حالت کلی تساوی جریان کمان‌ها در شبکه‌های جریان چندکالایی

حسن صالحی فتح‌آبادی*- محمدعلی رعایت‌پناه
دانشکده ریاضی و علوم کامپیوتر، پردیس علوم، دانشگاه تهران، تهران، ایران
* مسئول مکاتبات- آدرس الکترونیکی: *salehi@khayam.ut.ac.ir*
(دریافت: ۸۵/۹/۲۱ ؛ پذیرش: ۸۶/۶/۲۰)

چکیده

در این مقاله مسئله شبکه جریان چند کالایی با جریان‌های مساوی روی کمان‌های معین مطرح می‌شود. قیود تساوی ایجاب می‌کند که جریان کمان‌های عضو زیر مجموعه‌های معین، برای کالاهای مجزا و مشخص مساوی باشند. به‌منظور حل این مسئله ابتدا با استفاده از الگوریتم تخصیص ظرفیت یک جواب شروع، برای مسئله به‌وجود می‌آوریم. سپس با استفاده از تکنیک تخفیف لاگرانژین روی قیود کلی یک کران پایین، و بعد با استفاده از الگوریتم سیمپلکس شبکه محاط‌شده، یک کران بالا را برای مقدار تابع هدف محاسبه می‌کنیم آنگاه کران‌های بالا و پایین را تعدیل کرده تا به جواب بهینه یا جواب بسیار نزدیک به بهینه برسیم.

واژه‌های کلیدی: شبکه‌های جریان، جریان‌های چندکالایی، تخفیف لاگرانژین، الگوریتم سیمپلکس شبکه، جریان با هزینه محدب.

مقدمه

شبکه $G=(N,A)$ را در نظر بگیرید که N مجموعه گره ها با n عضو و A مجموعه کمان‌ها با m عضو است . تعریف می‌کنیم x_{ij}^k مقدار کالای نوع kام روی کمان (i,j) و c_{ij}^k و u_{ij}^k به‌ترتیب هزینه انتقال یک واحد از کالا و ظرفیت این کمان برای کالای kام باشند.

c^k, x^k را بردار جریان و بردار هزینه برای کالای kام و b^k را بردار عرضه و تقاضای کالای نوع kام می گیریم. تعداد انواع کالا در شبکه برابر K و u_{ij} را ظرفیت کلی کمان (i,j) در نظر می‌گیریم (به‌طوری‌که مجموع تمام جریان‌های روی کمان (i,j) از u_{ij} بیشتر نشود).

زیرمجموعه‌های $A_{kr} \subseteq A$ ، $r=1,2,...,S_k$ و $k=1,2,...,K$ با کمان مرجع $(p,q)_{kr} \in A_{kr}$ داده شده‌اند. این مجموعه‌ها شامل کمان‌هایی هستند که لازم است جریان نوع kام روی کمان‌های هر یک از آنها مساوی مقدار جریان کمان مرجع آن مجموعه باشد. حال با توجه به تعاریف داده شده حالت کلی مسئله چند کالایی با جریان مساوی را می‌توان به‌صورت فرمول زیر بیان کرد:

P1

$$\text{Min} \sum_{1 \le k \le K} c^k x^k \tag{1}$$

S.t

$$\sum_{1 \le k \le K} x_{ij}^k \le u_{ij} \quad , \quad (i,j) \in A \tag{1.a}$$

$$A^k x^k = b^k \quad , \quad k=1,...,K \tag{1.b}$$

$$x_{ij}^k = x_{(p,q)kr} \quad (i,j) \in A_{kr}, k=1,...,K, \ r=1,2,...,S_k \tag{1.c}$$

$$0 \le x_{ij}^k \le u_{ij}^k \quad (i,j) \in A \quad , k=1,...,K \tag{1.d}$$

قیود (1.a) قیود ظرفیت کلی، قیود (1.b) قیود تعادل جریان، قیود (1.c) قیود تساوی جریان و قیود (1.d) قیود ظرفیت مربوط به کالای نوع k ام و A^k ماتریس وقوع گره-کمان کالای kام می‌باشند.

ابتدا مسئله شبکه جریان مساوی در حالت تک کالایی توسط علی و همکارانش مطرح شد و سپس مسئله جریان با حداقل هزینه،- وقتی که جریان روی جفت‌های مشخصی از کمان‌ها مساوی بود- را حل کردند.(Ali *et al.*, 1988) . سپس آهوجا و همکارانش مسئله جریان مساوی ساده - که جریان روی یک زیر مجموعه دلخواهی از کمان‌ها مساوی بود- را مورد بررسی قرار دادند.(Ahuja *et al.*, 1999) . بالاخره کالوت مسئله جریان مساوی کلی را بیان کرد که جریان روی چند زیر مجموعه از کمان‌ها مساوی است (Calvete 2003). به‌طور کلی همه این بررسی‌ها شبکه‌های جریان تک کالایی را مورد توجه قرار می‌دادند در حالی‌که بسیاری از سیستم‌های واقعی، به‌خصوص سیستم‌های حمل‌ونقل، بیش از یک نوع کالا در شبکه، در جریان می‌باشد.

در این مقاله سعی بر این است که جریان‌های مساوی روی زیرمجموعه‌هایی از کمان‌ها برای شبکه جریان چند کالایی تجزیه و تحلیل شده و الگوریتمی برای حل آن ارائه شود. مسئله P1 را طی چهار مرحله کلی زیر بررسی و حل می‌کنیم:

مرحله اول : ساده سازی مسئله و به‌دست آوردن یک نقطه شروع

پیشرفته:

مرحله دوم : استفاده از تخفیف لاگرانـژین (lagrangian Relaxtion) و به‌دست آوردن یک کران پایین برای مسئله:

مرحله سوم : استفاده از الگوریتم سیمپلکس شبکه محاط شده (Embedded Network) برای به‌دست آوردن یک کران بالا برای مسئله:

مرحله چهارم : تکرار مرحله‌های دوم و سوم تا شرط توقف (نزدیکی کران‌های بالا و پایین) به‌دست آید، (بیان الگوریتم اصلی).

در بخش‌های بعدی به بیان جزئیات اجرای مراحل اول تا چهارم به‌ترتیب زیر می‌پردازیم. در بخش ۲ به نحوه ساده‌سازی مسئله و محاسبه یک جواب اولیه و در بخش ۳ روش تخفیف لاگرانژین را به‌کار گرفته و یک کران پایین برای مقدار تابع (۱) محاسبه می‌کنیم. در بخش ۴، کران بالای مقدار تابع (۱) کرده و در نهایت در بخش ۵ به بیان جزئیات الگوریتم اصلی می‌پردازیم.

