



دانشگاه صنعتی امیرکبیر  
(پلی تکنیک تهران)

دوره ۴۸، شماره ۴، زمستان ۱۳۹۵، صفحه ۳۷۵ تا ۳۸۵  
Vol. 48, No. 4, Winter 2016, pp. 375-385



نشریه علمی پژوهشی امیرکبیر - مهندسی عمران و محیط زیست

Amirkabir Journal of Science and Research  
Civil and Environmental Engineering  
(AJSR-CEE)

## ارائه الگوریتم ترکیبی حل مسئله مسیریابی وسیله نقلیه همراه با دریافت و تحویل همزمان کالا

امیر مسعود رحیمی<sup>۱\*</sup>، وحید رجبی توارات<sup>۲</sup>

۱- استادیار، دانشکده مهندسی (گروه عمران)، دانشگاه زنجان

۲- دانشجوی کارشناسی ارشد راه و ترابری، دانشکده مهندسی، دانشگاه بین‌المللی امام خمینی

(دریافت: ۱۳۹۱/۱۰/۲۷، پذیرش: ۱۳۹۴/۱۰/۱)

### چکیده

مسئله مسیریابی وسیله نقلیه از مباحث مهمی است که در چند دهه اخیر کاربرد زیادی در بهره‌وری و کارایی سیستم‌های حمل‌ونقل داشته است. یکی از توسعه‌های معروف و پرکاربرد این موضوع، مسئله مسیریابی وسیله نقلیه با دریافت و تحویل همزمان کالا است که در آن، عمل تحویل و جمع‌آوری کالا برای هر مشتری به صورت همزمان انجام می‌گیرد. الگوریتم پیشنهادی در این مقاله، ترکیبی از سه الگوریتم ابتکاری نزدیک‌ترین همسایگی، ارزان‌ترین الحاقی و ژنتیک است که دو الگوریتم اول به همراه یک روش تصادفی، جواب ابتدایی را برای الگوریتم سوم فراهم می‌کنند. در روش‌های نزدیک‌ترین همسایگی و ارزان‌ترین الحاقی، یک تابع احتمالی برای ایجاد جواب‌های بهتر ابداع شده است. همچنین عملگرهایی برای الگوریتم ژنتیک به منظور افزایش فضای جستجو و فرار از بهینه‌های محلی پیشنهاد شده و پس از آن، الگوریتم پیشنهادی بر روی چهل مثال استاندارد و متنوع اجرا شده و با مقایسه نتایج بدست آمده از آن و بهترین جواب‌های موجود از سایر الگوریتم‌ها، بهبود مناسبی نیز مشاهده گردیده است.

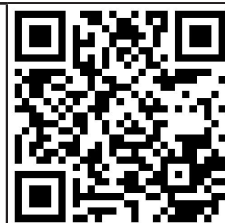
کلمات کلیدی:

الگوریتم فراابتکاری ترکیبی، مسئله مسیریابی وسیله نقلیه، دریافت و تحویل همزمان کالا، الگوریتم ژنتیک

برای ارجاع به این مقاله از عبارت زیر استفاده کنید:

Please cite this article using:  
Rahimi, A. M. and Rajabi-Tavarat, V., 2016. "A Hybrid Meta-heuristic Algorithm for the Vehicle Routing Problem with Simultaneous Delivery and Pick-up". *Amirkabir Journal of Civil and Environmental Engineering*, 48(4), pp. 375-385.  
DOI: 10.22060/ceej.2016.576  
URL: [http://ceej.aut.ac.ir/article\\_576.html](http://ceej.aut.ac.ir/article_576.html)

نویسنده مسئول و عهده‌دار مکاتبات: E-mail: amrahimi@znu.ac.ir



## ۱- مقدمه

مسئله مسیریابی وسیله نقلیه با دریافت و تحویل همزمان کالا<sup>۱</sup> دارای ناوگانی همسان از خودروها و مجموعه‌ای از مشتری‌ها بوده که هر کدام با دو تقاضای مشخص؛ یکی برای جمع‌آوری و دیگری برای تحویل کالا و یک انبار است و هدف در آن، یافتن مسیریابی با کمترین مسافت پیموده‌شده توسط وسایل نقلیه است؛ به صورتی که شرایط زیر برقرار باشد:

- ۱) هر وسیله نقلیه دقیقاً یک مسیر را طی می‌نماید.
- ۲) هر مشتری تنها یک بار توسط یکی از وسایل نقلیه به صورت کامل سرویس‌دهی می‌شود (دریافت و تحویل).
- ۳) کل تقاضای دریافت‌شده و نیز جمع‌آوری شده در هر مسیر، نباید از ظرفیت وسیله نقلیه تجاوز کند.

## ۲- تاریخچه مسئله

مسئله مسیریابی وسیله نقلیه<sup>۲</sup> اولین بار توسط دانزیگ و رامسر<sup>۳</sup> در سال ۱۹۵۹ میلادی معرفی شد. این مسئله بر روی تحویل کالا به مشتریان با تقاضای مشخص تمرکز می‌کند؛ به صورتی که شروع و پایان مسیر در انبار است [۱]. مین<sup>۴</sup> در سال ۱۹۸۹ میلادی محدودیت جدیدی را برای مسئله مسیریابی وسیله نقلیه تعریف نمود. او مسئله حمل‌ونقل کتاب بین کتابخانه‌ها را به شکل فرموله‌شده درآورد که به مسئله مسیریابی با تحویل و دریافت همزمان معروف شد [۲]. همچنین او از رویکرد دو مرحله‌ای برای حل این مسئله استفاده کرد. در اولین مرحله، مشتری‌ها با این شرط که مجموع تقاضاهای تحویل یا مجموع دریافت‌ها در داخل آن مسیر کمتر از ظرفیت وسیله نقلیه باشد، در گروه‌هایی خوشه‌بندی شدند. در دومین مرحله، مسئله فروشنده دوره‌گرد<sup>۵</sup> به طور کامل حل شد؛ به صورتی که اگر مسیر فروشنده دوره‌گرد محدودیت ظرفیت وسیله نقلیه را نیز نقض می‌کرد، مسئله دوباره حل می‌شد.

بعد از گذشت مدت زمانی طولانی از نتایج مین، سالچی و ناگی<sup>۶</sup> در سال ۱۹۹۹ میلادی و با استفاده از یک روش ابتکاری، مسئله مسیریابی وسیله نقلیه دریافت و تحویل همزمان کالا را حل کردند [۳]. آن‌ها در ابتدا مسئله مسیریابی وسیله نقلیه با محدودیت ظرفیت<sup>۷</sup> (CVRP) را با در نظر گرفتن مشتری‌هایی که فقط کالا را دریافت می‌کنند، حل کردند و سپس مشتری‌هایی که کالا تحویل می‌دهند را نیز در مسیر اضافه نمودند. دتلوف<sup>۸</sup> در سال ۲۰۰۱ میلادی کاربردهای مسئله مسیریابی وسیله نقلیه دریافت و تحویل همزمان کالا را برای اولین بار در مسائل لجستیک

<sup>1</sup> Vehicle Routing Problem with Simultaneous Pickup and Delivery

<sup>2</sup> Vehicle Routing Problem (VRP)

