الگوریتم ژنتیک[1]
آنچه کامپیوتر می‌تواند از تکامل بیاموزد

از: چارلز والبریج
ترجمه: محمود البرزی
عضو هیئت علمی دانشگاه نفت

چکیده

مقاله زیر ترجمه مقاله‌ای است تحت همین عنوان به قلم Charles T. walbridge از مجله Technology Review. صرفنظر از تفکر نویسنده در مورد چگونگی آغاز حیات و تکامل آن، به دلیل معرفی یک روش نو در برنامه‌ریزی با کامپیوتر، برای چاپ انتخاب شد. این نوع برخورد با مسئله نه تنها برای متخصصین کامپیوتر بلکه دست‌اندرکاران سایر رشته‌ها که با هوش مصنوعی ارتباط دارند می‌تواند مفید باشد. به هر صورت چاپ آن به هیچ عنوان به معنی تأیید نگرش اعتقادی نویسنده نسبت به چگونگی آغاز حیات و تکامل آن نیست، بلکه منظور معرفی یک الگوریتم است.

زیبایی الگوریتم معرفی شده (ژنتیک) در این است که با داشتن ضوابط قیاسی، راه‌حل‌های مختلف حل مسئله را از طریق آزمون و خطا می‌پیماید، رشته‌های پیروز را با یکدیگر جفت‌گیری می‌کند و هرازگاهی عمل جهش را انجام می‌دهد. این فرایند بارها تکرار

1 - Genetic Algorithm, Charles T. Walbridge, Technology Review, Jan. 1989.

می‌گردد تا نهایتاً، برنامه کار مطلوبی ایجاد شود. سرعت الگوریتم ژنتیک در پیدا کردن پاسخ مسائل پیچیده، به آن قابلیت فراگیری داده است.

❐ مقدمه :

چند میلیون سال پیش، پس از آنکه زمین آنقدر سرد شد که آب به حالت مایع درآید، اولین مولکول‌های حیات شروع به تکثیر کردند. در آن زمان مولکول‌ها عملاً فاقد هرگونه اطلاعات ژنتیکی بودند. با این وجود فرآیندی آغاز شده بود که به تنوع شگرف اطلاعات ژنتیکی منتهی می‌گشت، تنوعی که نهایتاً مشخصات کلیه موجودات کنونی این کره را تعیین می‌کرد.

نیروی محرکه این فرآیند یک الگوریتم است، یعنی مجموعه‌ای از دستورات که تکرار آنها به حل مسائل می‌انجامد؛ مانند دستوراتی که در یک عمل تقسیم دنبال می‌شود. مسأله‌ای که الگوریتم تکامل با آن روبرو است تولید گونه‌های متفاوت حیات است.

همین الگوریتم می‌تواند انواع مسائل دیگر را نیز حل کند. شیوه‌ای از برنامه‌ریزی کامپیوتر که بر پایه این الگوریتم بنا شده است هم‌اکنون با سایر سیستم‌های ماهر[1] برابری می‌کند. مثلاً در حال حاضر تحقیقاتی در دست است که کاربرد این الگوریتم را در حل مسائلی چون نحوه کنترل تلمبه‌خانه‌های نفت، تنظیم تصاویر اشعه ایکس و طراحی تراشه‌های کامپیوتر VLSI بررسی می‌کند.

برای مثال در طراحی یک تراشه VLSI[2] ابتدا روشی را برای نمایش ساختار تراشه بکار می‌بریم که از ردیفی از ارقام به شکل کروموزوم[3] استفاده می‌کند. بخش‌های این کروموزوم یعنی ژن‌ها[4] در واقع اجزاء مختلف تراشه می‌باشند. فرآیند طراحی تراشه را مانند فرآیند تکامل با انتخاب مجموعه‌ای تصادفی از کروموزوم‌ها آغاز می‌کنیم. هر کروموزم مبین طراحی کامل ولی نامنظم از مشخصات الکتریکی و

---

1 - Expert system.

2 - VLSI (very - large - scale integrated computer chips).

3 - Chromosome.

4 - Gene.

