

درس ۱۷:

الگوریتم فراابتکاری

الگوریتم ژنتیک

تهیه شده توسط گروه بهینه‌یاب



www.behinehyab.com

مقدمه

محاسبه راه حل بهینه یا *Optimal solution* برای اکثر مسایل بهینه‌سازی که در خیلی از زمینه‌های کاربردی و عملی مشاهده می‌گردند، کار دشوار و سختی است. در عمل، معمولاً به راه‌های خوب که از الگوریتم‌های **هیوریستیک** *Heuristic* یا **متاهیوریستیک** (همان **فراابتکاری**) *Metaheuristic* بدست می‌آید، اکتفا می‌گردد. متاهیوریستیک‌ها مجموعه‌ای از فنون بهینه‌سازی تقریبی *Approximate optimization techniques* را که عمدتاً در طول دو دهه گذشته شهرت پیدا کرده‌اند، در بر می‌گیرند. روش‌های فراابتکاری راه‌های قابل قبول در زمان معقول را برای مسایل پیچیده و سخت، در زمینه‌های مهندسی و علوم پایه ارائه می‌نمایند. برخلاف الگوریتم‌های بهینه‌سازی دقیق *Exact optimization algorithms*، فراابتکاری‌ها بهینه بودن جواب‌های بدست آمده را ضمانت نمی‌نمایند.

کلمه متاهیوریستیک از کلمه یونانی "*Heuriskein*" به معنای هنر کشف قواعد جدید برای حل مسایل گرفته شده است. پیشوند "متا" نیز از یک کلمه یونانی به معنای "متدولوژی" سطح بالا گرفته شده است. واژه "متاهیوریستیک" اولین بار توسط گلوور در سال ۱۹۸۶ ارائه گردید. روش جستجوی متاهیوریستیک را می‌توان به صورت متدولوژی‌های عمومی سطح بالایی که می‌توانند به عنوان یک استراتژی راهنما در طراحی هیوریستیک‌های اختصاصی برای حل مسایل بهینه‌سازی تخصصی به کار روند، تعریف کرد.

برخلاف روش‌های دقیق، متاهیوریستیک‌ها (فراابتکاری‌ها) برای مسایل با اندازه‌های بزرگ، کاربرد دشوار دارد و راه‌هایی راضی‌کننده‌ای در زمان معقولی ارائه می‌نمایند. در این الگوریتم‌ها، هیچ‌گونه ضمانتی برای یافتن جواب بهینه‌سراسری یا حدودی از آن وجود ندارد. متاهیوریستیک‌ها (فراابتکاری‌ها) در طول بیست سال گذشته شهرت زیادی پیدا کرده‌اند. کاربرد و استفاده از آن‌ها در خیلی از مسایل، کارایی و اثر بخشی آن‌ها را برای حل مسایل پیچیده و بزرگ نشان می‌دهد. فراابتکاری‌ها در خیلی از زمینه‌ها از قبیل موارد زیر کاربرد دارند:

✓ طراحی مهندسی، بهینه‌سازی توپولوژی، بهینه‌سازی مسایل الکترونیک، آئرودینامیک، دینامیک

سیالات، مخابرات، رباتیک

- ✓ یادگیری ماشین و کاوش داده‌ها
- ✓ مدل سازی سیستم‌ها، شبیه سازی و تحقیق در شیمی، فیزیک، بیولوژی، کنترل، سیگنال، و پردازش تصویر
- ✓ مسایل مسیره‌هی و برنامه‌ریزی، برنامه‌ریزی ربات، مسایل تولید و زمان بندی، حمل و نقل و لجستیک، مدیریت زنجیره تامین
- روش‌های مختلف الگوریتم‌های فراابتکاری یا متاهیوریستیک تا به حال پیشنهاد شده است که به صورت زیر هستند:

- ✓ بهینه‌سازی کلونی مورچگان *Ant colony optimization*
- ✓ بهینه‌سازی کلونی زنبوران *Bee colony*
- ✓ الگوریتم‌های ترتیبی *cultural algorithms*
- ✓ الگوریتم‌های با هم تکاملی *Co-evolutionary algorithms*
- ✓ الگوریتم ژنتیک *Genetic algorithm*
- ✓ جستجوی محلی تکراری *Iterated local search*
- ✓ بهینه‌سازی توده ذرات *particle swarm intelligent*
- ✓ انجماد تدریجی *Simulated Annealing*
- ✓ جستجوی ممنوع *Taboo search*
- ✓ جستجوی همسایگی متغیر *Variable neighbor search*

در طراحی یک فراابتکاری، دو معیار متناقض شامل **کاوش** *Exploration* در فضای جستجو (گوناگونی و تنوع) و **تبعیت** (*Exploitation*) از بهترین راه حل‌های پیدا شده، باید در نظر گرفته شوند. در کاوش در ناحیه‌های جستجو نشده بررسی صورت می‌گیرد. در تبعیت، در ناحیه‌های امید بخش که تا به حال در آن ناحیه یک جواب خوب پیدا شده است بررسی بیشتر صورت می‌گیرد. در صورتیکه به رفتار **کاوش** اهمیت بیشتری داده شود، الگوریتم رفتار تصادفی بیشتری خواهد داشت و به سمت رفتار تصادفی میل می‌کند و در

صورتی که به رفتار **تبعیت** توجه بیشتری شود، الگوریتم از رفتار تصادفی فاصله می‌گیرد و جستجو تنها در محدوده راه‌های خوب به جستجو می‌پردازد که می‌تواند به راه‌های محلی منجر شود.