<sup>3</sup> Dantzig and Ramser

<sup>4</sup> Min

<sup>5</sup> Travelling Salesman Problem

<sup>6</sup> Salhi and Nagy

<sup>7</sup> Capacitated Vehicle Routing Problem

<sup>8</sup> Dethloff

معکوس<sup>۹</sup> برجسته کرد و از روش ابتکاری ارزان‌ترین الحاقی<sup>۱۰</sup> برای حل مسئله استفاده نمود [۴]. مونتان و گالوو<sup>۱۱</sup> در سال ۲۰۰۲ میلادی، سه نوع متفاوت از مسئله مسیریابی وسیله نقلیه دریافت و تحویل همزمان کالا را تعریف کردند. آن‌ها مسئله‌های مزبور را با استفاده از الگوریتم‌های ابتکاری چرخه‌ای، حداقل شعاع پوشش درختی<sup>۱۲</sup> و الگوریتم ابتکاری ارزان‌ترین الحاقی حل کردند. عملگرهای تعویض گره‌ها به منظور غلبه بر مسیرهای امکان‌ناپذیر و بهبود کیفیت حل مسئله مورد استفاده قرار گرفتند [۵]. آنجلیا و مانسینی<sup>۱۳</sup> در سال ۲۰۰۲ میلادی، محدودیت زمانی را برای مسئله مسیریابی وسیله نقلیه دریافت و تحویل همزمان کالا معرفی نمودند. تحقیقات آن‌ها، اولین و تنها تحقیق انجام‌شده به منظور استفاده از یک الگوریتم دقیق برای حل نسخه اصلی مسئله است. آن‌ها الگوریتم شاخه و ارزش<sup>۱۴</sup> را به صورت مجموعه‌ی فرموله‌شده توسعه دادند [۶].

سالچی و ناگی در سال ۲۰۰۵ میلادی، پژوهش‌های قبلی خود را گسترش و توسعه دادند. آن‌ها روش ابتکاری را با افزودن عملگرهای گره‌ی بیشتر ابداع کردند که منجر به اصلاح حل شد [۷]. چن و هو<sup>۱۵</sup> در سال ۲۰۰۶ میلادی، یک روش الحاقی و الگوریتم فراابتکاری ترکیبی را بر اساس روش ترکیبی شامل الگوریتم لیست ممنوع و الگوریتم‌های بهبود دهنده پیشنهاد دادند [۸]. یکی دیگر از روش‌های حل مسئله مسیریابی وسیله نقلیه دریافت و تحویل همزمان کالا، الگوریتمی است که توسط تانگ و گالوو<sup>۱۶</sup> در سال ۲۰۰۶ میلادی ارائه شد. آن‌ها یک الگوریتم فراابتکاری جستجوی ممنوع<sup>۱۷</sup> که دارای مراحل جریمه مضاعف است را پیشنهاد دادند [۹]. بیانچسی و ریچینی<sup>۱۸</sup> در سال ۲۰۰۷ میلادی، روشی را از ترکیب الگوریتم جستجوی محلی و الگوریتم ممنوع بر روی مثال‌های دتلوف اعمال نمودند که منجر به نتایج بهتری شد [۱۰]. زاچاریادیس و همکاران<sup>۱۹</sup> در سال ۲۰۰۹ میلادی، یک چارچوب ترکیبی را بر اساس دو الگوریتم ابتکاری شناخته‌شده جستجوی ممنوعه و جستجوی محلی پیشنهاد دادند. این روش با هدف دستیابی به یک تعادل رضایت‌بخش بین شدت و تنوع جستجو ارائه شده است. با این منطق، این روش با شدت‌بخشیدن در مناطق به صورت امیدوارکننده‌تری به جستجوی فضای وسیعی از حل‌ها می‌پردازد [۱۱]. در همین سال، کچپل و همکارش<sup>۲۰</sup> در سال ۲۰۰۹ میلادی، از یک روش الگوریتم بهبودیافته مورچگان<sup>۲۱</sup> برای

<sup>9</sup> Reserve Logistic

<sup>10</sup> Cheapest Insertions

<sup>11</sup> Montane and Galvao

<sup>12</sup> Minimum Spanning Tree

<sup>13</sup> Angelelli and Mansini

<sup>14</sup> Branch and Price

<sup>15</sup> Chen and Wu

<sup>16</sup> Tang and Galvao

<sup>17</sup> Tabu-search Meta-heuristic Algorithm

<sup>18</sup> Bianchessi and Righini

<sup>19</sup> Emmanouil E. Zachariadis et al.

<sup>20</sup> Gajpal and Abad

<sup>21</sup> Improved Ant Algorithm

$$L_j \geq L_i - D_j + P_j - M \left( 1 - \sum_{v \in V} x_{ijv} \right); \quad (6)$$

$$i, j \in J, i \neq j$$

$$\sum_{i \in N} \sum_{j \in J} D_j x_{ijv} \leq Q; \quad v \in V \quad (7)$$

$$L_j \leq Q; \quad j \in J \quad (8)$$

$$z_j \geq z_i + 1 - n \left( 1 - \sum_{v \in V} x_{ijv} \right); \quad i, j \in J, i \neq j \quad (9)$$

$$z_j \geq 0; \quad j \in J \quad (10)$$

$$x_{ijv} \in \{0, 1\}; \quad i, j \in N, v \in V \quad (11)$$

$$q_{i,j,k} = q_{i-1,j,k} - d_i + p_j \quad (12)$$

متغیرهای مسئله عبارتند از:

$n$ : تعداد کل مشتریان

$C$ : مجموعه‌ای از مشتریان  $C = \{1, 2, \dots, n\}$

$V$ : تعداد خودرو

$Q$ : ظرفیت خودرو

$A$ : مجموعه‌ای از مشتریان و پایانه

$$A = \{0\} \cup C$$

$c_{ij}$ : مسافت بین مشتری  $i$  با مشتری  $j$  ام

$d_i$ : تقاضای دریافتی مشتری  $i$  ام

$p_i$ : تقاضای تحویلی مشتری  $i$  ام

$q_{ijk}$ : مقدار بار وسیله نقلیه بعد از ملاقات مشتری  $i$  ام و قبل از مشتری  $j$  ام

مشتری  $j$  ام

متغیرهای تصمیم‌گیری در مسئله عبارتند از:

$L_i$ : مقدار بار وسیله نقلیه پس از سرویس‌دهی مشتری  $i$  ام

$z_j$ : متغیر حذف زیر مسیر

#### ۴- الگوریتم پیشنهادی

الگوریتمی که برای حل مسئله مسیریابی وسیله نقلیه با تحویل و دریافت همزمان پیشنهاد شده، بر اساس چارچوب روش ژنتیک<sup>۶</sup> بوده که در آن، برای ساخت نسل اولیه از چند روش فراابتکاری استفاده شده است. در روش‌های ابتکاری در هر بار اجرا، یک جواب برای مسئله بدست می‌آید که این جواب ثابت است. اما در روش‌های فراابتکاری به دلیل تفاوت در ساختار آن، در هر بار تکرار جواب متفاوتی بدست می‌آید. در این الگوریتم‌ها به دلیل متغیرهای زیادی که باید توسط کاربر و به صورت تجربی تنظیم شود و همچنین توابع تصادفی که در آن وجود دارد، جواب‌ها یکسان نیستند و زمان متفاوتی برای هر بار تکرار نیاز دارد. به طور معمول برای رسیدن به بهترین جواب، تعداد زیادی تکرار انجام می‌شود. در ادامه

حل مسئله مسیریابی وسیله نقلیه دریافت و تحویل همزمان کالا استفاده کردند. در روش آن‌ها از یک ساختار مناسب برای افزایش کیفیت جواب‌ها و همچنین از دو جستجوی محلی چندگانه استفاده شده و مزیت این الگوریتم نسبت به الگوریتم‌های دیگر، توانایی حل نسخه‌های دیگر از مسئله مسیریابی است [۱۲].

