کاهش بعد یک مساله بهینه‌سازی فازی با استفاده از الگوریتم ABS

علی محرابیان[۱]، رضا قنبری[۲]، خاطره قربانی مقدم[۳]

[۱] دانشجوی دکتری تحقیق در عملیات، دانشکده علوم ریاضی، دانشگاه فردوسی مشهد؛ alimehr600@gmail.com

[۲] عضو هیئت علمی دانشکده علوم ریاضی، دانشگاه فردوسی مشهد؛ rghanbari@um.ac.ir

[۳] عضو هیئت علمی موسسه تحقیقات ریاضی دکتر مصاحب، دانشگاه خوارزمی؛ k.ghorbani@khu.ac.ir

چکیده

بسیاری از مدل‌های برنامه‌ریزی خطی و غیرخطی شامل پارامترها یا متغیرهایی هستند که مقدار آن‌ها را نمی‌توان طور دقیق محاسبه کرد. با توجه به کاربرد و تنوع مدل‌های برنامه‌ریزی خطی و غیرخطی فازی، این مدل‌ها از اهمیت زیادی برخوردار است. در دسته قیود مسایل برنامه‌ریزی خطی گاهی با دسته قیدهای $AX = b$ رو به رو هستیم. یکی از روش‌های حل این دستگاه استفاده از الگوریتم ABS است. این الگوریتم علاوه بر حل سیستم، کاربردهایی در مسائل بهینه‌سازی نیز دارد. در این مقاله، از الگوریتم ABS برای کاهش بعد و ساده‌سازی قیود تساوی مدل برنامه‌ریزی خطی با پارامترهای فازی استفاده خواهیم کرد. با به کارگیری این روش کل قیود مساله به قیود نامساوی تبدیل می‌شود و بعد فضای جواب تقلیل خواهد یافت.

کلمات کلیدی: الگوریتم ABS؛ برنامه‌ریزی خطی فازی؛ فضای پوچ؛ دستگاه معادلات.

۱- مقدمه

در بیشتر مسایل تصمیم‌گیری که منجر به یک مساله‌ی برنامه‌ریزی خطی فازی می‌شود، تصمیم‌گیرنده نمی‌تواند مقادیر ضرایب مساله‌ی تصمیم‌گیری را به طور دقیق بداند و این ابهام ممکن است از نوع احتمالی نباشد. در حقیقت، در برنامه‌ریزی خطی مرسوم معمولا ضرایب توسط افراد با تجربه با مقادیر دقیق تعیین می‌شوند. بنابراین، در محیط‌های نادقیق و صریحا فازی، فرض وجود اطلاعات دقیق بسیار دور از واقعیت است. بدین ترتیب، طرح مدل‌سازی فازی در مسایل تصمیم‌گیری واقعی با داده‌های نادقیق، منطقی و مناسب به نظر می‌رسد. در مسایل برنامه‌ریزی خطی قطعی، هدف ماکزیمم‌سازی یا مینیمم‌سازی تابع هدف بر اساس قیدهای خطی است. در موقعیت‌های واقعی، ممکن است تصمیم‌گیرنده قادر به تعیین توابع هدف و یا قیدها به طور دقیق نباشد، ولی قادر به تعیین آن ها از جنبه‌ی فازی باشد. در این موقعیت‌ها، منطقی است که از برخی مدل‌های برنامه‌ریزی خطی فازی استفاده شود که انعطاف‌پذیری بیشتری را برای تصمیم گیرنده فراهم می کنند .برنامه‌ریزی خطی با پارامترها یا متغیرهای فازی نقش بسیار مهمی در زمینه های مختلف از جمله در علوم مدیریت[۱۲]، نظریه بازی‌ها [۱۳,۳]، اقتصاد و مهندسی [۵] ایفا می کنند (کاربردهای دیگر را می توان در [۴-۱۱] دید). فازی بودن به صورت‌های مختلف در یک مساله‌ی برنامه‌ریزی خطی ظاهر می شود. برای نمونه، ممکن است نامساوی‌ها فازی باشند، متغیرها و یا پارامترهای مساله ممکن است به صورت فازی بیان شوند.

