فصلنامه مدیریت صنعتی دانشکده علوم انسانی، دانشگاه آزاد اسلامی
واحد سنندج – سال یازدهم / شماره۳۵/ بهار ۱۳۹۵

حل مسأله برنامه ریزی دو سطحی خطی با استفاده از الگوریتم ژنتیک

هاشم عمرانی

استادیار دانشکده مهندسی صنایع، دانشگاه صنعتی ارومیه، ارومیه، ایران

Email: h.omrani@uut.ac.ir

تاریخ دریافت: ۹۴/۸/۱ * تاریخ پذیرش: ۹۵/۴/۲۸

چکیده

مسأله برنامه ریزی دو سطحی (BLP) یکی از مسائل مهم در تئوری تصمیم گیری می باشد که زیر مجموعه مسائل برنامه ریزی چند سطحی به شمار می رود. این مسأله دارای دو سطح بیرونی و داخلی می باشد که فضای جواب مسأله بیرونی یا سطح اول توسط مسأله داخلی یا سطح دوم معین می شود. با توجه به اینکه BLP یک مسأله NP-hard می باشد، حل آن توسط روشهای سنتی به راحتی امکان پذیر نیست. در این مقاله ابتدا مسأله BLP و کاربردهای آن بررسی و سپس برای یافتن نقطه بهینه مسأله از روش شمارش نقاط رأسی استفاده می شود. در این مقاله برای جستجوی فضای اطراف نقاط رأسی و یافتن جواب بهینه از الگوریتم ژنتیک استفاده می گردد. همچنین با استفاده از یک پارامتر کنترلی، شعاع فضایی را که باید جستجو شود کنترل می شود تا از افزایش زمان حل مسأله اجتناب گردد. نتایج خروجی نشان می دهد که جواب بدست آمده از الگوریتم ژنتیک پیشنهادی در مقایسه با مطالعات قبلی قابل قبول می باشد.

کلمات کلیدی: برنامه ریزی دو سطحی، الگوریتم ژنتیک، تئوری تصمیم گیری.

۱- مقدمه

مسأله برنامه ریزی دو سطحی (BLP) بعنوان یک مسأله مهم در تئوری تصمیم گیری مطرح شده است. تصمیم گیرندگان معمولاً در یک ساختار سلسله مراتبی تصمیم گیری می کنند و احتمالاً تصمیمات آنها در تضاد با هم قرار دارد. باید توجه کرد که مدل های دو سطحی زیر مجموعه مسائل برنامه ریزی چند سطحی[1] (MLP) می باشند. در حقیقت BLP حالت خاصی از MLP می باشد که در ساختار آن فقط دو سطح وجود دارد. در مسأله BLP دو تصمیم گیرنده در یک ساختار سلسله مراتبی سعی می کنند که اهداف مورد نظر خود را که در بعضی موارد در تضاد با هم هستند، بهینه کنند. تصمیم گیرنده سطح دوم اهداف خود را تحت پارامترهای گرفته شده از تصمیم گیرنده سطح اول بهینه می کند.

مسأله BLP یک مسأله غیر محدب و NP-hard می باشد و بنابراین در صورتی که تعداد متغیرهای آن زیاد باشد، حل آن با روشهای سنتی و کلاسیک امکان پذیر نخواهد بود. البته باید توجه کرد که مدلهای BLP انواع مختلفی دارند که می توان به مدلهای خطی، عدد صحیح، غیرخطی و ... اشاره کرد. بن آید و بلایر (۱۹۹۰) و بارد (۱۹۹۱) ثابت کردند که مسأله BLP یک مسأله NP-hard است و هانسن، جامرد و ساورد (۱۹۹۲) ثابت نمودند که BLP یک مسأله شدیداً NP-hard می باشد.

همانگونه که گفته شد، BLP زیر مجموعه MLP می باشد. مسائل برنامه ریزی چند سطحی شامل چندین تابع هدف می باشند که باید بر روی یک ناحیه معین بهینه شوند. اگرچه کنترل بر روی متغیرهای تصمیم در بین سطوح مختلف وجود دارد، ولی متغیرهای تصمیم موجود در سطح اول بر همه توابع سطوح بعدی تأثیر می گذارند. ابزارهای تحت کنترل تصمیم گیرنده سطح اول (مثلاً تخصیص منابع به سطوح دیگر) به او این امکان را می دهد که بر سیاستگذاری سایر مدیران تأثیر بگذارد تا بتواند نسبت به بهینه کردن تابع هدف خود اقدام کند. بطور خلاصه در مسأله BLP دو نوع تصمیم گیری وجود دارد:

۱- تصمیم گیری در اولین سطح توسط مدیر ارشد یا تصمیم گیرنده اول

۲- تصمیم گیری در دومین سطح توسط مدیر پایین تر یا تصمیم گیرنده دوم

مسأله BLP دارای مشخصات عمومی زیر است (ون و سو، ۱۹۹۱):

۱- در یک ساختار سلسله مراتبی، تصمیمات مدیران بر هم تأثیر می گذارد و از هم تأثیر می پذیرد.

۲- اجرای تصمیمات بصورت سلسله مراتبی از سطح بالا به سطح پایین جریان دارد. در حقیقت تصمیم گیرنده سطح پایین یا مدیر میانی تصمیمات خود را با توجه به تصمیمات مدیر سطح اول اتخاذ می کند.

