا کلیک نمایید.



این مقاله، از سری مقالات ترجمه شده رایگان سایت ترجمه فا میباشد که با فرمت PDF در اختیار شما عزیزان قرار گرفته است. در صورت تمایل میتوانید با کلیک بر روی دکمه های زیر از سایر مقالات نیز استفاده نمایید:

لیست مقالات ترجمه شده ✓

لیست مقالات ترجمه شده رایگان ✓

لیست جدیدترین مقالات انگلیسی ISI ✓

سایت ترجمه فا ؛ مرجع جدیدترین مقالات ترجمه شده از نشریات معتبر خارجی