International Journal of Industrial Engineering & Production Management (2014)

May 2014, Volume 25, Number 1
pp. 99-108

http://IJIEPM.iust.ac.ir/

A Branch-and-Price Algorithm for Solving the Railroad Blocking Problem

M. Yaghini*, **M. Rahbar, M. Karimi, M. Khoshkroudian**

***Masoud Yaghini**, Assistant Professor, School of Rail Engineering, Iran University of Science and Technology, Tehran, Iran*
***Majid Khoshkroudian**, School of Railway Engineering, Iran University of Science and Technology, Tehran, Iran*
***Mohadeseh Rahbar**, School of Railway Engineering, Iran University of Science and Technology, Tehran, Iran*
***Mohammad Karimi**, School of Railway Engineering, Iran University of Science and Technology, Tehran, Iran*

Keywords

Railroad Blocking Problem,
Branch and Price Algorithm,
Column Generation,
Dantzig-Wolfe algorithm,
Optimization

ABSTRACT

The railroad blocking problem is one of the most important planning problem in freight railways. By solving this problem, once can minimize volume of switching process and total cost of delivering the commodities. This paper presents a method based on branch and price algorithm for solving the railroad blocking problem. This algorithm is a combination form of branch-and-bound and column generation algorithms. Branch and price is a variant of branch and bound, with bounds provided by solving linear programs using column generation at nodes of the branch and bound tree. In branch and price algorithm, re-optimize column generation algorithm in each of branches. Because the bound provided by the LP relaxation is weak, we suggest cuts to strengthen it and show the effect of adding them on the column generation procedure. Implementation of this algorithm has been done by Java programming language. To analyze the quality of the algorithm, some simulated problems with different size generated and are solved by CPLEX software. The results of our algorithm compared with CPLEX's results. The comparison shows the efficiency and accuracy the purpose algorithm.

* Corresponding author. Masoud Yaghini
Email: *yaghini@iust.ac.ir*

نشـــریه بین المللــــی مهنـــدسی صنـــایع و مدیـــریت تولیـــد

شمـــاره ۱، جلـــد ۲۵، خـــرداد ۱۳۹۳
صفحــــه ۱۰۸-۱۰۰

نشریه بین المللی مهندسی صنایع و مدیریت تولید
دانشگاه علم و صنعت ایران

ISSN: 2008-4870

http://IJIEPM.iust.ac.ir/

یک الگوریتم شاخه و قیمت برای حل مساله گروه بندی واگن‌های باری راه آهن

مسعود یقینی*، محدثه رهبر نودهی، محمد کریمی ارکوینی، مجید خشکرودیان

کلمات کلیدی

گروه‌بندی واگن‌ها، روش شاخه و قیمت، روش تجزیه دنتزینگ-ولف، بهینه سازی

چکیده:

مساله گروه‌بندی واگن‌های باری یکی از مسائل مهم برنامه‌ریزی در حوزه راه‌آهن باری است. در این مقاله جهت حل این مساله، یک الگوریتم شاخه و قیمت که یک روش بهینه‌سازی دقیق می‌باشد، ارائه شده است. این روش مشابه روش شاخه و کران است، با این تفاوت که برای حل مسائل در هر شاخه، بجای استفاده از آزادسازی خطی از روش تولید ستون استفاده می‌شود. در روش تولید ستون که مبتنی بر تجزیه دانتزینگ-ولف است، مساله اولیه گروه‌بندی واگن‌ها به یک مساله اصلی و بر اساس تعداد تقاضاها به چندین مساله فرعی تجزیه می‌شود. با حل مساله اصلی، مقادیر دوگان محاسبه و به مسائل فرعی ارسال شده و با حل مسائل فرعی، مسیرهای جدید تولید و به مساله اصلی اضافه می‌شود. جهت ارزیابی این روش حل، چندین مساله نمونه تولید و حل شده و نتایج با جواب های بدست آمده از نرم افزار CPLEX مقایسه شده است. نتایج بدست آمده نشان‌دهنده کارایی الگوریتم پیشنهادی است.

۱. مقدمه

در راه‌آهن‌های باری گروه‌بندی واگن‌ها با توجه به مبدا و مقصد تقاضاها انجام می‌شود و تقاضاهایی که در یک گروه قرار می‌گیرند، تا زمانی که به مقصد گروه نرسیده باشند، نمی‌توانند تفکیک شوند. مبدا یا مقصد یک گروه واگن، می‌تواند مبدا یا مقصد هیچ یک از تقاضاهای موجود در آن نباشد، به این دلیل که یک تقاضا از مبدا (و در طول مسیر خود) تا رسیدن به مقصد، ممکن است چندین گروه واگن عوض کند. راه‌آهن‌ها برای جابجایی تقاضاها از مبدا تا مقصد با هزینه‌های کمتر، به برنامه‌ریزی گروه‌بندی واگن‌ها می‌پردازند که این برنامه‌ریزی مشخص می‌کند در هر ایستگاه کدام گروه از واگن‌ها باید ایجاد شود و به هر گروه واگن چه تقاضاهای تخصیص داده شود. مساله گروه‌بندی واگن‌های باری، سعی در جواب دادن به این سوالات دارد: (۱) عملیات مانور چطور بین ایستگاه‌های بین راهی تقسیم شود؟، (۲) چه گروه واگن‌هایی باید تشکیل شود؟ و (۳) هر گروه واگن چه تقاضاهایی را حمل کند؟. مساله گروه‌بندی واگن‌های باری با پاسخ گویی به سوالات فوق، در هزینه‌ها صرفه جویی می‌کند و سعی می‌کند حمل تقاضا به مقصد، در کمترین زمان ممکن، صورت گیرد.برای اولین بار مدل مساله گروه‌بندی واگن‌ها توسط بودین و همکاران [۱] ارائه شد. این مدل، مساله گروه‌بندی واگن‌ها را به عنوان یک مساله MIP[2] غیر خطی در نظر گرفت. این مدل، گروه‌بندی بهینه را برای تمامی ایستگاه‌ها در یک سیستم ریلی به صورت هم زمان بدست می‌آورد. در این مدل محدودیت‌های تعداد گروه و حد بالا برای حجم عملیات مانور

