حل برنامه ریزی خطی دو ترازه با استفاده از الگوریتم ژنتیک

نگار صلاحی*
دانشگاه آزاد اسلامی واحد ابهر

چکیده

حالت خاصی از برنامه‌ریزی چندترازه[۱]، برنامه‌ریزی‌های دوترازه[۲] است که بعلت سهولت در نشان دادن تاثیر روش، بیشتر دو ترازه درنظر گرفته می‌شود. همچنین یکی از انواع مهم و اساسی سیستم‌های دو ترازه مسائل برنامه‌ریزی خطی دوترازه (Blpp) است.
هرچند که توابع هدف و قیود سطوح بالا و پایین این نوع مسائل همگی خطی هستند اما از طرفی این نوع برنامه‌ریزی برای توابع هدف مسائل تراز بالا در همه‌جا پیوسته و محدب نیستند و مهم هستند. جوابی از مسائل پایین‌تر را بدست آورند که بطور کلی خطی و مشتق‌پذیر نباشند.
در این مقاله قصد داریم طرحی از الگوریتم ژنتیک را برای حل (Blpp) با ساختن تابع برازندگی تراز بالای یک برنامه‌ریزی خطی برپایه تعریف درجه شدنی بودن ارائه دهیم. الگوریتم ژنتیکی که در اینجا بکار گرفته می‌شود از روش تابع جریمه قیود استفاده نمی‌کند بلکه با تغییر تصادفی نسل آغازین جمعیت به جای یک نسل آغازین صادق در قیود به منظور بهبود بخشیدن توانایی الگوریتم، جواب بهینه را بدست می‌آورد و سرانجام با نتایج عددی به دست آمده و تعداد مثال حل شده با این روش، توانایی و کارا بودن آن را نشان می‌دهد.

واژه‌های کلیدی: برنامه‌ریزی خطی دو ترازه، تصمیم‌گیری نا متمرکز[۳]، جواب بهینه[۴]، الگوریتم ژنتیک[۵].

۱ مقدمه

یك سیستم تصمیم نامتمرکز و غیر مشارکتی را در یك وضعیت تعادلی که یك رهبر و چندین پیرو با هم در ارتباط هستند در نظر می‌گیریم، فرض کنیم که رهبر و پیروها هر کدام متغیرهای تصمیم مربوط به خودشان را داشته باشند؛ رهبر می‌تواند از طریق متغیرهای تصمیم خودش به عکس‌العمل پیروها تأثیر بگذارد، (امرکند) در مقابل، پیروها در این‌که کاربردهای عینی خود را در قبال تصمیمات رهبر و پیروهای دیگر بهبود بخشند و به حداکثر برسانند، اختیار تام دارند، لذا ابزار نیرومند سیستم‌های تعیین نامتمرکز «برنامه‌ریزی چند ترازه» است، لازم به ذکر است که فرمول‌های برنامه‌ریزی خطی ممکن است، به‌طور قابل ملاحظه‌ای از یك مقاله تا مقاله دیگر تغییر کند.
بیالاس و کاروان در سال (۱۹۸۴) نامحدب بودن مسائل تراز بالا را با مثال‌های عددی اثبات نمودند [۱].
همچنین باردوبن–آید اثبات نمودند که هر (Blpp) یک مسأله NP-hard است. [۱] علاوه بر این در سالهای اخیر و بینسنت و ساوارد (۲۰۰۱) اذعان داشتند که هر مسأله NP-hard می‌تواند جواب بهینه کلی مسأله (Blpp) را پیدا کند

*سخنران

[۱] Multi level
[۲] Bi-level
[۳] Decenteralized Decision Making
[۴] Optimized solution
[۵] Geneic Algorithm

[۱].
در حال حاضر الگوریتم‌های منطقی برای حل برنامه‌ریزی خطی دوترازه ارائه شده است که می‌توان آنها را بطور خلاصه بصورت زیر دسته‌بندی نمود:

۱– روش پیدا کردن نقطه رأسی: این روش اساساً برگرفته از الگوریتم K– ام بهترین است [۱].