ساده‌سازی و به‌دست آوردن یک نقطه شروع پیشرفته :

برای ساده سازی مسئله ابتدا قیود تساوی را در بقیه قیود و تابع هدف جایگذاری می‌کنیم: یعنی قیود (1.c) را در قیود (1.d) و (1.b) و (1.a) و تابع هدف جایگذاری کرده و تعریف می‌کنیم:

$$A_k = A - \cup_{r=1}^{s_k} A_{kr} \quad , \quad k = 1,2,...,K$$

A_k زیر مجموعه‌ای از کمان‌های شبکه است که لازم نیست جریان روی کمان‌های آن در قیود تساوی (قیود 1.c) صدق کند.

از آنجایی که جریان روی کمان‌های مجموعه A_{kr} باید با هم رابطه تساوی داشته باشد پس جریان روی کمانها ی مجموعه A_{kr} در هر تکرار یا صفر خواهد شد یا برابر مقدار $x_{(p,q)_{kr}}$، که هزینه انتقال یک واحد جریان روی کمان‌های مجموعه A_{kr} برابراست با:

$$c_r^k = \sum_{(i,j)\in A_{kr}} c_{ij}^k \ , \ r = 1,2,...,s_k \quad k = 1,2,...,K$$

و حداکثر جریان مجاز روی کمان‌های مجموعه A_{kr} برابر است با :

$$u_r^k = \min\left\{ u_{ij}^k \middle| (i,j)\in A_{kr} \right\} \ , \ r = 1,2,...,S_k \quad k = 1,2,...,K \ .$$

یعنی $0 \le x_{(p,q)kr} \le u_r^k$.

ماتریس A دارای nk سطر و mK ستون است و به شکل زیر می‌باشد.

$$A = \begin{bmatrix} A^1 & & & 0 \\ & A^2 & & \\ & & \ddots & \\ 0 & & & A^K \end{bmatrix}$$

A^k، $k = 1,2,...,K$ ماتریس وقوع گره-کمان مربوط به کالای نوع kام است. با توجه به تعریف ماتریس وقوع گره -کمان، تعداد عناصر ۱+ در سطر iام معرف تعداد کمان‌های خارج‌شونده از گره i (درجه خروجی گره i) تعداد عناصر ۱- در سطر iام معرف تعداد کمان‌های واردشونده به گره i (درجه ورودی گره i) است. اگر A_{ij}^k ستون مربوط به کمان (i, j) و کالای k باشد، با توجه به تعریف ماتریس وقوع گره-کمان، درایه iام ۱+ و درایه jام ۱- و بقیه درایه ها صفر می‌شود. سپس اینگونه تعریف می کنیم:

$$A_r^k = \sum_{(i,j)\in A_{kr}} A_{ij}^k \quad , \quad k = 1,2,...,K \ , \quad r = 1,2,...,S_k$$

در واقع A_r^k بردار ستونی است که از جمع بردارهای ستونی ماتریس A متناظر با کمان‌های (i, j) در A_{kr} به‌دست آمده است. درایه iام بردار ستونی A_r^k متناظر است با مجموع ۱+ و۱-، در سطر iام که ۱+ مربوط به ضریب متغیر x_{ij}^k (یعنی کمانی از A_{kr} که از گره i خارج شده است) و۱- مربوط به ضریب متغیر x_{ji}^k (یعنی کمانی از A_{kr} که به گره i وارد شده است) می‌باشد. بنابراین اگر $out_r^k(i)$ را تعداد کمان‌های در A_{kr} خارج‌شونده از گره i و $in_r^k(i)$ را تعداد کمان‌های در A_{kr} را وارد شونده به گرهi و a_r^{ki} را iامین مؤلفه از A_r^k نمایش دهیم، داریم:

$$a_r^{ki} = out_r^k(i) - in_r^k(i)$$

$$k = 1,2,...,K \ , \ r = 1,2,...,s_k \ , \ i = 1,2,...,n,$$

با توجه به نمادهای بالا به آسانی لم زیر ثابت می شود.

لم۱ :

a) $-(n-1) \le a_r^{ki} \le (n-1)$

b) $\sum_{i\in N} a_r^{ki} = 0$

c) $-m_{kr} \le \sum_{i\in N'} a_r^{ki} \le m_{kr}$

که N' زیر مجموعه دلخواه از مجموعه گره‌ها و m_{kr} تعداد کمان‌ها در A_{kr} می‌باشد.

اثبات: صحت بندهای a و b به‌سادگی قابل قبول می‌باشد. بنابراین تنها به اثبات بند c می‌پردازیم.

باتوجه به تعریف a_r^{ki} داریم:

$$\sum_{i\in N'} a_r^{ki} = \sum_{i\in N'} out_r^k(i) - in_r^k(i) = \sum_{i\in N'} out_r^k(i) - \sum_{i\in N'} in_r^k(i)$$

پس برای گره‌های $i \in N'$ حداکثر ممکن است تمام کمان‌ها در A_{kr} ازاین گره‌ها خارج‌شونده باشند، یعنی:

$$\sum_{i\in N'} in_r^k(i) = 0 \quad و \quad \sum_{i\in N'} out_r^k(i) = m_{kr}$$

$$\Rightarrow \quad \sum_{i\in N'} a_r^{ki} = m_{kr} \quad (1)$$

و یا برای گره‌های $i \in N'$ حداکثر ممکن است تمام کمان‌ها در A_{kr} به این گره‌ها وارد شونده باشند، یعنی:

$$\sum_{i\in N'} in_r^k(i) = m_{kr} \qquad و \sum_{i\in N'} out_r^k(i) = 0$$

$$\Rightarrow \quad \sum_{i\in N'} a_r^{ki} = -m_{kr} \qquad (2)$$

از روابط ۱و۲ نتیجه مطلوب حاصل می‌شود.

پس از جای‌گذاری قیود (1.c) در قیود (1.b) و استفاده از نمادهای قبلی، قید گره i مربوط به کالای kام برابر:

$$\sum_{\{j:(i,j)\in A_k\}} x_{ij}^k - \sum_{\{j:(j,i)\in A_k\}} x_{ji}^k + \sum_{r=1}^{s_k} a_r^{ki} x_{(p,q)kr} = b_i^k \qquad i=1,\dots,n \qquad (3)$$

خواهد گردید. ماتریس D^kی به‌وجود آمده توسط n سطر بالا(3) را ماتریس ضرایب کالای k در حالت تساوی می‌نامیم. توجه کنید که ماتریس D^kمتشکل از دو زیر ماتریس به شکل قیود کلاسیک جریان شبکه (ماتریس‌های یک هنگ unimodular) و ماتریس متناظر با کمان‌های با جریان مساوی است. ماتریس D^k دارای n سطر و $|A_k|+s_k$ ستون است. به‌ازای هر کالا چنین ماتریسی به‌دست می‌آید. بنابراین ماتریس D را به‌صورت زیر تعریف می‌کنیم :

$$D = \begin{bmatrix} D^1 & & 0 \\ & D^2 & \\ & & \ddots \\ 0 & & D^K \end{bmatrix}$$