بولنت ساتای<sup>۱</sup> در سال ۲۰۱۰ میلادی، یک الگوریتم کلونی مورچه<sup>۲</sup> را توسعه داد که مجهز به یک تابع با دید صرفه‌جویی و روش بروزرسانی فرمون<sup>۳</sup> بود. نتایج ارائه‌شده نشان از قابلیت الگوریتم پیشنهادی در یافتن کوتاه‌ترین فاصله دارد [۱۳]. زاچارادیس در سال ۲۰۱۱ میلادی، یک الگوریتم فراابتکاری برای مسئله مسیریابی وسیله نقلیه دریافت و تحویل همزمان کالا را ارائه داد که در آن، با استفاده از یک روش جستجوی محلی کارآمد به اکتشاف همسایگی جواب می‌پردازد. او همچنین نتایج بدست آمده قبلی خود را در سال ۲۰۰۹ میلادی بهبود بخشید [۱۵].

در این مقاله، یک الگوریتم فراابتکاری ترکیبی برای حل مسئله مسیریابی وسیله نقلیه با دریافت و تحویل همزمان کالا ارائه شده است. این روش، ترکیبی از چند الگوریتم ابتکاری و فراابتکاری بوده که تاکنون در تحقیقات قبلی دیده نشده است. در ادامه ضمن تعریف ریاضی مسئله، این نوآوری‌ها توضیح داده می‌شوند.

### ۳- مدل‌سازی ریاضی

مسائل مسیریابی وسیله نقلیه را می‌توان به وسیله یک گراف  $G=(V,E)$  بیان نمود که در آن،  $V$  مجموعه گره‌ها<sup>۴</sup> و مقدار آن بین صفر تا  $n$  بوده و  $E$  مجموعه کمان‌ها<sup>۵</sup> در بین گره‌ها است.

تابع هدف این مسئله، به حداقل رساندن مسافت پیموده‌شده با وسیله نقلیه یا همان طول کل مسیر است که به صورت رابطه (۱) بیان می‌شود:

$$\text{Min} \sum_{i \in N} \sum_{j \in N} \sum_{v \in V} C_{ij} X_{ijv} \quad (1)$$

محدودیت‌های مسئله به صورت زیر مدل می‌شوند:

$$\sum_{j \in N} \sum_{v \in V} X_{ijv} = 1 \quad i \in J \quad (2)$$

$$\sum_{i \in N} \sum_{v \in V} X_{ijv} = 1 \quad j \in J \quad (3)$$

$$\sum_{i \in N} X_{ikv} - \sum_{j \in N} X_{kjh} = 0 \quad k \in J, \quad v \in V \quad (4)$$

$$L_j \geq \sum_{i \in N} \sum_{j \in J} D_j x_{ijv} - D_j + P_j - M(1 - x_{ojv}) \quad (5)$$

$$j \in J, v \in V$$

<sup>1</sup> Bulent Catay

<sup>2</sup> Ant Colony

<sup>3</sup> Pheromone

<sup>4</sup> Nodes

<sup>5</sup> Links

<sup>6</sup> Genetic

Repeat for each chromosome  
 Pick randomly the first seed  
 While Number of appended nodes  $\leq$  Total Number of nodes  
 If the candidate is a seed  
 CandidateNode  $\leftarrow$  first available node  
 If angular and capacity constraints are satisfied  
 NewNode  $\leftarrow$  CandidateNode  
 Else move to the next available node  
 Else (candidate is not a seed)  
 While Number of Nodes in RCL  $\leq \alpha$  and there exists available nodes  
 If the candidate satisfies capacity constraints  
 Add it to RCL  
 Move to the next available node that is closest to the NewNode  
 End of While

شکل (۱): شبه کد روش تصادفی الگوریتم ساخت

$$p_i = \left( \frac{1}{d_i} \right)^{2.5} \quad (13)$$

$$f_i = \frac{p_i}{\sum p_i} \quad (14)$$

که در آن،  $d_i$  فاصله مشتری  $i$  تا انبار و  $f_i$  احتمال انتخاب برای هر مشتری است. شکل (۲) نمودار الگوریتم نزدیکترین همسایگی را ارائه می‌دهد.

#### ۴-۳- روشی ارزان‌ترین الحاقی

در روش الحاقی برای ساخت مسیر، در ابتدا یک مسیر اولیه ایجاد می‌شود و در ادامه، یک یال از مسیر به صورت تصادفی انتخاب شده و نزدیکترین مشتری به آن یال مشخص و به مسیر اضافه می‌شود. این روند تا زمانی که به تمام مشتری‌ها سرویس داده شود، ادامه می‌یابد. در این روش نیز مانند روش‌های قبلی، در ابتدا زیرمسیرها ساخته شده و سپس با تجمیع زیرمسیرها، مسیر کلی ایجاد می‌شود. در این مقاله، ساخت مسیر اولیه برای روش الحاقی به صورت زیر پیشنهاد شده که با ادبیات پیشین متفاوت است:

ابتدا مسیر اولیه به صورت رابطه (۱۵) ایجاد می‌شود:

$$d_0 \rightarrow n_i \rightarrow d_0 \quad (15)$$

که در آن،  $d_0$  انبار و  $n_i$  مشتری است که بر اساس تابع احتمال انتخاب

توضیح روش‌های تصادفی، نزدیکترین همسایگی و ارزان‌ترین الحاقی که برای ساخت جواب استفاده شده آمده است.

#### ۴-۱- الگوریتم تصادفی<sup>۱</sup>

در روش تصادفی، انتخاب نقاط به صورت تصادفی بوده و زیرمسیرها به ترتیب ساخته می‌شوند. برای این کار، در ابتدا انبار به عنوان نقطه اول انتخاب می‌شود. سپس نقطه بعدی به صورت تصادفی انتخاب شده و به مسیر اضافه می‌شود. این کار به همین ترتیب ادامه می‌یابد تا ظرفیت وسیله نقلیه به اتمام برسد. پس از آن، زیرمسیر بعدی آغاز می‌شود. این روند ادامه می‌یابد تا هیچ نقطه‌ای باقی نماند.

در مسائل مسیریابی وسیله نقلیه دریافت و تحویل همزمان کالا، هر مشتری مقداری کالا را دریافت و تحویل می‌دهد. بنابراین هر وسیله نقلیه با توجه به ظرفیتی که دارد، در هر سفر می‌تواند به تعداد محدودی از مشتری‌ها سرویس بدهد. پس از آن که ظرفیت وسیله نقلیه تکمیل شد، به انبار برمی‌گردد و سفر جدیدی را آغاز می‌کند. این سفرها تا زمانی که از تمام نقاط عبور کند، ادامه دارند. به دلیل این که نقاط به صورت تصادفی انتخاب شده‌اند، حل‌های بدست آمده دارای جواب‌های خوبی نیستند. اما همیشه حل‌های قابل قبولی می‌دهد که از مزیت‌های این روش به شمار می‌رود. شبه کد این روش در شکل (۱) ارائه شده است.

#### ۴-۲- الگوریتم نزدیکترین همسایگی<sup>۲</sup>

شاید مشهورترین الگوریتم ابتکاری برای حل مسئله مسیریابی وسیله نقلیه، الگوریتم نزدیکترین همسایگی باشد. در این روش، در ابتدا زیرمسیرها ساخته می‌شوند. برای این کار با شروع از انبار، اولین مشتری که نزدیکترین فاصله را با انبار دارد، انتخاب می‌شود. سپس آن مشتری که نزدیکترین فاصله را با مشتری اول دارد، انتخاب می‌شود؛ با این شرط که مقدار کالای دریافتی و تحویلی از ظرفیت وسیله نقلیه بیشتر نشود. پس از آن، مشتری‌های بعدی نیز به همین ترتیب انتخاب می‌شوند و مسیر تکمیل می‌شود. این مرحله تا زمانی که ظرفیت وسیله نقلیه پر نشود، ادامه دارد. در پایان مسیر، وسیله نقلیه به انبار باز می‌گردد و زیرمسیر بعدی آغاز می‌شود و به همین ترتیب ادامه یافته تا مسیر تکمیل شود و هیچ مشتری باقی نماند. برای یافتن اولین مشتری بعد از انبار، یک تابع احتمالی پیشنهاد شده است. این تابع بر اساس فاصله مشتری‌ها تا انبار، احتمالی را به هر مشتری می‌دهد؛ به این صورت که نزدیکترین مشتری از شانس انتخاب بیشتری بهره می‌برد. احتمال انتخاب برای هر مشتری از رابطه‌های (۱۳) و (۱۴) بدست می‌آید:

<sup>1</sup> Randomize Algorithm

<sup>2</sup> Nearest Neighbor Algorithm

#### ۴-۴- الگوریتم ژنتیک

الگوریتم ژنتیک در واقع تلاشی برای شبیه‌سازی برخی از خصوصیت‌های تکامل و تغییرات بر روی کروموزوم است که همواره در طبیعت صورت می‌گیرد. به عبارت دیگر، این الگوریتم‌ها شدیداً موازی هستند و سعی می‌کنند تا با وراثت و جهش در طی نسل‌های متوالی عملکرد مورد نظری را در افراد یک جمعیت ایجاد کنند. این کار از طریق ایجاد یک جمعیت اولیه و فراهم آوردن شرایط تکامل در نسل‌های بعدی صورت می‌گیرد. در ادامه، روش کلی الگوریتم ژنتیک برای حل مسائل بهینه‌سازی آمده است.

#### ۵- تشکیل جمعیت اولیه

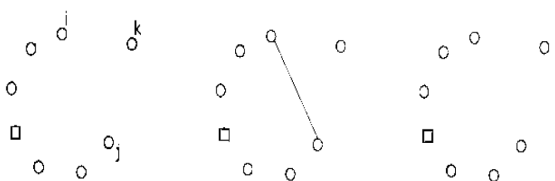
الگوریتم ژنتیک با مجموعه‌ای از کروموزوم‌ها به نام جمعیت اولیه شروع می‌شود. هر کروموزوم نشان‌دهنده یک راه حل برای مسئله است. جمعیت اولیه یا به صورت تصادفی تولید می‌شود که در این صورت برای رسیدن به راه حل بهینه زمان بیشتری را لازم دارد و یا الگوریتم‌های ابتکاری ساخت جواب استفاده می‌شوند که علاوه بر کاهش زمان رسیدن به حل، منجر به ارائه حل‌های بهتری می‌شود.

#### ۶- کدگذاری کردن

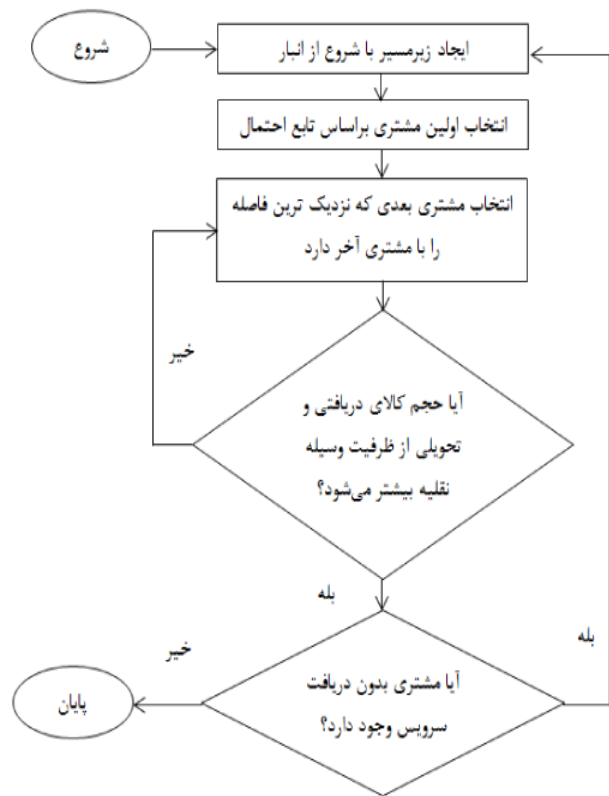
این مرحله از مهم‌ترین مراحل حل مسئله با استفاده از الگوریتم ژنتیک است. الگوریتم ژنتیک به جای این که بر روی متغیرهای مسئله کار کند، با شکل کد شده آن سر و کار دارد. برای حل مسئله مسیریابی چند روش کدگذاری وجود دارد. در این تحقیق، از رشته‌ای از اعداد که هر عدد نشان‌دهنده یک گره است، استفاده شد. برای مثال، رشته Route=۱۲۳۴ این مفهوم را دارد که وسیله نقلیه ابتدا از نقطه ۱ مسیر را آغاز می‌کند و سپس به نقطه ۲ می‌رود، از نقطه ۲ به نقطه ۳ و به همین ترتیب تا به نقطه شماره ۴ می‌رسد.

#### ۷- استراتژی برخورد با محدودیت‌ها

بحث مهم دیگری که در الگوریتم ژنتیک وجود دارد، چگونگی برخورد با محدودیت‌های مسئله است. عملگرهای موجود در ژنتیک سبب ساخت کروموزوم‌های غیرموجه می‌شود و این کروموزوم‌ها نمی‌توانند یک جواب قابل قبول برای مسئله باشند. به همین دلیل، چند روش برای برخورد با آن‌ها وجود دارد:



شکل (۳): روش ارزان‌ترین الحاقی



شکل (۲): نمودار الگوریتم نزدیک‌ترین همسایگی

می‌شود. این تابع بر اساس فاصله مشتری‌ها تا انبار، احتمالی را به هر مشتری می‌دهد؛ به این صورت که دورترین مشتری از انبار از شانس انتخاب بیشتری بهره می‌برد. احتمال انتخاب برای هر مشتری از رابطه‌های (۱۶) و (۱۷) بدست می‌آید:

$$p_i = (d_i)^{2.5} \quad (16)$$

$$f_i = \frac{p_i}{\sum p_i} \quad (17)$$

که در آن،  $d_i$  فاصله مشتری  $i$  تا انبار و  $f_i$  احتمال انتخاب برای هر مشتری است. پس از ایجاد مسیر اولیه، یک یال به صورت تصادفی انتخاب شده و نزدیک‌ترین مشتری به آن یال مشخص می‌شود. سپس این مشتری به نحوی به مسیر اولیه اضافه می‌شود که کم‌ترین طول را ایجاد نماید.

برای مثال، اگر یال انتخاب‌شده به صورت  $T: \{i, j\}$  در نظر گرفته شود، مشتری  $k$  به نحوی انتخاب می‌شود که مقدار  $c(T, k)$  کمینه باشد.

$$c(T, k) = c(i, k) + c(k, j) - c(i, j) \quad (18)$$

به همین ترتیب، مشتری بعدی اضافه می‌شود؛ تا زمانی که ظرفیت وسیله نقلیه پر شود. زیرمسیر بعدی نیز به همین نحو و پس از اتمام زیرمسیر با مشتری‌های سرویس داده نشده ساخته می‌شود تا تمام مشتری‌ها یک بار سرویس داده شود. شکل (۳) روش ارزان‌ترین الحاقی پیشنهادی را بیان می‌نماید.

والد اول	A	B	C	D	E	F	G
والد دوم	C	B	G	E	F	D	A
فرزند اول	?	?	C	D	E	?	?
فرزند دوم	?	?	G	E	F	?	?
فرزند اول	F	A	C	D	E	B	G
فرزند دوم	C	D	G	E	F	A	B

شکل (۴): نحوه ترکیب به روش ترتیبی

مشخصی از آن را با یکدیگر تعویض می‌کند و به این ترتیب، فرزندان جدید ایجاد می‌شود. هدف از این عملگر ایجاد نسل بهتر از یک مجموعه از بهترین‌های نسلی پیشین است. در حقیقت عملگر ترکیب، فضای حل‌ها را با حفظ اطلاعات موجود در کروموزوم‌ها جستجو می‌کند.