۲– روش تبدیلات: این روش برگرفته از الگوریتم مکمل خطی و الگوریتم شاخه و کران و روش کاربردی تابع جریمه می‌باشد.

۳– روش تکاملی: این روش براساس حل ژنتیکی معادله تعادل Nash و الگوریتم ژنتیک است.

۲ مدل ریاضی برنامه‌ریزی خطی دو ترازه (Blpp)

انواع مختلفی از مدلهای برنامه‌ریزی خطی و غیرخطی دو ترازه وجود دارد که در این مقاله مدل زیر را به عنوان مدل برنامه‌ریزی خطی معرفی می‌کنیم [۱]:

$$
\begin{aligned}
&P_1 : \max F(x,y) = a^t x + b^t y \\
&\qquad y \geq 0 \\
&P_2 : \max f(x,y) = c^t x + d^t y \\
&s.to : Ax + By \leq r \\
&\qquad x, y \geq 0
\end{aligned}
$$

بطوریکه:

$$
\begin{aligned}
&b, d, y \in \mathbb{R}^{n_2} \\
&a, c, x \in \mathbb{R}^{n_1}, b \in \mathbb{R}^{m \times n_2}, A \in \mathbb{R}^{m \times n_1}, r \in \mathbb{R}^m
\end{aligned}
$$

که اگر x ثابت فرض شود، جمله $c^t x$ در تابع هدف تراز پایین ثابت می‌شود. بنابراین تابع هدف مسأله تراز پایین بصورت زیر درمی‌آید:

$$\bar{f} = d^t y$$

حال فرض کنید ناحیه قیدی مسأله برنامه‌ریزی دوترازه خطی بصورت

$$S = \{(x,y) | Ax + By \leq r\}$$

باشد که این ناحیه ناتهی و کراندار است سپس مجموعه ناتهی و کراندار

$$Q = \{y | By \leq r - Ax, \quad y \geq 0\}$$

را درنظر بگیرید و فرض کنید $Y(x)$ مجموعه جوابهای بهینه مسأله

$$
\begin{aligned}
&\max \bar{f} = d^t y \\
&\quad y \in Q(x)
\end{aligned}
$$

باشد. عضو منحصربفردی از مجموعه $Y(x)$ موجود است بطوریکه:

$$\psi_f(s) = \{(x, y) \mid (x, y) \in S, y \in Y(x)\}$$

بنابراین مسأله $P_۱$ می‌تواند به فرم زیر درآید:

$$\begin{aligned} (P_۲) &: \max F(x, y) = c^t x + d^t y \\ s.to &: Ax + By \leq r \\ & y = Y(x) \end{aligned}$$

حال اگر نقطه (x, y) یک جواب مسأله زیر باشد:

$$\begin{aligned} &\max F(x, y) = c^t x + d^t y \\ &s.to : Ax + By \leq r \quad , \quad y = Y(x) \end{aligned}$$

بنابراین (x, y) جواب مسأله $(P_۱)$ نیز هست.

تعریف ۱.۲. نقطه (x, y) را شدنی گوییم: (feasibility point) اگر:

$$(x, y) \in \psi_f(s)$$

تعریف ۲.۲. نقطه شدنی $(x^*, y^*) \in \psi_f(s)$ را جواب بهینه مسأله (Blpp) می‌دانیم اگروفقط اگر برای هر نقطه $(x, y) \in \psi_f(s)$ داشته باشیم:

$$F(x^*, y^*) \geq f(x, y)$$

این مقاله روش عددی (Blpp) را براساس این تعاریف بحث می‌کند.

