



## Research Paper;

# Segmentation of Streaming Trajectory Data in Pharmaceutical Distribution Logistics in the field of Passive Defense

Eynollah Khanjari<sup>1</sup>, Amir Hossein Shokouhi<sup>2</sup>

1- faculty member, Faculty of Engineering, Computer Software Department

2- Software Department, Faculty of Computer Engineering, Iran University of Science and Technology

### Article Information

### Abstract

**Accepted:**  
2024/12/04

**Received:**  
2024/08/10

### Keywords:

*Pharmaceutical Logistics, Passive Defense, Genetic Algorithm, Trajectory Segmentation, Spatiotemporal Data*

### Corresponding Author:

**Email:**  
khanjari.e.iust@gmail.com

The management and preservation of supply chains for pharmaceuticals that are environmentally sensitive is critical in a world where food and pharmaceutical security are becoming essential components of passive defense. In this kind of environment, governments place a high premium on the function of pharmaceutical logistics in guaranteeing the availability of necessary drugs, especially in times of crisis and sanction. Due to the extreme sensitivity of biological materials and their susceptibility to heat, the drug distribution systems in nations such as Iran necessitate close and continuous monitoring. On the supply chain, delays brought on by traffic, transportation mishaps, and human mistake can be disastrous. Drug trafficking, medication loss, and other issues can result from improper drug storage and transportation, posing a major risk to public health.

This paper addresses one of the main issues in Iran's pharmaceutical logistics: the massive volume of spatiotemporal data from refrigerated drug distribution vehicles. It does this by introducing a method named SEGA to optimize the segmentation of distribution fleet routes, which is inspired by the genetic algorithm and offers a novel idea for saving time and enhancing performance. This data is essential for preserving the safety and caliber of pharmaceuticals being delivered. It includes details like the temperature within the refrigerator, the outside temperature, the speed of the vehicle, and the location. But because this data is sent continually and in real-time, the system has to deal with database overflow, which means there needs to be a way to manage and summarize this data.

Experiments and simulations show that the suggested approach effectively improves the genetic algorithm's performance by precisely and efficiently segmenting and summarizing the data. The results demonstrate that the system's speed and efficiency may be effectively increased by up to 92% with the suggested strategy.




## فصلنامه علمی دفاع هوافضایی

دوره ۳، شماره ۳

پاییز

صص ۸۹-۶۵

مجموعه مقالات

مقاله پژوهشی؛ 

# بخش بندی بهینه داده های جریانی خط سیر لجستیک توزیع دارو در حوزه پدافند غیر عامل

عین اله خنجری<sup>۱</sup>، امیرحسین شکوهی<sup>۲</sup>

۱- عضو هیات علمی دانشکده مهندسی کامپیوتر گروه نرم افزار

۲- گروه نرم افزار دانشکده مهندسی کامپیوتر دانشگاه علم و صنعت ایران

### چکیده

### اطلاعات مقاله

در جهانی که امنیت غذایی و دارویی به یکی از ارکان اساسی پدافند غیرعامل تبدیل شده است، حفظ و مدیریت زنجیره تأمین داروهای حساس به شرایط محیطی از اهمیت ویژه ای برخوردار است. در چنین شرایطی، نقش لجستیک دارویی در تأمین داروهای حیاتی برای جامعه، به خصوص در شرایط بحرانی و تحریم ها، از اولویتهای اصلی دولت ها به شمار می آید. سامانه های توزیع دارو در کشورهایی مانند ایران، به دلیل حساسیت بالای مواد بیولوژیکی و گرماپذیری آن ها، نیازمند نظارت دقیق و لحظه ای هستند. تأخیرهای ناشی از ترافیک، سوانح حمل و خطاهای انسانی می تواند اثرات مخربی بر زنجیره تأمین داشته باشد. عدم نگهداری و حمل صحیح داروها منجر به قاچاق دارو، از دست رفتن آن ها و در نتیجه تهدید جدی سلامت جامعه می شود.

این مقاله با الهام از الگوریتم ژنتیک و پیشنهاد ایده ای جهت کاهش زمان و عملکرد صحیح، روشی به نام SEGA جهت بهینه سازی بخش بندی خط سیر ناوگان توزیع دارو معرفی می کند و به بررسی و بهینه سازی یکی از چالش های عمده در لجستیک دارویی ایران می پردازد: حجم عظیم داده های مکانی-زمانی خودروهای یخچال دار توزیع دارو. این داده ها که شامل اطلاعاتی همچون دمای داخلی یخچال، دمای محیط خارجی، سرعت خودرو، محدوده و موقعیت جغرافیایی می شوند، برای حفظ امنیت و کیفیت داروهای حمل شده ضروری هستند؛ اما از آنجا که این داده ها به صورت لحظه ای و مستمر ارسال می شوند، سیستم با مشکل سرریز شدن پایگاه های داده مواجه شده و نیاز به راه حلی برای مدیریت و خلاصه سازی این داده ها احساس می شود.

نتایج شبیه سازی ها و آزمایش ها نشان می دهد که ارائه ایده مناسب جهت بهبود عملکرد الگوریتم ژنتیک، بخش بندی و خلاصه سازی داده ها را به درستی و با کیفیت بالا انجام دهد؛ به طوری که نتایج حاصل نشان می دهد که روش پیشنهادی می تواند به طور مؤثری سرعت و کارایی سیستم را در حد ۹۲ درصد بهبود بخشد.

تاریخ دریافت:

۱۴۰۳/۰۵/۲۰

تاریخ پذیرش:

۱۴۰۳/۰۹/۱۴

کلیدواژه ها:

لجستیک دارویی، پدافند غیرعامل، الگوریتم ژنتیک، بخش بندی خط سیر، داده های مکانی-زمانی

نویسنده مسئول:

ایمیل:

khanjari.e.iust@gmail.com

**استناد:** عین اله خنجری، امیرحسین شکوهی. بخش بندی بهینه داده های جریانی خط سیر لجستیک توزیع دارو در حوزه پدافند غیرعامل.

دفاع هوافضایی، دوره ۳ (شماره ۳)، صفحات ۸۹-۶۵

## ۱- مقدمه

یکی از مهم‌ترین جنبه‌های لجستیک دارویی، مدیریت دقیق و پیوسته محموله‌های دارویی حساس به شرایط محیطی است. مواد بیولوژیکی که بسیاری از آن‌ها به دما و رطوبت حساس هستند، نیازمند کنترل دقیق شرایط نگهداری و حمل‌ونقل هستند. هرگونه تأخیر در حمل، سوانح جاده‌ای یا خطاهای انسانی می‌تواند منجر به فساد داروها و از دست رفتن ارزش آن‌ها شود؛ موضوعی که نه تنها هزینه‌های مالی سنگینی را به دنبال دارد، بلکه می‌تواند زندگی افراد نیازمند به این داروها را به خطر بیندازد.