داده‌پردازی یک تراشه است. بیشتر طرح‌های تصادفی مذکور از نظر مهندسی بسیار ضعیف‌اند. در این هنگام از روتین ریاضی بسیار ساده‌ای استفاده کرده و تعداد قلیلی از بهترین طرح‌ها را از سایر طرح‌ها جدا می‌کنیم. آنگاه به همان صورت که ارگانیسم‌های زنده در طبیعت تولید مثل می‌کنند کروموزم‌های برگزیده شده را با یکدیگر جفت کرده و از ترکیب آنها کروموزم‌های جدیدی را تولید می‌کنیم، به طوری که این کروموزم‌ها خصوصیات خود را از والدین خود کسب می‌کنند. از نسل دوم مجدداً بهترین را جدا کرده و بقیه را حذف می‌کنیم. گاهی اوقات نیز تعدادی از کروموزم‌ها را با تغییر تصادفی ژن‌های آنها جهش می‌دهیم. فرآیند جفت‌گیری[1]، جهش[2] و انتخاب[3] تا حصول به طرح مناسب تراشه مورد نظر ادامه می‌یابد.

رشته‌ای در علوم محاسباتی که بر پایه الگوریتم ژنتیک بنا شده است، بیست‌وپنج سال پیش توسط جان هلند[4] دانشمند علوم کامپیوتر دانشگاه میشیگان پایه‌ریزی شد. او ابتدا به موضوعی علاقمند بود که خود آن را اتومات سلولی[5] می‌نامد. اتومات سلولی مدل بسیار ساده‌ای است که می‌توان آن را به راحتی حتی بر صفحه کاغذ شطرنجی نشان داد. خانه‌های شطرنجی بنا بر قاعده بسیار ساده‌ای اشغال یا تخلیه می‌گردند. بدین صورت شکلی که حاصل می‌شود در مسیرهایی که از پیش قابل پیش‌بینی نیست رشد می‌کند، حرکت می‌کند و یا محو می‌گردد، و در صورتی که مدل قدری پیچیده‌تر باشد حتی تولید مثل می‌کند. بدین صورت اتومات سلولی به نوعی ظهور حیات را در دریاهای اولیه نمایش می‌دهد. هلند به تدریج از رشد و تولید مثل برنامه‌ها به تکامل برنامه‌ها، یعنی محور اصلی تحقیقات جاری الگوریتم ژنتیک، گرایش یافت. امروزه گرچه الگوریتم ژنتیک هنوز در تسلط دانشمندان کامپیوتر است، لیکن به تدریج افراد دیگر را از حوزه‌های متفاوت چون مهندسی، ریاضیات و بیومتری به خود جلب کرده است.

---

1 - Mating.

2 - Mutating.

3 - Selecting.

4 - John Holland.

5 - Cellular automatons.

در بسیاری از مسائل کاربردی رشته‌های فوق از سیستم‌های ماهر استفاده می‌شود.

سیستم ماهر نوعی از سیستم‌های هوش مصنوعی[1] می‌باشد که سعی می‌کند بخش‌هایی از مهارت انسان‌ها را که به دقت تعریف شده است کامپیوتری کند. یک سیستم ماهر به پایه استنتاج‌های منطقی متعدد بنا می‌گردد و شامل قواعد پیچیده‌ای است که از شیوه تفکر انسان‌های ماهر مثلاً یک پزشک، یک زمین‌شناس و یا یک مکانیک استخراج می‌گردد. استخراج اطلاعات از اندیشه افراد ماهر و تغییر صورت آنها به طریقی که بتواند توسط یک کامپیوتر دنبال شود مشکل عمده‌ای می‌باشد.

در الگوریتم ژنتیک برنامه‌نویس‌ها به مهارت در رشته‌های خاص نیاز ندارند. آنها صرفاً به شیوه‌ای نیاز دارند که بتواند راه‌حل‌های متفاوت را سریعاً ارزیابی کند و بهترین راه حل را با توجه به ضوابط مورد نظر انتخاب کند.

روشن است که ارزیابی کارآیی یک طرح معین از تراشه، از طراحی آن تراشه، با توجه به حالات نامحدود طراحی، بسیار ساده‌تر است. زیبایی الگوریتم ژنتیک در این است که با داشتن ضوابط قیاس، راه‌های مختلف را از طریق آزمون و خطا می‌پیماید، بهترین تخمین‌ها را حفظ می‌کند و نهایتاً بر پایه آنها جواب مناسب را می‌یابد.

