برنامه ریزی خطوط هواپیمایی با در نظر گرفتن محدودیت‌های عملیاتی

شکوه خردمند[۱]، محمد محمدی[۲،*]، بهمن نادری[۳]

[۱] دانشجوی کارشناسی ارشد، دانشگاه خوارزمی، گروه مهندسی صنایع، البرز، ایران
[۲] دانشیار، دانشگاه خوارزمی، گروه مهندسی صنایع، البرز، ایران(عهده‌دار مکاتبات)
[۳] استادیار، دانشگاه خوارزمی، گروه مهندسی صنایع، البرز، ایران

تاریخ دریافت: شهریور ۱۳۹۵، اصلاحیه: آبان ۱۳۹۵، پذیرش: دی ۱۳۹۵

چکیده

از گذشته تا به امروز به علت بالا بودن هزینه‌های مربوط به صنعت هوایی و هواپیماها، برنامه‌ریزی برای این صنعت از اهمیت زیادی برخوردار بوده است. یکی از مسائل مربوط به صنعت هوایی، مسئله‌ی مسیریابی هواپیماها و تعیین تعداد هواپیماهای مورد نیاز در هر برنامه‌ی پروازی است. این مسائل، بخش قابل توجهی از هزینه‌های صنعت هوایی را برعهده دارند. از طرفی با توجه به یک سری محدودیت‌ها و قوانینی که شرکت‌های هواپیمایی در برنامه‌های پروازی برای هر هواپیما دارند، به دست آوردن یک برنامه‌ی پروازی مناسب به منظور تعیین تعداد هواپیماهای مورد نیاز، کمینه سازی هزینه‌های مربوط به هواپیماها و هزینه‌های ناشی از فروش از دست رفته دارای اهمیت زیادی است. از طرفی با توجه به NP-hard بودن این مسئله، حل دقیق مدل در سایزهای بزرگ امکان پذیر نمی‌باشد؛ لذا با کمک الگوریتم‌های فراابتکاری چون ژنتیک و شبیه سازی تبرید به حل مسئله در سایزهای بزرگ پرداختیم و نتایج حاصل از آن‌ها را با یکدیگر مقایسه کردیم و در نتیجه به برتری الگوریتم شبیه سازی تبرید نسبت به الگوریتم ژنتیک رسیدیم.

واژگان کلیدی: صنعت هوایی، مسیریابی هواپیما، برنامه‌ریزی خدمه، فروش از دست رفته، الگوریتم‌های فراابتکاری

۱- مقدمه

صنعت هوایی از گذشته تا به امروز همواره از گرانترین منابع عملیاتی محسوب شده است. هزینه‌های مربوط به سوخت، هزینه‌های مربوط به خدمه، تعمیر و نگهداری، موجودی و ... بخشی از هزینه‌های مرتبط به این صنعت می‌باشند. لذا برنامه‌ریزی برای کمینه‌سازی این هزینه‌ها از اهمیت زیادی برخوردار است. اگر بخواهیم یک دسته بندی کلی از مسائل موجود در صنعت هواپیمایی داشته باشیم، می‌توان گفت که این مسائل به دو دسته قابل تفکیک اند:

۱- مسائل مربوط به برنامه‌ریزی[۱] و انتساب خدمه[۲]
۲- مسائل مربوط به مسیریابی هواپیماها[۳] و تعیین تعداد هواپیمای مورد نیاز

بخش اول خود به دو قسمت برنامه‌ریزی خدمه و انتساب آن قابل تفکیک است. در برنامه‌ریزی خدمه هدف، بدست آوردن یک برنامه‌ی پروازی منظم در یک دوره مشخص برای خدمه است به گونه ای که هزینه‌های مربوط به خدمه کمینه شود. در انتساب خدمه نیز هدف تخصیص این خدمه‌ها به مسیرهای مشخص می‌باشد.

بخش دوم نیز شامل برنامه‌ریزی و زمانبندی هواپیما می‌باشد که توابع هدف مختلفی همچون کمینه سازی هزینه‌های مربوط به هواپیماها با تعیین تعداد هواپیمای مورد استفاده و یا تخصیص یک تعداد هواپیمای مشخص به مسیرهای مختلف جهت رفع تقاضاها دارد[۲,۱].

در هر کدام از این مسائل همواره محققان زیادی به تحقیق و مدل سازی پرداخته اند که در اینجا تاریخچه‌ی هر کدام از این مسائل را به اختصار بیان می‌کنیم.

مسائل مربوط به خدمه برای اولین بار توسط روبین [۴][۸] در سال ۱۹۷۳ بیان شد که بعد از ان محققان دیگری همچون گربراکت (۱۹۷۸)[۱۴]، لاوویه [۵] (۱۹۸۸)[۱۵]، چو[۶](۱۹۹۶)[۹] ، انبیل[۷] (۲۰۰۲)[۵]، ونگ [۸]

1 .Crew pairing
2 .Crew rostering
3 .Aircraft Routing

*mohammadi@khu.ac.ir
4 .Rubin
5 .lavoie
6 .Chu
7 .Anbil
8 .Wang

(۲۰۰۲)[۱۰] و ... به حل مسائل برنامه‌ریزی خدمه در قالب مسائل پوشش دهی مجموعه [1]و یا پوشش‌دهی جزئی[2] پرداختند. مسائل مربوط به انتساب خدمه نیز بخش مهمی از عملکرد هواپیما را بر عهده دارند و همواره جزء مسائل جالب از دید محققین بوده اند که از جمله‌ی آن‌ها می‌توان به سیستم انتساب خدمه کارمن[3] اشاره کرد که در سال ۱۹۹۹ و ۲۰۰۰ توسط هیجورینگ[4]، ویگلمنز [5] و کاریش[6]، [۷] از جنبه‌های مختلف این سیستم استفاده شد. همچنین در سال ۲۰۰۱ کونیگ[7]، دیوید[8] و اشتراوس[9][۴] به تحقیقاتی در این زمینه پرداختند. در زمینه‌ی مسائل مربوط به مسیریابی هواپیماها نیز محققان زیادی فعالیت نموده اند. این محققان در ابتدا با استفاده از روش‌های شاخه و کران و برش هواپیما به برنامه‌ریزی برای هواپیماها پرداختند. (۱۹۷۱-۱۹۶۹) لوین[10][۱۲]، تئودورویک[11] (۱۹۸۴) [۱۸]، (۱۹۸۶) لی[12][۱۳]، از ان جمله اند. هم چنین در سال ۱۹۸۱ فیشر[13][۶] و در سال ۱۹۹۶ یان و یونگ[14][۱۹] و در سال ۲۰۰۲ تسنگ[15]و یان [۲۰] با کمک تکنیک‌هایی چون آزاد سازی لاگرانژین به حل این مسائل پرداختند.