۳- هر واحد تصمیم گیرنده تابع هدف خود را مستقل از سایر واحدها بهینه می کند، اما با توجه به ساختار مسأله تصمیم هر واحد بر واحدهای دیگر تأثیر دارد و از آنها تأثیر می پذیرد.

۳- اثرات بیرونی می توانند بر تصمیم گیری در هر دو واحد اثر کنند.

در سالهای اخیر BLP کاربردهای زیادی پیدا کرده است که می توان به کاربردهای آن در مدلسازی بخش کشاورزی (Candler, and Norton, 1978) (Candler, and McCarl, 1978)، سیاستگذاری حکومتی (Cassidy, Kirby, 1971)، (Kyland, 1975) و (Miljkovic, 2002)، سیستمهای اقتصادی (Wen, 1981;Desilva, 1978; Parraga, 1981; Wen and Xiang, 1988; Bard, Plummer, Friedman, 1983)، مدلهای مالی (and Sourie, 2000)، مسائل جنگی (Ben-Aye, Boyce & Blair, 1986, 1988; Marcotte, Yang and Yagar, 1994, 1995;Clegg, Smith, & Xiang, 2001) حمل و نقل (Ben-Aye, Boyce & Blair, 1986, 1988; Marcotte, Yang and Yagar, 1994, 1995;Clegg, Smith, & Xiang, 2001)مسأله فلوشاپ (Karlof and Wang, 1996)، مدیریت و برنامه ریزی زنجیره تأمین Gupta, and

[1] Multi-level programming

(Maranas, 2003)، مسأله فروشنده دوره گرد (Marcotte, Savard and Semet, 2004)، مسأله پسرک روزنامه فروش (Ji and Shao, 2006)، و برنامه ریزی تولید (Lukač, Šorić, and Rosenzweig, 2008) اشاره کرد.

برای حل مسأله BLP الگوریتمهای زیادی توسعه داده شده اند که می توان آنها را در چهار دسته زیر قرار داد:

۱) روشهایی بر اساس شمارش کلیه نقاط رأسی: از کارهای انجام شده در این زمینه می توان به مطالعات فالک (۱۹۷۳)، بیالاس و کاروان (۱۹۸۲ و ۱۹۸۳)، بارد (۱۹۸۳)، اونلو (۱۹۸۷)، بارد و موری (۱۹۸۸) و هانسن و همکاران (۱۹۹۲) اشاره کرد.

۲) روشهایی بر اساس شرایط کاروش- کان – تاکر (KKT): در این روش سعی می شود که مسأله سطح دوم بصورت شرایط KKT به مسأله سطح اول اضافه شود و در حقیقت مسأله بصورت تک هدفه درآید. وقتی که مسأله سطح دوم بصورت شرایط KKT به مسأله سطح اول اضافه می شود، مسأله بصورت غیر خطی در می آید که می توان با استفاده از متغیرهای صفر و یک مسأله را به فرم مسأله برنامه ریزی عدد صحیح مختلط تبدیل کرد. از کارهایی که در این زمینه انجام شده است می توان به مقالات بارد و موری (۱۹۹۰)، آناندالینگام و وایت (۱۹۹۰) اشاره کرد.

۳) رویکردهای فازی : در این رویکرد، معمولاً برای توابع هدف سطوح درجه عضویت تعریف می شود. از کارهای نجام شده در این زمینه می توان به مقالات ساکاوا و نیشیزاکی (۱۹۹۷) و ساهین و سیریت (۱۹۹۸) اشاره کرد.

۴) روشهایی بر اساس متاهیورستیک ها: با توجه به اینکه مسأله BLP یک مسأله NP-hard می باشد، الگوریتمهای متاهیورستیک کاربرد وسیعی در این حوزه دارند. از کارهای انجام شده در این زمینه می توان به مقالات ژیندریا و همکاران (۱۹۹۶)، ساهین و سیریت (۱۹۹۸)، حجازی و همکاران (۲۰۰۲) و لانا و همکاران (۲۰۰۷) اشاره کرد.

در این مقاله برای برای یافتن نقطه بهینه از روش شمارش نقاط رأسی استفاده می شود. چون فضای جواب نامحدب می باشد، با استفاده از الگوریتم ژنتیک فضای اطراف نقاط رأسی جستجو می شود تا نقطه بهینه پیدا شود. با استفاده از یک پارامتر کنترلی، شعاع فضایی را که باید جستجو شود کنترل می گردد تا از افزایش زمان حل مسأله اجتناب شود. توجه شود که در این مقاله سعی می شود تا آنجا که ممکن است مقدار تابع هدف سطح اول بهینه گردد. ساختار مقاله در ادامه بصورت زیر است: در بخش ۲ فرمول بندی ریاضی مسأله BLP ارائه می گردد. در بخش ۳ الگوریتم ژنتیک پیشنهادی و در بخش ۴ نتایج حاصل از الگوریتم ارائه شده است. نتیجه گیری و جمع بندی مقاله هم در بخش ۵ آمده است.