در یک مساله‌ی برنامه‌ریزی خطی استاندارد دسته‌ای از قیدها به صورت $AX = b$ ممکن هست ظاهر شود. دستگاه معادلات خطی برحسب اینکه ماتریس ضرایب آن از چه میدانی (حقیقی، مختلط، عدد صحیح) انتخاب شود، دارای خواص و ویژگی‌های مختلفی است. به طور کلی دستگاه معادلات به دو دسته منفرد یا نامنفرد تقسیم‌بندی می شوند، که اکثر ریاضیدان‌ها علاقمند به مطالعه دستگاه‌هایی هستند که دارای محدودیت‌هایی مانند منفرد بودن، عدد صحیح بودن، مختلط بودن و غیره هستند. از این

رو روش‌های متعددی برای این چنین دستگاه‌هایی ارائه شده است. روش‌های ارائه شده به دو دسته روش مستقیم یا تکراری تقسیم‌بندی می‌شوند. روش‌های مستقیم معمولا با تجزیه ماتریس ضرایب دستگاه $AX = b$ انجام می‌شود. یکی از روش های حل دستگاه $AX = b$ رده الگوریتم‌های ABS است. الگوریتم‌های ABS در سال ۱۹۸۴ توسط ابافی[۱] برای حل دستگاه معادلات خطی ارائه شد. ابتدا این الگوریتم به عنوان رده الگوریتم‌های ABS مورد استفاده قرار می‌گرفت در ادامه به عنوان رده الگوریتم‌های ABS مقیاس شده تعمیم یافت. این الگوریتم کاربردهایی در مسائل کمترین مربعات خطی، معادلات غیرخطی و بهینه‌سازی دارد. این رده الگوریتم‌ها به دلیل وجود پارامترهای آزاد، مورد علاقه ریاضیدانان و محققان قرار گرفت که با اعمال شرایطی روی این پارامترها، روش‌های جدید و بهبودیافته‌تری نسبت به الگوریتم اولیه ارائه دادند. از جمله کاربردهای این الگوریتم، تجزیه مثلثی ماتریس‌ها، حل مسائل بهینه‌سازی و غیره بود.

در این مقاله به بررسی مدلی فازی می‌پردازیم که در آن همه قیود قطعی هستند. برخلاف تصور نحوه برخورد با قیود نامساوی کاری دشوارتر نسبت به قیود تساوی هست. در این‌جا به کمک الگوریتم ABS، قیود تساوی و نامساوی مدل را به قیود نامساوی تبدیل کرده و با این تبدیل بعد فضای جستجو کاهش خواهد یافت.

۲-تعریف‌های مقدماتی

در این بخش، تعریف‌های مقدماتی درباره مجموعه‌های را خواهیم داشت.

تعریف۱ (مجموعه فازی). [۲] فرض کنید X یک مجموعه مرجع باشد. $\tilde{A}$ را یک مجموعه فازی بر روی X گوییم و به صورت زیر تعریف می‌کنیم که در آن $\mu_{\tilde{A}}(x) \in [0,1]$:

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) | x \in X\}.$$

تعریف۲ (تکیه‌گاه). [۲] تکیه‌گاه یک مجموعه‌ی فازی به صورت زیر تعریف می‌شود:

$$supp(\tilde{A}) = \{x \in X \mid \mu_{\tilde{A}}(x) > 0\}.$$

تعریف۳ (مجموعه فازی محدب). [۲] یک مجموعه‌ی فازی $\tilde{A}$ محدب است هرگاه داشته باشیم:

$$\mu_{\tilde{A}}(\gamma x + (1-\gamma)y) \geq \min\{\mu_{\tilde{A}}(x), \mu_{\tilde{A}}(y)\}.$$

تعریف ۴ (عدد فازی LR). [۲] یک مجموعه‌ی فازی $\tilde{A}$ با تابع عضویت

$$\mu_{\tilde{A}}(x) = \begin{cases} L\left(\dfrac{a-x}{\alpha}\right), & x \leq a \\ R\left(\dfrac{a-x}{\beta}\right), & x \geq a \end{cases}$$

را یک عدد فازی LR می‌نامند که در آن، L (به طور مشابه R) تابعی نا افزایشی از $\mathbb{R}^+$ به بازه [۰ ۱] است. در نمایش عدد فازی LR، a مقدار میانی، α و β به ترتیب پهنای چپ و راست $\tilde{A}$ نامیده می‌شوند.