۳ طرح الگوریتم ژنتیک برای (Blpp)

شناخت تابع هدف تراز بالا در برنامه‌ریزی خطی دوترازه کار آسانی نیست و نمی‌توان فرمول آشکار و واضحی برای آن بدست آورد چون حل اینگونه توابع از حل توابع تراز پایینی ناشی می‌شود که خود نیز فرمول‌بندی مشخصی ندارد، بنابراین به کار بردن تعریف مشتق کار آسانی نبوده و لذا بحث روی وجود جوابهای بهینه با توجه به شرایط ذکر شده در الگوریتم‌های مذکور و همچنین الگوریتم ژنتیک در مسائل بهینه‌سازی با روشهای عددی کار بسیار مشکلی است. لذا در این نوشتار سعی داریم برنامه‌ریزی خطی دوترازه را با استفاده از الگوریتم ژنتیک که روشی کارا در بدست آوردن جواب بهینه است بکار ببریم.

۴ ایده اصلی الگوریتم (GA) ارائه شده برای حل (Blpp)

ابتدا یک جمعیت آغازین صادق در قیود انتخاب می‌کنیم، سپس متغیر تصمیم تراز پایین را متناظر با واکنش بهینه تراز بالا می‌سازیم و افراد را با محاسبه تابع برازندگی‌شان برمبنای درجه شدنی بودن، تا زمانیکه جواب بهینه توسط عملگر ژنتیک بیشتر و بیشتر و بهتر جستجو شود ارزیابی می‌کنیم.

۵ کدگذاری قیود

اغلب کدگذاریهای حاضر بدو صورت:

الف– کدگذاری بردار دودویی

ب– کدگذاری بردار شناور (حقیقی) تقسیم می‌شود.

تجربه نشان داده است روش دوم به فضای مسأله نزدیکتر بوده و با همگرایی زودتر و بدست آوردن جوابهای بهینه دقیق‌تری همراه است. در این نوشتار نیز از کدگذاری حقیقی استفاده می‌شود. بنابراین هر فرد بصورت یک بردار $V_K = (V_{K_۱}, V_{K_۲}, \cdots, V_{K_m})$ درنظر گرفته می‌شود. در الگوریتم‌های ژنتیک بطور کلی افراد جمعیت آغازین بصورت تصادفی تولید شده و تمایل به تولید فرزندانی را دارند که در ناحیه قیدی نباشند. در اینجا به طریق زیر با قیود سروکار داریم: ابتدا یک نسل از افراد، بطور تصادفی تولید و سپس افرادی که در قیود $Ax + By \leq r$ صادق هستند را بعنوان یک جمعیت آغازین نگه داشته و بقیه را بیرون می‌اندازیم، بنابراین همه‌ی افراد تولید شده به این طریق در قیود صدق می‌کنند و فرزندان نیز بوسیله عملگرهای ژنتیکی انتخاب شده صادق در قیود خواهند شد.

۶ محاسبه تابع برازندگی

برای حل مسأله $(P_۳)$ توسط الگوریتم GA با استفاده از تعریف درجه شدنی، ابتدا تابع برازندگی را بصورت زیر تعریف نموده و فرض می‌کنیم d یک جریمه بزرگ بازه‌ای از ناحیه شدنی برای هر $(x, y) \in S$ باشد.

تعریف ۱.۶. فرض کنیم $\theta \in [۰, ۱]$ درجه صدق ناحیه شدنی باشد که بصورت تابع زیر تعریف می‌شود:

$$\begin{cases} ۱ & \text{اگر } \|y - Y(x)\| = ۰ \\ ۱ - \left\|\dfrac{y - Y(x)}{d}\right\| & \text{اگر } \alpha\|y - Y(x)\| \leq d \\ ۰ & \text{اگر } \|y - Y(x)\| \geq d \end{cases}$$

بطوریکه $\|\|$ یک نرم باشد بنابراین طبق تعریف فوق تابع برازندگی بصورت زیر محاسبه میگردد:

$$eval\ (v_k) = (F(x, y) - F_{\min} * \theta \qquad (*)$$

در صورتیکه $F_{\min}$ کمترین مقدار تابع $F(x, y)$ روی s باشد.