### ۷-۳- روش ترتیبی

این روش که اولین بار توسط دیویس<sup>۱</sup> در سال ۱۹۸۵ میلادی معرفی شد، دو عدد تصادفی را به عنوان عدد برش انتخاب کرده و سپس ژن‌های بین این دو عدد در کروموزوم‌های والد ثابت نگه‌داشته شده و مابقی ژن‌ها جابجا می‌شوند [۱۴]. شکل (۴) نحوه ترکیب به روش ترتیبی را نشان می‌دهد.

<sup>۱</sup> Davis

روش اول جلوگیری از تولید کروموزوم‌های غیرموجه است که مستلزم آن بوده که عملگر ژنتیکی به صورتی تعریف شود که پس از ایجاد تغییرات بر روی کروموزوم‌ها، کروموزوم تولید شده یک جواب قابل قبول باشد. ایجاد چنین عملگرهایی برای مسئله مسیریابی بسیار مشکل و برای مسئله مسیریابی وسیله نقلیه دریافت و تحویل همزمان کالا شاید غیرممکن باشد. در نتیجه، از این روش نمی‌توان استفاده نمود. روش دیگری که برای الگوریتم ژنتیک بسیار مرسوم‌تر است، جلوگیری از ورود کروموزوم‌های غیرموجه به هر نسل بوده که این روش بسیار کارا و ساده است.

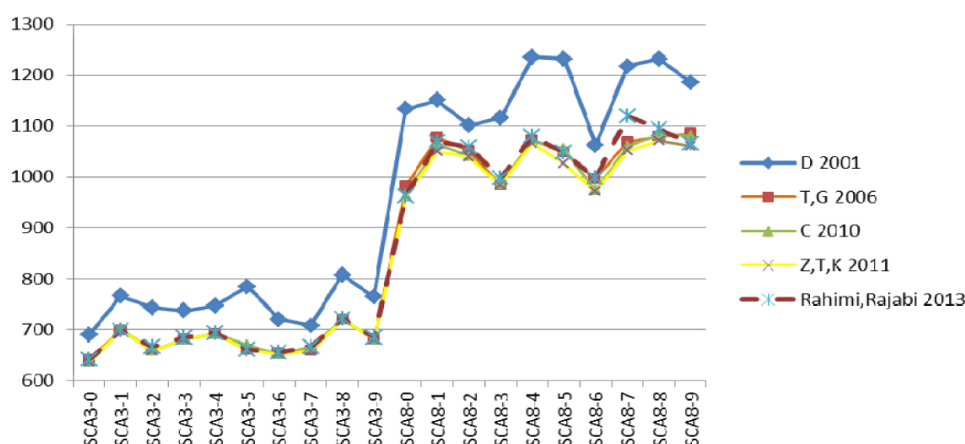
در این تحقیق، از یک تابع برای کنترل کروموزوم‌هایی که توسط عملگرها دچار تغییر شدند، استفاده شد. این تابع با کنترل محدودیت‌های تعریف‌شده برای مسئله مسیریابی وسیله نقلیه با دریافت و تحویل همزمان کالا، موجه و یا غیرموجه بودن آن را تشخیص می‌دهد. در صورتی که کروموزوم غیرموجه تشخیص داده شود، این عملگرها کروموزوم دیگری را تولید می‌کنند. سپس دوباره کروموزوم تولیدی کنترل می‌شود و این روند تا زمانی که کروموزوم موجه ایجاد شود، ادامه می‌یابد.

### ۷-۱- انتخاب

در مرحله انتخاب، یک جفت از کروموزوم‌ها برگزیده می‌شوند تا با یکدیگر ترکیب شوند. عملگر انتخاب، رابط بین دو نسل است و بعضی از اعضای نسل کنونی را به نسل بعدی منتقل می‌کند. بعد از انتخاب، عملگرهای ژنتیک روی دو کروموزوم که به عنوان والد انتخاب شده‌اند، اعمال می‌شوند. معیار در انتخاب اعضا، ارزش تطابق آن‌ها است. اما سازوکار انتخاب حالت تصادفی دارد.

### ۷-۲- ترکیب

این عملگر، دو کروموزوم را به عنوان والدین در نظر گرفته و ژن‌های



شکل (۵): نمودار مقایسه نتایج الگوریتم پیشنهادی با سایر الگوریتم‌ها برای سناریو SCA

۷-۴- جهش

برای اجرای الگوریتم از نرم‌افزار متلب<sup>۱</sup> با رایانه شخصی با پردازنده مناسب<sup>۲</sup> استفاده شده است.

در الگوریتم ژنتیک بعد از آن که یک عضو در جمعیت جدید به وجود آمد، جهش هر ژن آن با احتمال جهش رخ می‌دهد. در جهش، تغییر هر ژن به صورت تصادفی است و در نتیجه این تغییرات تصادفی در جمعیت کروموزوم‌ها فضای بیشتری از فضای حل جستجو می‌شود و از همگرایی بی‌موقع الگوریتم جلوگیری می‌شود.

در روش پیشنهادی این مقاله، از چهار عملگر جهش به منظور ایجاد ویژگی‌های جدید در نسل فعلی استفاده شده است. این عملگرها امکان جستجو در فضای جدید را برای الگوریتم ژنتیک مهیا می‌کنند و همچنین باعث اصلاح کروموزوم‌ها می‌شوند.

۸- اجرای الگوریتم پیشنهادی

در این مطالعه برای ارزیابی الگوریتم پیشنهادی، مثال‌های استاندارد دتلوف در سال ۲۰۰۱ میلادی استفاده شده است [۴]. این مثال‌ها دارای پنجاه مشتری با دو سناریو مختلف بر اساس موقعیت جغرافیایی است. در سناریو SCA مختصات مشتریان به صورت یکنواخت در بازه صفر تا صد توزیع شده است. در سناریو CON، نیمی از مشتریان به همان شیوه SCA و نیمی دیگر به صورت یکنواخت در طول بازه ۱۰۰/۳ تا ۲۰۰/۳ توزیع شده است. مقدار تحویل کالا از مشتریان ( $D_j$ ) به صورت یکنواخت از صفر تا صد توزیع شده است. مقدار دریافت کالا ( $P_j$ ) با استفاده از یک عدد تصادفی ( $R_j$ ) بین صفر تا یک و از رابطه (۱۹) بدست آمده است:

$$P_j = (0.5 + r_j) D_j \quad (19)$$

مثال‌ها با ظرفیت مختلف وسیله نقلیه، با انتخاب حداقل وسیله نقلیه ( $\mu$ ) تولید شده‌اند. ظرفیت مربوط به این مثال‌ها از رابطه (۲۰) بدست آمده است ( $\mu$  برابر با ۳ یا ۸ انتخاب شده است):

$$C = \sum_{S \in J} D_s / \mu \quad (20)$$

۹- ارزیابی و بحث در مورد نتایج

نتایج بدست آمده از الگوریتم پیشنهادی این مقاله (رحیمی و رحیمی در سال ۲۰۱۳) بر روی مثال‌های دتلوف با برترین روش‌های ارائه شده در دنیا تا سال ۲۰۱۲ میلادی در جدول‌های (۱) و (۲) نشان داده شده است. منظور از تعداد گره‌ها در این جدول‌ها همان تعداد مشتری‌ها و همچنین منظور از  $\mu$  همان وسایل نقلیه است. در ستون دوم نتایج بدست آمده توسط دتلوف (D ۲۰۰۱) [۴] نشان داده است که به علت مشخص نبودن تعداد وسیله نقلیه، مقدار آن نیز ارائه نشده است. در ستون چهارم نتایج تانگ و گالوو (T,G ۲۰۰۶) [۹]، ستون ششم نتایج ساتای (C ۲۰۱۰) [۱۳] و ستون هشتم نتایج زاچارادیس (Z,T,K ۲۰۱۱) [۱۵] نشان داده شده است.