با توجه به هزینه‌بر بودن نگهداری و حمل صحیح داروها و مواد بیولوژیکی و خطر فساد داروها و همچنین تأمین امنیت کالاهای دارویی، اهمیت یک سیستم جامع و پیشرفته برای مدیریت این فرآیندها را بیش از پیش آشکار می‌کند. در این راستا، سیستم‌های ناوبری و نظارت لحظه‌ای بر محموله‌های دارویی، یکی از ابزارهای حیاتی برای حفظ امنیت و کیفیت داروها به شمار می‌روند. این سیستم‌ها با استفاده از تکنولوژی‌های پیشرفته همچون GPS و سنسورهای متعدد، امکان کنترل دقیق شرایط حمل‌ونقل، از جمله دما، رطوبت و موقعیت مکانی را فراهم می‌کنند. داده‌های مربوط به سنسورها و GPS در یک پایگاه داده مکانی برای رصد و مانیتورینگ رانندگان ذخیره می‌گردند. شکل این داده‌ها و فناوری مولد آن‌ها شیوه‌های خاصی را جهت پردازش آن‌ها طلب می‌کند. پردازش این نوع داده‌ها با توجه به ماهیت و ویژگی‌های خاص آن‌ها مستلزم به کارگیری ایده‌ها و بسترهای مناسب می‌باشد، که روش‌های مرسوم ذخیره‌سازی، کشف دانش و استخراج الگو بر روی این داده‌ها قابل پیاده‌سازی نیست. به همین دلیل در دو دهه اخیر متخصصان علوم رایانه‌ای و مهندسی داده به دنبال طراحی و تولید روش‌ها و مدل‌های جدیدی به منظور مواجهه با این جریان داده گسترده بوده و موفق به ارائه مدل‌ها و الگوریتم‌هایی در این زمینه شده‌اند؛ اما به دلیل گسترش روزافزون سیستم‌های مکان-آگاه و همچنین ماهیت رشد نمایی این نوع داده‌ها، پاسخگوی نیازها و مسائل جدیدی که دچار دگرگونی می‌شوند، نیستند. برخلاف داده‌های سنتی، در پردازش داده‌های جریان‌ی، از نظر فضا و زمان محدودیت‌هایی وجود دارد [۱]. در دسترس نبودن کلیه داده‌ها، امکان مشاهده تنها یک بار داده‌ها، حجم بسیار زیاد داده‌ها و عدم امکان نگهداری آن‌ها در حافظه، نرخ تولید بالای داده‌ها و محدودیت زمان پردازش آن‌ها، لزوم پردازش مستمر داده‌ها، از جمله محدودیت‌هایی است که در پردازش داده‌های جریان‌ی وجود دارند [۲].

داده‌های جریان‌ی حاصل از ناوگان توزیع دارو یا همان "خط سیر"، شامل حجم زیادی از

داده‌های مرتبط با زمان و مکان اشیا متحرک است. اشیا متحرک در مسیرهای هندسی با الگوهای مختلف منظم و نامنظم خطی حرکت می‌کنند و مختصات و داده‌های توصیفی آن‌ها در طول زمان به طور پیوسته تغییر می‌کند. این ویژگی منحصر به فرد اشیا متحرک باعث می‌شود که سیستم‌های مدیریت پایگاه داده سنتی نتوانند به صورت مناسب خود را با داده‌های اشیا متحرک مطابقت دهند [۳]. در مجموع، آماده‌سازی این حجم زیاد از داده‌ها یک فرآیند پیچیده و مهم است. لذا پیش‌پردازش و آماده‌سازی داده‌ها برای تحلیل‌های بعدی نقش مهمی در ارتقا کیفیت داده‌ها داشته و باعث تسهیل فرآیند تحلیل داده‌ها، داده‌کاوی و کشف دانش می‌شود. همچنین، پیش‌پردازش داده‌ها از نظر کاهش حجم و سرعت ذخیره‌سازی داده‌های خط سیر اشیا متحرک، موضوعیت پیدا کرده است [۴]. دهه اخیر شاهد تلاش پژوهشگران عرصه مهندسی داده در زمینه ارائه مدل‌ها و الگوریتم‌های راهبردی برای آماده‌سازی داده‌های خط سیر در زمینه پالایش، فشرده‌سازی، قطعه‌بندی، تطبیق و تکمیل داده‌های جریانی خط سیر اشیا متحرک است [۵].

در میان روش‌های آماده‌سازی داده‌ها، روش "بخش‌بندی" به لحاظ دارا بودن قابلیت بالا در کاهش حجم و نمونه‌برداری خط سیر و همچنین کاربردی بودن آن در کشف دانش و استخراج الگو، نسبت به سایر روش‌ها از اهمیت ویژه‌ای برخوردار شده است؛ اما در مقابل با چالش‌هایی نیز روبرو می‌باشد؛ یکی از مهم‌ترین چالش‌های کارایی بخش‌بندی و خلاصه‌سازی، حفظ کیفیت داده‌هاست که تبدیل به موضوعی رقابتی در میان پژوهشگران این عرصه، به منظور ارائه مدل‌ها و الگوریتم‌های جدیدتر شده است. به منظور تبیین این موضوع، در این مقاله رویکردی جدید برای بهینه‌سازی فرآیندهای لجستیکی و بخش‌بندی داده‌های حاصل از سنسورهای ناوگان توزیع دارویی، با توجه به ماهیت جریانی بودن و محدودیت زمانی برای پردازش و حجم برای نگهداری داده‌ها، ارائه می‌گردد. به نظر می‌رسد این پژوهش از این نظر که بتواند به نتایج قابل قبولی از نظر ارائه رویکردی یکپارچه برای خلاصه‌سازی خط سیر با بخش‌بندی بهینه دست یابد، بسیار حائز اهمیت و جالب خواهد بود؛ چرا که روش‌ها و مدل‌های موجود با وجود کارآ بودن، از نظر سرعت و کیفیت پردازش پیچیده بوده و بهینه نیستند و این مقاله راهکاری نوین برای مواجهه با چالش‌های فعلی ارائه می‌دهد که می‌تواند به بهبود چشمگیر عملکرد سامانه‌های ناوگان توزیع دارویی در سطح کشور منجر شود.

ساختار مقاله به شرح زیر است: در بخش دوم، به مرور جامع و تحلیلی از مباحث و موضوعات مرتبط با پژوهش پرداخته شده است. بخش سوم به بررسی و تحلیل مطالعات پیشین اختصاص یافته است. در بخش چهارم، روش پیشنهادی به همراه ایده نوآورانه و الگوریتم

توسعه‌یافته مورد بحث و ارزیابی قرار می‌گیرد. بخش پنجم نتایج شبیه‌سازی‌ها و تحلیل‌های تجربی را ارائه می‌دهد. در نهایت، بخش پایانی به جمع‌بندی و نتیجه‌گیری از یافته‌های پژوهش می‌پردازد.