۷ عملگرهای ژنتیک

در مسائل بهینه‌سازی با متغیرهای پیوسته از عملگرهای پیوند بسیاری استفاده شده است، که از مهمترین آنها پیوند ساده– ضربدری– حسابی می‌باشد. ولی در اغلب جمعیت‌ها پیوند حسابی کاربرد بیشتری دارد. این مقاله نیز نوعی از پیوند حسابی را بکار می‌برد که (فرزندان) هنوز در ناحیه قیدی هستند و این امر سبب می‌شود تا سیستم پایدارتر شده و انحراف از بهترین جواب کمتر باشد. عملگر پیوند حسابی می‌تواند دو فرزند را بگونه‌ای تولید کند که بطور کلی ترکیب خطی از والدین باشند، اگر $v_۱, v_۲$ دو رشته‌ای باشند که تحت عملگر پیوند حسابی تولید فرزند کنند بنابراین فرزندان آنها بصورت زیر خواهد بود.

$$v'_۱ = \alpha * v_۱ + (۱ - \alpha)v_۲$$
$$v'_۲ = \alpha * v_۲ + (۱ - \alpha)$$

در صورتیکه $\alpha \in [0, 1]$ عددی تصادفی است و لازم به تذکر است که این نوع عملگر حسابی بستار $v-1, v_2 \in S$ نیز هست.
یکی دیگر از عملگرهای مهم بکار برده در الگوریتم ژنتیک جهش است که در این نوشتار نیز مورد استفاده قرارگرفته است. جهش انواع مختلفی دارد: جهش یکنواخت– غیریکنواخت و مرزی. که در اینجا با جهش مرزی برای حل مسائلی که جواب بهینه آنها در، یا نزدیکی ناحیه جستجوی قیود است سروکار داریم.
اگر فرد V_k جهش کند بنابراین: $V'_K = (V'_{K_1}, V'_{K_2}, \cdots, V'_{K_m})$ بدست می‌آید که هر کدام از V'_{K_i} ها می‌تواند جهت چپ یا جهش راست V_K با احتمال یکسان باشند.
و اما انتخاب برپایه این اصل صورت می‌گیرد که فرزندان کارا انتخاب شده و فرزندان غیرکارا حذف می‌شوند و سپس جستجو برای بهترین جواب در جمعیت صورت می‌گیرد و نتیجتاً تعدادی از بزرگترین (بالاترین) افراد با محاسبه تابع برازندگی‌شان بتدریج افزایش می‌یابند و دوره تکامل نیز در طول بیشتر بهینه سازی انجام می‌شود. گرچه عملگرهای انتخاب زیادی در الگوریتم GA وجود دارد اما در اینجا با انتخاب چرخ رولت را بکار می‌بریم که انتخاب ساده و منطقی است.

۸ معیارهای خاتمه

نتیجه‌گیری آخر در صورتی انجام می‌گیرد که نتیجه مطلوب حاصل شود و محاسبات متوقف گردد. تعداد بیشترین تکرار بعنوان شرط خاتمه درنظر گرفته می‌شود لذا الگوریتم ژنتیک بکارگرفته شده در مقاله حاضر را بصورت مراحل زیر بیان می‌نماییم.
گام (۱): (آغازسازی) انتخاب مبنای جمعیت (M)، احتمال پیوند (P_c)، احتمال جهش (P_m) و بیشترین تکرار نسل MAXGEN و فرض اینکه نسل از صفر شروع می‌شود $(t = 0)$
گام (۲): (آغازسازی نسل اولیه) افراد تصادفی شده در S (ناحیه قیدی) را M و ساخته شده از نسل آغازین تولید می‌کنیم.
گام (۳): محاسبه تابع برازندگی و برآورد مقدار برازندگی هر نسل مطابق با فرمول (*)
گام (۴): تولید نسل بعدی توسط عملگرهای GA و انتخاب فرد بوسیله جرخ دولت و برای تولید نسل بعدی نیز از عملگرهای پیوند و جهش ذکر شده استفاده می‌شود.
گام (۵): (شرط خاتمه) هنگامی‌که $(t > M)$ باشد الگوریتم متوقف می‌شود و جواب بهینه بدست می‌آید. به عبارت دیگر $t = t + 1$ سپس به گام (۳) می‌رویم.