همچنین سمردی [16]و همکارانش در سال ۲۰۰۴ یک مدل پرواز خروجی، زمانبندی مجدد و مسیریابی هواپیما را با هدف کمینه سازی تعداد مسافران از دست رفته مورد انتظار گسترش دادند [۱۶]. لن[17]و همکارانش نیز در سال ۲۰۰۶ تلاش کردند تا با ارائه‌ی یک برنامه‌ریزی قوی به تخمین تاخیر پروازها برای هر پرواز و کم کردن میزان تاخیرها بپردازند [۱۷]. لازم به ذکر است که علاوه بر موارد گفته شده گروهی از محققان نیز مقالاتی در مورد ترکیب چند نمونه از این نوع مسائل با هم ارائه کردند. تعدادی از آن‌ها که در این زمینه فعالیت نمودند به شرح زیر هستند:

- کوردئو[18] (۲۰۰۱) با تلفیق برنامه‌ریزی خدمه و برنامه‌ریزی هواپیما به ارائه‌ی یک مدل جدید دست یافت [۳].
- کلابجان [19]و همکارانش (۲۰۰۲) مدلی را حل کردند که در آن ابتدا مسئله برنامه‌ریزی خدمه و سپس مسیریابی هواپیما مدل شده بود و برای خدمه محدودیت تعداد هواپیما قائل شده بودند [۱۱].

1 .Set Covering Problem
2 .Set Partitioning Problem
3 .Carmen
4 .Hjorring
5 .wagelmans
6 .karisch
7 .konig
8 .Dawid
9 .Strauss
10. Levin
11. Teodorovics
12. Lee
13. Fisher
14. Yan & Yong
15. Tseng
16. somardi
17. Lan
18. Cordeau
19. Klabjan

امروزه مسئله‌ای که به شدت با آن روبرو هستیم بحث تعیین تعداد هواپیمای مورد نیاز و تخصیص این هواپیماها به پروازهاست؛ چرا که بخش عمده ای از هزینه‌های صنعت هوایی مربوط به هزینه‌ی استفاده و یا عدم استفاده از هواپیماست. با توجه به مطالعات و تحقیقات انجام شده از دهه‌ی ۵۰ تا کنون به این نتیجه می‌رسیم که اکثر مطالعات انجام شده پیرامون برنامه‌ریزی برای کاهش هزینه‌های خدمه و تخصیص هواپیماست، در صورتی که بخش اعظم هزینه‌ها مربوط به استفاده و یا عدم استفاده از هواپیما را در بر می‌گیرد که از جمله‌ی آن‌ها هزینه‌های مربوط به سوخت هواپیماست که در راس هزینه‌ها قرار دارد. مطالعات انجام شده تاکنون به ندرت به تعیین حداقل تعداد هواپیمای لازم و تخصیص این هواپیماها به مسیرهای مشخص برای یک مجموعه پرواز پرداخته است، به گونه ای که هزینه‌های مربوطه را کمینه کند. در واقع ما در این مقاله با ارائه‌ی یک مدل ترکیبی، حداقل تعداد هواپیماهای مورد نیاز برای یک مجموعه پرواز را به گونه‌ای که هزینه‌های پنهان و اشکار مربوط به هواپیما را حداقل کند، می‌یابیم. از طرفی موضوع دیگری که در واقعیت وجود دارد و در مقاله‌های انجام شده تاکنون در مورد آن صحبت نکرده اند، بحث فروش از دست رفته[20] و از دست دادن اعتبار شرکت[21] است. چرا که در بسیاری موارد، شرکت با برآورد مالی که انجام می‌دهد مشاهده می‌کند که تخصیص پرواز به یک مسیر اصلا به صرفه نیست و شرکت هواپیمایی حاضر است یک سری هزینه‌هایی ناشی از فروش از دست رفته متحمل شود ولی پروازی برای آن مسیر در نظر نگیرد. لذا ما در این مقاله با ارائه‌ی مدلی جدید به تعیین تعداد هواپیمای مورد نیاز در یک برنامه‌ی پروازی به گونه ای که حداکثر رضایت مشتری را با کمترین هزینه برآورد کند، پرداخته ایم.

ما در این مقاله در بخش دوم به معرفی مدل خود، تشریح متغیرها و پارامترهای مدل می‌پردازیم. در بخش سوم به معرفی اجمالی روش‌های حل مدل خود می‌پردازیم. در بخش چهارم به ارائه‌ی نتایج حاصل از حل مدل پرداخته می‌شود. در بخش پنجم نیز به نتیجه‌گیری و ارائه‌ی راهکارهایی برای توسعه‌ی آتی مدل پرداخته می‌شود.

۲- تعریف مسئله و بیان مدل ریاضی

هدف ما از ارائه‌ی این مدل به دست آوردن حداقل تعداد هواپیمای مورد استفاده و تخصیص این هواپیماها به مسیرهای مختلف مورد تقاضا و تعیین تعداد تقاضای از دست رفته در یک مجموعه برنامه‌ی پروازی است به گونه ای که هزینه‌ها کمینه شود.

در واقع تابع هدف مدل ما کمینه سازی تعداد هواپیماها و تعداد تقاضای از دست رفته به گونه‌ای است که کمترین هزینه و بیشترین سطح رضایت را داشته باشیم. در این مدل با توجه به قوانین و مقررات عمومی که

20 .Lost sale
21 .Good will

شرکت‌های هواپیمایی دارند یکسری محدودیت‌ها برای مدل خود قائل شده ایم که این محدودیت‌ها عبارتند از:

۱. محدودیت مسافت طی شده در هر مجموعه برنامه‌ی پروازی: هر هواپیما موظف است که از حداکثر مسافت مجاز پروازی در هر مجموعه برنامه‌ی پروازی تجاوز نکند. به عبارتی مجموع مسیرهای طی شده در هر هواپیما همواره باید از ماکزیمم مسافت مجاز کمتر باشد.

۲. محدودیت بازگشت به مبدأ: هر هواپیما موظف است در هر مجموعه برنامه‌ی پروازی به فرودگاه مبدأ خود بازگردد.

۳. محدودیت ورود و خروج یکسان: در هر فرودگاه پروازها باید به نحوی برنامه‌ریزی شوند که هر هواپیما هنگامی که وارد یک فرودگاه شد برای پرواز بعدی از همان فرودگاه موظف به حرکت باشد. به عبارتی مقصد پرواز همواره باید مبدا پرواز بعدی باشد.

حال به معرفی پارامترهای مدل می پردازیم:

n: تعداد شهرها

i,j :زیر شماره برای شهرها $i,j=\{1,2,3,\dots,n\}$

m: تعداد هواپیماهای در دسترس

l: زیر شماره برای هواپیماها $l=\{1,2,3,\dots,m\}$

P_{ij} : میزان تقاضای موجود بین شهر i و شهر j

C_{ij} :هزینه سفر از شهر i به شهر j

e_{ij} :هزینه کمبود از شهر i به شهر j

F_l : هزینه ثابت برای هر هواپیما

A : ظرفیت هر هواپیما

O : ماکزیمم ظرفیت مجاز پرواز برای هر هواپیما

متغیرها :

X_{ijl} : متغیر باینری برای استفاده یا عدم استفاده از هواپیمای l ام برای پرواز از شهر i به شهر j .

Y: ₁متغیر باینری که اگر از هواپیمای l ام استفاده بشود مقدار یک و در غیر اینصورت مقدار صفر می‌گیرد .