به‌منظور جایگذاری قیود تساوی (قیود 1.c) را در قیود ظرفیت کلی (قیود 1.d) جایگذاری کنیم. برای بازنویسی مجدد این قیود دو نماد زیر را اینگونه تعریف می‌کنیم که:

$$\varepsilon_{ij}^k = \begin{cases} 1 & (i,j)\in A_k \\ 0 & \text{O.W} \end{cases}$$

و

$$\delta_{r\,ij}^k = \begin{cases} 1 & (i,j)\in A_{kr} \\ 0 & \text{O.W} \end{cases}$$

بنابراین قیود ظرفیت کلی (قیود1.d) به شکل زیر در می‌آید:

$$\sum_{k=1}^{K}\sum_{r=1}^{s_k} \delta_{r\,ij}^k x_{(p,q)_{kr}} + \sum_{1\le k\le K} \varepsilon_{ij}^k x_{ij}^k \le u_{ij} \qquad (i,j)\in A$$

ماتریس حاصل از جایگذاری قیود (1.c) در (1.a) را B می‌نامیم.

اکنون مسئله‌ی P1 را می‌توان به‌صورت زیر نوشت:

P2:

$$\sum_{k=1}^{K}\sum_{(i,j)\in A_k} c_{ij}^k x_{ij}^k + \sum_{k=1}^{K}\sum_{r=1}^{s_k} c_r^k x_{(p,q)kr} \qquad (2)$$

S.t

$$\sum_{k=1}^{K}\sum_{r=1}^{s_k} \delta_{r\,ij}^k x_{(p,q)kr} + \sum_{1\le k\le K} \varepsilon_{ij}^k x_{ij}^k \le u_{ij} \quad (i,j)\in A \qquad (2.a)$$

$$D^k x^k = b^k \qquad k=1,2\dots,K \qquad (2.b)$$

$$0\le x_{ij}^k \le u_{ij}^k \qquad (i,j)\in A_k\;, \quad k=1,2,\dots,K \qquad (2.c)$$

$$0\le x_{(p,q)kr} \le u_r^k \qquad k=1,2,\dots,K, r=1,\dots,s_k \qquad (2.d)$$

حال می‌خواهیم یک نقطه شروع پیشرفته را برای مسئله P2 پیدا کنیم، این کار را با استفاده از روش تخصیص ظرفیت انجام می‌دهیم . (Mc Bride *et al.*1997) در ابتدا به ازاء کمان (i,j) و جریان k متغیر تخصیص ظرفیت y_{ij}^k ($y_{(p,q)kr}$) را نسبت می دهیم به‌طوری‌که:

$$x_{(p,q)kr} \le y_{(p,q)kr}$$
$$x_{ij}^k \le y_{ij}^k$$
$$\sum_{k=1}^{K}\sum_{r=1}^{s_k} \delta_{r\,ij}^k y_{(p,q)_{kr}} + \sum_{1\le k\le K} \varepsilon_{ij}^k y_{ij}^k \le u_{ij} \qquad (i,j)\in A$$

روش تخصیص ظرفیت:

اگر در مسئله P2 از قیود ظرفیت کلی (یعنی قیود 2.a) صرف نظر کنیم، مسئله P2 به K مسئله جریان مساوی کلی برای کالاهای متفاوت ومجزا تبدیل می‌شود که جریان بهینه هرکدام را می‌توان با استفاده از الگوریتم سیمپلکس شبکه برای مسئله جریان مساوی کلی به‌دست آورد .(Calvete, 2003) مسئله قابل بحث در اینجا چگونگی مقداردهی به متغیرهای تخصیص ظرفیت است به‌طوری‌که زیر مسئله نشدنی نشود همان‌طور که می‌دانیم در روش زیر- گرادیان(Subgradient) در هر تکرار با استفاده از روش تصویر(Projection)، مقدارها طوری تخصیص می یابند که زیر مسئله نشدنی نشود(Ahuja *et al.*, 1993). ولی در اینجا ما به‌جای روش تصویر از الگوی جریان با هزینه محدب(Convex Cost flow) استفاده می‌کنیم(Mcbride *et al.*, 1997; Ahuja *et al.*, 1993) . در واقع در هر تکرار متغیرهای X را باید طوری مقداردهی کنیم تا قیدهای (2.a) برقرار شوند. این شرط را با توجه به تعریف متغیرهای تخصیص ظرفیت و الگوی هزینه محدب، ایجاد می‌کنیم. یعنی مجموع جریان‌ها روی هر کمان می تواند از ظرفیت کلی بیشتر شوند اما این کار با یک جریمه بالا امکان‌پذیر است. مسئله تخصیص ظرفیت زیر را در نظر بگیرید:

P3:

$$Z(y)=\min\; CX+M(X-Y)^+ \qquad (3)$$

S.t

$$D^k x^k = b^k \qquad k=1,2\dots,K \qquad \begin{aligned} &0\le x_{ij}^k \le u_{ij}^k \\ &0\le x_{(p,q)kr} \le u_r^k \end{aligned}$$

$$(i,j)\in A_k \;, \quad k=1,2,\dots,K$$

$$k=1,2,\dots,K \;, \quad r=1,2,\dots,s_k$$

که CX تابع هدف مسئله p2 و M یک مقدار بسیار بزرگ است و درایه‌های بردار $(X-y)^+$ برابر است با $\mathrm{Max}(x_{ij}^k-y_{ij}^k,0)$ برای هر $(i,j)\in A_k$ و $\max(x_{(p,q)kr}^k-y_{(p,q)kr}^k,0)$ برای هر $r=1,2\dots,S_k$ و $k=1,2,\dots K$.

الگوریتم تخصیص ظرفیت، ابتدا به هر متغیر تخصیص ظرفیت متناظر با هر کمان, یک ظرفیت اختصاص می‌دهد (ظرفیت اولیه به‌عنوان کران بالای کمان منظور می‌شود) سپس با الگوریتم سیمپلکس شبکه برای مسئله جریان مساوی کلی (Calvete 2003) یک جریان بهینه به‌دست می آید. اگر این جریان حاصل، قیدهای کران بالای کمان‌ها (قیود2.a) را برآورده کند، الگوریتم متوقف و در غیر این صورت ظرفیت کمان‌ها را به‌هنگام کرده و ادامه می‌دهد. این روند به دفعات معینی ادامه می‌یابد تا یک جواب قابل قبول به‌دست آید.

الگوریتم تخصیص ظرفیت را با در نظر گرفتن یک n ثابت در زیر بیان می‌کنیم.

الگوریتم تخصیص ظرفیت:

(۱) مقداردهی اولیه:

ابتدا مقدار اولیه y را برابر با کران بالای کمان‌ها یعنی $y^0=U$ می‌گیریم. سپس به‌وسیله الگوریتم سیمپلکس شبکه برای مسئله جریان مساوی کلی $Z(y^0)$ را حل می‌کنیم. جواب بهینه به‌دست آمده رابا X^0 نمایش داده و $t=0$ می‌گیریم.