۹ نتایج عددی

در مثال‌های زیر کارایی روش حل این نوع الگوریتم ژنتیك برای برنامه‌ریزی خطی دوترازه نشان داده شده است:

مثال ۹.۱. [۶]

$$\max_{x \geq 0} \quad F(x, y) = x + 3y$$

برای y حل شده

$$\begin{aligned} \max_{y \geq 0} \quad & f(x, y) = x - 3y \\ \text{s.t.} \quad & \\ & -x - 2y \leq -10 \\ & x - 2y \leq 6 \\ & 2x - y \leq 21 \\ & x + 2y \leq 38 \\ & -x + 2y \leq 18 \end{aligned}$$

مثال ۲.۹.

$$\max_{x\geq 0} \quad F(x,y) = -8x_1 - 4x_2 - 4y_1 - 40y_2 - 4y_3$$

برای y حل شده

$$\begin{aligned} &\max_{y\geq 0} \quad f(x,y) = x_1 + 2x_2 + y_1 + y_2 + 2y_3 \\ &\text{s.t.} \\ &\qquad -y_1 + y_2 + y_3 \leq -1 \\ &\qquad 2x_1 - y_1 + 2y_2 + 0/5y_3 \leq 1 \\ &\qquad 2x_2 + 2y_1 - y_2 - 0/5y_3 \leq 1 \end{aligned}$$

مثال ۳.۹. [۷]

$$\begin{aligned} &\max_{x\geq 0} \quad F(x,y) = 8x_1 + 4x_2 - 4y_1 + 40y_2 + 4y_3 \\ &\text{s.t.} \\ &\qquad x_1 + 2x_2 - y_3 \leq \tfrac{1}{3} \\ &\qquad \max_{y\geq 0} \quad f(x,y) = -2y_1 - y_2 - 2y_3 \\ &\qquad \text{s.t.} \\ &\qquad\qquad -y_1 + y_2 + y_3 \leq 1 \\ &\qquad\qquad 4x_1 - 2y_1 + 4y_2 + y_3 \leq 2 \\ &\qquad\qquad 4x_2 + 4y_1 - 2y_2 - y_3 \leq 2 \end{aligned}$$

مقایسه نتایج به‌دست آمده از میان ۵۰۰ نسل به‌وسیله الگوریتم ژنتیک به‌کار برده شده و الگوریتم‌ها ژنتیک موجود در منابع ذکر شده در جدول زیر نشان داده شده است:

جدول ۱

	پارامترها			نتایج به‌دست آمده از الگوریتم ارائه شده			نتایج به‌دست آمده از الگوریتم‌های قبلی		
	M	p_c	p_m	(x,y)	F	f	(x,y)	F	f
۱	۵۰	۰/۷	۰/۱۵	(۱۵/۹۵۹, ۱۰/۹۷۲)	۴۸/۸۷۵	−۱۶/۹۵۷	(۱۶, ۱۱)	۴۹	−۱۷
۲	۱۰۰	۰/۶	۰/۲	(۰, ۰/۸۹۹, ۰, ۰/۶, ۰/۳۹۴)	۲۹/۱۷۲	−۳/۱۸۶	(۰, ۰/۹, ۰, ۰/۶, ۰/۴)	۲۹/۲	−۳/۲
۳	۱۰۰	۰/۶	۰/۳	(۰/۵۳۳, ۰/۸, ۰, ۰/۱۹۷, ۰/۸)	۱۸/۵۴۴	−۱/۷۹۷	(۰/۲, ۰/۸, ۰, ۰/۲, ۰/۸)	۱۸/۴	−۱/۸

توجه *: M میزان جمعیت، P_c احتمال پیوند، P_m احتمال جهش، F و f به‌ترتیب مقادیر توابع هدف تراز بالا و تراز پائین مسأله برنامه‌ریزی می‌باشند.