T_{ij} متغیر پیوسته برای میزان کمبود موجود از شهر iبه شهر j

در این بخش ما به ارائه‌ی مدل ریاضی مسئله خود می‌پردازیم و محدودیت‌ها و فرضیات حل خود را ذکر می‌کنیم:

$$\text{Min } \sum_{l=1}^{m} F_l Y_l + \sum_{i=1}^{n}\sum_{j=1|j\neq i}^{n} c_{i,j}\left(\sum_{=1}^{m} X_{i,j,l}\right) + \sum_{i=1}^{n}\sum_{j=1|j\neq i}^{n} e_{i,j}T_{i,j} \quad (1)$$

subject to:

$$T_{i,j} \geq p_{i,j} - \sum_{l=1}^{m} AX_{i,j,l} \qquad \forall_{i,j\neq i} \quad (2)$$

$$\sum_{j=1|j\neq i}^{n} X_{i,j,l} = \sum_{j=1|j\neq i}^{n} X_{j,i,l} \qquad \forall_{i,l=1,\dots,m} \quad (3)$$

$$OY_l \geq \sum_{i=1}^{n}\sum_{j=1|j\neq i}^{n} d_{i,j}X_{j,i,l} \qquad \forall_{l=1,\dots,m} \quad (4)$$

$$T_{i,j} \geq 0 \qquad \forall_{i,j\neq i} \quad (5)$$

$$X_{i,j,l} \in \{0,1\} \qquad \forall_{i,j\neq i,l=1,\dots,m} \quad (6)$$

$$Y_l \in \{0,1\} \qquad \forall_{l=1,\dots,m} \quad (7)$$

در این مدل تابع هدف ما از سه قسمت تشکیل شده است: بخش اول مربوط به هزینه‌های ناشی از استفاده یا عدم استفاده از هر هواپیما است. در واقع اگر از یک هواپیما استفاده شود یک هزینه‌ی ثابتی ناشی از استفاده از آن هواپیما برای ما ایجاد می‌شود که مستقل از مسیرهای پیموده شده توسط هر هواپیما است. بخش دوم مربوط به هزینه‌های هر هواپیما در هر مسیر پیموده شده می‌باشد. این هزینه‌ها نیز به تعداد مسیرهایی که هر هواپیما می‌پیماید بستگی دارد. در واقع بسته به اینکه این هواپیماها از چه تعداد مسیر عبور می‌کنند این هزینه‌ها بدست می‌آیند. بخش سوم مربوط به هزینه‌های ناشی از کمبود هر هواپیما است. در واقع هر شرکت هواپیمایی اگر نتواند تمام تقاضا را برطرف کند یک هزینه ای ناشی از، از دست رفتن اعتبار شرکت پرداخت می‌کند که به آن هزینه‌ی فروش از دست رفته گویند. این هزینه به میزان تقاضای برآورده نشده بستگی دارد.

در مدل بیان شده رابطه (۲) نشان از روش محاسبه میزان کمبود موجود بین دو شهر i و j دارد بدین صورت که اگر اختلاف بین میزان تقاضای کل بین دو شهر و تعداد تقاضاهایی که از شهر iبه شهر j برآورده شده اند را محاسبه کنیم همواره میزان کمبود ازین اختلاف بیشتر یا مساوی خواهد شد. رابطه (۳) برای بازگشت هر هواپیما به مبدا و همچنین ورود و خروج یکسان در هر فرودگاه می‌باشد. این محدودیت نشان می‌دهد که مقصد هر پرواز باید مبدا پرواز بعدی باشد. رابطه (۴) بیانگر ان است که هر هواپیما موظف است درصورت استفاده از آن هواپیما در هر مجموعه پروازی از ماکزیمم مسافت مجاز پرواز، بیشتر پرواز نکند. یعنی میزان مسافت طی شده در هر مجموعه‌ی پروازی برای هر هواپیما حداکثر به اندازه O است. رابطه (۵) نشان دهنده‌ی پیوسته بودن میزان کمبود و رابطه (۶) و (۷) نشان دهنده‌ی باینری بودن دو متغیر دیگر است.

۳- روش‌های حل

در این بخش به معرفی اجمالی الگوریتم ژنتیک و شبیه سازی تبرید می‌پردازیم و مدل خود را با این دو روش حل می‌کنیم:

۳-۱- الگوریتم ژنتیک

الگوریتم ژنتیک یکی از الگوریتم‌های جستجوی تصادفی است که ایده آن برگرفته از طبیعت می‌باشد. در واقع الگوریتم‌های ژنتیک از اصول انتخاب طبیعی داروین برای یافتن فرمول بهینه جهت پیش‌بینی یا تطبیق الگو استفاده می‌کنند. الگوریتم‌های ژنتیک با نماد اختصاری (GA) تکنیک جستجویی در علم رایانه برای یافتن راه‌حل تقریبی برای بهینه‌سازی و مسائل

جستجو است. الگوریتم ژنتیک نوع خاصی از الگوریتم‌های تکامل است که از تکنیک‌های زیست شناسی فرگشتی مانند وراثت و جهش استفاده می‌کند. این الگوریتم برای اولین بار توسط جان هلند در سال ۱۹۶۷معرفی شد. ایده اساسی این الگوریتم انتقال خصوصیات موروثی توسط ژن‌هاست.

درالگوریتم‌های ژنتیک ابتدا به طور تصادفی یا الگوریتمیک، چندین جواب برای مسئله تولید می‌شود. این مجموعه جواب را جمعیت اولیه و هر جواب را یک کروموزوم می‌نامند. سپس با استفاده از عملگرهای الگوریتم ژنتیک پس از انتخاب کروموزوم‌های بهتر، کروموزوم‌ها را باهم ترکیب کرده و جهشی در آن‌ها ایجاد می‌کنند. در نهایت نیز جمعیت فعلی را با جمعیت جدیدی که از ترکیب و جهش در کروموزوم‌ها حاصل می‌شود، ترکیب می‌کند.

در واقع در الگوریتم ژنتیک نخستین چیزی که باید مشخص شود نحوه‌ی انتخاب والد برای ترکیب و جهش و بهبود جواب اولیه است. برای هر کدام از عملگرهای ترکیب و جهش و نحوه‌ی انتخاب والد روش‌های مختلفی وجود دارد که ما در این بخش به طور مختصر عملگرهایی که ما در مقاله استفاده کرده ایم را معرفی می‌کنیم:

الف) نحوه‌ی انتخاب والد

برای انتخاب والد راه حل‌های مختلفی وجود دارد.یکی از این راه‌ها انتخاب دو کروموزوم به صورت تصادفی از مجموعه جواب اولیه است. در واقع ما با انتخاب دو کروموزوم به صورت تصادفی به عنوان دو والد به ترکیب و جهش روی آن‌ها می پردازیم و مجموعه جواب جدیدی به وجود می‌آوریم و سپس بهترین جواب این مجموعه جدید را با بهترین جواب جمعیت اولیه مقایسه می‌کنیم و در صورت بهبود جواب، آن را جایگزین جواب قبلی می‌کنیم و دوباره این مراحل را تا رسیدن به شرط توقف ادامه می‌دهیم.