## ۲- ادبیات موضوع

### ۲-۱- پایگاه داده اطلاعات مکانی (GIS)

سیستم اطلاعات مکانی یا جغرافیایی، سیستمی است متشکل از نیروی انسانی متخصص، سخت‌افزار و نرم‌افزار و داده‌های مکانی که برای اخذ، ذخیره‌سازی، تغییر و تحول، تحلیل، مدیریت و ارائه داده‌های مکانی به کار می‌رود [۶]. وقتی در مورد سیستم اطلاعات جغرافیایی صحبت می‌شود، اغلب منظور همان سیستم اطلاعات جغرافیایی از نوع ایستاست؛ یعنی سیستم اطلاعات جغرافیایی، سیستمی است که اشیا از لحاظ مکان هندسی تغییر نمی‌کنند.

### ۲-۲- داده‌های اشیا متحرک

اشیا متحرک پدیده‌هایی هندسی هستند که موقعیتشان و یا شکلشان به طور پیوسته در طول زمان تغییر می‌کند. این ویژگی منحصر به فرد اشیا متحرک، باعث می‌شود تا سیستم‌های مدیریت پایگاه داده سنتی برای اشیا متحرک سازگار نباشند. این ناسازگاری به دلیل این واقعیت است که پایگاه داده‌های سنتی داده‌ها را ثابت و بدون تغییر فرض می‌کنند [۲]. انواع پدیده‌های متحرک که تحلیل جابجایی آن‌ها همواره مورد توجه دانشمندان و جوامع علمی قرار گرفته است، عبارتند از: حرکت انسان‌ها، حرکت خودروها، حرکت حیوانات، حرکت پدیده‌های طبیعی

### ۲-۳- بخش‌بندی<sup>۱</sup>

در بسیاری از موارد برای پیش‌پردازش داده‌های خط سیر از روش "بخش‌بندی" که موضوع اصلی این مقاله است استفاده می‌شود. در این روش یک خط سیر به زیربخش‌هایی تقسیم می‌شود که هر یک از این زیربخش‌ها یک تکه یا قسمت نامیده می‌شوند. انتخاب این زیربخش‌ها به جای کل خط سیر کاملاً منطقی است، زیرا به لایه‌های زیرین داده‌های خط سیر مرتبط هستند؛ برای مثال، یک مسیر با چند بخش جاده‌ای.

<sup>1</sup> Segmentation

یک رویکرد بخش‌بندی و خلاصه‌سازی، تلاش می‌کند تا داده‌های خط سیر را به صورت توصیفی و قابل استفاده برای افراد تولید کند، که برای این کار خط سیر را درون چندین بخش براساس رفتارهای اشیا متحرک تقسیم‌بندی می‌کند [۴].

## ۲-۴- تعریف "خط سیر"

داده‌های حرکتی اشیا متحرک که داده‌های خط سیر نامیده می‌شوند به صورت دنباله‌ای از مختصات‌های مکانی در قالب سری زمانی گزارش می‌شوند [۷]. به بیان ریاضی، در یک فضای مرجع  $S$ ، مانند سیستم مختصات اقلیدسی و یک دامنه از زمان به نام  $Time$ ، یک خط سیرمکانی مانند  $T$  به طول  $n$ ، عبارت است از یک دنباله  $n$  تایی از نقاط مکانی-زمانی است به گونه‌ای که  $T = \{(p_i, t_i)\}$  که  $p_i \in S$  و  $t_i < t_{i+1} \in Time$ . با توجه به نوع کاربرد خط سیر، فاصله این نقاط متوالی می‌تواند از لحاظ مکان یا زمان برابر و یا غیر برابر باشند [۵].

## ۲-۵- الگوریتم ژنتیک

الگوریتم‌های تکاملی<sup>۱</sup> از مکانیزم‌ها و عملیات ابتدایی برای حل مسأله، استفاده می‌کنند و در طی یک سری از تکرارها به راه مناسب برای حل مسأله می‌رسند. این الگوریتم‌ها غالباً از یک جمعیت<sup>۲</sup> حاوی راه‌حل‌های تصادفی شروع می‌کنند و در طی هر مرحله تکرار، سعی در بهتر کردن مجموعه راه‌حل‌ها دارند. در آغاز کار تعدادی از اعضای جامعه به صورت تصادفی حدس زده شده سپس تابع هدف برای هر یک از این اعضا محاسبه و نخستین نسل ایجاد می‌شود. اگر هیچ‌یک از معیارهای خاتمه بهینه‌سازی برآورده نشوند، ایجاد نسل جدید آغاز خواهد شد. اعضا برحسب میزان شایستگی‌شان برای تولید فرزندان<sup>۳</sup> انتخاب می‌شوند [۸]. این افراد به عنوان والدین<sup>۴</sup> محسوب می‌شوند و باز ترکیب فرزندان را تولید می‌نمایند. سپس تمامی فرزندان با یک مقدار معینی از احتمال، جهش و تغییر ژنتیکی می‌یابند. اکنون میزان شایستگی فرزندان تعیین و جایگزین والدین شده و نسل جدید را ایجاد می‌نمایند. این چرخه آن قدر تکرار می‌شود تا یکی از معیارهای پایان بهینه‌سازی برآورده شود [۹].

<sup>1</sup> Evolutionary algorithm

<sup>2</sup> Population

<sup>3</sup> offspring

<sup>4</sup> parents

### ۳- کارهای پیشین

در این بخش کلیه مقالات و پژوهش‌های انجام شده در زمینه "بخش‌بندی خط سیر" مورد بررسی و مرور قرار گرفته و ایده‌های کلی و استراتژی‌های مربوط به آن‌ها استخراج گردیده است. هدف از این کار شناسایی رویکردهای کلی در بخش‌بندی خط سیر و تجزیه و تحلیل آن‌ها به منظور اتخاذ استراتژی مناسب برای مدل پیشنهادی خواهد بود.