الف) مدل ریاضی مسأله برنامه ریزی دو سطحی

فرمول بندی مسأله BLP توسط آمات و مک کارل (۱۹۸۱) و همچنین کاندلر و تاون سلی (۱۹۸۲) صورت گرفت. اما فرمول بندی هایی هم توسط محققان دیگری ارائه شده است که می توان به بارد و فالک (۱۹۸۲)، بیالاس و کاروان (۱۹۸۲ و ۱۹۸۳) و بارد (۱۹۸۵) اشاره کرد. می توان مسأله BLP را در ساده ترین حالت بصورت زیر نوشت:

$$\begin{aligned}
&Max_y\, f(x,y) = c_1 x + d_1 y \\
&Max_y\, f(x,y) = c_2 x + d_2 y \\
&s.t. \\
&A_1 x + A_2 y \le b \\
&x, y \ge 0
\end{aligned} \tag{1}$$

$F(x,y)$: تابع هدف مدیر ارشد (تصمیم گیرنده سطح اول)

$f(x,y)$: تابع هدف مدیر میانی (تصمیم گیرنده سطح دوم)

x : بردار $n_1 \times 1$ که شامل متغیرهای تحت کنترل تصمیم گیرنده سطح اول می باشد.

y : بردار $n_2 \times 1$ که شامل متغیرهای تحت کنترل تصمیم گیرنده سطح دوم می باشد.

A_1 : ماتریس $m \times n_1$

A_2 : ماتریس $m \times n_2$

همانطور که ذکر شد مسأله BLP بصورتهای مختلف دیگری هم مورد بحث قرار گرفته است و شکل بالا تنها فرم ساده ای از این مسأله می باشد. همانگونه که اشاره شد، در این مقاله برای حل مسأله BLP از رویکرد اول یعنی شمارش نقاط رأسی استفاده می کنیم. برای یک جواب x داده شده، چون مقدار $c_2 x$ مقدار ثابتی است، بنابراین تابع هدف مسأله دوم فقط شامل جزء $d_2 y$ می باشد. فرض کنید که $Y(x)$ مجموعه جوابهای بهینه مسأله داخلی (مسأله سطح دوم) باشد:

$$\max_{y\in Q(x)} \tilde{f}(y) = d_2 y \quad (۲)$$

که $Q(x)$ بصورت زیر تعریف می شود:

$$Q(x) = \{y \mid A_2 y <= b - A_1 x\} \quad (۳)$$

$Q(x)$ فضای جواب تصمیم گیرنده سطح بالاتر را نشان می دهد و در حقیقت مجموعه ای از تصمیمات f را روی S نشان می دهد این تصمیمات بصورت زیر نوشته می شوند:

$$\Psi_f(S) = \{(x,y) \mid (x,y)\in S, y\in Y(x)\} \quad (۴)$$

فرض می شود که S و $Q(x)$ محدود و غیرتهی هستند. تعریف های شدنی بودن و بهینگی برای مسأله BLP بصورت زیر می باشند:

تعریف ۱: نقطه (x,y) شدنی است اگر $(x,y)\in\Psi_f(S)$

تعریف ۲: نقطه شدنی (x^*, y^*) بهینه است اگر برای همه $y^*\in Y(x^*)$ مقدار $c_1 x^* + d_1 y^*$ منحصر بفرد باشد و ضمناً برای همه جفتهای شدنی $(x,y)\in\Psi_f(S)$ داشته باشیم:

$$c_1 x^* + d_1 y^* \ge c_1 x + d_1 y \quad (۵)$$

بنابراین وقتی که متغیر تصمیم گیری x مساوی مقداری مانند $\bar{x}$ می شود، مسأله سطح دوم بصورت زیر در می آید که براحتی قابل حل است:

$$\begin{aligned} &\max\ f(y) = d_2 y \\ &s.t.\ \ A_2 y \le b - A_1\bar{x} \end{aligned} \quad (۶)$$

جواب بهینه این مسأله بصورت $\bar{y}$ می باشد و بنابراین $(\bar{x},\bar{y})\in\Psi_f(S)$. چون $Y(x)$ مجموعه جوابهای بهینه مسأله داخلی می باشد، بنابراین مسأله بیرونی را می توان بصورت زیر نوشت:

$$\begin{aligned} &Max\ F(x,y) = c_1 x + d_1 y \\ &s.t. \\ &A_1 x + A_2 y \le b \\ &y = Y(x) \\ &x, y >= 0 \end{aligned} \quad (۷)$$

اگر نقطه (x,y) جواب بهینه مسأله زیر باشد و همچنین $y = Y(x)$، آنگاه (x,y) یک نقطه بهینه است:

$$\begin{aligned} &Max\ F(x,y) = c_1 x + d_1 y \\ &s.t. \\ &A_1 x + A_2 y \le b \\ &x, y \ge 0 \end{aligned} \quad (۸)$$

ب) طراحی الگوریتم ژنتیک برای حل مسأله BLP

الگوریتم ژنتیک یکی از روشهای متاهیورستیک است که کاربردهای فراوانی در حل مسائل پیچیده یافته است. این روش تاکنون توسط چندین محقق برای حل مسأله BLP مورد استفاده قرار گرفته است که می توان به مقالات حجازی و همکاران (۲۰۰۲)، جی و شائو (۲۰۰۶)، کالوته و همکاران (۲۰۰۷) اشاره کرد. گامهای مهم برای اجرای الگوریتم ژنتیک عبارتند از: کدینگ جوابها و

تبدیل آنها به کروموزم، طراحی تابع برازندگی، عملگر تقاطع و عملگر جهش. این مراحل در الگوریتم پیشنهادی بصورت زیر می باشند:

کدینگ جوابها: در این تحقیق هر کروموزم بصورت رشته ای به طول $n_1 + n_2$ در نظر گرفته شده است که n_1 و n_2 تعداد متغیرهای سطح اول و دوم می باشند. هر کروموزم در حقیقت دربردارنده جوابهای مسأله می باشد که می توان آن را به شکل زیر نمایش داد:

$$x_1 \quad x_2 \quad \ldots \quad x_{n_1} \quad y_1 \quad y_2 \quad \ldots \quad y_{n2}$$

بنابراین هر کروموزم در حقیقت شامل یک رشته $n_1 + n_2$ تایی از اعداد اعشاری می باشد.