(۲) محاسبه γ^t:

تعریف می‌کنیم $\gamma^t= BX^t$ ، در واقع γ_{ij}^t مجموع جریان‌های X^t روی کمان (i,j) در مرحله tام است.

(۳) به‌هنگام کردن متغیرهای تخصیص ظرفیت:

متغیرهای تخصیص ظرفیت را باید به‌گونه‌ای به‌هنگام کرد که در تکرارهای بعد قیود ظرفیت کلی برقرار باشند.

بنابراین اگر $\gamma_{ij}^t \le u_{ij}$ باشد، با ظرفیت تخصیص یافته قبلی، قید کلی کمان (i,j) برقرار است، اما اگر $\gamma_{ij}^t > u_{ij}$ باشد به‌معنای آن است که مجموع جریان‌های روی کمان(i,j) از حد مجاز تجاوز می‌کند پس باید متغیرهای تخصیص ظرفیت را بنحوی تعدیل کرد. این تعدیل بستگی به مقدار جریان کالای نوع kام روی کمان(i,j) در مرحله tام یعنی x_{ij}^{kt} دارد. حالت $x_{ij}^{kt}=0$ ، در مقدار γ_{ij}^t بی تاثیراست لذا ظرفیت تخصیص یافته متناظر با آن یعنی y_{ij}^{kt} تغییر نمی کند.اما حالت $x_{ij}^{kt}>0$ ، در مقدار γ_{ij}^t تاثیرگذار است وباید y_{ij}^{kt} متناظر با آن کاهش یابد .برای این منظور متغیر $z_{ij}^{k(t+1)}$ را به صورت زیر تعریف می‌کنیم:

$$z_{ij}^{k(t+1)}=\begin{cases}\dfrac{x_{ij}^{kt}u_{ij}}{\gamma_{ij}^t} & \text{if} \quad x_{ij}^{kt}>0 \quad \text{and} \quad \gamma_{ij}^t>u_{ij}\\ y_{ij}^{kt} & \text{if} \quad x_{ij}^{kt}=0 \quad \text{or} \quad \gamma_{ij}^t\le u_{ij}\end{cases}$$

حال متغیرهای تخصیص ظرفیت کالای kام متناظر باکمان(i,j) در مرحله بعد یعنی $y_{ij}^{k(t+1)}$ را به‌صورت زیر به‌هنگام می‌کنیم:

$$y_{ij}^{k(t+1)}=\min\left\{y_{ij}^{kt}, z_{ij}^{k(t+1)}\right\}$$

(۴) حل مسئله بامتغیرهای تخصیص ظرفیت جدید و چک کردن شرط توقف:

$Z(y^{t+1})$ را با الگوریتم سیمپلکس شبکه برای مسئله جریان مساوی کلی حل کرده ومقدار بهینه x_{ij}^{t+1} را به دست می آوریم. اگر $BX^{t+1}\le U$ یا t>n الگوریتم خاتمه می‌یابد در غیر این صورت متغیر t را یک واحد اضافه کرده (t=t+1) و از مرحله ۲ الگوریتم تکرار می‌کنیم.

توجه کنید که این الگوریتم یک روش ابتکاری بوده و ممکن است به جواب شدنی مسئله P2 منجر نشود اما با توجه به شرط پایانی این الگوریتم امکان حذف یک جواب شدنی به‌طور اتفاقی در یک تکرار وجود ندارد. بنابراین این الگوریتم یک جواب شدنی یا یک جواب نزدیک به شدنی بودن را می‌دهد، جواب نهایی به‌دست آمده از الگوریتم تخصیص ظرفیت را x_0 می نامیم.

مسئله P3 یک مسئله با جریان هزینه محدب است. جواب پایه‌ای برای مسائل برنامه‌ریزی خطی با هزینه محدب در مقابل مسائل با هزینه خطی یک جواب توسعه یافته است. یعنی اینکه متغیرهای غیر پایه‌ای می‌توانند در نقاط انفصال (نقطه ای بین کران بالا و پایین) باشند. درحالیکه در روش استاندارد سیمپلکس، متغیرهای غیر پایه‌ای باید حتماً در کران بالا یا پایین خود باشند. بنابراین جواب x_0 اگر چه یک جواب پایه‌ای مسئله p3 است اما یک جواب پایه ای برای مسئله p2 نیست، زیرا متغیرهای غیر پایه ای x_0 می‌توانند در نقاط انفصال باشند (Ahuja *et al*., 1993).

استفاده از تخفیف لاگرانژین و به‌دست آوردن یک کران پایین برای مسئله

تکنیک تخفیف لاگرانژین را برای مسئله چند کالایی با جریان مساوی (مسئلهp2) به‌کار می‌بریم و به هر قید 2.a ضریب لاگرانژین نامنفی w_{ij} را نسبت می‌دهیم W را بردار m تایی از ضرایب لاگرانژین درنظر می‌گیریم، زیر مسئله لاگرانژین عبارت است از:

P4:

$$l(W)=\mathrm{Min}\sum_{k=1}^{K}\sum_{(i,j)\in A_k} c_{ij}^k x_{ij}^k+\sum_{k=1}^{K}\sum_{r=1}^{S_k} c_r^k x_r^k+W(BX-U) \quad (4.a)$$

S.t

$$D^k x^k=b^k \qquad k=1,2\ldots,K \qquad (4.b)$$

$$0\le x_{ij}^k\le u_{ij}^k \qquad (i,j)\in A_k,\quad k=1,2,\ldots,K \qquad (4.c)$$

$$0\le x_{(p,q)kr}\le u_r^k \quad r=1,2,..,s_k\ ,\ k=1,2,\ldots,K \qquad (4.d)$$

در ابتدا در مسئله P4 در قسمت (4.a) ، قسمت W(BX-U) را باز کنیم. (می دانیم ماتریس B از جایگذاری قیود 1.c در1.a به‌دست آمده

است).

$$W(BX-U)=\sum_{k=1}^{K}\sum_{(i,j)\in A_k} w_{ij}x_{ij}^k-\sum_{(i,j)\in A} w_{ij}u_{ij}$$
$$+\sum_{k=1}^{K}\sum_{r=1}^{s_k}\Big(\sum_{(i,j)\in A_{kr}} w_{ij}\Big)x_{(p,q)kr}$$

به‌عبارت دیگر سطر (4.a) را می‌توان به‌صورت زیر نوشت:

$$\sum_{k=1}^{K}\sum_{(i,j)\in A_k}(c_{ij}^k+w_{ij}^k)x_{ij}^k+\sum_{k=1}^{K}\sum_{r=1}^{s_k}\Big(\big(\sum_{(i,j)\in A_{kr}} w_{ij}\big)+c_r^k\Big)x_{(p,q)kr}$$
$$-\sum_{(i,j)\in A} w_{ij}u_{ij}$$

برای یک ماتریس ضرایب لاگرانژین ثابت داده شده W ، مقدار $-\sum_{(i,j)\in A} w_{ij}u_{ij}$ در تابع هدف مسئله یک مقدار ثابت است. بنابراین در مسئله می‌نیمم سازی با w_{ij} ثابت می‌توان از آن صرف نظر کرد.