برای بخش‌بندی خط سیر اشیا متحرک در دهه اخیر، رویکردها و روش‌های متعددی ارائه شده است. این روش‌ها را می‌توان به دسته‌های مختلفی تقسیم کرد. یک دسته از روش‌های موجود به بخش‌بندی داده‌های مکانی خط سیر ذخیره شده در پایگاه داده اشیا متحرک<sup>۱</sup> پرداخته و داده‌های خط سیر اشیا متحرک در پایگاه داده را مورد بخش‌بندی قرار می‌دهند [۱۰] در کنار این رویکردها، روش‌هایی هم وجود دارد که بر روی داده‌های خط سیر جریان‌ی بدون امکان ذخیره‌سازی داده‌ها که با محدودیت‌های خاصی مواجه هستند مربوط می‌شود [۱۱]؛ همچنین روش‌های بخش‌بندی را می‌توان به صورت باناظر<sup>۲</sup> [۱۲، ۱۳]، شبه باناظر<sup>۴</sup> [۱۴] و بدون ناظر<sup>۵</sup> [۱۵، ۱۶] طبقه‌بندی کرد. در روش باناظر مقادیر مختلفی از پارامترهای ورودی برای ایجاد بخش‌های مختلف مورد استفاده قرار می‌گیرد. در اینجا معیارهای بخش‌بندی مبتنی بر تعدادی قواعد از پیش تعریف شده هستند. در روش شبه باناظر ترکیبی از روش‌های باناظر و بدون ناظر مورد استفاده قرار می‌گیرد و در این روش سعی می‌شود مزایای هر دو روش قبلی مدنظر قرار گیرد. در روش بدون ناظر، هیچ معیار از پیش تعیین شده‌ای در فرآیند بخش‌بندی اعمال نمی‌شود و تقسیم سگمنت‌ها بر اساس ویژگی‌ها و صفات داده‌های خط سیر است [۱۴، ۱۵]. در روش‌های بدون ناظر به منظور دستیابی به حداکثر همگنی<sup>۶</sup> در سگمنت‌ها و حداقل شدن تعداد آن‌ها تابعی تحت عنوان تابع هزینه<sup>۷</sup> بر مبنای اصل MDL<sup>۸</sup> [۱۷] تعریف می‌شود. برای بهینه‌سازی و کمینه‌سازی تابع هزینه، الگوریتم‌های مختلف کلاسیک، ابتکاری<sup>۹</sup> و فرا ابتکاری<sup>۱۰</sup> پیشنهاد شده است. در الگوریتم‌های کلاسیک تابع هزینه توسط یک الگوریتم قطعی و در زمان چندجمله‌ای قابل حل می‌باشند که با جستجو و مشاهده همه داده‌ها، می‌توانند پاسخ کاملاً دقیق و قطعی را به

<sup>1</sup> MOD: Moving Objects Database

<sup>2</sup> Stream

<sup>3</sup> Supervised

<sup>4</sup> Semi Supervised

<sup>5</sup> Unsupervised

<sup>6</sup> Homogeneity

<sup>7</sup> Cost Function

<sup>8</sup> MDL: Minimum Description Length

<sup>9</sup> Heuristic

<sup>10</sup> MetaHeuristic

دست آورند [۱۸]. این الگوریتم‌ها، سرعت پایینی دارند و طبیعتاً برای مسائلی با داده‌ها و ابعاد زیاد، کاربرد چندانی ندارند. اگرچه حالت بهینه بخش‌بندی در این روش دارای جواب قطعی است ولی در مواردی که داده‌های پیوسته و جریانی با محدودیت زمانی وجود دارد، این روش جوابگو نیست؛ به همین دلیل روش‌ها و الگوریتم‌های ابتکاری ارائه شده است [۵]. روش‌های ابتکاری، می‌توانند قسمت‌های مشخصی از داده‌ها را جستجو کنند و با این کار سرعت تولید جواب را بالا می‌برند. ولی الگوریتم‌های ابتکاری، تضمین نمی‌دهند که جواب به دست آمده، بهترین جواب ممکن باشد. یعنی اگر محدودیت زمانی نباشد، با جستجوی بیشتر احتمالاً پاسخی بهتر نیز پیدا خواهد شد. یک الگوریتم ابتکاری برای یک مسئله خاص پاسخگو خواهد بود. این جاست که یک سری دیگر از الگوریتم‌ها توسعه داده شده‌اند که بدون آگاهی از مسئله می‌توانند به جستجو برای پاسخ بهینه بپردازند. این روش‌ها الگوریتم‌های فرا ابتکاری یا Metaheuristic می‌باشند. در واقع الگوریتم‌های فرا ابتکاری قادر هستند بدون آگاهی از مسئله، با ارائه یک راه‌حل عمومی مسئله را با سرعت و دقت معقولی حل کنند. این روش‌ها در اصطلاح مستقل از مسئله<sup>۱</sup> هستند. نمونه‌هایی از الگوریتم‌های فرا ابتکاری در مقالات [۱۵، ۱۹] ارائه شده است. در الگوریتم مقاله [۱۵] که با نام GRASP-UTS شناخته می‌شود؛ از یک الگوریتم حریمانه<sup>۲</sup> فرا ابتکاری استفاده می‌شود که جزو الگوریتم‌های Local search است. این الگوریتم ابتدا تعدادی نقطه با عنوان Landmark به صورت تصادفی انتخاب کرده و سپس شباهت نقاط داخل سگمنت را بر پایه فاصله اقلیدسی<sup>۳</sup> صفات تعیین می‌کند. به این صورت یک جواب ممکن و اولیه<sup>۴</sup> برای مسأله بدست می‌آید. سپس بر پایه Local Search نقاط بهینه محلی تعیین می‌شوند. این الگوریتم تعدادی ورودی شامل حداقل زمان برای بخش‌بندی، ضریب شباهت برای متصل کردن سگمنت‌های متوالی، یک ضریب تقسیم برای اجرای جستجوی محلی در سگمنت و همچنین یک ضریب آستانه‌ای برای میزان حریمانه بودن الگوریتم برای ساخت سگمنت دریافت می‌کند. برای جلوگیری از گیر افتادن الگوریتم در نقاط محلی از جایجایی Landmark‌ها بهره می‌گیرد. با توجه به اینکه این متد به صورت Local Search عمل می‌کند و برای اینکه در نقاط محلی گیر نیفتد مدام جای Landmark را جابجا می‌کند بنابراین دارای داده‌های تکراری و فراخوانی‌های متعدد و حلقه‌های تودرتو بوده و این موضوع زمان اجرای الگوریتم را افزایش می‌دهد. به همین دلیل در حل مسائل به جای استفاده از روش‌های هیوریستیک مانند

<sup>1</sup> Problem Independent

<sup>2</sup> Greedy algorithm

<sup>3</sup> Euclidian Distance

<sup>4</sup> Feasible Solution



GRASP، از الگوریتم ژنتیک استفاده شده است [۲۰]. با توجه به بررسی‌های انجام شده در روش‌های قبلی بهینه‌سازی تابع هزینه، در پژوهش حاضر، برای نخستین بار از الگوریتم فرا ابتکاری ژنتیک با توجه به ویژگی‌های خاص برای بهینه‌سازی تابع هزینه در بخش‌بندی بدون ناظر خط سیر استفاده شده است.