طراحی تابع برازندگی**:** برای طراحی تابع برازندگی از روش وانگ و همکاران (۲۰۰۲) استفاده می کنیم. همانطور که قبلاً گفته شد ابتدا مسأله سطح دوم حل شده و جوابهای آن بصورت $y = Y(x)$ در نظر گرفته می شود. فرض می کنیم با مقدار دلخواه $\bar{x}$ مدل (۶) را حل کرده و جواب $\bar{y}$ را بدست می آوریم. حال اگر (x_1,y_1) جواب مسأله (۸) باشد، d را می توان بصورت رابطه (۹) تعریف کرد:

$$d = \| (x_1, y_1) - (\bar{x}, \bar{y}) \|^{\beta} \tag{۹}$$

در رابطه (۹)، d مقدار جریمه و β پارامتری می باشد که با تغییر آن می توان ناحیه جستجو حول $(\bar{x}, \bar{y})$ را تغییر داد. مقدار β بسته به شرایط مسأله تنظیم شده و مقدار آن مساوی ۱ $(l_1 - norm)$ و ۲ $(l_2 - norm)$ در نظر گرفته می شود. در حقیقت اگر جوابهای حاصل از حل مدل (۸) شرط $y = Y(x)$ را ارضا نمودند، این جوابها بهینه هستند، در غیر اینصورت با استفاده از تابع جریمه از جوابهای نزدیک به جوابهای بهنه استفاده می کنیم. برای طراحی تابع برازندگی ابتدار مقدار q را بصورت زیر تعریف می کنیم:

$$q = \begin{cases} 1 & \| (x,y) - (\bar{x},\bar{y}) \| = 0 \\ 1 - \| \dfrac{(x,y) - (\bar{x},\bar{y})}{d} \| & 0 < \| (x,y) - (\bar{x},\bar{y}) \| \le d \\ 0 & \| (x,y) - (\bar{x},\bar{y}) \| > d \end{cases} \tag{۱۰}$$

که (x,y) نقاط یافته شده در هر مرحله و q درجه نزدیکی به گوشه های حاصل از حل مسأله سطح دوم می باشد. بنابراین تابع برازندگی را می توان بصورت زیر در نظر گرفت:

$$fitness(x_k) = F(x_k, y_k) \times q \tag{۱۱}$$

که $F(x_k, y_k)$ مقدار تابع هدف تصمیم گیرنده سطح اول می باشد.

عملگر تقاطع: اگر چه عملگرهای تقاطع زیادی در مطالعات و مقالات مختلف بکار رفته اند (برای نمونه می توان به عملگرهای تقاطع CX، PMX و... اشاره کرد)، ولی در این مطالعه با توجه به ساختار اعشاری کروموزمها از عملگرهای تقاطع ریاضیاتی استفاده می شود. در حقیقت برای تولید نوزادان از ترکیبات خطی والدین با هم استفاده می کنیم. اگر v_1 و v_2 والدهای انتخابی باشند، نوزادان v_1' و v_2' را می توان بصورت زیر بدست آورد:

$$\begin{aligned} v_1' &= \alpha_1 v_1 + \alpha_2 v_2 \\ v_2' &= \alpha_1 v_2 + \alpha_2 v_1 \end{aligned} \tag{۱۲}$$

که α_1 و α_2 اعداد تصادفی می باشند. برای اینکه حالت تنوع در فضای جواب را داشته باشیم، مقادیر α_1 و α_2 بصورت دو عدد تصادفی در فاصله $[0,10]$ انتخاب می شوند. شایان ذکر است که فاصله ذکر شده با استفاده از روش سعی و خطا حاصل شده است.

عملگر جهش: در این مقاله، برای اعمال عملگر جهش بر روی کروموزمها از یک روش ابتکاری استفاده شده است. در این روش ابتدا بزرگترین و کوچکترین مقدار ژن ها را در کروموزم پیدا می کنیم. سپس ضرایب تابع هدف سطح اول را در نظر می گیریم.

هر جا که مقدار این ضریب عددی غیر منفی باشد، در کروموزم بجای ژن معادل آن، بیشترین مقدار پیدا شده در کروموزم را قرار می دهیم. همچنین هر جا که مقدار ضریب تابع هدف سطح اول منفی باشد، کمترین مقدار ژن موجود در کروموزم را در ژن معادل آن قرار می دهیم. این عمل سبب می شود که همواره در توابع بیشینه سازی، بزگترین ضریب تابع هدف در بیشترین متغیر موجود در کروموزم ضرب شود. برای مثال فرض کنید که ضرایب تابع هدف بصورت رشته زیر هستند:

[4 -6 -1 2 -11]

همچنین کروموزم انتخابی بصورت زیر باشد:

[1 3 5 4 2]