تاریخ وصول: ۹۰/۶/۳۰

تاریخ تصویب: ۹۱/۴/۱۷

*__نویسنده مسئول مقاله:__ **دکتر مسعود یقینی**، استادیار دانشکده مهندسی راه آهن، دانشگاه علم و صنعت ایران. yaghini@iust.ac.ir

محدثه رهبر نودهی، کارشناسی ارشد دانشکده مهندسی راه آهن، دانشگاه علم و صنعت ایران. m_rahbar@rail.iust.ac.ir

محمد کریمی ارکوینی، کارشناسی ارشد دانشکده مهندسی راه آهن، دانشگاه علم و صنعت ایران. mohammad_karimi@rail.iust.ac.ir

مجید خشکرودیان، کارشناسی ارشد مهندسی دانشکده مهندسی راه آهن، دانشگاه علم و صنعت ایران. m_khoshkroudian@rail.iust.ac.ir

[2] *Mixed Integer Programming*

در هر ایستگاه در نظر گرفته می‌شود. ون‌دایک [۲] رویکردی تعاملی و مبتکرانه را ارائه کرد که در آن سعی شده بود تا یک برنامه گروه‌بندی از قبل تهیه شده را توسط حل یک سری از مسائل کوتاه‌ترین مسیر[۱] که در آن کمان‌ها بیانگر گروه‌های موجود بودند را بهبود بخشد. از مهم‌ترین پژوهش‌های صورت گرفته در زمینه گروه‌بندی واگن‌ها می‌توان به مطالعات بارنهارت و نیوتن [۳] اشاره کرد. آنها مدل گروه‌بندی واگن‌ها را بصورت یک مدل طراحی شبکه[۲] ارائه و برای حل آن یک روش شاخه و قیمت[۳] طراحی کردند.

بارنهارت و همکاران [۴] نیز فرمولی مشابه نیوتن برای مساله گروه-بندی واگن‌ها در نظر گرفتند و با ارائه روش آزادسازی لاگرانژی[۴] این مساله را به دو زیر مساله تجزیه کردند. هر دو روش جوابی نزدیک به بهینه تولید می‌کنند. زمانی که اندازه مساله بسیار بزرگ است، استفاده از این روش‌ها عملی نیست. بطور خلاصه الگوریتم-هایی که تا به حال برای مساله گروه‌بندی واگن‌ها ارائه شده است با صرف زمان زیاد جواب‌هایی نزدیک به بهینه تولید می‌کند و در عمل فقط برای شبکه‌های کوچک یا خیلی ساده مورد استفاده قرار می‌گیرند.

آهوجا [۵] در سال ۲۰۰۷ یک مدل MIP ارائه کرد و برای حل مدل خود، از الگوریتمی به نام جستجوی همسایگی استفاده کرد. این الگوریتم با یک جواب شدنی آغاز می‌شود و بصورت متناوب جواب را بهبود می‌دهد و این کار با تغییر همسایگی در هر مرحله انجام می‌شود. او روش خود را بر روی دو راه‌آهن بزرگ آمریکا با نام‌های CSX و BNSF پیاده‌سازی کرد و ادعا کرد که روش وی قادر خواهد بود تا هزینه‌های عملیاتی این دو راه‌آهن بزرگ را تا حدودی کاهش دهد.

آخرین کار جدی انجام شده، در سال ۲۰۰۸ توسط جی‌ها [۶] صورت گرفت که مدل خود را با الگوریتم شمارش مسیر[۵] حل کرد. مساله گروه‌بندی واگن‌ها جزء مدل‌های طراحی شبکه‌های چند کالایی[۶] است. مگنانتی، ونگ [۷]، بالاکریشنان [۸] و کراینیک و همکاران [۹] مطالعات جامعی بر روی این مدل‌ها ارائه نموده‌اند. برای حل مسائل طراحی شبکه چند کالایی، روش‌های حل دقیق و روش‌های حل مبتنی بر الگوریتم‌های فرا ابتکاری ارائه شده است. از جمله مقالاتی که اخیرا ارائه شده می‌توان به مارتین و گنزالز [۱۰]، پدرسون و همکاران [۱۱] و هاف و همکاران [۱۲] که الگوریتم های فرا ابتکاری برای حل مسائل ارائه کردند و همچنین به کارهای

[۱] Shortest Path
[۲] Network Design Model
[۳] Branch and Price
[۴] Lagrangian Relaxation
[۵] Path Enumeration Algorithms
[۶] Multicommodity Network Design

فرانجیونی و همکاران [۱۳]، کاتایاما و همکاران [۱۴] و هلمبرگ و همکاران [۱۵] که روش‌های دقیق و ترکیب روش های تقریبی برای حل این مسائل ارائه کردند، اشاره نمود.

در صورتیکه بخواهیم حوزه حمل و نقل بار را با حمل نقل مسافر مقایسه کنیم، برنامه ریزی خطوط مسافری[۷] در حمل و نقل مسافر شبیه به گروه بندی واگنها در حمل و نقل بار است، با این تفاوت که در برنامه ریزی خطوط مسافری، هدف تعیین مسیرهای بهینه مسافری است.

برای مطالعه بیشتر به یقینی و همکاران [۱۶ و ۱۷] مراجعه شود.

در ادامه، مقاله بدین صورت سازماندهی شده است. در بخش بعد، مدل ریاضی مساله گروه بندی واگن‌ها ارائه می‌شود. در ادامه آن به بیان روش پیشنهادی شاخه و قیمت و ارزیابی عملکرد روش حل، با حل مسائل شبیه‌سازی شده و مقایسه جواب روش پیشنهادی با جواب بدست آمده از نرم‌افزار CPLEX پرداخته می‌شود. نتیجه‌گیری و جمع‌بندی مطالب در بخش پایانی ارائه شده است.