ب) ترکیب

برای ترکیب دو والد روش‌های مختلفی موجود است. هر کدام از این روش‌ها بسته به اینکه جواب اولیه ما به صورت اعداد تصادفی، جایگشتی و یا باینری باشد، استفاده می‌شود. در این مدل با توجه به اینکه مدل ما به صورت باینری می‌باشد، از یکی از روش‌های مربوط به باینری استفاده کردیم که ترکیب تک نقطه‌ای [1]نام دارد. در این نوع از عملگرها در ابتدا یک نقطه به تصادف روی دو والد جدا کرده و این دو والد را در آن نقطه می‌بریم و با هم ژن‌هایشان را جابه جا می‌کنیم تا فرزند جدید ایجاد شود. به عنوان مثال داریم:

۱	۰	۰	۱	۰	۱	۰	۱	۱	۱

۰	۱	۰	۱	۰	۱	۱	۱	۰	۰

حال فرزند

۱	۰	۰	۱	۰	۱	۱	۱	۰	۰

ج) جهش

در قسمت قبل، فهمیدیم که به کمک ترکیب، کروموزوم جدیدی از دو والد تولید شد. سپس اگر شرایط وارد شدن به فرایند جهش را نیز داشت این کروموزوم، وارد فرایند جهش می‌شود. عملگرهای جهش نیز همانند ترکیب با توجه به باینری، تصادفی و یا جایگشتی بودن متغیرها انواع مختلفی دارند که ما در این مقاله برای حل مدل خود از عملگر شیفت[2] استفاده کردیم. در این نوع عملگر در یک مجموعه جواب، دو ژن را به تصادف انتخاب می‌کنیم سپس ژن مربوط به نقطه اول را به جایگاه بعد از ژن دوم منتقل می‌کنیم. به عنوان مثال فرض کنید مجموعه جواب ما یک مجموعه باینری به صورت زیر باشد.

۱	۱	۰	۱	۰	۱	۱	۰	۱	۱

دو ژن به تصادف روی این مجموعه انتخاب شد، حال ژن اول را به جایگاه بعد از ژن دوم منتقل می‌کنیم و داریم:

۱	۱	۱	۰	۱	۱	۰	۰	۱	۱

به این ترتیب یک جواب جدید تولید می‌شود که جایگزین یکی از والدها می‌شود. این روند به همین صورت ادامه می‌یابد تا به شرط توقف برسیم. این الگوریتم شرایط توقف مختلفی وجود دارد که عبارتند از :

۱- رسیدن به تعداد ثابتی از نسل‌ها
۲- رسیدن به یک زمان مشخص
۳- رسیدن به تعداد تکرار و شخص
۴- و ...

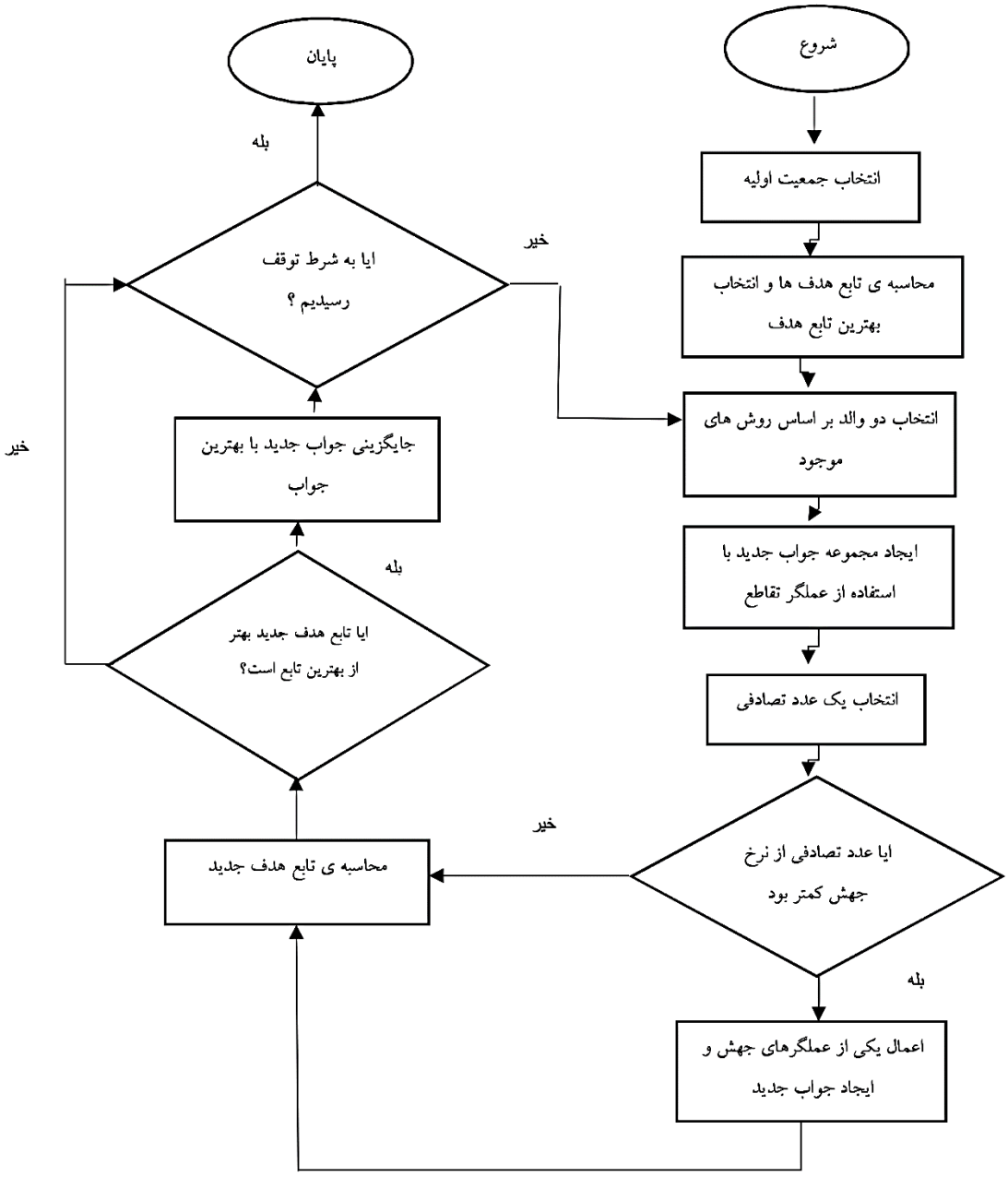
اگر بخواهیم یک دید کلی نسبت به الگوریتم ژنتیک در کد خود داشته باشیم می‌توانیم از فلوچارت و شبه کد این الگوریتم استفاده کنیم. در شکل (1) شبه کد و در شکل (۲) فلوچارت الگوریتم ژنتیک را مشاهده می‌کنید.

```
The procedure: the proposed GA
Initialization mechanism
While the stopping criterion is not met do
    Selection mechanism
    Crossover mechanism
    If rand <Mutation_rate do
    Mutation mechanism
    End if
Endwhile
```