از نتایج عددی به‌دست آمده توسط الگوریتم ژنتیک ارائه شده و الگوریتم ژنتیک موجود در منابع ذکر شده چنین استنباط می‌شود که روش ارائه شده بسیارکاراتر از روش‌های قبلی است و از طرفی تجربه نیز نشان می‌دهد که میزان جمعیت و سرعت جهش، بر کارایی و سرعت همگرایی الگوریتم تأثیر دارد، اما عمل پیوند تأثیر کمتری بر این دو پارامتر دارد، لذا هر چه اندازه جمعیت بزرگتر شود، برای آن‌که سرعت همگرایی افزایش یابد باید از سرعت جهش بیشتری استفاده کنیم.

۱۰ نتیجه‌گیری

این مقاله، روشی از الگوریتم ژنتیک را برای حل برنامه‌ریزی خطی دو ترازه‌ای که جواب بهینه مسأله تراز پائین وابسته با مساله تراز بالا است را پیشنهاد می‌کند، همچنین نتایج عددی نشان در جدول ۱ نیز نشان می‌دهد که این روش در مقایسه با روش‌های قدیمی دارای خصوصیات ریز است:

۱. روش مذکور نیاز خاصی به همه متغیرهای تابع نداشته و بحث مشکل قیود و به‌دست آوردن جواب بهینه در همه الگوریتم‌ها با وجود تعریف مشتق‌پذیری توابع آنها را بر طرف می‌سازد.

۲. روش الگوریتم ژنتیك ارائه شده از به‌کار بردن روش تابعه جریمه قیود دوری نموده و با تغییر تصادفی نسل آغازین صادق در قیود برای بهبود توانائی الگوریتم تلاش می‌کند.

سپاس‌گزاری

این مقاله برگرفته از طرح پژوهشی به‌عنوان «بهینه‌سازی سطوح تصمیم‌گیری» در دانشکده فنی و مهندسی دانشگاه آزاد واحد ابهر به‌عنوان یک مدل برنامه‌ریزی خطی دو ترازه توسط الگوریتم ژنتیک می‌باشد.

مراجع

[۱] سید رضا حجازی طاقانکی، *روشی برای حل برنامه‌ریزی چند سطحه*، دانشگاه تربیت مدرس، رساله دکتری (۱۳۷۹).

[۲] نرگس مولایی‌نژاد، *ارائه مدلی تحت عنوان GDEA و کاربرد آن در بهینه‌سازی چند هدفی با استفاده از الگوریتم GA* ، دانشگاه آزاد اسلامی، واحد زاهدان (۱۳۸۳).

[3] Jenkins, W.M. (1991), *Toward structural optimization via the genetic algorithm (Technical note)*, Computers & Structures, 40 (5).

[4] Y. H. Liu and T. H. Spencer, *Solving a bi-level linear program when the inner decision maker controls few variables*, European Journal of Operation Research, 81 (2003), 644-651.

[5] U. P. Wen and S. T. Hsu, *A note on a linear bi-level programming algorithm based on bi criteria programming*, Computer & Operation Research, 16 (2010), 79-83.

[6] J.B Cruz, *Leader-follow strategies for multi-level systems*, IEEE, Iran actions on Automatic control 23(2), 244-255, (1978).

[7] W. Candler and R. Townsly, *A Linear two-level programming problem*, Computers & Operations research 9 (1999), 59-76.

پست الکترونیکی: n_salahi57@yahoo.com