۲. مدل ریاضی مساله گروه بندی واگن‌ها

مساله گروه‌بندی واگن‌ها را می‌توان بر مبنای گره-کمان[۸] و بر مبنای مسیر[۹] مدلسازی کرد. G = (N, A) گرافی با مجموعه ایستگاه‌های N و مجموعه گروه واگن‌های A است. در این مقاله از مدل مبتنی بر مسیر استفاده شده است [۳]. با توجه به اینکه مدل مبتنی بر مسیر بر اساس مدل مبتنی بر گره-کمان ساخته می‌شود، بنابراین ابتدا مدل گره-کمان و سپس مدل مبتنی بر مسیر ارائه می‌شود.

در مدل بر مبنای گره-کمان، پارامترها و متغیرهای مورد استفاده در مدل ارائه شده، به شرح زیر است.

k: اندیس تقاضاها، $k \in K$.

K: مجموعه تقاضاها، هر تقاضا با یک زوج گره مبدأ- مقصد تعریف می‌شود.

v^k: حجم تقاضای k.

$orig\,(a)$: ایستگاه مبدأ کمان a.

$dest\,(a)$: ایستگاه مقصد کمان a.

u_a: ظرفیت کمان a.

c_a: هزینه جریان هر واحد تقاضا بر روی کمان a.

$B(i)$: تعداد گروه واگن‌هایی که در ایستگاه i می‌توان تشکیل داد.

[۷] Line Planning
[۸] Node-Arc Formulation
[۹] Path-Based Formulation

$V(i)$: تعداد واگن‌هایی که می‌توان در ایستگاه i روی آن‌ها عملیات مانور انجام داد.

x_a^k: متغیر تصمیم پیوسته که مشخص کننده نسبت حمل تقاضای k بر روی کمان a است.

y_a: متغیر تصمیم صفر- یک، که مشخص کننده انتخاب کمان a است.

با تعاریف فوق مدل ریاضی مبتنی بر گره-کمان برای مساله گروه‌بندی واگن‌ها، به صورت زیر بیان می‌شود.

$$\min: \quad Z = \sum_{k\in K}\sum_{a\in A} c_a v^k x_a^k \tag{۱}$$

$s.t.$

$$\sum_{\substack{a\in A\\ orig(a)=i}} x_a^k - \sum_{\substack{a\in A\\ dest(a)=i}} x_a^k = \begin{cases} 1 & orig(a)=i \\ -1 & dest(a)=i \\ 0 & otherwise \end{cases} \qquad \forall i\in N,\ k\in K \tag{۲}$$

$$\sum_{k\in K} v^k x_a^k \le u_a y_a \qquad \forall a\in A \tag{۳}$$

$$\sum_{\substack{a\in A\\ orig(a)=i}} y_a \le B(i) \qquad \forall i\in N \tag{۴}$$

$$\sum_{k\in K}\sum_{a\in A/orig(a)=i} v^k x_a^k \le V(i) \qquad \forall i\in N \tag{۵}$$

$$y_a \in \{0,1\} \qquad \forall a\in A \tag{۶}$$

$$0 \le x_a^k \le 1 \qquad \forall a\in A, k\in K \tag{۷}$$

تابع هدف مساله (۱) عبارتست از حداقل کردن هزینه ارسال کلیه تقاضاها از مبدا به مقصد، که این هزینه شامل هزینه ارسال تقاضا بر روی گروه واگن‌ها است. در این مدل، هزینه‌های عملیات مانور در مبدا و مقصد تقاضاها در نظر گرفته نشده است، زیرا این عملیات برای همه تقاضاها باید انجام پذیرد و تاثیری در انتخاب متغیرهای تصمیم ندارد.

محدودیت (۲)، محدودیت‌های توازن را نشان می‌دهد که در آن تفاضل مجموع تقاضاهای خروجی از یک ایستگاه به مجموع تقاضاهای ورودی به همان ایستگاه برای ایستگاه‌های مبدا، مقصد و میانی به ترتیب برابر v^k، $-v^k$ و صفر است. محدودیت (۳) تنها به گروه‌هایی اجازه تشکیل شدن می‌دهد که حتماً تقاضایی را حمل کنند.

به دلیل اینکه حجم تقاضاهایی که بوسیله یک گروه واگن حمل می‌شود، محدودیتی ندارد، بنابراین بجای u_a، یک مقدار بزرگ در نظر گرفته می‌شود. محدودیت (۴) تعداد گروه واگن‌هایی که در هر ایستگاه تشکیل می شود را محدود می کند و محدودیت (۵) حجم عملیات مانور قابل انجام در هر ایستگاه را مشخص می‌کند.

در ادامه بدلیل انطباق ساختار مدل با روش شاخه و قیمت، مدل مبتنی بر مسیر ارائه می‌شود که روش حل پیشنهادی نیز بر اساس این مدل ارائه شده است. برای مدل سازی ریاضی مبتنی بر مسیر، از مدل ارائه شده توسط بارنهارت و همکاران [۴] استفاده شده است. علائم و نمادهای اضافه شده نسبت به مدل مبتنی بر گره-کمان به شرح زیر است.

$Q(K)$: مجموعه مسیرهای مجاز برای تقاضای K.

PC_q^k: هزینه مسیر برای حمل یک واحد از تقاضای k.

δ_a^q: پارامتر صفر-یک، زمانی یک می‌شود که مسیر q شامل کمان a باشد.

f_q^k : متغیر تصمیم پیوسته، که مشخص کننده نسبت حمل تقاضای k روی مسیر q است.