بنا به این روش، مسئله p4 به K مسئله مجزای جریان مساوی کلی تبدیل می شود که هزینه هر متغیر که در قیود تساوی نیست برابر $c_{ij}^k+w_{ij}^k$ و هزینه هر متغیر که در قیود تساوی است برابر با: $\big(\sum_{(i,j)\in A_{rk}} w_{ij}\big)+c_r^k$) می‌باشد. بنابراین حل زیر مسئله لاگرانژین از دو مرحله تشکیل شده است.

مرحله ۱. حل مسئله جریان مساوی کلی برای هر کالا:

اگر ماتریس ضرایب کالای k در حالت تساوی را با D^k و بردار نیازمندی کالای kام را با b^k نمایش دهیم ، برای ماتریس ثابت داده شده W^p در تکرار pام $z^k(W^p)$ را به‌صورت زیر تعریف کنیم:

P5:

$$z^k(W^p)=Min\sum_{(i,j)\in A_k}(c_{ij}^k+w_{ij}^p)x_{ij}^p$$
$$+\sum_{r=1}^{S_k}\Big(\big(\sum_{(i,j)\in A_{kr}} w_{ij}^p\big)+c_r^k\Big)x_{(p,q)kr}$$

S.t (5)

$$D^k x^k = b^k \qquad (5.a)$$
$$0\le x_{ij}^k\le u_{ij}^k \qquad (i,j)\in A_k \qquad (5.b)$$
$$0\le x_{(p,q)kr}\le u_r^k \qquad r=1,2,\dots,s_k \qquad (5.c)$$

حال با استفاده از الگوریتم سیمپلکس شبکه برای مسئله جریان مساوی کلی مسئله p5 را برای یک بردار ثابت داده شده Wp حل می‌کنیم .حل مسئله p4 برای یک ماتریس ثابت داده شده W^p برابر است با:

$$Z(W^p)=\sum_{k=1}^{K} z^k(W^p)$$

مرحله ۲. به‌هنگام کردن بردار W^p :

اگر v_{ij}^{kp} , $v_{(p,q)kr}^{p}$ را جواب بهینه مسئله جریان مساوی کلی (مسئله p5) برای کالای kام در نظر بگیریم برای به‌هنگام کردن w_{ij}^{p+1} طبق فرمول زیرگرادیان داریم:

$$w_{ij}^{p+1}=[w_{ij}^p+\theta_p(\sum_{1\le k\le K} v_{ij}^{kp}-u_{ij})]^+ \qquad \forall(i,j)\in A$$

که در این فرمول $v_{ij}^{kp}=v_{(p,q)kr}^{p}$, $\forall\,(i,j)\in A_{kr}$ و $r=1,2\dots,S_k\,,k=1,2,\dots,K$, است. و θ_p طول گام است.

در اینجا دو بحث به‌وجود می‌آید. بحث اول مربوط به مقدار اولیه W و بحث دوم مربوط به شرط توقف برای روش تکراری لاگرانژین است.

برای مقدار اولیه W جواب به‌دست آمده از الگوریتم تخصیص ظرفیت، یعنی x_0 را در نظر بگیرید همانطور که بیان شد x_0 جواب پایه ای مسئله p3 است. ولی جواب پایه ای مسئله p2 نیست، زیرا متغیرهای غیر پایه ای x_0 می‌توانند در نقاط انفصال باشند. ما جواب y^0 را از جواب x_0 بدین صورت می‌سازیم که برای متغیرهای غیرپایه‌ای x_0 که در نقاط انفصالشان هستند y^0 متناظر را در کران پایین قرار داده و بقیه متغیرها را برابر x_0 درنظربگیریم سپس $W^0=By^0-U$ می‌گیریم. بدین‌ترتیب مقدار اولیه را برای W به‌دست می آوریم. (از آنجایی که x_0 یک جواب شدنی یا یک جوابی نزدیک به شدنی برای مسئلهP2 است و با توجه به نحوه محاسبه W^0 و y^0، اکثر درایه W^0 منفی است بنابراین اکثر درایه‌های W^0 در تکرار بعد تبدیل به صفر می‌شود).

اکنون چند قاعده برای شرط توقف رویه تکراری تخفیف لاگرانژین براساس کران پایین بیان می‌نماییم.

قضیه ۱ : فرض کنید که تکنیک تخفیف لاگرانژین در مسئله $p^{\le}$: Min. $\{cx\,;Ax\le b, x\in X\}$ با تخفیف قیود $Ax\le b$ اجرا شده باشد. اگر که برای بعضی مقادیر ماتریس ضرایب لاگرانژینW , جواب به‌دست آمده y^* از مسئله لاگرانژین، یک جواب شدنی مسئله $p^{\le}$ باشد که در شرط مکمل زائد $W(Ay^*-b)=0$ صدق کند، آنگاه y^* یک جواب بهینه برای مسئله بهینه سازی $p^{\le}$ است (Ahuja *et al.*, 1993).

قاعده بعدی توقف، مربوط به اختلاف کران بالا و کران پایین تابع هدف است , اگر LB را مقدار کران پایین تابع هدف و UB را مقدار کران بالا تابع هدف بنامیم , چون در الگوریتم اصلی هر دو کران بالا و پایین را به کار می بریم بنابراین اگر شرط زیر صادق باشد جواب به‌دست آمده بسیار نزدیک بهینگی است.

قضیه ۲: $\frac{UB-LB}{|LB|+1}\le TOL$ که معمولاً $TOL=10^{-6}$ است

(Gulpinar *et al.* 2002).

شرط سوم توقف مربوط به اندازه تغییر W^p می باشد. اگر اندازه تغییر W^p بسیار کوچک باشد، بهبود در مقدار کران تابع هدف نیز بسیار کم است، پس اگر شرط زیر نیز برقرار باشد توقف می‌کنیم.

قضیه۳: $\text{Max}_{(i,j)\in A}(\theta_q(\sum_{1\le k\le K} v_{ij}^k - u_{ij})) \le 0/005$

که:

$$\forall\ (i,j)\in A_{kr}\ ,\ v_{ij}^{kp} = v_{(p,q)kr}^p$$
$$r = 1,2\ldots,S_k, k = 1,2,\ldots,K\ ,$$

(Ali 1988).