❐ یک بازی نمونه :

اگر نسبت به کارآیی الگوریتم ژنتیک در حل مسائل مشکوک می‌باشید به عنوان مثال بازی "کشتی‌های جنگی[2]" را در نظر بگیرید. در این بازی هر طرف دارای تعدادی کشتی جنگی است که موقعیت مکانی آنها را در نقشه از طرف دیگر پنهان می‌دارد. در هر نوبت یکی از طرف‌های بازی محل برخورد گلوله توپ خود را به طرف مقابل اعلام می‌کند و طرف دیگر وقوع تصادم و یا عدم تصادم گلوله توپ مذکور را اعلام می‌دارد. این خبر همانند انتخاب طبیعی باعث می‌شود که هر طرف بهترین موقعیت‌ها را حفظ کند و آنها را بهبود دهد تا بتواند کشتی‌های طرف مقابل را نابود کند.

---

1 - Anticficial Intelligence.

2 - Battleship.

بازی دیگری به نام "بردار[1]" طرح شده است که نحوه عمل الگوریتم ژنتیک را با جزئیات بیشتر نشان می‌دهد. این بازی به دو طرف نیاز دارد. ابتدا رقیب شما رشته‌ای ۶ رقمی متشکل از ارقام ۰ یا ۱ را در نظر می‌گیرد. شما عددی را حدس می‌زنید و رقیب شما امتیاز شما را بر حسب ارقامی که صحیح گفته‌اید اعلام می‌دارد. هدف بازی این است که در کمترین دفعات کلیه ۶ رقم را صحیح اعلام دارید.

بازی را با نوشتن چهار رشته تصادفی زیر آغاز می‌کنیم. فرض کنید امتیازات بدست آمده از طریق زیر باشد.

	رشته	امتیاز
(A)	۰۱۰۱۰۱	۱
(B)	۱۱۱۱۰۱	۱
(C)	۰۱۱۰۱۱	۳
(D)	۱۰۱۱۰۰	۳

حال دو رشته اول و دوم (A) و (B) را که امتیاز کمتری دارند حذف می‌کنیم و دو رشته دیگر یعنی (C) و (D) را مانند انتخاب طبیعی تحت عملیات ژنتیکی قرار می‌دهیم. اول آنها را تکثیر می‌کنیم. برای سهولت هر کروموزم را دوباره تکرار می‌کنیم و چهار کروموزم بدست آمده را به صورت دو دسته دو تایی که والدین کروموزم‌های بعدی خواهند بود دسته‌بندی می‌کنیم.

	تکثیر[2]	
(C)	۰۱۱۰۱۱	دسته اول
(D)	۱۰۱۱۰۰	
(C)	۰۱۱۰۱۱	دسته دوم
(D)	۱۰۱۱۰۰	

1 - Vector.

2 - Multiplication.

سپس کروموزم‌های فوق را به صورت ضربدر [1] جفت‌گیری می‌کنیم. دو کروموزم دسته اول را در محل تصادفی قطع می‌کنیم. آنگاه قسمت اول کروموزم (C) را به قسمت دوم کروموزم (D) و قسمت دوم کروموزم (C) را به قسمت اول کروموزم (D) وصل می‌کنیم. همین عمل را در مورد دسته دوم در محل دیگری تکرار می‌کنیم.

فرض کنید هشت کروموزم با امتیازهای زیر بدست می‌آید.

تکثیر و جفت‌گیری		نتیجه		امتیاز
۰۱:۱۰۱۱	(C)	۰۱:۱۱۰۰	(E)	۳
۱۰:۱۱۰۰	(D)	۱۰:۱۰۱۱	(F)	۳
۰۱۱۰:۱۱	(C)	۰۱۱۰:۰۰	(G)	۴
۱۰۱۱:۰۰	(D)	۱۰۱۱:۱۱	(H)	۳

حال دو کروموزم جدید (F) و (G) را با بالاترین امتیاز انتخاب کرده و بقیه را حذف می‌کنیم. عملیات تکثیر و جفت‌گیری را مجدداً تکرار می‌کنیم.