مدل ریاضی گروه‌بندی واگن‌ها بر مبنای مسیر، بصورت زیر است.

$$\min: \quad \sum_{k\in K}\sum_{q\in Q(k)} v^k PC_q^k f_q^k \qquad (۸)$$

$$s.t.$$

$$\sum_{k\in K}\sum_{q\in Q(k)} v^k f_q^k \delta_a^q - u_a y_a \le 0 \qquad \forall a\in A \qquad (۹)$$

$$\sum_{q\in Q(k)} f_q^k = 1 \qquad \forall k\in K \qquad (۱۰)$$

$$\sum_{\substack{a\in A\\ orig(a)=i}} y_a \le B(i) \qquad \forall i\in N \qquad (۱۱)$$

$$\sum_{k\in K}\sum_{q\in Q(k)}\sum_{\substack{a\in A\\ orig(a)=i}} v^k f_q^k \delta_a^q \le V(i) \qquad \forall i\in N \qquad (۱۲)$$

$$0\le f_q^k \le 1 \qquad \forall q\in Q(k), \forall k\in K \qquad (۱۳)$$

$$y_a\in\{0,1\} \qquad \forall a\in A \qquad (۱۴)$$

تابع هدف این مدل (۸) کمینه کردن هزینه‌های کل حمل تقاضاها بر روی مسیرها است. محدودیت شماره (۹) تضمین می‌کند که تقاضاها حتماً بر روی گروه واگن‌هایی جریان داشته باشند که آن گروه واگن‌ها در جواب مساله ساخته شده باشند. محدودیت (۱۰)، محدودیت توازن بوده و رسیدن تقاضاها را به مقصد تضمین می‌کند. در نهایت محدودیت‌های (۱۱و ۱۲) به ترتیب مربوط به تعداد گروه‌هایی است که می‌توان در ایستگاه تولید کرد و تعداد واگن‌هایی است که می‌توان در ایستگاه، بر روی آن‌ها، عملیات مانور انجام داد.

۳. روش حل پیشنهادی

در این بخش، روش پیشنهادی برای حل مساله گروه‌بندی واگن‌ها را بطور کامل معرفی می‌کنیم. روش پیشنهادی، روش شاخه و قیمت است که مشابه روش شاخه و کران[1] بوده، با این تفاوت که در هر گره به جای حل مساله به صورت آزادسازی[2] خطی از الگوریتم تولید ستون[3] استفاده می‌شود. این روش حل، یک روش حل دقیق[4] بهینه‌سازی است [۱۸].

همان طور که در شکل (۱) مشاهده می‌شود، در ابتدا مساله با الگوریتم تولید ستون به طور کامل حل می‌شود. اگر جواب شدنی نباشد الگوریتم خاتمه پیدا می‌کند. در صورت شدنی بودن جواب، کنترل می‌شود که آیا جواب بدست آمده بصورت عدد صحیح است یا خیر. اگر جواب بدست آمده عدد صحیح باشد، الگوریتم خاتمه پیدا کرده و جواب بدست آمده بعنوان خروجی الگوریتم ارائه می‌شود. اگر جواب شدنی و غیر عدد صحیح باشد به منزله گره صفر در درخت شاخه و قیمت است. سپس با توجه به قانون انتخاب متغیر، برای ثابت کردن به مقدار صفر و یک، متغیری از میان متغیرهای y_a که مقداری بین صفر و یک دارند، انتخاب می‌شود. در این مطالعه استراتژی‌های مختلفی نظیر انتخاب متغیر با بیشترین مقدار، کمترین مقدار و یا نزدیکترین مقدار به صفر و یک بکار رفته است. با توجه به نتایج بدست آمده، بهترین قانون که انتخاب متغیر با بیشترین مقدار بین صفر و یک می‌باشد، انتخاب گردید. به ازای هر y_a انتخابی، دو شاخه و گره جدید به درخت شاخه و قیمت اضافه شده و الگوریتم تولید ستون یکبار برای $y_a=1$ و بار دیگر برای $y_a=0$ به طور کامل اجرا می‌شود [۱۹]. $y_a=1$ بدین مفهوم است که تشکیل کمان a در شبکه حتمی بوده و الگوریتم را ملزم به استفاده از این کمان در تشکیل مسیر برای حمل تقاضاها می‌نماید. با این کار مساله فرعی مسیر جدیدی برای حمل تقاضای مربوط به خود ایجاد کرده و به مساله اصلی اضافه می‌کند. بعد از اجرای کامل الگوریتم تولید ستون برای هر دو شاخه، باید وضعیت جواب‌های تولید شده مشخص گردد. هر شاخه می‌تواند دارای یکی از سه وضعیت: (۱) تولید جواب نشدنی، (۲) رسیدن به جواب غیر عدد صحیح و (۳) رسیدن به جواب عدد صحیح شدنی باشد. اگر جواب تولید شده شدنی نباشد، گره جاری بسته می‌شود و الگوریتم به مرحله انتخاب یک گره کاندید برای شاخه زدن می‌رود. در صورتی که جواب شدنی غیر عدد صحیح باشد، بدون انجام عملیات خاصی الگوریتم به مرحله انتخاب یک گره کاندید برای شاخه زدن می‌رود.

در صورتی که جواب تولید شده، شدنی عدد صحیح باشد، بررسی می‌شود که آیا جواب تولید شده از بهترین جواب عدد صحیح تولید شده تاکنون، بهتر است یا خیر. در صورت بهتر نبودن جواب جاری، شاخه بسته شده و الگوریتم به مرحله انتخاب یک گره کاندید برای شاخه زدن می‌رود. در صورتی که جواب تولید شده بهتر باشد، بهترین جواب عدد صحیح تولید شده بروز رسانی می‌شود.

سپس کلیه شاخه‌هایی که جواب بدست آمده آنها از مقدار جواب عدد صحیح بدست آمده، بیشتر باشد، بسته می‌شوند. بعد از بررسی وضعیت جواب‌های تولید شده در هر شاخه، الگوریتم باید گره‌ای را از میان گره‌های کاندید برای شاخه زدن جدید انتخاب کند. در این مطالعه انتخاب گره برای شاخه زدن نیز از استراتژی‌های مختلفی نظیر گره‌ای با بهترین مقدار تابع هدف و گره‌ای با