حال با توجه به مقدار اولیه W و بیان شرط توقف الگوریتم مربوط به کران پایین تابع هدف را بیان می‌کنیم. در این الگوریتم ما یک طول گام اولیه θ را در نظر می گیریم مقدار بعدی θ وابسته به بهبود کران پایین و پارامتر m^* (پارامتر m^* یک مقدار ثابت اولیه است) دارد. در صورتیکه مقدار تابع هدف در طول m^* تکرار بهبود یابد، مقدار طول گام بدون تغییر باقی خواهد ماند، ولی اگر تابع هدف در طول m^* تکرار بهبود نیافت طول گام نصف می شود و تکرار از نقطه‌ای که بهترین مقدار برای کران پایین به‌دست آمده است ادامه می‌یابد.

الگوریتم کران پایین تابع هدف

(۱) مقدار دهی اولیه:

یک کران بالا اولیه UB , طول گام اولیه θ و مقدار اولیه m^* وTol را در نظر بگیرید .

$$p \leftarrow 0 \quad , \quad m' \leftarrow 0$$

(۲) به دست آوردن بردار زیرگرادیان:

- از زیر مسئله p5 ، برای مقدار ثابت داده شده W^p برای $k = 1,2,\ldots,K$ جواب V^{kp} را به‌دست می‌آوریم و قرار می‌دهیم :

$$\Delta_{ij}^p = [\sum_{1\le k\le K} v_{ij}^{kp} - u_{ij}] \quad \forall (i,j)\in A$$

که

$$r = 1,2\ldots,S_k, k = 1,2,\ldots,K\ ,\ \forall\ (i,j)\in A_{kr}\ ,\ v_{ij}^{kp} = v_{(p,q)kr}^p$$

- اگر $W^p\Delta^p = 0$ و V^p یک جواب شدنی مسئله P2 بود آنگاه V^p در قضیه (۱) صدق می‌کند و جواب بهینه حاصل می‌شود.

- اگر $Z(W^p) < LB$ آنگاه $m' \leftarrow m'+1$

- اگر $m' = m^*$ آنگاه

$$\theta \leftarrow \frac{\theta}{2} \quad , \quad W^p \leftarrow W^* \quad , \quad m' \leftarrow 0$$

در غیر این صورت $m' \leftarrow 0$ و $LB \leftarrow Z(W^p)$ و $W^* \leftarrow W^p$

- اگر $\frac{UB - LB}{|LB|+1} \le TOL$ بنا به قضیه ۲ الگوریتم خاتمه می‌پذیرد.

(۳) حرکت به یک نقطه جدید:

- قرار دهید $w_{ij}^{p+1} \leftarrow [w_{ij}^p + \theta\Delta_{ij}^p]^+$

- اگر $\text{Max}_{(i,j)\in A}\ \theta\,\Delta_{ij}^p \le 0/005$ بنا به قضیه ۳ الگوریتم خاتمه می پذیرد.

در غیر این صورت $p \leftarrow p+1$ و به مرحله (۲)بروید.

توجه کنید که طول گام θ تعریف شده دراین الگوریتم در شرایطی که(Ahuja *et al.*, 1993) برای همگرایی بیان کرده است صدق می‌کند.

به دست آوردن یک کران بالا برای تابع هدف مسئله

برای به‌دست آوردن یک کران بالا برای مسئله می‌توان از روش هدایت منابع(Resource Directive) استفاده کرد که مسئله را به دو مسئله اصلی و زیر مسئله تبدیل می‌کند و با تخصیص یک ظرفیت به متغیرهای تخصیص ظرفیت، زیر مسئله را حل می‌کند. (Ahuja *et al.*, 1993) . اشکال عمده این روش بزرگ شدن اندازه مسئله اصلی است.

برای حل این مشکل ما از یک روش مستقیم، گرفته شده از روش EMNET(Embedded Network) (Mc Bride 1995) و حل مسئله چند کالای با استفاده از الگوریتم سیمپلکس شبکه محاط (Mc Bride *et al.*, 1997) استفاده می‌کنیم. برای این منظور ابتدا مسئله P2 به شکل استاندارد می‌نویسیمT یعنی فقط به قیود (2.a) متغیرهای کمبود را می افزاییم و بقیه قیود مسئله P2 تغییری نمی‌کند. پس قیود (2.a) به شکل زیر در می آیند:

$$\sum_{k=1}^{K}\sum_{r=1}^{s_k}\delta_{r\,ij}^k x_{(p,q)_{kr}} + \sum_{1\le k\le K}\varepsilon_{ij}^k x_{ij}^k + s_{ij} = u_{ij} \qquad (2.a')$$
$$(i,j)\in A$$

مرحله‌های معمول در روش EMNETعبارتند از:

مرحله اول: حل مسئله p2 با حذف قیود (2.a) با استفاده از الگوریتم سیمپلکس شبکه برای مسئله جریان مساوی کلی و به‌دست آوردن یک جواب اولیه می‌باشد.

مرحله دوم: جواب بهینه به‌دست آمده از مرحله اول را در قیود (2.a) کنترل می‌کنیم یا این جواب در قیود (2.a) صدق می‌کند که در این حالت جواب مرحله اول جواب بهینه مسئله است یا این‌که جواب مرحله اول در قیود (2.a) صدق نمی‌کند که در این حالت متغیرهای فرضی را به قیود('2a) اضافه کرده و این متغیرهای فرضی را، وارد پایه می‌کنیم. سپس یک جواب شدنی را با حذف متغیرهای فرضی از پایه به‌دست می‌آوریم.EMNET از روش M بزرگ به‌جای برنامه‌ریزی دو فازی برای بیرون کردن متغیرهای فرضی از پایه استفاده می کند.

مرحله سوم : با جواب شدنی که از مرحله دوم به‌دست آمده شروع می‌کند و به‌سمت بهینگی می‌رود.

یک مشکلی که در EMNET با آن برخورد می‌کنیم این است که

جواب به‌دست آمده از مرحله اول، می‌تواند خیلی دور از شدنی بودن باشد. در این حالت ما به‌جای مرحله اول EMNET از جواب x_0 به‌دست آمده از روش تخصیص ظرفیت استفاده می‌کنیم که جواب x_0 بسیار نزدیک به شدنی بودن است. همانطور که می‌دانیم مسئله تخصیص ظرفیت به‌عنوان یک مسئله با هزینه محدب حل می‌شود. وقتی یک جواب شدنی پیدا شد ما باید آن را به یک جواب پایه‌ای، برای مسئله اصلی تبدیل کنیم. در این حالت از الگوی هزینه محدب جریان یک‌بار دیگر استفاده می‌کنیم و یک جواب پایه‌ای اولیه برای مسئله اصلی به‌وجود می‌آوریم (Mc Bride *et al.*, 1997). برای این منظور، ما برای هر متغیر غیر پایه‌ای (به‌دست آمده از الگوریتم تخصیص ظرفیت) که در کران پایین (صفر) یا بالای اصلی خود نیست دو قطعه هزینه محدب یکسان به‌وجود می‌آوریم، قطعه اول از صفر تا مقدار اولیه متغیر غیر پایه‌ای و قطعه دوم از مقدار اولیه متغیر غیر پایه تا کران بالای اصلی است و با بقیه متغیرها به‌عنوان متغیر ها با هزینه خطی رفتار می‌کنیم. بردار q را به‌صورت زیر تعریف می‌کنیم:

$q_{ij}^k = 0$ برای هر متغیر پایه‌ای و متغیرهای غیر پایه‌ای که در کران بالا یا پایین خود هستند.

q_{ij}^k را برای متغیرهای غیر پایه که در نقاط انفصال (بین کران بالا و پایین هستند) برابر با مقدار نقطه انفصال می‌گذاریم. و به همین ترتیب $k=1,2,\dots,K, q_r^k$ و $r=1,2,\dots,s_k$ را تعریف می‌کنیم.