تکثیر و جفت‌گیری		نتیجه		امتیاز
۱:۰۱۰۱۱	(F)	۱:۱۱۰۰۰	(I)	۳
۰:۱۱۰۰۰	(G)	۰:۰۱۰۱۱	(J)	۵
۱۰۱:۰۱۱	(F)	۱۰۱:۰۰۰	(K)	۴
۰۱۱:۰۰۰	(G)	۰۱۱:۰۱۱	(L)	۴

باکسب امتیاز ۵ تقریباً به جواب رسیده‌ایم. مجدداً اعمال قبل را انجام می‌دهیم.

1 - Crossover.

جفت‌گیری		نتیجه		امتیاز
۰۰۱۰:۱۱	(J)	۰۰۱۰:۰۰	(M)	۵
۱۰۱۰:۰۰	(K)	۱۰۱۰۰۱۱	(N)	۴
۰۰۱۰۱:۱	(J)	۰۰۱۰۱:۰	(O)	۶
۱۰۱۰۰:۰	(K)	۱۰۱۰۰:۰	(P)	۳

مقصود حاصل شد و جواب کامل ۶ بدست آمد. الگوریتم ژنتیک با ۱۶ سؤال جواب پنهان را یافت. لزومی نداشت امتیاز کلیه ۶۴ حالت مختلف مسئله را در نظر بگیریم. تعداد سؤالات حتی از ۳۲ حالت که قاعدتاً در جستجوی تصادفی مسئله موردنیاز می‌بود نیز کمتر است. در بازی "بردار" و یا "کشتی‌های جنگی"، اطلاعات بدست آمده در هر مرحله (امتیاز بدست آمده و یا تعداد تصادم‌های مؤثر) به اطلاعات قبلی اضافه می‌شود و بدین صورت نیازی به جستجوی کامل فضای جواب‌های ممکن نخواهد بود.

در واقع تنها ذکر این نکته که به این طریق اطلاع حاصل، به اطلاعات قبلی اضافه می‌شود حق مطلب را ادا نمی‌کند، چه این افزایش بیشتر بطریق هندسی است، به همان صورت که همگی در بازی بیست سؤالی با آن آشنا هستیم. در این بازی که خود نمونه‌ای از الگوریتم ژنتیک است هدف رسیدن به جواب مسئله با پرسش‌های بله یا خیر است. فرض کنید سؤال اول همانطور که اغلب معمول است این باشد که "جاندار" است؟ هر جواب بلافاصله به حذف دسته عظیمی از امکانات منتهی می‌شود. هر چه پرسش‌ها امکانات بیشتری را حذف کند رسیدن به جواب نهایی سریع‌تر می‌شود. اگر افزایش اطلاعات تنها به طریق حسابی بود امکان نداشت جواب صحیح را از میان حالات بی‌شمار، تنها با ۲۰ سؤال، تشخیص داد. این قدرت به علت "توازی درونی" الگوریتم ژنتیک است. وقتی والدین ضعیف را حذف می‌کنیم به طور خودکار کلیه زاد و ولدهای احتمالی آنها و زاد و ولدهای زاد و ولدهای آنها را حذف کرده‌ایم. بدین طریق در هر تکرار، اهداف بسیاری هم زمان دنبال می‌شود و نتایج حال و آینده، تماماً به موازات

یکدیگر ارزیابی می‌گردد.

به همین طریق در تکامل، وقتی اعضاء ضعیف‌تر یک نسل، قبل از رسیدن به سن بلوغ می‌میرند از خود نسل‌های دیگر بجا نمی‌گذارند. ترکیب‌های ضعیف ژن‌ها حذف می‌شوند و انتخاب طبیعی تنها بر دسته کوچکتری از ژن‌های مناسب عمل می‌کند.

این قدرت "توازی ذاتی" است که الگوریتم ژنتیک را تا به این حد مؤثر کرده و به آن توان تقریب سریع به جواب‌های مناسب داده است.

❑ نقش جهش (موتاسیون)[1] :

در بازی بردار یکی از عملیات ژنتیکی یعنی جهش را کنار گذاشته بودیم. جهش در الگوریتم ژنتیک به معنی تغییر تصادفی در یکی از ارقام رشته است، به همان صورت که جهش در طبیعت به معنی تغییری تصادفی در یکی از ژن‌ها می‌باشد. گاه برای احتراز از توقف الگوریتم در نقاط زیر بهینه، به جهش نیاز است. فرض کنید در بازی بردار چهار رشته ارقام اولیه را با رقم ۰ در ستون سوم انتخاب کرده بودیم، هیچ تعداد تکثیر و جفت‌گیری نمی‌توانست هرگز به عدد ۱ یعنی جواب اصلی در آن ستون دست یابد، در این صورت ناگزیر از بکارگیری جهش بودیم.