بیشترین و کمترین مقدار ژن ها در این کروموزم ۵ و ۱ می باشند؛ بنابراین برای همه ژن های معادل با مقادیر غیر منفی در تابع هدف مقدار ۵ و برای همه ژن های معادل با مقادیر منفی در تابع هدف مقدار ۱ قرار می دهیم. پس کروموزم جدید به شکل زیر در می آید:

[5 1 1 5 1]

جستجوی محلی: برای بهینه سازی محلی، ابتدا بزرگترین ضریب موجود در تابع هدف سطح دوم را پیدا کرده و سپس بزرگترین مقدار ژن ها را در داخل هر کروموزم پیدا می کنیم. حال بزرگترین مقدار ژن داخل کروموزم را با توجه به موقعیت بزرگترین مقدار ضریب موجود در تابع هدف سطح دوم جابجا می کنیم. برای مثال ضرایب تابع هدف زیر را در نظر می گیریم:

[5 2 0 -1 1]

حال کروموزم زیر را در نظر می گیریم:

[3 8 0 1 4]

بیشترین مقدار ضریب تابع هدف سطح دوم در موقعیت اول می باشد؛ یعنی *max(f2)=5*. بزرگترین ژن موجود در کروموزم نیز در موقعیت ۲ می باشد که مقدار آن ۸ است. حال باید جای ۸ و ۳ را عوض کنیم تا عدد ۸ به موقعیت اول منتقل شود. همین روند را برای تابع هدف سطح اول نیز تکرار می کنیم.

با توجه به موارد بالا الگوریتم پیشنهادی را بصورت زیر می نویسیم:

گام اول– تولید جمعیت اولیه: برای تولید جمعیت اولیه مسأله زیر را حل می کنیم:

$$\begin{aligned} &Max \quad rx + d_2 y \\ &s.t. \\ &A_1 x + A_2 y \le b \\ &x, y \ge 0 \end{aligned} \qquad (۱۳)$$

که r یک بردار تصادفی می باشد. با تغییر r می توان به اندازه دلخواه جمعیت اولیه را تولید کرد. باید توجه کرد که در مدل بالا با توجه به محدودیت مسأله، شرط شدنی بودن همواره برای کروموزومها حفظ می شود و در ضمن شرط بهینگی هم با توجه به وجود مقدار $d_2 y$ برقرار می ماند. بعد از تولید جمعیت اولیه، جوابهای تولید شده به کروموزم تبدیل می شوند و مقدار تابع هدف سطح اول برای کروموزومها محاسبه می شود.

گام دوم– تقاطع**:** برای عمل تقاطع از روش بیان شده در قسمتهای قبلی استفاده می شود. ابتدا ۷۰٪ درصد بهترین کروموزومها را از جمعیت انتخاب کرده و سپس عدد تصادفی $p_c \in [0,1]$ را تولید می کنیم. این عدد درصد جمعیتی را نشان می دهد که باید عمل تقاطع روی آنها صورت گیرد. همانگونه که گفته شد اگر v_1 و v_2 والدهای انتخابی باشند، با استفاده از رابطه (۱۲) می توان نوزادان را بدست آورد.

گام سوم- جهش: برای اعمال عملگر جهش بر روی کروموزمها از یک روش ابتکاری استفاده می شود که در قسمتهای قبلی شرح داده شده است.

گام چهارم- جستجوی محلی: جستجوی محلی برای بهبود جوابها نیز در قسمتهای قبلی شرح داده شده است.

گام پنجم- انتخاب نسل بعدی: انتخاب نسل بعدی بر اساس مقدار تابع برازندگی صورت می پذیرد. برای محاسبه مقدار تابع برازندگی از رابطه (۱۱) استفاده می شود. این مقدار بر اساس q تغییر می کند که خود تابعی از d و β می باشد. چون d در ابتدای مسأله تنظیم می شود، بنابراین با تغییر β می توان جوابهای متنوعی را تولید کرد. بعد از مرتب کردن کروموزومها بر حسب مقدار تابع برازندگی، به اندازه جمعیت اولیه از آنها را به نسل بعدی منتقل می کنیم.

گام ششم- توقف: فرآیند در هر مرحله ۵۰۰ بار تکرار می شود. الگوریتم ۱۰ مرتبه اجرا شده و بهترین جواب به عنوان جواب خروجی در نظر گرفته می شود.

د) مثال های عددی

در این قسمت چند مثال عددی با الگوریتم پیشنهادی حل شده اند. این مسائل از مقالات و مطالعات پیشین گرفته شده اند.

مثال ۱) این مثال توسط ون و سو (۱۹۸۹ و ۱۹۹۱) ارائه شده است:

$$
\begin{aligned}
&\max_x F = -2x + 11y\\
&\max_y f = -x - 3y\\
&s.t.\quad x - 2y \le 4\\
&\qquad 2x - y \le 24\\
&\qquad 3x + 4y \le 96\\
&\qquad x + 17y \le 126\\
&\qquad -4x + 5y \le 65\\
&\qquad x + 4y \ge 8\\
&\qquad x, y \ge 0
\end{aligned}
$$

نتایج خروجی آنها بصورت زیر می باشد:

جدول شماره (۱): نتایج خروجی ون و سو (۱۹۹۱)