#### ۴- روش پیشنهادی

در این مقاله مشابه مقاله [۱۵] از اصل MDL به منظور اطمینان از صحت تقسیم‌بندی خط سیر مربوط به داده‌های ناوگان توزیع دارو، استفاده می‌شود. در تقسیم‌بندی خط سیر،  $H$  مجموعه نقاط نماینده سگمنت‌های خط سیر است و  $D$  داده‌های اصلی خط سیر است.  $L(H)$  مجموع فاصله همه نقاط نماینده سگمنت‌هاست.  $L(D|H)$  نشان‌دهنده فاصله بین نقاط نماینده سگمنت‌ها و داده‌های اصلی خط سیر است. براساس اصل MDL و تئوری اطلاعات،  $L(H)$  به صورت زیر تعریف می‌شود:

$$L(H) = \log_2 \left( 1 + \sum_{(j=1)}^{(T-1)} \max\_dist - distance(lm_j, lm_{(j+1)}) \right) \quad (۱)$$

$$\max\_dist = distance(p_1, p_2) = \sqrt{\sum_{i=1}^M (p_{1i} - p_{2i})^2}$$

از رابطه ۱ برای محاسبه فاصله اقلیدسی بین نماینده دو بخش متوالی استفاده می‌شود. اگر میزان تشابه نماینده دو بخش کم باشد، مقدار  $L(H)$  کاهش خواهد یافت. برای محاسبه فاصله همه نقاط درون یک بخش با نقطه نماینده آن بخش، از رابطه ۲ استفاده می‌شود.

$$L(D|H) = \log_2 \left( 1 + \sum_{(k=1)}^T cohesiveness(lm_k, [p_u, \dots, p_v]_k) \right) \quad (۲)$$

$$cohesiveness(lm, p_u, \dots, p_v) = \sum_{k=u}^v distance(lm, p_k)$$

از  $L(H)$  برای اندازه‌گیری حداکثر جدایی میان سگمنت‌ها و از  $L(D|H)$  برای اندازه‌گیری میزان تراکم درون سگمنت‌ها استفاده می‌شود. در اصل MDL به دنبال کمینه‌سازی حاصل جمع  $L(H)$  و  $L(D|H)$ ، همان تابع هزینه یا  $f$ ، به صورت رابطه ۳ هستیم [۱۵].

$$\text{Cost\_function} = \text{MDL} = L(H) + L(D|H) \quad (۳)$$

در این رابطه مقدار کمینه تابع هزینه، بهترین مصالحه بین حداکثر تمایز میان سگمنت‌ها و حداکثر میزان شباهت درون سگمنت‌ها خواهد بود تا بتوانیم نقاط داخل خط سیر را به گونه‌ای بخش‌بندی کنیم که نقاط داخلی شبیه‌ترین حالت و نقاط بیرونی غیرمشابه باشند؛ یعنی نقاط داخلی هر بخش بیشترین شباهت را از نظر صفات داشته باشند درعین حال نقاط نماینده هر بخش بیشترین تفاوت و تمایز را با هم داشته باشند؛ از آنجاکه مسأله این پژوهش یک مسأله بهینه‌سازی غیرخطی و غیرقطعی است، باید از روش‌های غیرقطعی استفاده شود. همچنین این مسأله یک نوع مسأله برنامه‌ریزی پویاست اما به دلیل زیاد بودن ابعاد مسأله، روش برنامه‌ریزی پویا نیز در یافتن پاسخ بهینه دچار مشکل می‌شود [۲۱].

#### ۴-۱- الگوریتم پیشنهادی SEGA

در میان تمام روش‌های موجود برای بهینه‌سازی، روش الگوریتم‌های تکاملی (فرا ابتکاری) از جمله الگوریتم ژنتیک می‌تواند زمان محاسبات کمتری را در هنگام افزایش تعداد متغیرها نسبت به دیگر روش‌ها داشته باشد [۲۲]. در واقع الگوریتم ژنتیک روشی فرا ابتکاری می‌باشد که با الهام از اصل بقای شایسته‌ترین در موجودات زنده عمل می‌کند [۸، ۲۳]. در این الگوریتم اعضای شایسته یک نسل، برای ایجاد نسلی از اعضای احتمالاً شایسته‌تر ترکیب می‌شوند. این الگوریتم با ساختار متغیرهای گسسته همخوانی دارد و دیگر آن که ابعاد بزرگ مسأله در این الگوریتم مشکل حادی ایجاد نمی‌کند و این الگوریتم می‌تواند در زمان قابل قبول، جوابی "تقریباً بهینه" را پیدا کند [۸، ۲۳].

ما در این مقاله الگوریتمی بر پایه‌ی الگوریتم ژنتیک ارائه می‌دهیم، به طوری که در زمان بهتر و با کارایی بهتر، نتیجه مورد نظر را به دست آورد. برای پیاده‌سازی الگوریتم SEGA (SEGmentation based on Genetic Algorithm)، لازم است ابتدا ساختار رشته‌ها و جمعیت ابتدایی در رابطه با بخش‌بندی تعریف و سپس ایده‌ی موردنظر در بخش ۴-۲ مطرح می‌گردد. روند کار در ادامه نشان داده شده است.

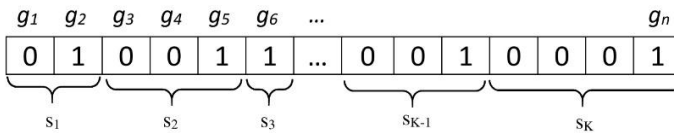
#### ۴-۱-۱- ساختار یک نمونه کروموزوم برای بخش‌بندی خط سیر - سطر ۱

##### الگوریتم اصلی

در ابتدا نیاز است راه‌حل‌های مناسب به صورت رشته‌های کروموزومی تعریف شوند. در این پژوهش یک کروموزوم معادل یک خط سیر است که بخش‌بندی خط سیر توسط مقادیر "صفر" و "یک" مشخص می‌گردد. یک کروموزوم در واقع مجموعه‌ای از ژن‌ها است به طوری که یک راه‌حل پیشنهادی را برای مسأله‌ای که سعی در حل آن داریم، تعریف می‌نماید.

در این پژوهش، ما یک نمونه جواب مسأله را به عنوان کروموزوم به شکل زیر تعریف کرده‌ایم: تعداد ژن‌های هر کروموزوم را برابر با تعداد نقاط یک نمونه خط سیر در نظر می‌گیریم که این ژن‌ها شامل مقادیر "صفر" و "یک" خواهد بود. بنا به "صفر" و "یک" بودن این ژن‌ها، در مورد حضور نقاط در هر سگمنت تصمیم‌گیری می‌شود؛ یعنی زمانی که مقدار یک ژن (نقطه از خط سیر) "یک" باشد نشان‌دهنده آن است که کلیه نقاط قبل از آن تا نقطه مورد نظر، یک سگمنت را تشکیل می‌دهند. هر ژنی که مقدار "یک" دارد، انتهای سگمنت را مشخص می‌کند. شکل ۱ نمایش یک کروموزوم را نشان می‌دهد، به طوری که `Sample_Chromosome`، یک نمونه جواب پیشنهادی برای مسأله مورد نظر است.