۳-بیان یک مساله بهینه‌سازی فازی و ساده‌سازی آن با استفاده از الگوریتم ABS

ما در این بخش یک مساله بهینه‌سازی فازی را معرفی کرده و سپس با استفاده از الگوریتم ABS قیدهای کل قیود آن را به قیدهای نامساوی تبدیل می‌کنیم. با این کار بعد فضای جستجو کاهش می‌یابد.

---

[۱] Abaffy

مساله بهینه سازی فازی زیر را در نظر بگیرید:

$$\begin{gathered} \min \tilde{c}^T x \\ s.t \\ Ax = b \\ x \geq 0 \end{gathered} \qquad (۱.۳)$$

که در آن $\tilde{c}$ ضرایب تابع هدف و از نوع فازی، $A \in \mathbb{R}^{m\times n}$، $b \in \mathbb{R}^m$ و $x \in \mathbb{R}^n$. لذا در این مساله مقدار تابع هدف یک جواب شدنی، فازی هست. قیود این مساله قطعی و شامل قیود تساوی و نامساوی هست. برخلاف تصور، نحوه برخورد با قیود نامساوی کاری دشوارتر هست.

۳-۱- الگوریتم ABS

الگوریتم ABS توسط ابافی، برویدن[۲] و اسپدیکاتو[۳] در سال ۱۹۸۴ گسترش یافت[۱]. دستگاه معادلات زیر را در نظر بگیرید:

$$Ax = b \qquad (۲.۳)$$

که در آن $A \in \mathbb{R}^{m\times n}$، $b \in \mathbb{R}^m$، $\operatorname{rank}(A) = m$. فرض کنید $A = (a_1, a_1, \dots, a_m)^T$ که $a_i \in \mathbb{R}^n$ برای $i = 1,2,\dots,m$ و $b = (b_1, b_2, \dots, b_m)^T$. همچنین در نظر بگیرید $A_i = (a_1, \dots, a_i)$ و $b^{(i)} = (b_1, \dots, b_i)^T$

فرض کنید $x_1 \in \mathbb{R}^n$ دلخواه و $H_1 \in \mathbb{R}^{n\times n}$ (پارامتر اسپدیکاتو) دلخواه و نامنفرد داده شده است. توجه کنید که برای هر $x \in \mathbb{R}^n$ می‌توانیم x را به صورت $x = x_1 + H_1^T q$ بنویسیم.

الگوریتم ABS شامل یک دستورالعمل تکراری برای پیدا کردن جواب عمومی (۲.۳) هست[۶]. درابتدای تکرار iام که $i \geq 1$، جواب عمومی $(i-1)$ معادله در دسترس هست. واضح است که اگر x_i یک جواب $(i-1)$ معادله و $H_i \in \mathbb{R}^{n\times n}$، به صورتی باشد که $\operatorname{rank}(H_i) = n - i + 1$ و ستون‌های H_i^T فضای پوچی A_{i-1}^T را تولید کند. آن‌گاه:

$$x = x_i + H_i^T q \qquad (۳.۳)$$

با یک $q \in \mathbb{R}^n$ دلخواه، جواب عمومی $(i-1)$ معادله را تشکیل می‌دهد و داریم:

$$A_{i-1}^T x = b^{(i-1)} \qquad (۴.۳)$$

که در **(۴.۳)** x به دست آمده از **(۳.۳)** هست. حال از آن‌جایی که $\operatorname{rank}(H_i) = n - i + 1$ و H_i^T تولیدکننده پوچی A_{i-1}^T هست (که به صورت بدیهی برای $(i = 1)$ برقرار هست). اگر در نظر بگیریم:

$$p_i = H_i^T z_i$$

با یک $z_i \in \mathbb{R}^n$ دلخواه (پارامتر برویدن)، آن‌گاه $A_{i-1}^T p_i = 0$ و

$$x(\alpha) = x_i - \alpha\, p_i$$

که برای هر α، در $(i-1)$ معادله اول صدق می‌کند. اگر قرار دهیم $x_{i+1} = x(\alpha_i)$ که iامین معادله را نیز حل کند. در این صورت داریم:

$$\alpha_i = \frac{a_i^T x_i - b_i}{a_i^T p_i}$$

با این فرض که $a_i^T p_i \neq 0$ آن‌گاه:

$$x_{i+1} = x_i - \alpha_i p_i$$

یک جواب برای i معادله اول هست. حال برای کامل شدن روند الگوریتم، کافی است H_i به H_{i+1} بروزرسانی شود به طوری که $H_{i+1} A_i = 0$. کافی است قرار دهیم:

---

[۲] Broyden

[۳] Spedicato

$$H_{i+1} = H_i - u_i v_i^T$$

و u_i، v_i را طوری انتخاب کنیم که $H_{i+1}a_j = 0$ برای $j = 1,2,\ldots,i$. همچنین رتبه H_i نیز باید یک واحد کاهش یابد. در الگوریتم ABS اولیه $u_i = H_i a_i$ و $v_i = H_i^T w_i / w_i^T H_i a_i$ انتخاب می‌شود که w_i (پارامتر ابافی) در رابطه زیر صدق می‌کند.

$$w_i^T H_i a_i \neq 0$$

بنابراین، فرمول بروزرسانی زیر را داریم:

$$H_{i+1} = H_i - \frac{H_i a_i w_i^T H_i}{w_i^T H_i a_i}$$

حال می‌توانیم الگوریتم کلی ABS را به صورت الگوریتم زیر ارایه کنیم[۶] . در این الگوریتم r_{i+1}، رتبه ماتریس A_i هست و در نتیجه رتبه H_i برابر $n - r_{i+1} + 1$ خواهد بود.

(۱) $x_1 \in \mathbb{R}^n$ دلخواه و $H_1 \in \mathbb{R}^{n\times n}$ دلخواه و نامنفرد در نظر بگیرید. قرار دهید $i = 1$ و $r_i = 0$.

(۲) $t_i = a_i^T x - b_i$ و $s_i = H_i a_i$ را حساب کنید.

(۳) **اگر** ($t_i = 0$ و $s_i = 0$) **آن‌گاه** قرار بده $x_{i+1} = x_i$، $H_{i+1} = H_i$، $r_{i+1} = r_i$ و به گام ۷ برو (معادله i زاید هست).

اگر ($t_i \neq 0$ و $s_i = 0$) **آن‌گاه** متوقف‌شو (iامین معادله و در نتیجه دستگاه معادلات ناسازگار هست).

(۴) ($s_i \neq 0$) جهت جستجو $p_i = H_i^T z_i$ را حساب کن، طوری که $z_i \in \mathbb{R}^n$ دلخواه و در رابطه $z_i^T H_i a_i = z_i^T s_i \neq 0$ صدق کند. حساب کن:

$$\alpha_i = \frac{t_i}{a_i^T p_i}$$

و قرار بده:

$$x_{i+1} = x_i - \alpha_i p_i$$

(۵) H_i را به H_{i+1} بروزرسانی کن:

$$H_{i+1} = H_i - \frac{H_i a_i w_i^T H_i}{w_i^T H_i a_i}$$

که $w_i \in \mathbb{R}^n$ بردار دلخواه هست به طوری که $w_i^T s_i \neq 0$.

(۶) قرار بده $r_{i+1} = r_i + 1$

(۷) **اگر** $i = m$ متوقف‌شو (x_{m+1} جواب هست) **در غیر این‌صورت** $i = i + 1$ و به گام ۲ برو.

الگوریتم *ABS*

در الگوریتم *ABS*، $s_i = 0$ به معنی آن هست که قید جدید در پوچی قیود قبلی هست و اطلاعات جدیدی را اضافه نمی‌کند. همچنین شرط $z_i^T H_i a_i \neq 0$ در گام ۴ به دلیل آن هست که جهت حرکت، در جهت پوچی قید a_i نباشد تا بتوانیم به صورت محدود در آن جهت حرکت کنیم و برای α_i جوابی داشته باشیم. توجه کنید اگر (۲.۳) سازگار باشد، آن‌گاه جواب کلی معادله به صورت زیر هست:

$$x = x_{m+1} + H_{m+1}^T q \qquad (۵.۳)$$

که $q \epsilon \mathbb{R}^n$ دلخواه هست. به علاوه ما در این‌جا $H_1 = I$ (پارامتر اسپدیکاتو)، z_i (پارامتر برویدن) و w_i (پارامتر آبافی) را برابر $H_i a_i$ قرار می‌دهیم.