شکل (1): شبه کد الگوریتم ژنتیک

1 .One-point crossover

2 .Shift



شکل(۲): فلوچارت الگوریتم ژنتیک

۲.۳. الگوریتم شبیه سازی تبرید

یک الگوریتم بهینه‌سازی فراابتکاری ساده و اثربخش در حل مسائل بهینه‌سازی است. منشأ الگوریتم تبرید شبیه‌سازی شده، کارهای کریک پاتریک و کرنی و همکارانشان در سال‌های ۱۹۸۳ و ۱۹۸۵ است. تکنیک تبرید تدریجی، به وسیله‌ی متالورژیست‌ها برای رسیدن به حالتی که در آن ماده جامد، به خوبی مرتب و انرژی آن کمینه شده باشد، استفاده می‌شود. این تکنیک شامل قرار دادن ماده در دمای بالا و سپس کم کردن تدریجی این دما است. معمولا الگوریتم های بهبود دهنده، که با یک جواب اولیه شروع شده و در طی مراحلی بهبود داده می‌شوند، ممکن است بعد از چند تکرار در نقطه بهینه محلی قرار بگیرند که گاهی اوقات نیز از ناحیه جواب نهایی خیلی دور است. فرق الگوریتم تبرید تدریجی با الگوریتم‌های بهینه‌سازی محلی در این است، که در الگوریتم بهینه‌سازی محلی، یک جواب در همسایگی جواب قبلی ایجاد می‌شود، اگر تابع هدف به واسطه جواب جدید بهتر شود، جواب جدید قبول شده و در غیر اینصورت جواب جدید رد می‌گردد. این عمل ممکن است منجر به قرار گرفتن در نقطه بهینه محلی شود و دیگر نتواند از آن خارج شود. درحالی که در روش تبرید تدریجی شبیه سازی شده، از توقف در ناحیه بهینه محلی اجتناب کرده و به طور گذرا از آن رد می‌شود. این حالت با پذیرفتن احتمالی جواب های بد انجام می شود، تا از نقطه‌ی بهینه محلی خارج شود. این احتمال برابر است با:

$$e^{-df/T} \qquad (8)$$

که در آن T درجه حرارت و df میزان تغییر در تابع هدف است. اگر این احتمال از یک عدد تصادفی بین [۰ ۱] بیشتر باشد جواب نامناسب هم پذیرفته می‌شود. در این الگوریتم، حلقه خارجی فرآیند انجماد درجه حرارت را کاهش می‌دهد و حلقه داخلی روند رسیدن به تعادل در هر درجه حرارت را بیان می‌کند. به عبارتی تعداد تعویض‌های همسایگی را که باید در یک درجه حرارت انجام شود مشخص می‌نماید. در این الگوریتم نیز همانند الگوریتم ژنتیک یک عملگر حرکت برای بهبود جواب اولیه وجود دارد که با توجه به نوع متغیرها انواع مختلفی دارد. در این مقاله با توجه به باینری بودن دو تا از متغیرها و پیوسته بودن یکی از آن‌ها از عملگر شیفت برای بهبود جواب اولیه استفاده کردیم. در این نوع عملگر در یک مجموعه جواب دو نقطه را به تصادف انتخاب می‌کنیم، سپس عدد مربوط به نقطه اول را به جایگاه بعد از نقطه دوم منتقل می‌کنیم. به عنوان مثال داریم:

۱	۱	۰	۱	۰	۱	۱	۰	۱	۱

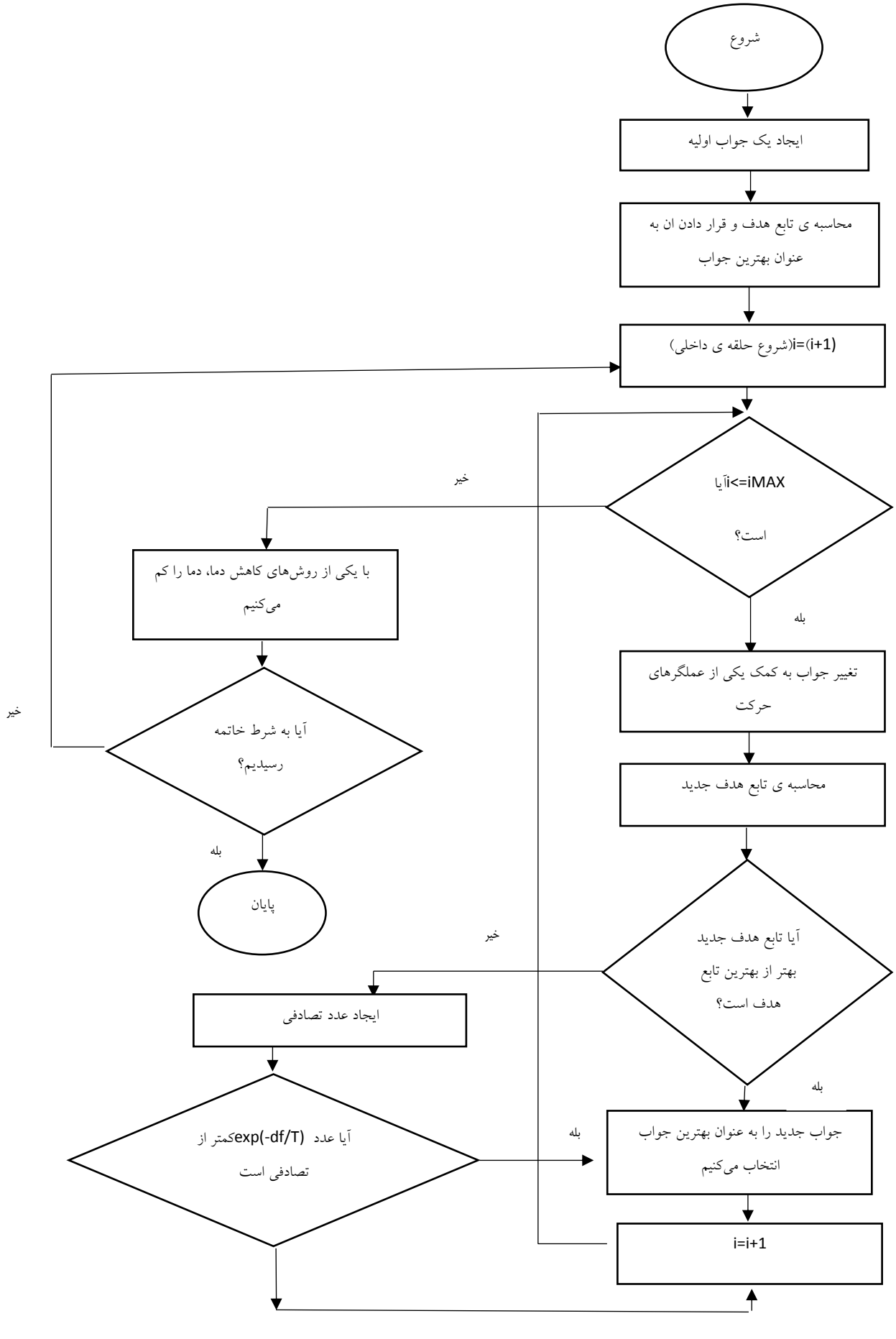
دو نقطه به تصادف روی این مجموعه انتخاب شد حال نقطه اول را به جایگاه بعد از نقطه دوم منتقل می‌کنیم و داریم:

۱	۱	۱	۰	۱	۱	۰	۰	۱	۱

شرایط توقف الگوریتم شبیه سازی تبرید نیز مختلف است که از جمله‌ی آن‌ها، رسیدن به یک زمان مشخص، رسیدن به تعداد تکرار مشخص و رسیدن به یک زمان تعریف شده است.