$g_i$  ژن‌های کروموزوم است که  $1 < i < n$  و  $n$  نشان‌دهنده تعداد نقاط یک خط سیر است. هر  $k$  در شکل ۱ نشان‌دهنده یک سگمنت است.  $1 < k < K$  (تعداد بخش‌های یک خط سیر است). همان‌طور که در شکل ۱ نیز نشان داده شده است، پایان هر سگمنت با ژن "یک" خاتمه یافته است و از ژن بعدی، سگمنت بعد آغاز می‌شود. در هر راه‌حل مسأله که توسط الگوریتم ژنتیک و با استفاده از عملگرهای جهش و تقاطع (بُرش) به وجود می‌آید انواع  $S_k$  (مدل‌های مختلف بخش‌بندی) ایجاد می‌شود که هر کدام می‌تواند به عنوان یک جواب باشند.



شکل ۱: نمونه‌ای از یک کروموزوم برای مسأله بخش‌بندی

زمانی که ترکیب هر کروموزوم مانند شکل ۱ مشخص شد جمعیت ابتدایی به صورت تصادفی و به صورت دودویی با  $n$  ژن در هر کروموزوم، ایجاد می‌شود ( $n$  تعداد نقاط یک خط سیر). سپس هر ژن کروموزوم به انتهای نقاط یک خط سیر مربوط می‌شود و مشخص می‌کند که کدام نقاط، داخل کدام سگمنت‌ها قرار بگیرد. در شکل ۲ به عنوان مثال نقطه ۱ (Point 1) و نقطه ۲ (Point 2) در داخل یک بخش قرار می‌گیرند. چرا که بیت  $g_k$  این نقاط در دومین نقطه "یک" است؛ یعنی انتهای یک سگمنت را نشان می‌دهد.

	$g_1$	$g_2$	$g_3$	$g_4$	$g_5$	$g_6$	...	...	...	...	...	$g_n$	
Sample_Chromosome	0	1	0	0	1	1	...	0	0	1	0	0	1
	...	...	...									...	
	$f_{13}$	$f_{23}$	$f_{33}$	...			...					$f_{n3}$	
	$f_{12}$	$f_{22}$	$f_{32}$	...			...					$f_{n2}$	
	$f_{11}$	$f_{21}$	$f_{31}$	...			...					$f_{n1}$	
	$y_1$	$y_2$	$y_3$	...			...					$y_n$	
	$x_1$	$x_2$	$x_3$	...			...					$x_n$	
	$p_1$	$p_2$	$p_3$	...			...					$p_n$	

### شکل ۲: ساختار نقاط خط سیر

$x_i$  و  $y_i$  مختصات نقاط خط سیر و  $f_{ir}$  صفات نقاط خط سیر هستند

$$1 < n < i \quad (n \text{ تعداد نقاط یک خط سیر})$$

$$1 < m < j \quad (m \text{ تعداد صفات نقاط یک خط سیر})$$

در ادامه با استفاده از رابطه ۴ نقطه نماینده هر سگمنت تعریف می‌شود.  $x_r$  و  $y_r$  مربوط به مختصات نقطه نماینده و  $f_{ir}$  مربوط به صفات نقطه نماینده می‌باشد.

$$\begin{aligned} x_r &= \text{mean}(x_u \dots x_v) \\ y_r &= \text{mean}(y_u \dots y_v) \\ f_{ri} &= \text{mean}(f_{ru} \dots f_{rv}) \end{aligned} \quad (۴)$$

$i$ : اندیس مربوط به نقطه نماینده

و در انتها، با استفاده از رابطه ۳ مقدار تابع هدف به ازای هر کروموزوم ارزیابی می‌شود.

### ۴-۱-۲- تعریف جمعیت اولیه در الگوریتم SEGA

تعداد کروموزوم‌ها در یک جمعیت معرف اندازه آن جمعیت است، بعضی از تحقیقات اخیر نشان داده‌اند که برای مسائل با پیچیدگی متوسط، مناسب‌ترین حالت داشتن جمعیتی برابر با تعداد متغیرهاست (تعداد نقاط خط سیر) [۲۴]. اگر تعداد کروموزوم‌ها کم باشد امکان انجام صحیح عملیات جابجایی کم است و تمام فضای جستجو کاوش نمی‌شود [۲۵، ۲۶] و اگر تعداد آن‌ها زیاد باشد الگوریتم کند می‌شود [۲۷، ۲۸]. بنابراین، اندازه بهینه جمعیت با آزمایش و خطا تعیین می‌شود. در این مقاله جهت افزایش سرعت عملیات بخش‌بندی، جمعیت اولیه

را هوشمندانه انتخاب کرده‌ایم. از آنجاکه می‌توان حدس زد که بعد از اجرای الگوریتم تعداد سگمنت‌ها از یک مقداری تجاوز نمی‌کند، در تعریف جمعیت اولیه، تعداد "صفر"ها و "یک"ها را به گونه‌ای در نظر می‌گیریم که تعداد سگمنت‌ها از آن مقدار تجاوز نکند.

#### ۴-۱-۳- محاسبه تابع هدف - سطر ۴ الگوریتم اصلی

برای محاسبه مقدار شایستگی هر جواب (هر کروموزوم) لازم است مقدار تابع هدف (رابطه ۳) را برای آن جواب محاسبه کنیم. با به دست آوردن نقطه نماینده مربوط به هر سگمنت و داشتن مشخصات نقاط با استفاده از رابطه ۳ تابع هدف محاسبه می‌شود.

#### ۴-۱-۴- تعریف عملگرهای الگوریتم ژنتیک در SEGA:

##### (a) عملگر Selection - سطر ۵ الگوریتم اصلی

برای ایجاد نسل‌های بعدی از عملگر "انتخاب" استفاده می‌شود. برای هر کروموزوم یک جفت والد انتخاب می‌شود. انتخاب‌ها به گونه‌ای است که مناسب‌ترین عناصر انتخاب شوند تا حتی ضعیف‌ترین عناصر هم شانس انتخاب داشته باشند تا از گیر افتادن در جواب‌های محلی جلوگیری شود. در این مقاله از مکانیزم چرخ رولت [۹] که پرکاربردترین مکانیزم انتخاب در الگوریتم ژنتیک است، استفاده می‌شود.

##### (b) عملگر تقاطع - سطر ۷ الگوریتم اصلی

در عملگر تقاطع با استفاده از نرخ تقاطع، درصد جمعیت فعال حاضر، جهت شرکت در عمل تقاطع را می‌توان تعیین کرد. با استفاده از این عملگر، با ترکیب ژن‌های یک جفت والد برگزیده، کروموزوم جدید به وجود می‌آید. در این عملگر ابتدا با استفاده از نرخ تقاطع دو والد انتخاب شده سپس محلی برای ادغام در طول رشته کروموزوم انتخاب می‌شود و نهایتاً مقدار دو رشته با توجه به محل ادغام جابجا می‌شود. در این مقاله از روش تقاطع دو نقطه‌ای [۹] استفاده شده است.