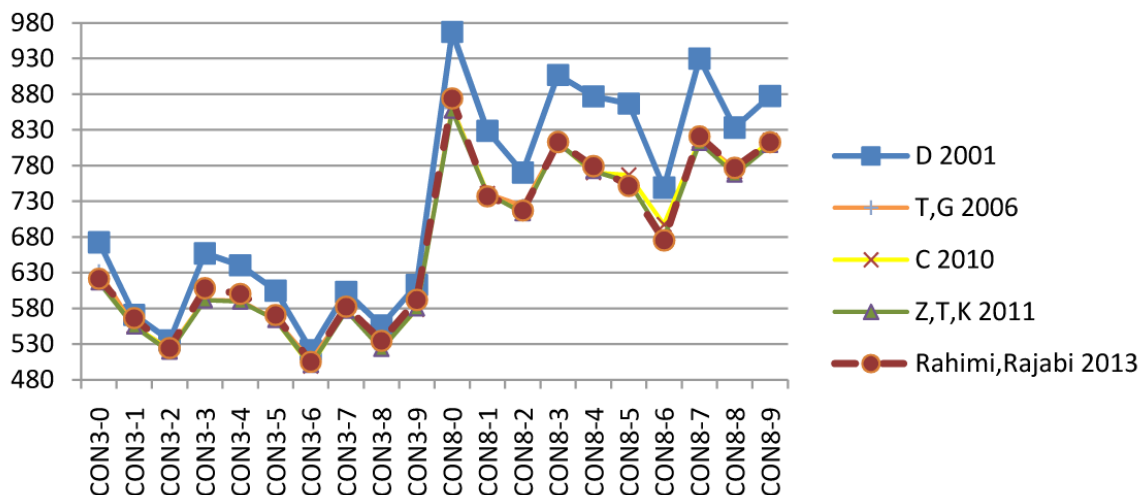
نمودار مقایسه نتایج الگوریتم پیشنهادی با سایر الگوریتم‌ها برای سناریو SCA نیز در شکل (۵) آورده شده است. نمودار نمایش داده شده در شکل‌های (۶) و (۷) نشان از نزدیکی نتایج بدست آمده از الگوریتم پیشنهادی نسبت به سایر الگوریتم‌ها دارد.

درصد اختلاف جواب‌ها<sup>۳</sup> با بهترین جواب‌های موجود، در جدول (۳) نشان داده شده است. با مقایسه نتایج با بهترین جواب‌های موجود در مطالعات قبلی، ملاحظه می‌شود که الگوریتم پیشنهادی در سه مثال، CON ۱-۸ و CON ۵-۸ و CON ۶-۸ به ترتیب برابر با ۰/۵۵، ۰/۴۴ و ۰/۵۵ درصد بوده و جواب بهتری نسبت به بهترین جواب‌های موجود در دنیا حاصل شده است. میانگین ۱/۱۸ و بهبود بدست آمده در جواب‌ها، نشان از کیفیت مناسب جواب‌های بدست آمده در این مقاله دارد.

<sup>1</sup> MATLAB

<sup>2</sup> Intel Core™ i3 (2.27 GHz)

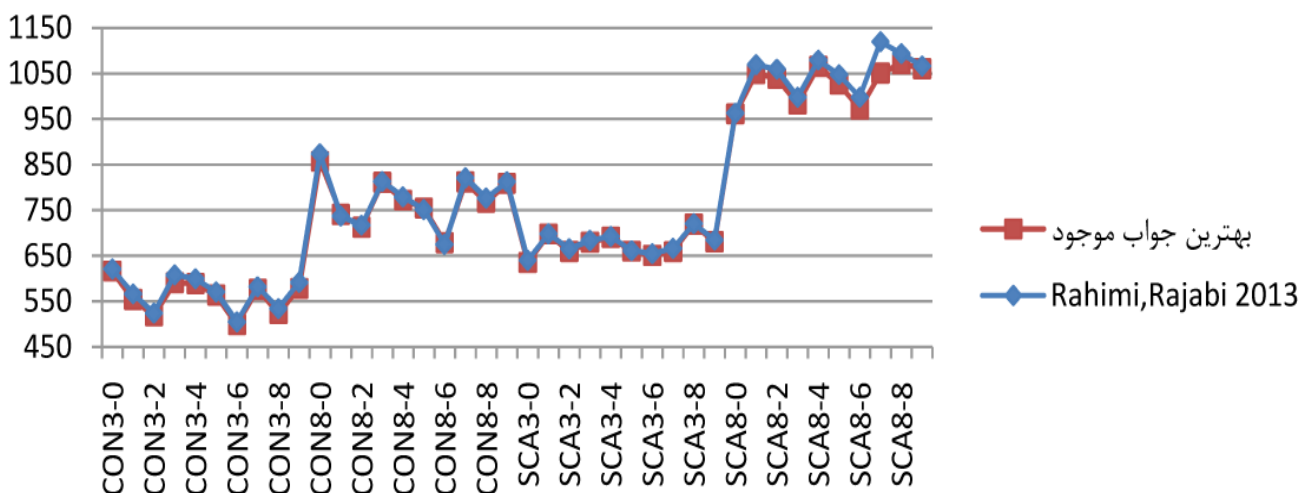
<sup>3</sup> Gap



شکل (۶): نمودار مقایسه نتایج الگوریتم پیشنهادی با بهترین جواب‌های موجود در مطالعات گذشته

جدول (۱): مقایسه الگوریتم پیشنهادی با سایر روش‌های فراابتکاری (حل مسائل دتلوف در سناریو SCA)