در این مقاله از یک شرط توقف یکسان (رسیدن به یک زمان مشخص) برای مقایسه دو الگوریتم استفاده می کنیم.

شبه کد و فلوچارت کلی الگوریتم شبیه سازی تبرید همانطور که در شکل شماره (۳) و (۴) می بینید به این صورت است:



شکل(۳): فلوچارت الگوریتم شبیه سازی تبرید

```
Procedure: Simulated annealing
Initialization (initial state and temperature)
Whilethe stopping criterion is not metdo
        For iteration = 1 to Max iteration do
                Generate new state s' from the current from the
current state    Susing the move operator
                Apply local search on new state S' (optional)
                Implement acceptance criterion
        Endfor
        Decrease the temperature
        Apply restart phase(optional)
Endwhile
```

شکل(٤): شبه کد الگوریتم شبیه سازی تبرید

کیفیت جواب الگوریتم شبیه سازی تبرید نسبت به پارامترهای کنترلی حساسیت بالایی دارد و تعیین پارامترهایی که جواب‌های خوب را ایجاد نماید، مشکل می‌باشد.

٤- حل مدل و ارائه‌ی نتایج محاسبات

برای حل مدل دو دسته مثال شامل مسائل سایز کوچک و مسائل سایز بزرگ تعریف می‌کنیم. در سایز کوچک ابتدا با تولید مثال به حل مدل خود به کمک نرم افزار گمز می‌پردازیم. این نرم افزار را برای هر مسئله در مدت زمان ۳۶۰۰ ثانیه اجرا می‌کنیم و نتایج حاصل از حل این مدل را یادداشت کرده سپس مدل خود را با کمک دو الگوریتم فراابتکاری ژنتیک و شبیه سازی تبرید حل می‌کنیم و نتایج حاصل از حل آن‌ها را نیز در جدول شماره‌ی (۱) یادداشت می‌کنیم. سپس به کمک شاخص RPD که از طریق رابطه شماره (۹) بدست می‌آید، میزان انحراف هر جواب را از کمترین مقدار ممکن آن مسئله حساب کرده و به جدول شماره ۲ می‌رسیم.

$$RPD = \frac{Alg_{\text{sol}} - Min_{\text{sol}}}{Min_{\text{sol}}} \times 100 \quad (9)$$

همانطور که مشاهده می‌کنید در مدت زمان ۳۶۰۰ ثانیه به ازاء N=5 فقط مقدار GAP صفر شد و در بقیه سایزهای مسئله نرم‌افزار گمز نتوانست در مدت زمان ۳۶۰۰ ثانیه به جواب بهینه برسد. منتها تا سایز ۲۰ این نرم افزار قادر به حل مدل با داشتن مقداری GAP بود و از سایز ۲۰ به بعد اصلا نمی‌تواند وارد حل الگوریتم بشود و ما مجبوریم برای سایزهای بزرگتر فقط از الگوریتم‌های فراابتکاری استفاده کنیم.

از طرفی همانطور که در جدول بالا ملاحظه می‌کنید، نرم افزار گمز بعد از مدت زمان ۳۶۰۰ ثانیه علاوه بر جواب داده شده یک مقدار GAP به ما میدهد که در واقع میزان انحراف جواب از کمترین جواب ممکن موجود تا آن لحظه است که ما این کمترین جواب را به صورت بهترین جواب در یک ستون برای نرم افزار گمز در جدول بالا قرار دادیم و برای ارزیابی عملکرد الگوریتم‌های فراابتکاری از این مقادیر نرم افزار گمز، برای مقایسه استفاده کردیم به این ترتیب که مطابق شاخص عملکرد RPD میزان انحراف هر جواب از کمترین مقدار نرم افزار گمز را حساب کرده و به جدول شماره (۲) رسیدیم و مشاهده کردیم که در ابعاد کوچک الگوریتم شبیه سازی تبرید در مجموع انحراف کمتری نسبت به الگوریتم فراابتکاری ژنتیک با نرم افزار گمز دارد.

جدول (۱): نتایج حل مدل در سایز کوچک

GAMS			سایز کوچک	مسئله			
GAP	زمان حل	تابع هدف	کمترینجواب گمزتاکنون	GA	SA	M	N
۰	۰	٤٦٠١,٠٢	٤٦٠١,٠٢	٤٦٠١,٠٢	٤٦٠١,٠٢	٥	٥
۰	۰	٤٦٠١,٠٢	٤٦٠١,٠٢	٤٦٠١,٠٢	٤٦٠١,٠٢	١٠	٥
۰	۰	٤٦٠١,٠٢	٤٦٠١,٠٢	٤٦٠١,٠٢	٤٦٠١,٠٢	١٥	٥
۰,۰۱۷۲٤۹	٣٦٠٠	٢٠١١٤,٢	٢٠١١٤,٢	٢٠٤٠٤,١٨	٢٠١٨٤,١٨	٢٠	١٠
۰,۰۱۷۲٤۹	٣٦٠٠	٢٠١١٤,٢	١٩٩٦٤,٢٤	٢٠٧٢٤,٠٦	١٩٩٦٤,٢٤	٣٠	١٠
۰,۰۸۱٦٣	٣٦٠٠	٢١٥٢٤,٣	٢٠١٠٤,١٦	٢٠١٥٤,٢٤	٢٠١٠٤,١٦	٤٠	١٠
۰,١٤٨٤٨١	٣٦٠٠	٦٢١٨٨,٥	٥٦٩٩٨,٥٤	٥٦٩٩٨,٥٤	٥٧٠٠٨,٤٢	٣٠	١٥
۰,۰٧٥١٥٦	٣٦٠٠	٥٨٤١٩,٤	٥٦٩٩٨,٨٨	٥٧٠٣٨,٨٤	٥٧٣٧٨,٦٢	٤٠	١٥
۰,۰٦٣١٢٨	٣٦٠٠	٥٧٦٦٩,٤	٥٧٢٤٨,٦٥	٥٧٢٩٨,٦٢	٥٧٢٤٨,٦٤	٥٠	١٥
۰,١١٦٩٦٨	٣٦٠٠	١٧٣٥٨٢,٧٤	١٧٣٥٨٢,٧٤	١٨٦١٦١,٤	١٨٢٤٨١,٤	٤٠	٢٠
۰,٢	٣٦٠٠	١٦٢٩٧٢,٩٨٠	١٤٣٥٠٢,٩٢	١٤٧٧٩٢,٩	١٤٤٠١٢,٨	٤٥	٢٠
۰,٢٢٨٦٢٥	٣٦٠٠	١٥٠٣٤٤,٠٤	١٢٢٣٦٧,٧٢	١٢٥٦٩٤,٢	١٢٥٧١٤,٢	٥٠	٢٠