##### (c) عملگر جهش - سطر ۶ الگوریتم اصلی

در عملگر جهش، کروموزوم به تنهایی و بدون نیاز به ادغام با کروموزوم دیگر براساس قوانین احتمال می‌تواند تغییر کند. برای انجام این کار، نیاز به یک نرخ جهش می‌باشد. با استفاده از نرخ جهش، احتمال حضور کروموزوم در عملگر جهش مشخص می‌شود. نرخ جهش یک احتمال کوچک بین "صفر" تا "یک" است. در این مقاله از روش جهش بیتی استفاده می‌شود [۹].

#### ۴-۲- ایده پیشنهادی جهت بهبود عملگر جهش

در این پژوهش در صورتی که تعداد داده‌ها کمتر از ۱۰۰ باشند عملگر جهش به روش جهش بیتی که در بالا معرفی شد عمل می‌شود؛ اما وقتی که تعداد داده‌ها افزایش می‌یابد، هیوریستیکی برای انجام عملگر جهش پیشنهاد می‌شود؛ یعنی با افزایش داده‌ها، برای ایجاد تنوع در فضای جستجو مناسب‌ترین حالت این است که تعداد جمعیت اولیه برابر با تعداد کروموزوم‌ها باشد [۲۴]، این امر باعث می‌شود که سرعت عمل بهینه‌سازی کاهش یابد. همچنین با افزایش داده‌ها، تعداد ژن‌های یک کروموزوم نیز افزایش می‌یابد که انجام روش جهش بیتی فقط در یک ژن، تنوع کمتری ایجاد می‌کند. بنابراین جهت افزایش تنوع و جلوگیری از کاهش سرعت برای هر ژن یک احتمال بین "صفر" و "یک" تعیین می‌شود و براساس یک آستانه، جهش بیتی در ژن‌ها رخ می‌دهد. انتخاب آستانه (که به صورت مکاشفه‌ای در نظر گرفته می‌شود) برای تعیین تعداد ژن‌هایی است که دچار جهش خواهند شد. در این پژوهش در صورتی که تعداد سگمنت‌ها (هر ژن با مقدار "یک" در کروموزوم که نشان دهنده پایان یک سگمنت است) زیاد باشد (مقدار تابع هدف تعریف شده در رابطه ۳) به جای کاهش، افزایش می‌یابد. بنابراین در جهت رفع این مسأله برای عمل جهش بیتی، هیوریستیکی به صورت رابطه ۵ (عملگر جهش بهبود یافته) تعریف می‌شود:

(۵)

$$new\_g_i = \begin{cases} g_i & \text{if } \tau_i < threshold \\ 0 & \text{if } \tau_i > threshold \text{ and } g_i = 1 \text{ and } (g_{i-1} = 1 \text{ or } g_{i+1} = 1) \end{cases}$$

$\tau_i$  احتمال جهش هر ژن که به صورت تصادفی بین "صفر" و "یک" تولید می‌شود. ( $1 < n < i$ )

$n$ : تعداد نقاط خط سیر

این جهش به این صورت تعریف می‌شود که اگر مقدار  $\tau_i$  برای ژن والد، پایین‌تر از مقدار آستانه باشد، آن ژن مقدار خود را بدون تغییر، در کروموزوم فرزند حفظ می‌کند اما در صورتی که مقدار  $\tau_i$  برای ژن والد بالاتر یا مساوی مقدار آستانه باشد و مقدار ژن والد برابر "یک" باشد، ژن‌های همسایه خود را با فاصله یک بررسی می‌کند اگر مقدار حداقل یکی از آن‌ها برابر با "یک" باشد، پس مقدار ژن کروموزوم فرزند "صفر" خواهد بود (شکل ۳). بررسی کردن همسایه به این دلیل است که سگمنت‌های پشت سر هم باعث افزایش مقدار رابطه ۳ یا همان تابع هدف خواهد شد. بنابراین عملیات جهش باعث تغییر در تعداد سگمنت‌ها می‌شود. ما در این پژوهش مقدار آستانه را برابر با ۰.۸۵ در نظر می‌گیریم تا

احتمال جهش ژن‌ها تا حدی افزایش یابد.

	$g_1$	$g_2$	$g_3$	$g_4$	$g_5$	$g_6$	...	$g_n$
Parent	0	0	1	0	1	1	...	1
	$\tau_1$	$\tau_2$	$\tau_3$	$\tau_4$	$\tau_5$	$\tau_6$	...	$\tau_n$
Probability	0.35	0.15	0.86	0.17	0.9	0.65	...	0.5
	$g_1$	$g_2$	$g_3$	$g_4$	$new\_g_5$	$g_6$	...	$g_n$
Offspring	0	0	1	0	0	1	...	1

شکل ۳: نمایش عملگر جهش بهبود یافته

#### Algorithm . Extended Mutation

**Input:** 1- A set of population that select from initial population (p)

2- A population number (pop\_num)

3- A rate for mutation operation (mute\_rate)

4- A gen number of chromosome (gen\_num)

**Output:** A set of population that generate by cross-over (new\_pop)

1. threshold = 0.85

2. mut\_count = mutation\_rate \* (gen\_num) \* pop\_num

3. rnd = generate a integer random for choosing a chromosome for mutation

4. for i = 1 : mut\_count

5. chrom = new\_pop (rnd)

6. rnd = generate random number between [0 , 1] for any gen of chromosome

7. for k = 1 : num\_g

8. if rnd (p<sub>k</sub>) > threshold and gen<sub>k</sub> = 1 and (gen<sub>k-1</sub> = 1 or gen<sub>k+1</sub> = 1)

9. chrom<sub>k</sub> = 1

10. end

11. end

12. new\_pop = insert chrom<sub>k</sub> to population

13. end

14. return new\_pop

#### الگوریتم Extended Mutation

ابتدا با استفاده از نرخ پرش و یک عدد تصادفی، تعداد و شماره کروموزم‌هایی که برای پرش انتخاب خواهند شد، مشخص می‌شود (خطوط ۲ و ۳). برای هر یک از ژن‌های کروموزم‌های انتخاب شده، یک عدد تصادفی بین "صفر" و "یک" تولید می‌شود (خط ۴). سپس با استفاده از آستانه معرفی شده در خط ۱ و با استفاده از هیوربستیک معرفی شده در رابطه ۵ برای هر یک از این کروموزم‌های انتخابی عمل پرش را انجام می‌دهیم (خطوط ۷ تا ۱۱). کروموزم‌های جدید وارد جمعیت خواهند شد و جمعیت جدید از عمل پرش تولید می‌شوند (خط ۱۲).