معیارهای طبقه‌بندی زیادی ممکن است برای طبقه‌بندی فراابتکاری‌ها استفاده شود که در زیر به بعضی از آن‌ها اشاره می‌شود:

الهام گرفته از طبیعت در مقابل عدم الهام از طبیعت

خیلی از الگوریتم‌های فراابتکاری از فرایندهای طبیعی الهام گرفته شده‌اند: از قبیل الگوریتم‌های اجتماع مورچگان و زنبور عسل که از هوش توده‌ای از گونه‌های مختلف مورچگان و زنبوران استفاده می‌کنند. برخی از الگوریتم‌های این چنین نبوده و از یک رفتار منطقی پیروی می‌کنند که می‌توان به الگوریتم جستجوی ممنوع نام برد.

نحوه استفاده از حافظه

بعضی از الگوریتم‌های فراابتکاری از قبیل انجماد تدریجی یا گرم و سرد کردن شبیه‌سازی شده بدون حافظه هستند و حرکت‌های قبلی را در جایی ذخیره نمی‌کنند. در مقابل الگوریتم جستجوی ممنوع از یک حافظه که بعضی از اطلاعات را در طول جستجو بدست می‌آورد، استفاده می‌کند.

قطعی در مقابل احتمالی

یک الگوریتم قطعی، یک مسئله بهینه‌سازی را از طریق تصمیم‌گیری قطعی حل می‌نماید (برای مثال الگوریتم جستجوی محلی و جستجوی ممنوع). در الگوریتم‌های فراابتکاری احتمالی، بعضی از قواعد احتمالی در فرایند جستجو مورد استفاده قرار می‌گیرد که می‌توان به الگوریتم انجماد تدریجی و الگوریتم ژنتیک اشاره کرد. در الگوریتم‌های قطعی، با داشتن یک راه حل اولیه و اجراهای متفاوت، تنها یک جواب نهایی بدست می‌آید و در حالی که در الگوریتم‌های تصادفی، با داشتن یک راه حل اولیه و اجراهای متفاوت، ممکن است که جواب‌های نهایی متفاوتی بدست آید.

تکراری در مقابل حریصانه

در الگوریتم‌های تکراری، الگوریتم با یک راه حل کامل شروع شده و در هر تکرار راه حل یا راه حل‌ها تغییر پیدا می‌کنند. در الگوریتم‌های حریصانه یک راه حل کامل در اختیار نبوده بلکه با یک راه حل ساخته نشده شروع شده و در هر مرحله، یک متغیر تصمیم از مسئله یک قسمت از راه حل را می‌سازد. اغلب الگوریتم‌های فراابتکاری، الگوریتم‌های تکراری هستند.

در ادامه **الگوریتم ژنتیک** که الگوریتمی است که از طبیعت الهام گرفته است و در رده الگوریتم‌های احتمالی، و تکراری با حافظه قرار می‌گیرد صحبت می‌شود.

الگوریتم ژنتیک

بر اساس تئوری انتخاب طبیعی، گیاهان و موجودات زنده ای که در حال حاضر وجود دارند، نتیجه میلیون‌ها سال تطابق با تقاضای‌های محیط می‌باشند. در هر زمانی، تعدادی از جانداران مختلف، ممکن است که با هم دیگر در یک اکوسیستم زندگی نمایند و برای دستیابی به منابع مشترک با هم دیگر به رقابت بپردازند. موجوداتی که توانایی بیشتری در دستیابی به منابع و تولید مثل داشته‌اند، در طبیعت نیز بیشتر مشاهده می‌شوند. موجوداتی که توانایی کمتری داشته‌اند، به دلایل مختلف، به مرور تعداد آن‌ها کم و کمتر شده و حتی به نابودی کشیده شده‌اند. در این حالت، گفته می‌شود که گونه‌های اول نسبت به گونه‌های بعدی شایسته‌تر می‌باشند و مشخصه‌هایی که موجب شده است تا بعضی گونه‌های شایسته‌تر باشند مشخصه‌های ارجح می‌نامند. در طول زمان، گفته می‌شود که کل جمعیت اکوسیستم دچار تکاملی شده است به طور متوسط، شامل موجوداتی شده است که شایسته‌تر از موجودات قبلی آن می‌باشند، زیرا این موجودات مشخصاتی داشته‌اند که منجر به بقای بیشتر آن‌ها شده است.