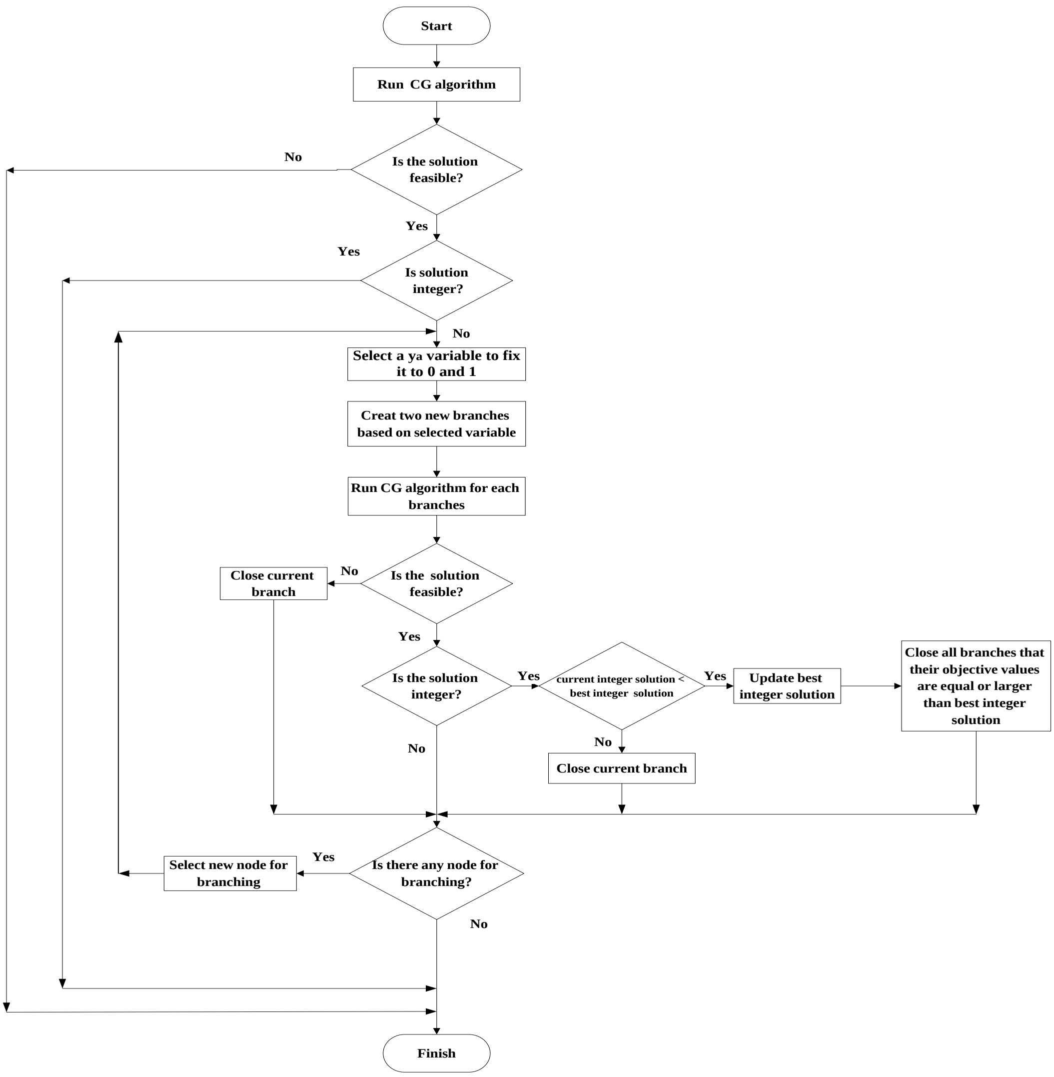
[1] *Branch and Bound*
[2] *Relaxation*
[3] *Column Generation*
[4] *Exact*

بیشترین تعداد متغیر ثابت شده استفاده شده است. با توجه به نتایج بدست آمده، بهترین قانون که انتخاب گره‌ای با بیشترین تعداد متغیر ثابت شده می‌باشد، انتخاب گردید. گره‌ای از میان گره‌های کاندید انتخاب می‌شود و الگوریتم ادامه می‌یابد. در صورتی که گره کاندیدی برای شاخه زدن موجود نباشد، الگوریتم شاخه و قیمت به پایان رسیده است. پس از پایان الگوریتم، اگر جواب عدد صحیح پیدا نشده باشد، بدین مفهوم است که برای مساله مورد نظر هیچ جواب عدد صحیح شدنی وجود ندارد.

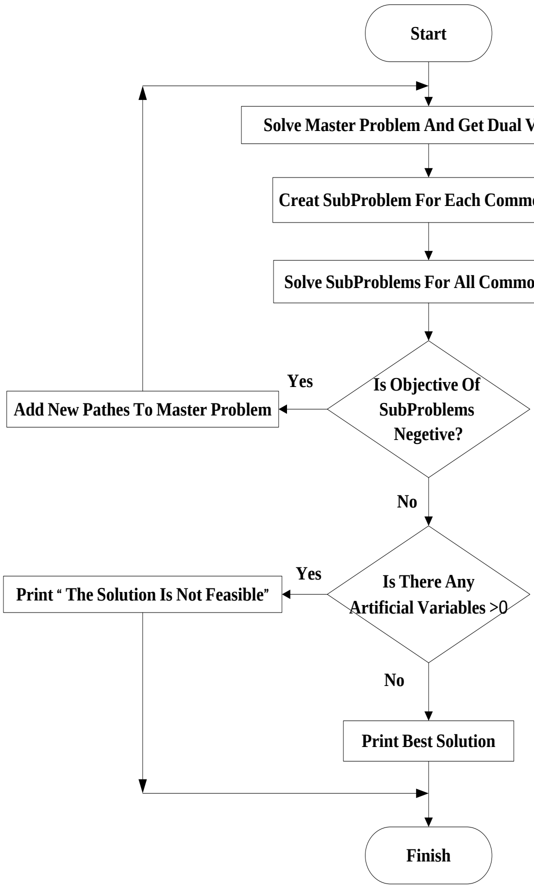
۳-۱. الگوریتم تولید ستون

الگوریتم تولید ستون مبتنی بر روش تجزیه دانتزیگ-ولف می‌باشد [۲۰]. این روش مساله اولیه را به یک مساله اصلی و مجموعه‌ای از چندین مساله فرعی تقسیم می‌کند. در این روش حل جواب مسائل فرعی و مساله اصلی به نحوی با یکدیگر تعامل پیدا می‌کنند که از حل مسائل به ظاهر جدا از یکدیگر همان جواب بهینه مساله اولیه حاصل می‌گردد. تعداد متغیرهای مساله اصلی با هر بار اجرای مسائل فرعی افزایش می‌یابد. پس از حل مساله اصلی با جواب‌های اولیه داده شده به آن و ارسال مقادیر دوگان حاصل از آن به مسائل فرعی، این مسائل حل می‌شوند. اگر مساله فرعی کمینه‌سازی باشد تنها با منفی بودن مقدار تابع هدف و درصورت بیشینه‌سازی بودن مساله، تنها با مثبت بودن مقدار تابع هدف امکان تولید متغیر جدید وجود خواهد داشت [۲۱]. نمودار جریان الگوریتم تولید ستون در الگوریتم پیشنهادی شکل (۲) نشان داده شده است.