Solution Concept	x	y	$F(x,y)$	$f(x,y)$	α
LBLP	17.45	10.91	85.09	-50.18	-
Nash	0	12.23	134.52	-36.69	0.5
l_1-norm/max F	3.29	15.36	165.37	-50.18	1
l_2-norm	1.7	14.36	154.57	-44.78	0.87
l_∞-norm	0.65	13.52	147.36	-41.21	0.78
Kalai-Smorodinsky	0	11.76	129.39	-35.29	0.45
max f	0	7.73	85.09	-23.21	0

در روش پیشنهادی آنها تابع برازندگی بصورت زیر تعریف شده است:

$$\max = [F(x,y) - F(x^*,y^*)]^{\alpha} \times [f(x,y) - f(x^*,y^*)]^{1-\alpha}$$
$$s.t. \ (x,y) \in S \tag{۱۴}$$

همانطور که در جدول ۱ مشاهده می شود با تغییر α جوابهای متنوعی تولید می شوند. نتایج خروجی الگوریتم پیشنهادی برای مثال ۱ در جدول ۲ نشان داده شده است. همانگونه که قبلاً اشاره شد، در الگوریتم پیشنهادی مقدار β مساوی ۱ و ۲ در نظر گرفته می شود.

جدول شماره(۲): نتایج خروجی الگوریتم پیشنهادی برای مثال ۱

β	x	y	$F(x,y)$	$f(x,y)$
1	5.77	11.64	116.55	-40.71
2	5.33	17.03	176.71	-56.42

همانگونه که مشاهده می شود با تغییر مقدار β از ۱ به ۲ مقادیر توابع هدف هم تغییر می کنند. این جوابها بر حسب مقدار تابع هدف سطح اول از مقادیر کم شروع می شوند و به مقادیر بزرگ می رسند. بر حسب مقدار تابع هدف سطح اول، بهترین جواب در $\beta = ۲$ روی می دهد که (۱۷.۰۳ و ۵.۳۳)=(x,y) می باشد و در آن مقادیر توابع هدف (۵۶.۴۳− و ۱۷۶.۷۱)= (F^*,f^*) می باشند. از دیدگاه تابع هدف سطح اول این جوابها از بهترین جوابهای تولید شده در جدول ۱ بهتر می باشد. ولی همانگونه که مشاهده می شود این جواب، مقدار تابع هدف سطح دوم را بدتر کرده است. در حقیقت می توان با تغییر پارامتر کنترلی β به قیمت بدتر شدن تابع هدف سطح اول، به جواب بهتری برای تابع هدف سطح دوم دست یافت.

مثال ۲) این مثال توسط وایت و آناندالینگام (۱۹۹۰) و وانگ و همکاران (۲۰۰۳) ارائه شده است و بصورت زیر می باشد:

$$\max_x = x + 3y$$
$$\max_y = x - 3y$$
$$s.t. \ -x - 2y \le -10$$
$$x - 2y \le 6$$
$$2x - y \le 21$$
$$x + 2y \le 38$$
$$-x + 2y \le 18$$
$$x, y \ge 0$$

نتایج خروجی آنها در جدول ۳ نشان داده شده است:

جدول شماره (۳): نتایج خروجی وایت و آناندالینگام (۱۹۹۰) و وانگ و همکاران (۲۰۰۳)

			Wang et al.			Anandalingam & White		
Pop. size	*Pc*	*Pm*	*(x,y)*	*F*	*f*	*(x,y)*	*F*	*f*
50	0.7	0.15	(15.959,10.972)	48.875	-16.957	(16,1)	49	-17

نتایج خروجی مدل پیشنهادی برای مسأله فوق در جدول ۴ آمده است.

جدول شماره (۴): نتایج خروجی الگوریتم پیشنهادی برای مثال ۲

β	x	y	$F(x,y)$	$f(x,y)$
1	11.13	5.83	28.63	-6.37
2	12.66	12.66	50.63	-25.31

همانطور که مشاهده می شود از منظر تابع هدف سطح اول بهترین جواب در ۲ = β حاصل می شود که در آن نقطه بهینه (۱۲.۶۶ و ۱۲.۶۶)= (x,y) و مقادیر توابع هدف (۲۵.۳۱- و ۵۰.۶۳)= (F^*,f^*) می باشند. مانند مثال اول، در ۱ = β مقدار تابع هدف سطح دوم بهتر است، ولی این امر با بدتر شدن تابع هدف سطح اول حاصل شده است.

مثال ۳) در این مثال، مسأله ای با استفاده از تعداد بیشتری متغیر مورد بررسی قرار می گیرد. این مثال توسط هانسن و همکاران (۱۹۹۲) و وانگ و همکاران (۲۰۰۳) ارائه شده است:

$$\max_{x\geq 0} = 8x_1 + 4_2 - 4y_1 + 40y_2 + 4y_3$$

$$s.t.\quad x_1 + 2x_2 - y_3 \leq 1.3$$

$$\max_{y\geq 0} = -2y_1 - y_2 - 2y_3$$

$$s.t.\quad -y_1 + y_2 + y_3 \leq 1$$

$$4x_1 - 2y_1 + 4y_2 + y_3 \leq 2$$

$$4x_2 + 4y_1 - 2y_2 - y_3 \leq 2$$

نتایج خروجی آنها بصورت زیر می باشد:

جدول شماره (۵): نتایج خروجی هانسن و همکاران (۱۹۹۲) و وانگ و همکاران (۲۰۰۳)

			Wang et al.			Hansen et al.		
Pop. size	*Pc*	*Pm*	*(x,y)*	*F*	*f*	*(x,y)*	*F*	*f*
100	0.6	0.3	(0.533, 0.8, 0, 0.197, 0.8)	18.544	-1.797	(0.2, 0.8, 0 0.2, 0.8)	18.4	-1.8

نتایج خروجی الگوریتم پیشنهادی برای مسأله فوق در جدول ۶ آمده است.