برای احتراز از چنین حالات بن‌بست، صورت‌های کامپیوتری الگوریتم ژنتیک معمولاً به طریقی برنامه‌ریزی می‌شوند که با احتمال کم مثلاً یک در هزار تعمداً اشتباهی در عملیات تکثیر آنها وارد شود. مثلاً یکی از ارقام از ۰ به ۱ و یا به عکس تبدیل می‌شود و بدین صورت تکثیر ژنتیکی به طور کامل صورت نمی‌پذیرد. در طبیعت نیز به طور مشابه در عملیات تکثیر جهش‌های ژنتیکی با احتمال کم مشاهده می‌شود.

گرچه جهش نقش مفیدی دارد لیکن نباید بیش از حد در مورد آن اغراق شود. هنوز بسیاری از بیولژیست‌ها بر این باورند که تکامل، صرفاً از طریق تغییرات ناگهانی در ماده ژنتیکی صورت می‌پذیرد. جهش موجب تولید ارگانیسم‌های کاملاً متفاوت با اجداد خود می‌شود. به عبارت دیگر ساده‌ترین اشکال تک‌سلولی از طریق جهش‌های

---

1 - Motation.

تصادفی به انسان تکامل یافته‌اند. این روند به طور باور نکردنی کند است.

یک محاسبه ساده علت مسئله را نشان می‌دهد. DNA از چهار اسید نوکلئوتید تشکیل یافته و در واقع یک رمز چهاربیتی است. ۴۸ کروموزمی که یک انسان را تعریف می‌کنند متشکل از ۳ بیلیون از بیت‌های مذکور است. در نتیجه اگر طبیعت تنها از طریق جهش به تولید انسان دست یافته بود، می‌بایستی ۴ به توان ۳ بیلیون ترکیب مختلف را آزمون می‌کرد. چون وقوع جهش امری نسبتاً نادر است، فرآیند تکامل به زمانی بسیار فراتر از عمر کنونی جهان نیاز داشت. عمر زمین که جای خود دارد! و اگر میزان وقوع جهش به اندازه‌ای بالا بود که می‌توانست تنوع کنونی انواع حیات را در زمان کوتاه ظهور حیات بر کره زمین موجب شود، آنگاه ارگانیسم‌های حاصله توان تولید مثل مؤثر کنونی را نمی‌داشتند.

بنابراین مولد اصلی تکامل، جهش نیست، بلکه ترکیب‌های دوباره ژنتیکی است. این هم شامل تولیدمثل جنسی است که از ترکیب کروموزم‌های کامل والدین صورت می‌گیرد و هم شامل جفت‌گیری کروموزم‌ها می‌باشد که ژن‌های کروموزم‌های مختلف را جابجا و دوباره ترکیب می‌کند. در طبیعت نیز مانند عملیات محاسباتی، آنچه فرآیند تکامل را به پیش می‌برد ترکیب بهترین جلوه‌های جمعیت موجود و حذف (از طریق توازی درونی الگوریتم) نتیجه‌هایی است که از ترکیب جلوه‌های نامطلوب به وجود می‌آید. موضوع جهش توجه بسیاری را جلب می‌کند زیرا امری مشهود است و اغلب نتایج شگفت‌آوری دارد، ولی نقش اصلی آن تنها برای شروع مجدد فرآیندی است که به بن‌بست افتاده است.

❑ بکارگیری تکامل :

به همان طریق که تکامل از الگوریتم ژنتیک در بهبود انواع موجودات بهره می‌گیرد، ما نیز می‌توانیم در حل مسائل علمی - مهندسی آن را بکار گیریم. شاید کمتر از ۵۰۰ پژوهشگر در این زمینه مشغول تحقیق‌اند لیکن تاکنون موارد استفاده بسیار گسترده‌ای یافته‌اند. یک نمونه کاربردی آن طراحی خرپا است. خرپا ساختاری برای نگهداری پل، سقف و یا دیگر ساختمان‌های مهندسی است. این برنامه که توسط

دیوید گلدبرگ[1] و منوهر صمدانی[2] در دانشگاه آلاباما نوشته شده است از الگوریتم ژنتیک برای تولید طرحی که بتواند با کمترین مواد، توان باربری لازم را داشته باشد استفاده می‌کند.