مثال	تعداد گره‌ها	$\mu$	$T, G$ 2006	$\mu$	$C$ 2010	$\mu$	$Z, T, K$ 2011	$\mu$	Rahimi, Rajabi 2013	$\mu$
SCA3-0	۵۰	۶۸۹	-	۶۴۰/۵۵	۴	۶۳۶/۱	۴	۶۳۶/۰۶	۴	۶۴۰/۵۵
SCA3-1	۵۰	۷۶۵/۶	-	۶۹۷/۸۴	۴	۷۰۰/۱	۴	۶۹۷/۸۴	۴	۸۴/۶۹۷
SCA3-2	۵۰	۷۴۲/۸	-	۶۵۹/۳۴	۴	۶۵۹/۳	۴	۶۵۹/۳۴	۴	۶۶۵/۷۱
SCA3-3	۵۰	۷۳۷/۲	-	۶۸۰/۰۴	۴	۶۸۰/۰	۴	۶۸۰/۰۴	۴	۶۸۴/۱
SCA3-4	۵۰	۷۴۷/۱	-	۶۹۰/۵۰	۴	۶۹۰/۵	۴	۶۹۰/۵۰	۴	۶۹۲/۵۷
SCA3-5	۵۰	۷۸۴/۴	-	۶۵۹/۹۰	۴	۶۷۰/۱	۴	۶۵۹/۹۰	۴	۶۶۱/۰۷
SCA3-6	۵۰	۷۲۰/۴	-	۶۵۳/۸۱	۴	۶۵۱/۱	۴	۶۵۱/۰۹	۴	۶۵۴/۴۷
SCA3-7	۵۰	۷۰۷/۹	-	۶۵۹/۱۷	۴	۶۶۶/۱	۴	۶۵۹/۱۷	۴	۶۶۶/۶
SCA3-8	۵۰	۸۰۷/۲	-	۷۱۹/۴۷	۴	۷۱۹/۵	۴	۷۱۹/۴۷	۴	۷۲۰/۵۷
SCA3-9	۵۰	۷۶۴/۱	-	۶۸۱/۰۰	۴	۶۸۱/۰	۴	۶۸۱/۰۰	۴	۶۸۴/۶۶
SCA8-0	۵۰	۱۱۳۲/۹	-	۹۸۱/۴۷	۹	۹۶۱/۶	۹	۹۶۱/۵۰	۹	۹۶۳/۰۲
SCA8-1	۵۰	۱۱۵۰/۹	-	۱۰۷۷/۴۴	۹	۱۰۶۳/۰	۹	۱۰۵۰/۲۰	۹	۱۰۶۹/۱
SCA8-2	۵۰	۱۱۰۰/۸	-	۱۰۵۰/۹۸	۱۰	۱۰۴۰/۶	۹	۱۰۳۹/۶۴	۹	۱۰۵۹/۱
SCA8-3	۵۰	۱۱۱۵/۶	-	۹۸۳/۳۴	۹	۹۸۵/۹	۹	۹۸۳/۳۴	۹	۹۹۷/۷۵
SCA8-4	۵۰	۱۲۳۵/۴	-	۱۰۷۳/۴۶	۹	۱۰۷۱/۰	۹	۱۰۶۵/۴۹	۹	۱۰۷۹/۱
SCA8-5	۵۰	۱۲۳۱/۶	-	۱۰۴۷/۲۴	۹	۱۰۵۴/۳	۹	۱۰۲۷/۰۸	۹	۱۰۴۷/۲
SCA8-6	۵۰	۱۰۶۲/۵	-	۹۹۵/۵۹	۹	۹۷۲/۵	۹	۹۷۱/۸۲	۹	۹۹۷/۷
SCA8-7	۵۰	۱۲۱۷/۴	-	۱۰۶۸/۵۶	۱۰	۱۰۵۹/۷	۹	۱۰۵۲/۱۷	۹	۱۱۱۹/۳
SCA8-8	۵۰	۱۲۳۱/۶	-	۱۰۸۰/۵۸	۹	۱۰۸۲/۷	۹	۱۰۷۱/۱۸	۹	۱۰۹۳/۵
SCA8-9	۵۰	۱۱۸۵/۶	-	۱۰۸۴/۸۰	۹	۱۰۸۱/۴	۹	۱۰۶۰/۵۰	۹	۱۰۶۶/۱



شکل (۷): نمودار مقایسه نتایج الگوریتم پیشنهادی با سایر الگوریتم‌ها برای سناریو CON



جدول (۲): مقایسه الگوریتم پیشنهادی با سایر روش‌های فراابتکاری (حل مسائل دتلوف در سناریو CON)

مثال	تعداد گره‌ها	$D$ 2001	$T, G$ 2006	$C$ 2010	$Z, T, K$ 2011	$Rahimi,$ $Rajabi$ 2013	$\mu$
CON3-0	۵۰	۶۷۲/۴	-	۶۳۱/۳۹	۴	۶۱۶/۵	۴
CON3-1	۵۰	۵۷۰/۶	-	۵۵۴/۴۷	۴	۵۵۵/۶	۴
CON3-2	۵۰	۵۳۴/۸	-	۵۲۲/۸۶	۴	۵۲۱/۴	۴
CON3-3	۵۰	۶۵۶/۹	-	۵۹۱/۱۹	۴	۵۹۱/۲	۴
CON3-4	۵۰	۶۴۰/۲	-	۵۹۱/۱۲	۴	۵۸۹/۳	۴
CON3-5	۵۰	۶۰۴/۷	-	۵۶۳/۷۰	۴	۵۶۳/۷	۴
CON3-6	۵۰	۵۲۱/۳	-	۵۰۶/۱۹	۴	۴۹۹/۲	۴
CON3-7	۵۰	۶۰۲/۸	-	۵۷۷/۶۸	۴	۵۷۷/۵	۴
CON3-8	۵۰	۵۵۶/۲	-	۵۲۳/۰۵	۴	۵۲۳/۱	۴
CON3-9	۵۰	۶۱۲/۸	-	۵۸۰/۰۵	۴	۵۷۸/۲	۴
CON8-0	۵۰	۹۶۷/۳	-	۸۶۰/۴۸	۹	۸۵۸/۹	۹
CON8-1	۵۰	۸۲۸/۷	-	۷۴۰/۸۵	۹	۷۴۰/۹	۹
CON8-2	۵۰	۷۷۰/۲	-	۷۲۳/۳۲	۹	۷۱۴/۳	۹
CON8-3	۵۰	۹۰۶/۷	-	۸۱۱/۲۳	۱۰	۸۱۲/۳	۱۰
CON8-4	۵۰	۸۷۶/۸	-	۷۷۲/۲۵	۹	۷۷۰/۱	۹
CON8-5	۵۰	۸۶۶/۹	-	۷۵۶/۹۱	۹	۷۶۶/۶	۹
CON8-6	۵۰	۷۴۹/۱	-	۶۷۸/۹۲	۹	۶۹۷/۲	۹
CON8-7	۵۰	۹۳۹/۸	-	۸۱۴/۵۰	۹	۸۱۴/۸	۹
CON8-8	۵۰	۸۳۳/۱	-	۷۷۵/۵۹	۹	۷۷۱/۳	۹
CON8-9	۵۰	۸۷۷/۳	-	۸۰۹/۰۰	۹	۸۱۵/۱	۹

### ۱۰- نتیجه گیری

در این مقاله، روش فراابتکاری ترکیبی برای حل مسئله مسیریابی وسیله نقلیه با دریافت و تحویل همزمان کالا ارائه شده است. در این روش، سه الگوریتم فراابتکاری تصادفی، نزدیک‌ترین همسایگی و ارزان‌ترین الحاقی برای ساخت جواب بکار رفته است. سپس از جواب‌های بدست آمده برای جمعیت اولیه الگوریتم ژنتیک استفاده شد. در هر کدام از این الگوریتم‌ها، از ابتکاراتی برای رسیدن به جواب بهتر استفاده گردید. پس از آن، این جواب‌ها با استفاده از یک الگوریتم ژنتیک کارا، بهبود داده شدند. جواب‌های خوبی برای مثال‌های استاندارد این مسئله توسط روش پیشنهاد شده بدست آمده و در سه مثال، بهترین جواب‌های مربوط مطالعات گذشته را بهبود بخشیده است و رقابت نزدیکی با سایر الگوریتم‌های ارائه شده دارد.

### ۱۱- فهرست علائم

- $A$ : مجموعه‌ای از مشتریان و پایانه
- $C$ : مجموعه‌ای از مشتریان
- $n$ : تعداد کل مشتریان
- $Q$ : ظرفیت خودرو
- $V$ : تعداد خودرو
- $\mu$ : انتخاب حداقل وسیله نقلیه
- $C_{ij}$ : مسافت بین مشتری  $i$  با مشتری  $j$  زام
- $d_i$ : انبار
- $d_i$ : تقاضای دریافتی مشتری  $i$  زام
- $D_j$ : مقدار تحویل کالا به مشتری  $j$
- $f_i$ : احتمال انتخاب برای هر مشتری
- $L_j$ : مقدار بار وسیله‌ی نقلیه پس از سرویس‌دهی مشتری  $j$  زام

جدول (۳): بهترین جواب‌های موجود برای مسائل دتلوف (تا پایان سال ۲۰۱۲ میلادی)