در ادامه با الهام از مدل بارنهارت و همکاران [۴]، به تقسیم‌بندی این مدل به مساله اصلی و مسائل فرعی می‌پردازیم



شکل ۱. نمودار جریان روش شاخه و قیمت



شکل ۲. نمودار جریان الگوریتم تولید ستون

مساله اصلی که همان مدل مبتنی بر مسیر (۱۴-۸) می‌باشد، با اضافه کردن متغیرهای دوگان به صورت زیر است.

$$(Master)\min \quad \sum_{k\in K}\sum_{q\in Q(k)} v^k PC_q^k f_q^k \qquad (Duals) \qquad (۱۵)$$

$s.t.$

$$\sum_{k\in K}\sum_{q\in Q(k)} v^k f_q^k \delta_a^q - u_a y_a \le 0 \qquad \forall a\in A \qquad (\beta_a) \qquad (۱۶)$$

$$\sum_{q\in Q(k)} f_q^k = 1 \qquad \forall k\in K \qquad (\chi_k) \qquad (۱۷)$$

$$\sum_{\substack{a\in A\\ orig(a)=i}} y_a \le B(i) \qquad \forall i\in N \qquad (\theta_i) \qquad (۱۸)$$

$$\sum_{k\in K}\sum_{q\in Q(k)}\sum_{\substack{a\in A\\ orig(a)=i}} v^k f_q^k \delta_a^q \le V(i) \quad \forall i\in N \qquad (v_i) \qquad (۱۹)$$

$$y_a \le 1 \qquad \forall a\in A \qquad (\mu_a) \qquad (۲۰)$$

$$f_q^k \ge 0 \qquad \forall q\in Q(k), \forall k\in K \qquad (۲۱)$$

$$y_a \ge 0 \qquad \forall a\in A \qquad (۲۲)$$

در شروع حل مساله، به تعداد تقاضاها مسیرهای مستقیم به مساله اصلی داده می‌شود. این مسیرها ممکن است باعث شود مساله نشدنی گردد. جهت تضمین شدنی بودن مساله، به محدودیت (۱۸) متغیرهای مصنوعی اضافه می‌کنیم که در نهایت این محدودیت به صورت محدودیت (۲۳) تبدیل می‌شود.

$$\sum_{\substack{a\in A\\ orig(a)=i}} y_a - a_i \le B(i) \qquad \forall i\in N \tag{۲۳}$$

برای تعریف مسائل فرعی ابتدا مساله دوگان مساله اصلی به صورت زیر بیان می‌شود.

$$(Dual)\max \quad \sum_{k\in K}\chi_k + \sum_{i\in N}(B(i)\theta_i + V(i)v_i) + \sum_{a\in A}\mu_a \tag{۲۴}$$

$$s.t.$$

$$\forall k\in K, \forall q\in Q(k)\ \sum_{a\in A}[(v^k\delta_a^q)(\beta_a + v_{orig(a)})] + \chi_k - v^k PC_q^k \le 0 \tag{۲۵}$$

$$\sum_{a\in A}(\mu_a - u_a\beta_a) + \sum_{i\in N}\theta_i \le 0 \tag{۲۶}$$

$$\beta_a, \mu_a, v_i, \theta_i \le 0 \tag{۲۷}$$

$$\chi_k \quad free \tag{۲۸}$$

برای کمینه کردن مساله اصلی بدنبال یافتن مسیرهایی (f_q^k) با کمترین هزینه هستیم. هدف مسائل فرعی یافتن این مسیرها است. بدین منظور محدودیت (۲۵) که محدودیت مساله دوگان مربوط به تقاضای f_q^k از مساله اصلی می‌باشد، باید بیشینه شود. تابع هدف مسائل فرعی به صورت زیر تعریف می‌شود.

$$\max:\ \sum_{a\in A}[(v^k\delta_a^q)(\beta_a + v_{orig(a)})] + \chi_k - v^k PC_q^k \tag{۲۹}$$

در الگوریتم استفاده شده با ضرب (۱-) در این تابع هدف، مسائل فرعی به کمینه‌سازی تبدیل شده که در نهایت به صورت زیر تعریف می‌شود.

$$(Subproblem)\min:\ \sum_{a\in A}[(v^k\delta_a^q)(-\beta_a - v_{orig(a)})] - \chi_k + v^k PC_q^k \tag{۳۰}$$

$$s.t.$$

$$Balance \quad \sum_{\substack{a\in A\\ orig(a)=i}}\delta_a^q - \sum_{\substack{a\in A\\ dest(a)=i}}\delta_a^q = \begin{cases}1 & orig(a)=i\\ -1 & dest(a)=i\\ 0 & otherwise\end{cases} \quad \forall i\in N \tag{۳۱}$$