در این برنامه برای مثال یک خرپای ۱۰ شاخه‌ای توسط یک رشته عدد ده رقمی نشان داده می‌شود که هر رقم نمایانگر قطر یکی از شاخه‌های خرپا است. برنامه با تولید تصادفی صدها خرپا با قطر شاخه‌های مختلف شروع می‌شود. سپس یک برنامه ساده کمک طراحی کامپیوتری، هر رشته را از نظر وزن و توان باربری ارزیابی می‌کند. اگر خرپایی که با رشته معین نشان داده می‌شود نتواند بار مورد نظر را تحمل کند آن رشته نمره صفر خواهد گرفت. اگر خرپا توانست بار لازم را تحمل کند آنگاه هر چه وزن آن کمتر باشد نمره بیشتر می‌گیرد.

البته رشته‌های اولیه معمولاً نمره کم می‌گیرند، بعضی از خرپاها شدیداً ضعیف‌اند و بقیه بسیار سنگین هستند. لیکن به تدریج که الگوریتم ژنتیک رشته‌های بهتر را حفظ و آنها را ترکیب می‌کند، سطح نمرات افزایش می‌یابد. نهایتاً سطح نمرات به حالت تعادل می‌رسد که نشان می‌دهد بهبود بیشتر در طراحی به دست آمده محتمل نمی‌باشد. توازی درونی باعث می‌شود که الگوریتم ژنتیک در زمان معقول به جوابی مطلوب نزدیک شود که نهایتاً خرپای سبک و خوبی خواهد بود. سایر برنامه‌های طراحی خرپا نیز به نتایج مشابهی خواهند رسید ولی آنها پیچیدگی بیشتری دارند و به تجارب مهندسی بیشتری نیازمندند.

از الگوریتم ژنتیک می‌توان برای طراحی برنامه‌های پیچیده عملیاتی نیز استفاده کرد. میکائیل هیلیارد[3] و گونار لیپینس[4] که در زمینه سیستم‌های ادراکی در لابراتوار اوک‌ریج‌ناشنال[5] تحقیق می‌کنند به مسئله برنامه‌ریزی کار در یک کارخانه پرداخته‌اند. در یک کارگاه نمونه، از ماشین‌های مته، تراش و نورد و سایر ابزار برای

---

1 - David Goldberg.

2 - Manohar Samatani.

* (مترجم در مورد املاء این نام با توجه به صورت انگلیسی آن مطمئن نمی‌باشد.)

3 - Michail Hilliard.

4 - Gunar Liepins.

5 - Ook Rich National Laboratory.

ساخت قطعات فلزی جهت مشتریان صنعتی استفاده می‌شود. محصولات یا سفارشات مسیرهای متفاوتی را در میان کارگاه طی می‌کنند و اغلب دو یا تعداد بیشتری محصول در آن واحد به ماشین مشترکی نیاز دارند که مشکل عمده برنامه‌ریزی کار است. فردی مسئول باید هر روز و یا هر گاه دستگاهی از کار افتاد برنامه کار جدیدی را طراحی کند.

برای اینکه یک برنامه کار بتواند با رشته‌ای از ارقام مشخص شود باید محدودیت‌های معینی در نظر گرفته شود به صورتی که برنامه‌ها تماماً قابل قبول باشند. برای مثال هیچ دستگاهی در آن واحد نمی‌تواند برای دو کار مورد استفاده قرار گیرد. بدین طریق رشته‌های ارقام متعدد که هر کدام نماینده یک برنامه کار می‌باشد تولید می‌شود و برنامه‌هایی که بالاترین ارزش تولید در ساعت را برای کارگاه داشته باشند انتخاب می‌شوند.

رشته‌های پیروز با یکدیگر جفت‌گیری می‌شوند و هر از گاهی عمل جهش انجام می‌شود. این فرآیند بارها تکرار می‌گردد تا نهایتاً برنامه کار مطلوبی ایجاد شود. از مرحله برنامه‌ریزی کار کارخانه‌ها تا سایر انواع برنامه‌ریزی‌ها مانند کمک به دانشگاه‌ها برای تخصیص حداکثر دانشجویان به کلاس‌های موجود، قدم کوچکی بیش نیست.