تکنیک **محاسبات تکاملی** یا **Evolutionary computation** یا **EC** قواعد تکامل را در الگوریتم‌هایی

برای جستجوی راه حل‌های بهینه مسایل به کار می‌گیرد. در واقع، تکنیک محاسبات تکاملی، توسعه علوم تکامل بیولوژیکی به حوزه علوم محاسباتی و کامپیوتر بوده است. در یک الگوریتم جستجو، تعدادی از راه حل‌های ممکن از یک مسئله در اختیار بوده و هدف یافتن بهترین راه حل ممکن در زمان محدود می‌باشد.

تفاوت اصلی الگوریتم‌های تکاملی نسبت به الگوریتم‌های دیگر این است که این الگوریتم‌ها مبتنی بر جمعیت عمل می‌نمایند. در این الگوریتم‌ها، معمولا یک جمعیت اولیه ایجاد و سپس تکامل داده می‌شود.

الگوریتم ژنتیک یا **Genetic Algorithm** مشهورترین تکنیک در تحقیقات الگوریتم‌های تکاملی است.

این الگوریتم یک تکنیک جستجو را برای یافتن راه حل‌های نزدیک به بهینه در زمان قابل قبول برای مسایل بهینه‌سازی آرایه می‌نماید.

الگوریتم ژنتیک با یک جمعیت اولیه از راه حل‌ها شروع می‌شود. هر راه حل از طریق یک کروموزوم نمایش داده می‌شود و تمامی راه حل‌های ممکن باید با استفاده از یک سیستم کدگذاری، تبدیل به کد شوند. سپس مجموعه‌ای از اپراتورهای تولید مثل، باید تعیین گردد. اپراتورهای تولید مثل، مستقیما روی کروموزوم‌ها عمل نموده و سپس کروموزوم‌ها تحت اپراتور جهش و ترکیب قرار می‌گیرند. طراحی ساختار کدگذاری و اپراتورها باید با حوصله و دقت فراوان انجام گردد. این طراحی، تاثیر بسیار زیادی روی عملکرد الگوریتم ژنتیک خواهد داشت.

رویه انتخاب برای رقابت افراد در داخل جمعیت به کار می‌رود که براساس یک تابع شایستگی عمل می‌نماید. برای هر کروموزوم، یک مقدار مرتبط با شایستگی راه حل که نمایش می‌دهد، وجود دارد. الگوریتم ژنتیک به دنبال حداکثر کردن مقدار تابع شایستگی است. البته در صورتی که تابع هدف به صورت حداقل سازی یک تابع هدف باشد، به هنگام کردن الگوریتم برای حداقل سازی کاری ساده‌ای است. هر تابع هزینه به راحتی قابل تبدیل به یک تابع شایستگی است مثلا با معکوس کردن تابع هزینه، می‌توانی یک تابع شایستگی متناسب آن ساخت.

بعد از این که مراحل تولید مثل و تابع برازندگی تعریف شد، یک الگوریتم ژنتیک براساس یک ساختار مشابه و پایه طراحی می‌گردد. این ساختار ساده با تولید یک جمعیت اولیه از کروموزوم‌ها شروع می‌گردد. معمولا، جمعیت اولیه به صورت تصادفی تولید می‌گردد.

مراحل الگوریتم ژنتیک

الگوریتم ژنتیک یک رویه تکراری را به منظور تکامل جمعیت انجام می دهد که در آن فعالیت‌های زیر انجام می شود.

انتخاب

اولین قدم شامل انتخاب افرادی از جمعیت برای تولید مثل است. این انتخاب به صورت تصادفی، با استفاده از یک احتمال متناسب با تابع برازندگی افراد انجام می گردد. بنابراین، همواره بهترین فرد شانس بیشتری برای انتخاب نسبت به فرد ضعیف تر دارد.

تولید مثل

در قدم دوم، عملگر ترکیب و جهش روی افراد انتخاب شده به کار گرفته شده و کروموزوم‌های جدید تولید می گردند. در این مرحله، فرزندان جدید تولید می شوند.

ارزیابی

در این مرحله، میزان برازندگی فرزندان جدید تولید شده تعیین می شود.

جابه جایی

در این مرحله، افرادی از جمعیت قبلی حذف و با افراد جدیدی که به تازگی تولید شده جایگزین می شوند.

برای مطالعه ادامه این درس، حل تمرین و پیاده سازی الگوریتم به زبان ویژال بیسیک، جزوه این درس را تهیه کنید.

برای دریافت بسته‌های آموزشی گروه **بهینه‌یاب** به وب سایت ما به نشانی

www.behinehyab.com مراجعه کنید.

در صورت هر گونه سوال از طریق ایمیل به نشانی behinehyab@gmail.com و یا

بخش تماس با ما وب سایت گروه **بهینه‌یاب** با ما در تماس باشید.

با تشکر از توجه شما

گروه آموزشی **بهینه‌یاب**