$$\delta_a^q \in \{0,1\} \tag{۳۲}$$

مجموعه محدودیت‌های (۳۱) جهت یافتن مسیر جدید به مسائل فرعی اضافه شده است و با وجود یک مساله فرعی برای هر تقاضا، تضمین می‌کند که تقاضا از مبدا خارج شده و به مقصد وارد گردد و همچنین برای ایستگاه‌های بین راهی در صورت ورود تقاضا به آن، حتماً از آن خارج شود. در مساله گروه‌بندی واگن‌ها، کمان‌ها که همان گروه واگن‌ها هستند، دارای ظرفیت نیستند. بنابراین به جای ظرفیت کمان (u_a) عدد بزرگی در نظر گرفته می‌شود. با توجه به این موضوع، مساله اصلی (۲۲-۱۵) یک مساله برنامه‌ریزی خطی آزاد شده ضعیف[1] است. همان طور که در محدودیت (۱۶) مشاهده می‌شود میان ظرفیت کمان (u_a) و متغیر تصمیم تشکیل کمان (y_a) رابطه معکوسی وجود دارد. بدین مفهوم که با بزرگتر شدن u_a، مدل مقادیر کوچکتری برای متغیر y_a تخصیص می‌دهد. جهت تقویت حدود متغیر y_a، یکسری محدودیت را به مساله اصلی اضافه می‌کنیم. این محدودیت‌ها "محدودیت‌های فورسینگ[2]" نامیده می‌شود. تکنیک استفاده از این نوع محدودیت‌ها در بارنهارت و همکاران [۲۲] توضیح داده شده است. تنها مشکل بکارگیری این محدودیت‌ها، بزرگتر شدن مدل می‌باشد. محدودیت‌های فورسینگ استفاده شده در روش حل پیشنهادی بصورت زیر بیان می‌شود.

$$\sum_{q\in Q(k)} f_q^k\delta_a^q - y_a \le 0 \qquad \forall a\in A, \forall k\in K \tag{۳۳}$$

با اضافه کردن این محدودیت‌ها به مساله، تعداد شاخه‌های درخت شاخه و قیمت کاهش یافته و سریع‌تر به جواب بهینه خواهیم رسید. با توجه به اینکه به ازای هر کمان یک محدودیت فورسینگ در مدل خواهیم داشت، بنابر این به تعداد تقاضا ضرب در تعداد کمان‌ها محدودیت فورسینگ اضافه می‌شود. در شبکه‌های بزرگ با تعداد تقاضای زیاد ، تعداد این محدودیت‌ها بسیار زیاد خواهد بود. با اضافه کردن محدودیت‌های فورسینگ به مساله اصلی، مقادیر دوگان حاصل از این محدودیت‌ها را با $\phi_a^k \le 0$

1 *Weak LP Relaxation*

2 *Forcing Constraints*

نشان می‌دهیم، این مقادیر که حاصل از حمل تقاضای k روی کمان a می‌باشد، به مسائل فرعی ارسال می‌گردد. در نهایت محدودیت (۲۵) با اضافه شدن محدودیت‌های فورسینگ بصورت زیر تغییر می‌کند.

$$\sum_{a\in A}[\delta_a^q(v^k\beta_a + v^k v_{orig(a)} + \phi_a^k)] + \chi_k - v^k PC_q^k \le 0 \qquad \forall k \in K, \forall q \in Q(k) \qquad (۳۴)$$

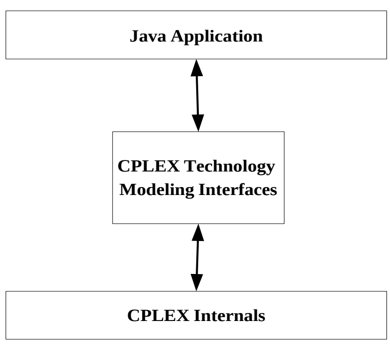
در نتیجه تابع هدف مساله فرعی تقاضای k بصورت زیر تغییر خواهد کرد.

$$\min \quad \sum_{a\in A}[\delta_a^q(-v^k\beta_a - v^k v_{orig(a)} - \phi_a^k)] - \chi_k + v^k PC_q^k \qquad (۳۵)$$

۴. ارزیابی تجربی

در این بخش به ارزیابی اثربخشی یا دقت روش حل پیشنهادی می‌پردازیم. بدین منظور چندین مساله شبیه‌سازی شده با ابعاد مختلف تولید کرده و مورد استفاده قرار دادیم که مشخصات آن‌ها در جدول (۱) آورده شده است.

لازم به ذکر است مسائل نمونه با این فرض که تمام ایستگاه‌ها دارای ظرفیت تشکیل حداقل یک بلاک را دارا باشند، تولید شده است. روش حل پیشنهادی با استفاده از زبان برنامه نویسی جاوا کد نویسی شده است. زبان برنامه نویسی جاوا یکی از قدرتمندترین زبان‌های برنامه‌نویسی برای محاسبات مهندسی بوده و دارای سرعت حل بسیار بالایی است. مسائل نمونه بر روی یک کامپیوتر با مشخصات Intel Core 2 Duo CPU 4GB RAM اجرا شده است. در برنامه جاوا از موتور حل نرم افزار CPLEXبرای حل مسائل برنامه‌ریزی آزاد شده خطی استفاده شده است. همانطور که در شکل (۳) نشان داده شده است، برنامه نوشته شده به زبان جاوا از طریق رابطهای مدل سازی نرم‌افزار CPLEX که شامل مجموعه‌ای از کلاس‌ها و متدهای از پیش نوشته شده است، با موتور حل CPLEX ارتباط برقرار می‌کند. بدین مفهوم که برنامه نوشته شده در صورت نیاز به حل مساله برنامه‌ریزی خطی از کلاس‌ها و متدهای ILOG CPLEX استفاده می‌کند.



شکل ۳. طرح کلی استفاده از موتور حل CPLEX

جدول ۱. مشخصات نمونه‌های شبیه‌سازی شده

Instances	Station	Demands	Constraints	y_a Variable(	f_q^k Variable(
1	5	3	52	20	60
2	5	5	66	20	100
3	5	10	101	20	200
4	10	5	171	90	450
5	10	10	231	90	900
6	10	30	471	90	2700
7	20	40	1301	380	15200

برای مقایسه دقت الگوریتم پیشنهادی برای حل مساله گروه‌بندی واگن‌ها با روش‌های دقیق حل مساله، مسائل تولید شده را به فرمت mps تبدیل کرده و در محیط تعاملی[1] نرم افزار CPLEX و با الگوریتم حل مسائل MIP این نرم افزار، حل شده است.