از جمله کاربردهای نویدبخش دیگر تشخیص الگوها می‌باشد. یکی از اهداف تحقیقات هوش ماشینی، ساخت سیستم‌هایی است که بتواند یک الگو را از زمینه متن تشخیص دهد چه مسئله تشخیص چهره‌ای از میان جمعیت باشد و یا درک زبان محاوره‌ای. پژوهشگران کشف کرده‌اند که الگوریتم ژنتیک از اکثر فنون کامپیوتری دیگر در ردیف کردن تصاویر درهم کارآیی بیشتری دارد. مثلاً در دانشگاه وندربیلت[1] مایکل فیتزپتریک[2] و همکاران او از الگوریتم ژنتیک برای مقایسه تصاویر اشعه ایکس قبل و بعد از عمل استفاده می‌کنند. امکان دارد بتوان گرفتگی رگ‌ها را با تزریق رنگ مخصوص و مقایسه تصاویر اشعه ایکس قبل و بعد از تزریق رنگ بررسی کرد.

---

1 - Vanderbilt.

2 - Michail Fitzpatrick.

لیکن مقایسه دقیق مشکل است زیرا رگ‌ها در تصویر دوم هرگز در محل قبلی خود در تصویر اول نخواهند بود. ممکن است تغییر محل داده و یا قدری تاب خورده باشند.

و اما پیچیدگی تصویر دوم را می‌توان توسط معادله تبدیل از بین برد. با معادله تبدیل هر نقطه دیجیتال شده را درست به اندازه‌ای تغییر محل می‌دهند تا با نقطه متناظر خود در تصویر مرجع منطبق شود. مسئله یافتن معادله تبدیل مناسب برای تصاویر معین است.

گروه وندربیلت دریافتند که با استفاده از برنامه‌ای که بر اساس الگوریتم ژنتیک نوشته شده است می‌توان به سرعت به معادله تبدیل مناسب دست یافت. برنامه از تعداد زیادی معادلات ممکن شروع می‌کند و با استفاده از اپراتورهای ژنتیکی، جستجو را تا دست‌یابی به معادله‌ای ادامه می‌دهد که بتواند تصویر دوم را بر تصویر اول به خوبی منطبق کند.

سرعت الگوریتم ژنتیک در یافتن جواب مسئله به آن قدرت فراگیری می‌دهد، به صورتی که بتواند به شیوه‌ای با محیط سازگار گردد که از عهده سیستم‌های ماهر برنمی‌آید. برای مثال دیوید گلدبرگ از دانشگاه آلاباما برنامه‌ای را برای تطبیق تقاضاهای بسیار متغیر تلمبه خانه‌های نفت طراحی کرده است.

اداره تلمبه‌خانه علم نیست، هنر است. مسئول تلمبه‌خانه در حالی که به دلایل مختلف چون خرابی تلمبه‌ها، نشت خطوط نفت، تغییرات فشار روزانه و فصلی با میزان ورودی‌های متغیر روبرو می‌باشد باید بتواند با تنظیم فشار تلمبه‌ها خروجی‌های یکنواختی را تولید کند. برنامه گلدبرگ ترکیب مناسب تلمبه‌ها و فشار لازم را برای دست‌یابی به خروجی مورد نظر می‌یابد. هم‌زمان که ورودی‌ها تغییر می‌کند الگوریتم ژنتیک به طور خودکار ترکیب جدیدی را جستجو کرده و مجدداً جریان خروجی قبلی برقرار می‌شود. الگوریتم می‌آموزد که چگونه جبران کند و در واقع خود، آموزگار خود است.

□ **هیچ چیز کامل نیست :**

در کلیه موارد استفاده طبیعی یا مصنوعی، الگوریتم ژنتیک به جواب‌های مناسب

می‌رسد ولی جواب‌ها هرگز کامل نیستند. این حالت عمدتاً به این دلیل است که ضوابط توفیق مرتباً در حال تغییر است. مسائل عمده ممکن است همیشگی باشند لیکن نحوه پاسخ به مسائل در حال تغییر است. حیات در شرایطی کاملاً متفاوت با شرایط امروز شروع شده است. محیط دائماً در حال تغییر است و هرگاه‌گونه جدیدی از موجودات ظهور می‌کند قواعد حاکم را برای بسیاری دیگر از موجودات تغییر می‌دهد. موجودات زنده می‌توانند با محیط سازگار شوند لیکن این تطابق هرگز کامل نخواهد بود.