P6:

$$\text{Min} \sum_{k=1}^{K} \sum_{(i,j)\in A_k} c_{ij}^k \quad (q_{ij}^k - (q_{ij}^k - x_{ij}^k)^+ + (x_{ij}^k - q_{ij}^k)^+) +$$

$$\sum_{k=1}^{K} \sum_{r=1}^{s_k} c_r^k (q_r^k - (q_r^k - x_r^k)^+ + (x_r^k - q_r^k)^+)$$

$$\text{S.t} \qquad (6)$$

$$BX \le U$$

$$D^k x^k = b^k \qquad k=1,2,\dots,K$$

$$0 \le x_{ij}^k \le u_{ij} \qquad (i,j)\in A_k \, , k=1,2,\dots,K$$

$$0 \le x_{(p,q)kr} \le u_r^k \qquad k=1,2,\dots,K, r=1,2,\dots,s_k$$

مسئله بالا با استفاده از تبدیل مسئله جریان هزینه محدب به مسئله مینیمم هزینه جریان به‌دست می آید (Ahuja *et al.*, 1993). جواب x_0 را می‌توان به‌عنوان یک جواب اولیه این مسئله به کار برد.

سر انجام یک جواب شدنی پایه‌ای به‌دست می آید و روند بقیه مرحله‌ها مانند روش EMNETادامه می‌یابد. تنها تفاوتی که هست، ما دیگر در زیر مسئله‌ها از روش سیمپلکس شبکه معمولی استفاده نمی‌کنیم بلکه از روش سیمپلکس شبکه برای مسئله مساوی درحالت کلی استفاده می‌کنیم.

در هر بار تکرار روش EMNET برای مسئله چند کالایی با نقطه اولیه بسیار خوب به‌دست آمده از p6 ما یک کران بالا UB برای تابع هدف مسئله به‌دست می‌آوریم. (زیرا هر جواب به‌دست آمده ازEMNET برای مسئله چند کالایی یک جواب شدنی مسئله p2 است).

شرط توقف در این‌جا دو حالت دارد :

۱) برقراری شرط بهینگی در مرحله سوم روش EMNET که همان شرط بهینگی روش سیمپلکس است.

۲) برقراری شرط قضیه (۲).

الگوریتم کلی

الگوریتم کلی کران بالا وکران پایین تابع هدف را تعدیل می‌کند تا به جواب بسیار نزدیک به جواب بهینه برسد. در این الگوریتم دو مقدار IL و IU را به‌طور دلخواه در ابتدا مقداردهی می‌کنیم. وقتی الگوریتم اصلی‌کران پایین را صدا می‌زند، الگوریتم کران پایین IL بار اجرا می شود و هم‌چنین وقتی الگوریتم کران بالا را فرا می‌خوانیم این الگوریتم نیز IU بار اجرا می‌شود.

در الگوریتم کلی ما ابتدا الگوریتم تخصیص ظرفیت را فرا می‌خوانیم زیرا آن یک نقطه شروع خوب نزدیک به جواب بهینه تولید می‌کند که می‌توانیم این مقدار به‌عنوان جواب اولیه در الگوریتم کران پایین و کران بالا مورد استفاده قرار دهیم. حال الگوریتم کلی را این‌گونه بیان می‌کنیم.

الگوریتم کلی:

(۱) یک مقدار اولیه

دو مقدار IU و IL را در نظر بگیرید. (این دو مقدار تعداد تکرارهای کران بالا و پایین را در هر مرحله بیان می‌کند).

$R \leftarrow 0 \quad , \quad T \leftarrow 0 \quad , \quad UB \leftarrow +\infty \quad , \quad LB \leftarrow -\infty$ الگوریتم

تخضیص ظرفیت را اجرا کنید:

(۱) محاسبه کران پایین برای مسئله

a) الگوریتم کران پایین را اجرا کنید.

b) $T \leftarrow T+1$ اگر $T \le IL$ آنگاه به مرحله 1.a بروید.

(۲) محاسبه کران بالا

a) الگوریتم کران بالا را اجرا کنید .

b) $R \leftarrow R+1$ اگر $R < IU$ آنگاه به مرحله 2.a بروید.

(۳) تکرار دوباره مرحله‌ها

$T \leftarrow 0, R \leftarrow 0$ و به مرحله (۱) بروید.

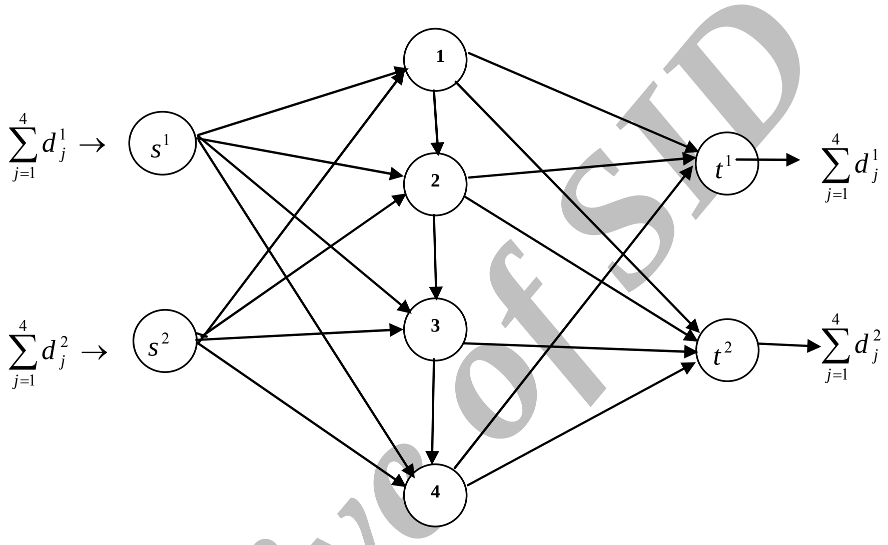
کاربرد مسئله چند کالایی با جریان مساوی روی کمان‌ها بنکداری تولیدات فصلی (Warehousing of Seasonal Products).