Gap%	Rahimi, Rajabi 2013	بهترین جواب موجود	مثال	Gap%	Rahimi, Rajabi 2013	بهترین جواب موجود	مثال
-۰/۷۶	۶۲۱/۲۲	۶۱۶/۵۲	CON3-0	-۰/۷۸	۶۴۰/۵۵	۶۳۵/۶۲	SCA3-0
-۲/۱۵	۵۶۶/۴	۵۵۴/۴۷	CON3-1	۰/۰۰	۶۹۷/۸۴	۶۹۷/۸۴	SCA3-1
-۱/۱۹	۵۲۴/۱۴	۵۱۸	CON3-2	-۰/۹۷	۶۶۵/۷۱	۶۵۹/۳۴	SCA3-2
-۲/۸۵	۶۰۸/۰۱	۵۹۱/۱۹	CON3-3	-۰/۶۰	۶۸۴/۱	۶۸۰/۰۴	SCA3-3
-۱/۹۰	۶۰۰	۵۸۸/۷۹	CON3-4	-۰/۳۰	۶۹۲/۵۷	۶۹۰/۵	SCA3-4
-۱/۲۳	۵۷۰/۶۴	۵۶۳/۷	CON3-5	-۰/۱۸	۶۶۱/۰۷	۶۵۹/۹	SCA3-5
-۱/۱۹	۵۰۴/۹۷	۴۹۹/۰۵	CON3-6	-۰/۵۲	۶۵۴/۴۷	۶۵۱/۰۹	SCA3-6
-۰/۹۸	۵۸۲/۱۵	۵۷۶/۴۸	CON3-7	-۱/۱۳	۶۶۶/۶	۶۵۹/۱۷	SCA3-7
-۲/۱۸	۵۳۴/۴۷	۵۲۳/۰۵	CON3-8	-۰/۱۵	۷۲۰/۵۷	۷۱۹/۴۷	SCA3-8
-۲/۳۱	۵۹۱/۶۳	۵۷۸/۲۵	CON3-9	-۰/۵۴	۶۸۴/۶۶	۶۸۱	SCA3-9
-۱/۹۶	۸۷۳/۹۶	۸۵۷/۱۷	CON8-0	-۰/۱۶	۹۶۳/۰۲	۹۶۱/۵	SCA8-0
۰/۵۵	۷۳۶/۷۸	۷۴۰/۸۵	CON8-1	-۱/۸۵	۱۰۶۹/۱	۱۰۴۹/۶۵	SCA8-1
-۰/۵۸	۷۱۷/۰۵	۷۱۲/۸۹	CON8-2	-۱/۸۷	۱۰۵۹/۱	۱۰۳۹/۶۴	SCA8-2
-۰/۲۲	۸۱۲/۸۸	۸۱۱/۰۷	CON8-3	-۱/۴۷	۹۹۷/۷۵	۹۸۳/۳۴	SCA8-3
-۰/۸۷	۷۷۹	۷۷۲/۲۵	CON8-4	-۱/۲۸	۱۰۷۹/۱	۱۰۶۵/۴۹	SCA8-4
۰/۴۴	۷۵۱/۵۵	۷۵۴/۸۸	CON8-5	-۱/۹۶	۱۰۴۷/۲	۱۰۲۷/۰۸	SCA8-5
۰/۵۵	۶۷۵/۱۶	۶۷۸/۹۲	CON8-6	-۲/۶۶	۹۹۷/۷	۹۷۱/۸۲	SCA8-6
-۱/۰۹	۸۲۰/۸	۸۱۱/۹۶	CON8-7	-۶/۴۷	۱۱۱۹/۳	۱۰۵۱/۲۸	SCA8-7
-۱/۱۹	۷۷۶/۶۷	۷۶۷/۵۳	CON8-8	-۲/۰۸	۱۰۹۳/۵	۱۰۷۱/۱۸	SCA8-8
-۰/۴۴	۸۱۲/۵۵	۸۰۹	CON8-9	-۰/۵۳	۱۰۶۶/۱	۱۰۶۰/۵	SCA8-9

میانگین اختلاف ۱/۱۸-

Problem with Simultaneous Delivery and Pick-up Points,” *Transportation Research-Part A*, Vol. 23, No. 5, pp. 377–386, 1989.

- [3] Salhi, S. and Nagy, G.; “A Cluster Insertion Heuristic for Single and Multiple Depot Vehicle Routing Problems with Backhauling,” *Journal of the Operational Research Society*, Vol. 50, No. 10, pp. 1034–1042, 1999

- [4] Dethloff, J.; “Vehicle Routing and Reverse Logistics: The Vehicle Routing Problem with Simultaneous Delivery and Pick-up,” *OR Spektrum*, Vol. 23, No. 1, pp. 79–96, 2001.

- [5] Montane, F. A. T. and Galvao, R. D.; “Vehicle

$P_i$ : تقاضای تحویلی مشتری  $i$ ام

$P_j$ : مقدار دریافت کالا

$R_j$ : عدد تصادفی

$q_{ijk}$ : مقدار بار وسیله نقلیه بعد از ملاقات مشتری  $i$ ام و قبل از

مشتری  $j$ ام

$z_j$ : متغیر حذف زیر مسیر

## ۱۲- مراجع

- [1] Dantzig, G. B. and Ramser, R. H.; “The Truck Dispatching Problem,” *Management Science*, Vol. 6, No. 1, pp. 80–91, 1958.

- [2] Min, H.; “The Multiple the Vehicle Routing

- with Simultaneous Pick-up and Delivery,” *Computers and Operations Research*, Vol. 34, pp. 2578–594, 2007.
- [11] Zachariadis, E.; Tarantilis, D. and Kiranoudis, T.; “A Hybrid Metaheuristic Algorithm for the Vehicle Routing Problem with Simultaneous Delivery and Pick-up Service,” *Expert Systems with Applications*, Vol. 36, No. 2, pp. 1070–1081, 2009.
- [12] Gajpal, Y. and Abad, P.; “An Ant Colony System (ACS) for Vehicle Routing Problem with Simultaneous Delivery and Pick-up,” *Computers and Operations Research*, Vol. 36, No. 12, pp. 3215–3223, 2009.
- [13] Catay, B.; “A New Saving-based Ant Algorithm for the Vehicle Routing Problem with Simultaneous Pick-up and Delivery,” *Expert Systems with Applications*, Vol. 37, No. 10, pp. 6809–6817, 2010.
- [14] Davis, L.; “Applying Algorithms to Epistatic Domains,” *International Joint Conferences on Artificial Intelligence*, Vol. 85, pp. 162–164, 1985.
- [15] Zachariadis, E. E. and Kiranoudis, C. T.; “A Local Search Meta-heuristic Algorithm for the Vehicle Routing Problem with Simultaneous Pick-ups and Deliveries,” *Expert Systems with Applications*, Vol. 38, No. 3, pp. 2717–2726, 2011.
- Routing Problems with Simultaneous Pick-up and Delivery Service,” *OPSEARCH*, Vol. 39, No. 1, pp. 19–33, 2002.
- [6] Angelelli, E. and Mansini, R.; “The Vehicle Routing Problem with Time Windows and Simultaneous Pick-up and Delivery,” *Quantitative Approaches to Distribution Logistics and Supply Chain Management*, Lecture Notes in Economics and Mathematical Systems, Springer-Verlag, pp. 249–267, 2002.
- [7] Nagy, G. and Salhi, S.; “Heuristic Algorithms for Single and Multiple Depot Vehicle Routing Problems with Pickups and Deliveries,” *European Journal of Operational Research*, Vol. 162, No. 1, pp. 126–141, 2005.
- [8] Chen, J. F. and Wu, T. H.; “Vehicle Routing Problem with Simultaneous Deliveries and Pickups,” *Journal of the Operational Research Society*, Vol. 57, No. 5, pp. 579–587, 2006.
- [9] Tang-Montane, F. A. and Galvao, R. D.; “A Tabu Search Algorithm for the Vehicle Routing Problem with Simultaneous Pick-up and Delivery Service,” *Computers and Operations Research*, Vol. 33, No. 3, pp. 595–619, 2006.
- [10] Bianchessi, N. and Righini, G.; “Heuristic Algorithms for the Vehicle Routing Problem