جدول شماره(۶): نتایج خروجی الگوریتم پیشنهادی برای مثال ۳

β	x	y	$F(x,y)$	$f(x,y)$
1	(0,0)	(0.11,0.55,0)	21.70	-0.78
2	(0,0)	(0.187,0.55,0)	27.68	-2.01

همانگونه که مشاهده می شود الگوریتم پیشنهادی در این مثال که تعداد متغیرهای آن بیشتر می باشد، جوابهای بسیار خوبی تولید می کند. بدترین جواب این الگوریتم از جوابهای بدست آمده از وانگ و همکارن (۲۰۰۳) بهتر است. از نقطه نظر تابع هدف سطح اول، بهترین جواب الگوریتم برای این مثال در نقطه (۰ و ۰/۵۵ و ۰/۱۸۷ و ۰ و ۰)= (x_1,x_2,y_1,y_2,y_3) حاصل می شود که در آن ۲ = β و (۲/۰۱- و ۲۷/۶۸)= (F^*,f^*) می باشند.

٤- نتایج و بحث

در این تحقیق از روش الگوریتم ژنتیک برای حل مسأله BLP استفاده گردید. ابتدا مسأله BLP و کاربردهای آن مطرح شد و سپس روشهای موجود برای حل این مسأله مورد بررسی قرار گرفت که این روشها در چهار دسته روشهایی بر اساس شمارش نقاط رأسی، شرایط KKT، رویکردهای فازی و متاهیورستیها تقسیم بندی شدند. سپس یک الگوریتم پیشنهادی بر اساس ژنتیک برای حل مسأله BLP ارائه گردید. در این الگوریتم به منظور حل مسأله BLP از شمارش نقاط رأسی استفاده گردید. نتایج خروجی برای چند مثال مختلف نشان داد که این الگوریتم جوابهای خوبی را تولید می کند. وجود پارامتر کنترلی β برای کنترل جوابهای خروجی بر انعطاف پذیری این الگوریتم افزوده است.

٤- منابع

1. Anandalingam, G., and White, D.J. (1990). A solution for the linear static Stackelberg problem using penalty functions. IEEE Transactions on Automatic Control, 35: 1170–1173.

2. Bard, J.F. (1990). Some properties of the bilevel linear programming. Journal of Optimization Theory and Applications, 68: 371-378.
3. Bard, J.F., Plummer, J., and Sourie, J.C. (2000). A bilevel programming approach to determining tax credits for biofuel production. European Journal of Operational Research, 120(1): 30-46.
4. Ben-Ayed, O., and Blair, C.E. (1990). Computational difficulties of bilevel linear programming. Operations Research, 38: 556-560.
5. Ben-ayed, O., Boyce, D.E., and Blair, C.E. (1988). A general bilevel linear programming formulation of the network design problem. Transportation Research Part B: Methodological, 22: 311-318.
6. Ben-ayed, O., Boyce, D.E., Blair, C.E. (1988). Solving a real world highway design problem using bilevel linear programming. Faculty Working Paper No. 1463, College of Commerce and Business Administration, University of Illinois, Urbana-Champaign. 1988.
7. Bialas, W.F., and Karwan, M.H. (1982). On two-level optimization. IEEE Transactions on Automatic Control, 27: 211-214.
8. Bialas, W.F., and Karwan, M.H. (1983). Two-level linear programming. Managment Sciences, 30: 1004-1020.
9. Bracken, J., and McGill, J.T. (1973). Mathematical programs with optimization problems in the constraints. Operations Research, 21: 37-44.
10. Bracken, J., and McGill, J.T. (1974). A method for solving mathematical programs with nonlinear programs in the constraints. Operations Research, 22: 1097-1101.
11. Bracken, J., and McGill, J.T. (1974). Defense application of mathematical programs with optimization problems in the constraints. Operations Research, 22: 1086-1096.
12. Bracken, J., Falk, J. E., and Miercort, F.A. (1977). A strategic weapons exchange allocation model. Operations Research, 25: 968-976.
13. Brad, J.F. (1983). An efficient point algorithm for a linear two-stage optimization problem. Operations Research, 31: 670-684.
14. Brad, J.F. (1985). Geometric and algorithmic developments for a hierarchical planning problem. European Journal of Operational Research, 19: 372-383.
15. Brad, J.F., and Falk, J.E. (1982). An explicit solution to the multi-level programming problem. Computers & Operations Research, 9: 77-100.
16. Brad, J.F., and Moore, J.T. (1988). A branch and bound algorithm for the bilevel programming problem. Presented at the TIMSIORSA Joint National Meeting, Washington, DC, April.
17. Calvete, H.I., Gale´, C., and Mateo, P.M. (2008). A new approach for solving linear bilevel problems using genetic algorithms. European Journal of Operational Research, 188 (1), 14-28.
18. Candler, W., and McCarl, B. (1981). The potential role of multilevel programming in agricultural economics. American Journal of Agricultural. Economics, 63: 521-531.
19. Candler, W., and Norton, R.D. (1977). Multi-level programming. World Bank Development Research Center Discussion Paper No. 20, Washington, DC.
20. Candler, W., and Townsley, R. (1982). A linear two-level programming problem. Computers & Operations Research, 9: 59-76.
21. Cassidy, R.G., Kirby, M.J.L., anRaike, W.M. (1971). Efficient distribution of resources through three levels of government. Managment Sciences, 17: 462-473.
22. Clegg, J., Smith, M., Xiang, Y., and Yarrow, R. (2001). Bilevel programming applied to optimising urban transportation Transportation Research Part B: Methodological, 35(1): 41-70.