یک شرکت تولیدکننده محصولات چند گانه را در نظر بگیرید به طوری که تولید و تقاضای محصولات به‌صورت دوره‌ای انجام می‌پذیرد.

این شرکت دارای انباری با ظرفیت ثابت R است که در این انبار همه محصولات تولید شده ذخیره می‌شود.

این مسئله یک مسئله چند کالایی و چند دوره‌ای (multi –period) است که دوره های آن معمولاً هفتگی، ماهانه یا سه ماه یکبار است. این مسئله را می‌توان با استفاده از تکنیک شبکه بسط یافته زمانی (Time-expanded) (Balázs Kotnyek Septembre 2003) به یک مسئله ایستا تبدیل کرد. مسئله تصمیم‌گیری در اینجا شناسای سطح تولیدات برای همه محصولات در هر دوره است به طوری که الف) تقاضا در هر دوره برای هر محصول را برآورده سازد و ب) میزان تولید برای بعضی ازکالاها در هر دوره ثابت بماند. ج) محصولات تولید شده در هر دوره از ظرفیت انبار بیشتر نشوند د) که هزینه تولید و هزینه نگهداری در انبار حداقل شود.

در حالت ساده فرض کنید شرکت دو نوع محصول را تولید می‌کند و شرکت نیاز به تهیه فهرستی از تولیدات برای ۴ دوره سه ماه یک سال را دارد. فرض کنید d_j^2 ، d_j^1 نمایش برای تقاضای محصولات ۱ و۲ در jامین دوره باشد. گنجایش (ظرفیت)محصول برای jامین دوره u_j^1 و u_j^2، هزینه هر واحد محصول برای این دوره c_j^1 و c_j^2 و h_j^1 و h_j^2 هزینه‌های ذخیره (نگهداری) دو محصول از دوره j به دوره j+۱ باشد. شبکه بسط یافته زمانی این مسئله به‌صورت شکل ۱ می‌باشد.



نمودار ۱. بهینه سازی بنکداری تولیدات فصلی

شبکه شامل یک گره برای هردوره همچنین یک گره منبع و یک گره تقاضا برای هرکالا می باشد.

(شبکه به شبکه تک عرضه‌ای و تک تقاضایی برای کالای نوع اول و دوم تبدیل شده است) میزان عرضه و تقاضا برابر مجموع میزان تقاضا برای کالا در چهار دوره است. هر گره منبع s^k دارای چهار کمان خارج شونده متناظر با چهارفصل است که فقط یک کالا روی هریک از این کمانها جریان دارد. هزینه c_j^k و ظرفیت u_j^k را به کمان (s^k, j) وابسته می‌کنیم، به طور مشابه گره تقاضای دارای چهار کمان وارد شونده است. به کمان (j, t^k) هزینه صفر و گنجایش d_j^k وابسته می‌کنیم.کمان‌های باقی مانده به صورت (j و j+۱) برای j=۱,۲,۳ می‌باشد که جریان روی این کمان‌ها میزان کالای ذخیره شده را از دوره j به دوره j+۱ نمایش می‌دهد هر یک از این کمان‌ها دارای ظرفیت R، با هزینه انتقال هر واحد h_j^1 برای کالای ۱ و h_j^2 برای کالای۲ می‌باشد. فرض کنید در این مثال می‌خواهیم جریان روی کمان $(s^1, ۱)$ و $(s^1, ۴)$ برای کالای ۱ و جریان روی کمان $(s^2, ۲)$ و $(s^2, ۳)$ برای کالای ۲ با همدیگر یکسان باشند.

این مثال را با متناظر کردن اعداد در نگاره ۱ و ۲ مربوط به کالای اول و دوم و با استفاده از الگوریتم توضیح داده شده تا ۱۰ مرحله در نگاره شماره ۳ حل می‌کنیم. درضمن فرض می‌کنیم که:

$$h_j^1 = h_j^2 = 5 \quad j = 1,2,3,4$$

$$s^1 = \sum_{j=1}^{4} d_j^1 = 20 \quad \text{و} \quad s^2 = \sum_{j=1}^{4} d_j^2 = 30$$

نگاره ۱. داده ها مربوط به کالای اول

پارامترها	u_1^1	u_2^1	u_3^1	u_4^1	d_1^1	d_2^1	d_3^1	d_4^1	c_1^1	c_2^1	c_3^1	c_4^1
مقادیر	10	5	9	12	3	5	9	3	5	4	3	4

نگاره ۲. داده ها مربوط به کالای دوم

پارامترها	u_1^2	u_2^2	u_3^2	u_4^2	d_1^2	d_2^2	d_3^2	d_4^2	c_1^2	c_2^2	c_3^2	c_4^2
مقادیر	20	8	15	14	4	8	4	14	6	4	2	3

نگاره ۳. جواب‌های محاسبه شده با استفاده از الگوریتم‌های کلی ارایه شده با ۱۰مرحله

تکرارام	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰
کران بالا	۳۶۳	۳۳۲,۳۶	۲۵۰,۷	۲۲۲	۲۰۶,۸۶	۱۹۸,۳۶	۱۹۲,۶۷	۱۸۸,۶۲	۱۸۴,۵۸	۱۸۲,۶۵
کران پایین	۹۶,۵۳	۱۰۵,۵	۱۶۵,۴۶	۱۷۰	۱۷۵	۱۷۷,۸۱	۱۷۸,۶۹	۱۷۹,۵	۱۷۹,۶۶	۱۷۹,۷۸

منابع:

Ahuja R.K., Magnanti T.L., Orlin J.B. 1993: Network Flows Theory, Algorithms and Applications: Prentice Hall, Englewood Cliffs, NJ.

Ahuja R.K., Orlin J.B., Sechi G.M., Zuddas, P. 1999: Algorithms for the simple equal flow problem. *Management Science.* **45:**1440–1455.

Ali A.I., Kennington J., Shetty B. 1988: The equal flow problem. *European Journal of Operational Researc.* **36:** 107–115.

Balázs Kotnyek Septembre 2003: An annotated overview of dynamic network flows.

Calvete H.I. 2003: Network simplex algorithm for the general equal flow problem. *European Journal of Operational Research*. **150:** 585 – 600.

Gulpinar N., Mitra G., Maros I. 2002: Creating Advanced Bases For Large Scale Linear Programs Exploiting Embedded Network Structure. *Computational Optimization and Application.* **21:** 71–93.

Mcbride R.D. 1985: Solving embedded generalized network problem. *European Journal of Operational Research.* **21:** 82–92.

Mcbride R.D., Mamer J.W. 1997: Solving multicommodity flow problems with a primal embedded network simplex algotithm: INFORMS Journal on computing. 9: 154–163.