فرض کنید $\mathrm{rank}(A) = r \leq m$ و دستگاه معادلات (۲.۳) سازگار باشد، در این صورت جواب عمومی همان (۵.۳) خواهد بود که H_{m+1}^T دارای $n - r$ ستون مستقل خطی هست. بنابراین در پایان الگوریتم، می‌توانیم سطرهای وابسته خطی H_{m+1}

را حذف کنیم. در این صورت $q\epsilon\mathbb{R}^{n-r}$ خواهد بود و بعد فضای جستجو r واحد کاهش می‌یابد. این مورد زمانی که m و n نزدیک هم هستند، بسیار کاربردی هست و بعد فضای جواب را به شدت کاهش می‌دهد. پس در نهایت تبدیل یافته مدل فازی (۱.۳) به فرم زیر هست:

$$\min \tilde{c}^T(x_{m+1} + H_{m+1}^T\, q)$$
$$\text{s.t}$$
$$x_{m+1} + H_{m+1}^T q \geq 0$$

چون در این روش با استفاده از الگوریتم ABS بعد فضای جواب مساله بهینه‌سازی فازی را کاهش می‌دهیم، این مورد در زمانی که بعد مساله بالا باشد بسیار حایز اهمیت هست.

۵-نتیجه و جمع‌بندی

در این مقاله به بیان یک مساله برنامه‌ریزی خطی فازی پرداختیم و سپس با استفاده از الگوریتم ABS مساله جدیدی به وجود آمدکه نسبت به مساله اولیه بعد کمتری دارد. روش پیشنهادی در این مقاله زمانی که با مسائلی با ابعاد بالا رو به رو می‌شویم بسیار کاربرد دارد، با استفاده از این روش می‌توان مساله را تبدیل به مساله‌ای با بعدی پایین تبدیل کرد.

۶-مراجع

[1] Abaffy, J., Broyden, C. and Spedicato, E., “*A class of direct methods for linear systems*”, Numerische Mathematik 45, 361–376, 1984.

[2] Adamo, J. M., *Fuzzy decision trees,* Fuzzy Sets and Systems, 4, 207-219, 1980.

[3] Bector, C. R., Chandra, S. and Vijiay, V., *“Duality in linear programming with fuzzy parameters and matrix games with fuzzy pay-offs”,* Fuzzy Sets and Systems 146, 253- 269, 2004.

[4] Buckely, J. J. and Jowers, L. J., *Monte Carlo Method in Fuzzy Optimization*, Studies in Fuzziness and Soft Computing, Springer, 2007.

[5] Bukley, J. J, Eslami, E and Feuring, T., *Fuzzy Mathmatics in Economics and Engineering*, Physica-Verlag, Heidelberg, 2002.

[6] Chang, W., *“Ranking of fuzzy utilities with triangular membership functions”*, Proceedings of International Conference on Policy Analysis and Systems, 263–272, 1981.

[7] Esmaeilli, H., Mahdavi-Amiri, N. and Spedicato, E., “*Explicit abs solution of a class of linear inequality systems and lp problems*”, Bulletin of the Iranian Mathematical Society 30, 2, 21–38, 2011

[8] Ghanbari, R., Ghorbani-Moghadam Kh. and Mahdavi-Amiri, N., “*A variable neighborhood search algorithm for solving fuzzy number linear programming problems using modified Kerres method* ”, IEEE Transactions on Fuzzy Systems 27, 1286-1294, 2019.

[9] Ghanbari, R., Ghorbani-Moghadam Kh. and Mahdavi-Amiri, N. , “*A variables neighborhood search algorithm for solving fuzzy quadratic programming problems using modified Kerres method*”, Soft Computing 23, 12305-12315, 2019.

[10] Ghanbari, R., Ghorbani-Moghadam Kh. and Mahdavi-Amiri, N and De Beats, B., “*Fuzzy Linear Programming Problems: Models and Solutions*”, Soft Computing 24, 10043–10073,2020.

[11] Nakahara, Y. and Gen, M., “*A method for solving linear programming problems with triangular fuzzy coefficients using new ranking index*”, Computers & industrial engineering 25, 1, 1–4, 1993.

[12] Negoitaa, C.V., *Fuzziness in Management,* OPSA/TIMS, Miami, 1976.

[13] Vijay, V., Mehra, A., Chandra S and Bector, C. R., “*Fuzzy matrix games via a fuzzy relation approach*”, Fuzzy Optimization and Decision Making 6, 299-314 ,2007.