جدول(۲) نتایج بدست آمده از الگوریتم پیشنهادی شاخه و قیمت و الگوریتم MIP نرم افزار CPLEX را نشان می‌دهد. همانطور که مشاهده می‌شود، الگوریتم ارائه شده در همه نمونه‌های شبیه-سازی شده به جواب بهینه رسیده است که این مساله نشان دهنده درستی و اثربخشی الگوریتم پیشنهادی است. زمان حل مسائل نمونه با الگوریتم پیشنهادی بسیار کوتاه بوده و در حد چند ثانیه الگوریتم به جواب بهینه رسیده است.

لازم به ذکر است که حل مسائل با ابعاد بزرگ و با ترکیب الگوریتم پیشنهادی با روش‌های فرا ابتکاری توسط نویسندگان این مقاله در حال انجام بوده و در مقالات بعدی منتشر خواهد شد.

[1] *Interactive area*

جدول ۲. نتایج محاسباتی

Instances	Station	Demands	Branch & Price	CPLEX	GAP (%)
1	5	3	1400	1400	0
2	5	5	3252	3252	0
3	5	10	3648	3648	0
4	10	5	3125	3125	0
5	10	10	4194	4194	0
6	10	30	14795	14795	0
7	20	40	37185	37185	0

مراجع

[1] Bodin, L.D., Golden, B.L., Schuster, A.D., Romig, W., *"A Model for the Blocking of Trains."* Transportation Research, Vol. 14, No. 1, PP. 115-120.

[2] Van Dyke, C.D., "*The Automated Blocking Model: A Practical Approach to freight Railroad Blocking Plan Development*." Transportation Research Forum, Vol. 27, No. 1, 1986, pp. 116-121.

[3] Newton, H.N., Barnhart, C., Vance, P.H., "*Constructing Railroad Blocking Plans to Minimize Handling Costs*." Transportation Science, Vol. 32, No. 4, 1998, pp. 330-345.

[4] Barnhart, C., Jin, H., Vance, P.H., *"Railroad Blocking: A Network Design Application*." Operations Research, Vol. 48, No. 4, 2000, pp. 603-614.

[5] Ahuja, R.K., Jha, K C., Liu, J., "*Solving Real-Life Railroad Blocking Problems."* Interfaces, Vol. 37, No. 5, 2007, pp. 404-419.

[6] Jha, K.C., Ahuja, R.K., Guvenc. *"New Approaches for Solving the Block-to-Train Assignment Problem*." Networks, Vol. 51, pp. 48-62.

[7] Magnanti, T.L., Wong, R.T., *"Network Design and Transportation Planning: Models and Algorithms*." Transportation Science, Vol. 18, 1984, pp. 1-55.

[8] Balakrishnan, A., Magnanti, T.L., Wong, R.T., *"A Dual-Ascent Procedure for Large-Scale Uncapacitated Network Design*." Operations Research, Vol. 37, No. 5, 1989, pp. 716-740.

[9] Crainic, T.G., Gendreau, M., Farvolden, J.M., *"A Simplex-Based Tabu Search for Capacitated Network Design*." INFORMS Journal on Computing, Vol. 12, No. 3, 2000, pp. 223-236.

[10] Rodríguez-Martín, I., JoséSalazar-González, J., "*A Local Branching Heuristic for the Capacitated Fixed-Charge Network Design Problem*." Computers & Operations Research, Vol. 37, No. 3, 2010, pp. 575-581.

[11] Pedersen, M. B., Crainic, T.G., Madsen, O.B.G., *"Models and Tabu Search Meta-Heuristics for Service Network Design with Asset-Balance Requirements*." Transportation Science, Vol. 43, No. 2, 2009, pp. 158-177.

[12] Hoff, A., Lium, A.G., Lokketangen, A., Crainic, T.G., *"A Metaheuristic for Stochastic Service Network Design*." Journal of Heuristics, Vol. 16, No. 5, 2009, pp. 653-679.

[13] Frangioni, A., Gendron, B.,. *"0 –1 Reformulations of the Multicommodity Capacitated Network Design Problem.*" Discrete Applied Mathematics, Vol. 157, No. 6, 2009, pp. 1229-1241.

[14] Katayama, N., Chen, M., Kubo, M., *"A Capacity Scaling Heuristic for the Multicommodity Capacitated Network Design Problem.*" Journal of Computational and Applied Mathematics, Vol. 232, No. 1, 2009, pp. 90-101.

[15] Holmberg, K., Joborn, M., Melin, K., *"Lagrangian Based Heuristics for the Multicommodity Network Flow Problem with Fixed costs on Paths*." European Journal of Operational Research, Vol. 188, No. 1, 2007, pp. 101-108.

[16] Yaghini, M., Alimohammadian, A.R., Sharifi, S., *"A Goal Programming Technique for Railroad Passenger Scheduling*." Management Science Letters, Vol. 2, No. 2, 2012, pp. 535-542.

[17] Yaghini, M., Alimohammadian, A.R., Sharifi, S., "*A Hybrid Method to Solve Railroad Passenger Scheduling Problem*. Management Science Letters, Vol. 2, No. 2, pp. 543-548.

[18] Barnhart, C., Johnson, E.L., Nemhauser, G.L., Savelsbergh, M.W.P., Vance, P.H., *"Branch-and-Price: Column Generation for Solving Huge Integer Programs."* Operations Research, Vol. 46, No. 3, 2012, pp. 316-329.

[19] Vanderbeck, F., *"On Dantzig-Wolfe Decomposition in Integer Programming and Ways to Perform Branching in a Branch-and-Price Algorithm*." Operations Research, Vol. 48, No. 1, 2000, pp. 111-128.

[20] Dantzig, G.B., Wolfe, P., "*Decomposition Principle for Linear Programs*." Operations Research, Vol. 8, No. 1, 1960, pp. 101-111.

[21] Desrosiers, J., Lubbecke, M.E., *"A Primer in Column Generation*." G. Desaulniers, J. Desrosiers and M. M. Solomon, eds., Springer-Verlag, New York, Chapter.1, 2005, pp. 1-32.

[22] Barnhart, C., Johnson, E.L., Nemhauser, G.L., Sigismondi, G., Vance, P.H., "*Formulating a Mixed Integer Programming Program to Improve Solvability."* Operations Research, Vol. 41, No. 1, 1993, pp. 1013-101.