جدول (۲): مقایسه RPDهای دو الگوریتم در سایز کوچک

مسئله		سایز کوچک	
M	N	SA	GA
۵	۵	۰	۰
۱۰	۵	۰	۰
۱۵	۵	۰	۰
۲۰	۱۰	۰/۰۰۳٤۷۹	۰/۰۱٤٤۱۷
۳۰	۱۰	۰	۰/۰۳۸۰۵۹
٤۰	۱۰	۰	۰/۰۰۲٤۱۹
۳۰	۱۵	۰/۰۰۰۱۷۳	۰
٤۰	۱۵	۰/۰۰٦٦٦۲	۰/۰۰۰۷۰۱
۵۰	۱۵	۰	۰/۰۰۰۸۷۳
٤۰	۲۰	۰/۰۵۱۲٦۵	۰/۰۷۲٤٦۲
٤۵	۲۰	۰/۰۰۳۵۵۳	۰/۰۲۹۸۹۵
۵۰	۲۰	۰/۰۰۳٤۳۱	۰/۰۰۳۲۷۱
جمع		۰/۰٦۸۵٦٤	۰/۱٦۲۱۷۲
میانگین		۰/۰۰۵۷۱٤	۰/۰۱۳۵۲٤

برای مسائل سایز بزرگتر نیز با توجه به عدم توانایی برنامه گمز در حل مدل در مدت زمان یک ساعت از دو الگوریتم فرا ابتکاری استفاده می‌کنیم و به جدول شماره ۳ می‌رسیم. سپس با کمک شاخص RPD میزان انحراف هر جواب از بهترین جواب را محاسبه کرده و به جدول شماره ۴ می‌رسیم و نتایج را باهم مقایسه می‌کنیم.

جدول ۳. نتایج حل مدل در سایز بزرگ

سایز بزرگ(RPD)		مسئله		
GA	SA	M	N	
۳۵۷۰۷۳۳	۳۶۱۸۲۳,۳	۵۰	۲۵	۱
۳۳۷۰۳۶,۴	۳۳۸۱۸۶,۲	۶۰	۲۵	۲
۱۵۵۸۹۸۸	۱۵۵۸۱۸,۸	۷۰	۲۵	۳
۶۹۹۸۲۶,۶	۷۰۸۹۲۶,۴	۶۰	۳۰	۴
۶۷۸۱۱۹,۴	۶۷۸۰۸۹,۵	۷۰	۳۰	۵
۴۶۱۷۳۲۲	۴۶۶۵۹۱,۹	۸۰	۳۰	۶
۱۱۲۵۱۹۹	۱۱۳۲۳۹,۹	۷۰	۳۵	۷
۱۱۲۰۷۲۲	۱۱۳۰۵۴۲	۸۰	۳۵	۸
۸۷۸۷۳۵	۸۸۰۴۸۴,۹	۹۰	۳۵	۹
۱۵۷۹۸۸۲	۱۵۹۰۱۴۲	۸۰	۴۰	۱۰
۱۴۸۲۵۲۸	۱۴۹۰۲۲۸	۱۰۰	۴۰	۱۱
۱۲۰۹۶۸۱	۱۲۱۸۲۵۱	۱۱۰	۴۰	۱۲
۲۴۰۵۵۰۴	۲۴۱۴۵۴۴	۱۰۰	۴۵	۱۳
۲۱۸۵۳۷۹	۲۱۹۳۳۷۹	۱۱۰	۴۵	۱۴
۱۹۹۶۳۶۴	۲۰۰۷۶۴۴	۱۲۰	۴۵	۱۵
۲۷۲۷۷۲۰	۲۷۲۶۸۹۱	۱۱۰	۵۰	۱۶
۲۷۸۰۲۶۴	۲۷۸۲۹۸۴	۱۲۰	۵۰	۱۷
۲۴۷۲۵۰۷	۲۴۷۳۷۵۶	۱۳۰	۵۰	۱۸
۳۸۵۰۸۱۰	۳۸۶۰۱۲۰	۱۲۰	۵۵	۱۹
۳۶۵۷۱۲۵	۳۶۶۳۱۴۵	۱۳۰	۵۵	۲۰
۳۴۵۵۱۳۰	۳۴۶۶۴۶۰	۱۴۰	۵۵	۲۱
۴۱۵۲۴۸۷	۴۱۵۳۳۹۷	۱۳۰	۶۰	۲۲
۴۲۵۶۴۶۰	۴۲۵۴۰۶۰	۱۴۰	۶۰	۲۳
۳۹۰۰۲۳۳	۳۹۰۶۸۵۳	۱۵۰	۶۰	۲۴

جدول(٤): مقایسه RPDهای دو الگوریتم				
متوسط RPD		مسئله		
GA	SA	M	N	
۰,۰۱۳۳۰۳	۰	۵۰	۲۵	۱
۰,۰۰۳۴۱۲	۰	۶۰	۲۵	۲
۰	۰,۰۰۰۵۱۳۴۱۷	۷۰	۲۵	۳
۰,۰۱۳۰۰۳	۰	۶۰	۳۰	۴
۰	۰	۷۰	۳۰	۵
۰,۰۱۰۵۲۵	۰	۸۰	۳۰	۶
۰,۰۰۶۳۹۹	۰	۷۰	۳۵	۷
۰,۰۰۸۷۶۲	۰	۸۰	۳۵	۸
۰,۰۰۱۹۹۱	۰	۹۰	۳۵	۹
۰,۰۰۶۴۹۴	۰	۸۰	۴۰	۱۰
۰,۰۰۵۱۹۴	۰	۱۰۰	۴۰	۱۱
۰,۰۰۷۰۸۵	۰	۱۱۰	۴۰	۱۲
۰,۰۰۳۷۵۸	۰	۱۰۰	۴۵	۱۳
۰,۰۰۳۶۶۱	۰	۱۱۰	۴۵	۱۴
۰,۰۰۵۶۵	۰	۱۲۰	۴۵	۱۵
o	۰,۰۰۰۳۰۴۰۰۹	۱۱۰	۵۰	۱۶
۰,۰۰۰۹۷۸	۰	۱۲۰	۵۰	۱۷
۰,۰۰۰۵۰۵	۰	۱۳۰	۵۰	۱۸
۰,۰۰۲۴۱۸	۰	۱۲۰	۵۵	۱۹
۰,۰۰۱۶۴۶	۰	۱۳۰	۵۵	۲۰
۰,۰۰۳۲۷۹	۰	۱۴۰	۵۵	۲۱
۰,۰۰۰۲۱۹	۰	۱۳۰	۶۰	۲۲
۰	۰,۰۰۰۵۶۴۱۶۷	۱۴۰	۶۰	۲۳
۰,۰۰۱۶۹۷	۰	۱۵۰	۶۰	۲۴
۰,۰۹۹۹۷۹	۰,۰۰۱۳۸۱۵۹۳			جمع
۰,۰۰۴۱۶۶	۰۵E-۵,۷۵۶۶۴			میانگین

۵- نتیجه‌گیری

صنعت هوایی از گذشته تا به امروز همواره به علت گران بودن هزینه‌های مربوط به این صنعت، از اهمیت زیادی در بخش برنامه‌ریزی برخوردار بوده است. در این صنعت برنامه‌ریزی برای کمینه‌سازی هزینه‌های مربوط به خود هواپیما، خدمه و ... همواره در اولویت و رأس کار برنامه‌ریزان و محققان بوده است. این مقاله به معرفی مدل ترکیبی جدیدی برای مسائل برنامه‌ریزی خطوط هواپیمایی به منظور تعیین تعداد هواپیمای بهینه‌ی مورد استفاده برای یک برنامه‌ی پروازی به منظور حداقل کردن هزینه‌های مربوط به هواپیما و به دست اوردن حداکثر سطح رضایت مشتری پرداخت. برای حل این مدل ترکیبی در این مقاله یکسری فرضیاتی نیز ارائه کردیم که یکسان بودن ظرفیت تمام هواپیماهای در دسترس، یکسان بودن هزینه‌ی سفر در هر مسیر (مستقل بودن از نوع مسیر)، یکسان بودن هزینه‌ی ثابت استفاده از هر هواپیما، یکسان بودن هزینه‌ی کمبود ناشی از عدم برآورده نشدن تقاضا در هر مسیر (مستقل بودن از نوع مسیر) از جمله‌ی این فروض اند.