23. Desilva, A.H. (1978). Sensitivity formulas for nonlinear factorable programming and their application to the solution of an implicitly defined optimization model of US crude oil production. DSc. Dissertation, George Washington University, Washington, DC.
24. Falk, J.E. (1973). A linear max-min problem. Mathematical Programming, 5: 169-188.
25. Fortuny-amat, J., and McCarl, B. 1981. A representation and economic interpretation of a two-level programming problem. Journal of Operation Research Society, 32: 783-792.
26. Friedman, J. (1983). Oligopoly Theory. Cambridge University Press, Cambridge.
27. Gendreau, M., Marcotte, P., and Savard, G. (1996). A hybridTabu-Ascent algorithm for the linear bilevel programming problem. Journal of Global Optimization, 8: 217–33.
28. Gupta, A., and Maranas, C.D. (2002). Managing demand uncertainty in supply chain planning. Computers & Chemical Engineering 27: 1219-1227.
29. Hansen, P., Jaumard, B., and Savard, G. (1992). New branch-and-bound rules for linear bilevel programming.SIAM Journal on Science and Statistical Computing, 13(5): 1194-1217.
30. Hejazi, S.R., Memariani, A. Jahanshahloo, G., and Sepehri, M.M. (2002). Linear bilevel programming solution by genetic algorithm. Computers & Operations Research, 29: 1913–1925
31. Ji, X., and Shao, Z. (2006). Model and algorithm for bilevel newsboy problem with fuzzy demands and discounts Applied Mathematics and Computation, 172(1): 163-174.
32. Karlof, J., K., and Wang, W. (1996). Bilevel programming applied to the flow shop scheduling problem. Computers & Operations Research, 23(5): 443-451.
33. Kyland, F. (1975). Hierarchical decomposition in linear economic model. Management Sciences, 21: 1029-1039.
34. Lana, K.M., Wena, U.P., Shihb, H-S., and Leec, E.S. (2007). A hybrid neural network approach to bilevel programming problems. Applied Mathematics Letters, 20: 880–884.
35. Lukač, Z., Šorić, K., and Rosenzweig, V.V. (2008). Production planning problem with sequence dependent setups as a bilevel programming problem European Journal of OperationalResearch, 187(3): 1504-1512.
36. Marcotte, P. (1986). Network design problem with congestion effects: a case of bilevel programming. Mathematical Programming, 34: 142-162.
37. Marcotte, P., Savard, G., and Semet, F. 2004. A bilevel programming approach to the travelling salesman problem Operations Research Letters, 32(3): 240-248.
38. Miljkovic, D. (2002). Privatizing state farms in Yugoslavia. Journal of Policy Modeling, 24(2): 169-179.
39. Parraga, F.A. (1981). Hierarchical programming and applications to economic policy. PhD. Dissertation, Department of Systems and Industrial Engineering, University of Arizona, Tucson.
40. Sahin, KH, and Cirit, AR. (1998). A dual temperature simulated annealing approach for solving bilevel programming problems. Computers and Chemical engineering, 23: 11–25.
41. Sakava, M., Nishizaki, I., and Uemura, Y. (1997). Interactive fuzzy programming for multilevel linear programming problem. Computers & Mathematics with Applications, 36(2), 71–86.
42. Unlu, G. (1987). A linear bilevel programming algorithm based on bicriteria programming. Computers & Operations Research, 14: 173-1 79.
43. Wang, G., Wan, Z., and Wang, X. (2003). Solving Method for a Class of Bi-level

Linear Programming based on Genetic Algorithms. In www. Optimization Online.org/DB-HTML/2003/03/617.html.
44. Wen, U. P. (1981). Mathematical methods for multilevel linear programming. PhD. Dissertation, Depaltment of Industrial Engineering, State University of New York at Buffalo, New York.
45. Wen, U. P., and Hsu, S. T. (1989). A note on a linear bilevel programming algorithm based on bicriteria programming. Computers & Operations Research, 16: 79-83.
46. Wen, U. P., and Hsu, S. T. (1991). Linear bi-level programming problems- a review. Journal of Operation Research Society, 42(2): 125-133.
47. Wen, U. P., and Jiang, C.F. (1988). A multilevel programming approach in commission rate setting problem. Journal of the Chinese Institute of Industrial Engineers, 5: 43-49.
48. Wen, U. P., Hsu, S. T. (1991). Efficient solutions for the linear bi-level programming problem. European Journal of Operational Research, 62: 354-362.
49. Yang, H., and Yagar, S. (1994). Traffic assignment and traffic control in general freeway-arterial corridor systems. Transportation Research Part B: Methodological, 28(6): 463-486.
50. Yang, H., and Yagar, S. (1995). Traffic assignment and signal control in saturated road networks. Transportation Research Part A: Policy and Practice, 29(2): 125-139.