از نظر تئوریک می‌توان دوره‌های جفت‌گیری و جهش الگوریتم ژنتیک را بدون حد تا دست‌یابی به طرح کامل تراشه و یا طرح کامل برنامه تلمبه‌های نفت ادامه داد، لیکن در همان زمان بازار تراشه و یا شرایط جریان نفت در لوله‌ها تغییر خواهد کرد. بدین صورت جواب کامل شما کهنه خواهد شد. تا آنجا که به تکامل مربوط می‌شود وضعیت خیلی خوب به اندازه کافی خوب است. در واقع از وضعیت کامل هم بهتر است زیرا بیشتر قابل انعطاف است. البته عده‌ای باور نخواهند داشت که حتی بتوان با روشی که تنها بر اساس ترکیبات کور و ثبت دقیق اتفاقات عمل می‌کند به جواب‌های مناسب دست یافت. انتظار نمی‌رود سیستمی با چنین میزان اطلاعات ناچیز بتواند به خوبی آنچه را که کشف می‌کند حفظ و از آن استفاده کند. علیهذا از طریق فرآیندی مشابه همین الگوریتم است که ما آنچه را از طبیعت می‌دانیم، آموخته‌ایم.

دانشمندان فرضیه خلق می‌کنند. فرضیه‌ها به آزمون محیط سپرده می‌شوند. دانشمندان دیگر آن را نقد می‌کنند و آنچه از بوته آزمون، موفق بیرون می‌آید با عملیات ژنتیکی متعدد روبرو می‌شود. فرضیه‌ها توسط انتشارات علمی تکثیر می‌شوند، فرضیه‌های مفید مجدداً منتشر می‌شوند و ماخذ مراجعه افراد بیشتری قرار می‌گیرد.

سپس دانشمندان با بهره‌گیری از بخش‌هایی از فرضیه‌ها مجدداً آنها را ترکیب می‌کنند و به این طریق فرضیه‌ها جفت‌گیری می‌کنند. دانشمندان همچنین مشابه عملیات جهش تغییرات عمده در فرضیه‌ها وارد می‌کنند. ما معمولاً این اتفاقات را ابتکارهای زاییده بصیرت فردی می‌نامیم ولی برداشت دیگری از این ابتکار چیزی جز تغییرات کور، ثبت دقیق تغییرات و نگهداشت بهترین جواب‌ها نیست.

واقعیت این است که روش حل موفقیت‌آمیز یک مسئله لزوماً نباید زیرکانه باشد، بلکه باید توان فراگیری از تجربه را داشته باشد. آنچه را عملی است حفظ کند و آنچه را عملی نیست طرد نماید. درسی از تاریخ در این نهفته است. ما از اجداد خود باهوش‌تر نیستیم. ما تنها به این دلیل متمدن هستیم که اتفاقات را بهتر ثبت کرده‌ایم. اگر آنها اتفاقات را بهتر ثبت می‌کردند متمدن‌تر بودند.

همزمان که تحقیقات در تکامل مصنوعی ادامه می‌یابد ما این فرصت را داریم که در مورد فراگیری، بیشتر فراگیریم. موضوع هوش مصنوعی معمولی و از خارج برنامه‌ریزی شده احتمالاً دارای چنین زمینه باروری در تحقیقات نخواهد بود. توان آن از اندیشه خود ما به مراتب عقب‌تر است. اگر سؤالی کنیم که جزیی خارج از تجربه محدود سیستم باشد هوش مصنوعی به سرعت به کودنی مصنوعی تبدیل می‌شود. در مقابل الگوریتم ژنتیک فرا می‌گیرد و فرا می‌گیرد و فرا می‌گیرد و این توان را در بوته آزمایش‌های میدانی ۳/۵ میلیون سال گذشته نشان داده است.

پژوهشگاه علوم انسانی و مطالعات فرهنگی
پرتال جامع علوم انسانی