سپس به تولید دو دسته مثال در سایز بزرگ و کوچک پرداختیم. ابتدا ۱۲ مثال(۴ دسته شهر (فرودگاه) که برای هرکدام ۳ تعداد هواپیمای در دسترس مختلف در نظر گرفتیم) در سایز کوچک تولید کردیم و سپس مدل خود را با نرم افزار دقیق گمز و الگوریتم‌های فراابتکاری شبیه سازی تبرید و ژنتیک حل نمودیم و مشاهده کردیم که نرم افزار دقیقی چون گمز در یک مدت زمان مشخص یک ساعته، قادر به حل دقیق مسئله از سایز ۵ شهر به بعد نیست. در واقع در بقیه سایزها در مدت زمان یک ساعت جوابی که به ما می‌دهد با مقداری GAP هست لذا ما از کمترین مقدار به دست آمده توسط نرم افزار گمز در این مدت زمان مشخص برای مقایسه دو الگوریتم فراابتکاری به کمک شاخص RPD می‌پردازیم و مشاهده می‌کنیم با توجه به متوسط RPDهای به دست آمده برای دو الگوریتم، الگوریتم شبیه سازی تبرید عملکرد بهتری دارد.

از طرفی با تولید ۲۴ مثال (۸ دسته شهر(فرودگاه) که برای هرکدام ۳ تعداد هواپیمای در دسترس مختلف در نظر گرفتیم) در سایز بزرگ به حل مدل توسط نرم افزار گمز، الگوریتم‌های فراابتکاری شبیه سازی تبرید و الگوریتم ژنتیک پرداختیم و مشاهده کردیم در مدت زمان یک ساعت نرم افزار گمز قادر به حل مدل در سایز بزرگ نمی‌باشد. لذا به مقایسه‌ی دو الگوریتم فراابتکاری ژنتیک و شبیه سازی تبرید به کمک شاخص RPD پرداختیم و مشاهده کردیم در سایز بزرگ نیز الگوریتم شبیه سازی تبرید عملکرد بهتری نسبت به الگوریتم ژنتیک دارد چرا که میزان انحراف آن بطور متوسط در سایز بزرگ از الگوریتم زنتیک کمتر است.

برای توسعه آتی مدل نیز راهکارهای مختلفی وجود دارد که از جمله‌ی آن‌ها می‌توان به موارد زیر اشاره کرد :

۱- حل مدل در فضای احتمالی به کمک عدم قطعیت در میزان تقاضا

۲- عدم یکسان بودن هزینه‌های کمبود و فروش در مسیرهای مختلف

۳- ترکیب مدل پیشنهادی با مدل‌های مربوط به خدمه استفاده از سایر الگوریتم‌های فراابتکاری و ...

٦- منابع و مأخذ

[1] Mercier ,A. , Cordeau ,J. , Soumis, F. , (2005) , **A Computational Study of Benders Decomposition for the Integrated Aircraft Routing and Crew Scheduling Problem** , Computer & Operations Research, 32:1451-1476

[2] Gopalakrishnan ,B. , Johnson , W. , (2005) , **Airline Crew Scheduling : State-of -the-Art**,Annals of Operations Research , 140:305-337

[3] Cordeau , J. , Stojkovi´c, G., Soumis, F., and Desrosiers, J. , (2001), **Benders Decomposition for Simultaneous Aircraft Routing and Crew Scheduling** ,Transportation Science, 35:375–388

[4] Dawid, H., König, J. , and Strauss, C. , (2001), **An Enhanced Rostering Model for Airline Crews**, Computers and Operations Research ,28(7): 671–688

[5] Barahona , F. ,Anbil, R. , (2002), **On Some Difficult Linear Programs Coming from Set Partitioning**, Discrete Applied Mathematics,118:3-11

[6] Fisher, M. L., (1981), **The Lagrangian relaxation method for solving integer programming problems**, Management Science, 27: 1–18.

[7] Hjorring, C.A., Karisch, S.E. , and Kohl,N. , (1999), **Carmen Systems' Recent Advances in Crew Scheduling** , In Proceedings of the 39th Annual AGIFORS Symposium, New Orleans, USA, October 3-8:404–420

[8] Rubin, J. ,(1973), **A Technique for the Solution of Massive Set Covering Problems with Application to Airline Crew Scheduling** ,Transplantation science ,7:34-48

[9] Beasley ,J.E , Chu ,P.C ,(1996) , **A genetic algorithm for the set covering problem** , European Journal of Operational Research,94: 392-404.

[10] Wang, K.C. , (2002), **Application of Constraint Programming and Mathematical Programming Techniques to Solve Large-scale Cabin Crew Scheduling Problems**",

[11] Klabjan, D., Johnson,E.L. , Nemhauser,G.L. , Gelman,E. , and Ramaswamy,S. , (2002), **Airline Crew Scheduling with Time Windows and Plane Count Constraints**, Transportation Science 36: 337-348.

[12] Levin, A., (1969), **Some Fleet Routing and Scheduling Problems for Air Transportation Systems Flight Transportation Laboratory Report R**, Massachusetts Institute of Technology, MA, 68-5.

[13] Lee, B.C. (1986), **Routing Problem with Service Choices**, flight transportation laboratory report, Cambridge, MA: MIT, R86-4.

[14] Gerbracht , R., (1978) , **A New Algorithm for Very Large Crew Pairing Problems**" , in : 18th AGIFORS Symposium Proceedings , Vancouver , Canada , PP: 315-341.

[15] Lavoie,S. ,Minoux,M. , Odier, E. , (1988), **A New Approach for Crew Pairing Problems by Column Generation with an Application to Air Transportation**,European Journal of Operational Research,35:45-58

[16] Sarmadi, S. ,(2004), **Minimizing Airline Passenger Delay through Integrated Flight Scheduling and Aircraft Routing**, Master's thesis, Operations Research Center, 83-86.

[17] Lan,S. , Clarke,J. , Barnhart,C. , (2006),**Planning for Robust Airline Operations Optimizing Aircraft Routings and Flight Departure Times to Minimize Passenger Disruptions**, Transportation Science, 40(1):15–28.

[18] Teodorovic, D., Guberinic, S, (1984), **Optimal Dispatching Strategy on an Airline Network after a Schedule Perturbation**, European Journal Operation Research, 15: 178-182

[19] Yan, S., Young, H.F.,(1996), A Decision Support Framework for Multi-fleet Routing and Multi-stop Flight Scheduling, Transportation Research A 30: 379–398

[20] Yan, S., Tseng, C.H., (2002), A Passenger Demand Based Model for Airline Flight Scheduling and Fleet Routing, Computers and Operations Research 29:1559–1581