در این پژوهش، جهت تسریع در الگوریتم ژنتیک، بهتر است عمل جهش بیشتر از عمل

برش انجام پذیرد، چرا که عمل جهش کمک می‌کند فضای مسأله را با وجود هیوریستیک معرفی شده در تعریف جمعیت اولیه، بهتر و کامل‌تر جستجو کنیم. از آنجا که تعداد سگمنت‌ها در عمل برش تغییر نمی‌کند؛ این تغییر در استفاده از عمل جهش اتفاق می‌افتد. هردو فرآیند جهش و برش باعث به وجود آمدن نسل جدیدی از کروموزوم‌هایی می‌شود که با نسل قبلی متفاوت‌اند. کل این فرآیند برای نسل بعدی هم تکرار می‌شود و این فرآیند تا این که به آخرین مرحله (همان شرط خاتمه الگوریتم) برسیم، ادامه پیدا می‌کند. شرط خاتمه الگوریتم در صورتی خاتمه می‌یابد که تغییری در بهترین جمعیت در طول چند نسل متوالی رخ ندهد. فرآیند بخش‌بندی با روش SEGA در قالب الگوریتم اصلی انجام می‌شود.

---

**MainAlgorithm (SEGA)** Pseudo code for select best segmentation base on minimum cost function

---

**Input:** 1- A set of initial population (pop)

2- A population number (pop\_num)

3- Mutation rate (mut\_rate)

4- A rate for cross over operation (cross\_rate)

5- A gen number of chromosome (gen\_num)

6- A segment counter (seg\_count)

7- Points

**Output:** best\_fitness

1. Choose encode method

2. Generate the initial population

3. while  $i < \text{MaxIteration}$  and  $\text{best\_fitness} < \text{Maxfitness}$  do

4.  $[c_{f_1}, \dots, c_{f_n}] = \text{calculate}(\text{pop}, \text{pop\_num}, \text{seg\_count}, \text{points}, \text{gen\_num})$

5.  $\text{new\_pop} = \text{Selection}(\text{pop}, [c_{f_1}, \dots, c_{f_n}], \text{pop\_num})$

6.  $\text{Extended\_Mutation}(\text{new\_pop}, \text{pop\_num}, \text{gen\_num}, \text{mut\_rate})$

7.  $\text{Crossover}(\text{new\_pop}, \text{pop\_num}, \text{cross\_rate})$

8. end

9. decode the individual with maximum fitness

10. return the best\_fitness

---

### الگوریتم اصلی

در الگوریتم اصلی، ابتدا پارامترهای اولیه مربوط به الگوریتم SEGA معرفی می‌شوند. سپس روش نمایش و انتخاب کروموزوم‌ها با توجه به مسئله مربوطه مشخص می‌شود (خط ۱). سپس در خط ۲ جمعیت اولیه را تعریف می‌کنیم. تا زمانی که شرط خاتمه برقرار نشود، برای جمعیت اولیه، تابع هدف محاسبه شده سپس با استفاده از نتایج تابع هدف و چرخ رولت، عملگر انتخاب اجرا می‌شود. سرانجام عملگر برش و پرش انجام می‌شود (خطوط ۳ تا ۸) در نهایت در خط ۹، بهترین کروموزوم (بهترین جواب) با توجه به نیازمان بازنمایی می‌شود.



### ۴-۳- داده‌های مورد بررسی

داده‌هایی که برای ارزیابی در این مقاله به کار می‌روند داده‌های ناوگان توزیع دارو، داده طوفان آتلانتیک و داده GeoLife دارای یک فیلد با مقدار مشخص هستند، که در ادامه به طور جزئی بیان شده‌اند:

### الف) داده‌های ناوگان توزیع دارو

این داده‌ها به یک شرکت داروسازی در ایران تعلق دارد که اختصاصاً در این مقاله به کار گرفته شده‌اند. این شرکت سیستم نظارتی پیشرفته‌ای برای مدیریت ناوگان توزیع داروی خود در سطح کشور راه‌اندازی کرده است. این سیستم به عنوان یک پلتفرم جامع مکانی-زمانی طراحی شده و به طور خاص برای مانیتورینگ دقیق خودروهای یخچال‌دار حامل دارو به کار می‌رود. در این سامانه، از یک دستگاه GPS سفارشی به همراه مجموعه‌ای از حسگرهای پیشرفته بهره گرفته شده است که وظیفه جمع‌آوری داده‌های حیاتی مانند سرعت خودرو و دمای داخلی یخچال را بر عهده دارند.

اطلاعات جمع‌آوری شده توسط این حسگرها به صورت بسته‌های داده‌ای، پس از ثبت ۱۵ رکورد، از طریق بستر ارتباطی GPRS به مرکز کنترل ارسال می‌شوند. این اطلاعات شامل مقادیر اندازه‌گیری شده از حداکثر ۶ حسگر مختلف است. برای افزایش کارایی و حذف داده‌های غیرضروری، سیستم به گونه‌ای برنامه‌ریزی شده است که در صورت کاهش سرعت خودرو به کمتر از ۳ کیلومتر بر ساعت، از ارسال داده جلوگیری کند. این رویکرد نه تنها از بار ترافیک داده‌ها در شبکه می‌کاهد، بلکه دقت و کارآمدی فرآیند نظارت را نیز بهبود می‌بخشد. داده‌های جمع‌آوری شده در بازه یک ماهه (۳۰ روز) برای هر خودرو در یک مسیر ۸ ساعته ۱۵ هزار نقطه بوده که در ماه اول با ۳۰۰ دستگاه خودرو، ۴۵۰ هزار رکورد است که حدود ۱۶۰ گیگابایت فضا اشغال کرده است. قابل ذکر است که برای هر نقطه یک رکورد که شامل داده‌های مختصات دریافتی از GPS، سرعت در آن نقطه، دمای درون یخچال، دمای بیرونی، ساعت و تاریخ ذخیره می‌گردد. در سخت‌افزار اختصاصی طراحی شده، قابلیت اتصال ۶ سنسور وجود دارد و براساس الگوهای تعریف شده در سامانه نرم‌افزاری، امکان تعریف و رصد محدوده حرکت، حداکثر سرعت، حداقل و حداکثر دما برای هر خودرو وجود دارد.



شکل ۴: واسط کاربری سامانه ناوگان توزیع دارویی

فرمت داده‌ها به شکل زیر است:

شماره	8 عدد نما و وضعیت		وضعیت سیستم	وضعیت سیم‌ریز	LAT	LONG	SPEED	سیگنال سیم‌ریز GSM	سیگنال GPS	ID	شماره باقیمانده
#1	A	B	C	D	E	F	G	H	I	J	
2	64	8	8	6	6	3	2	2	4	8	
#1	102350351023503510235035102350351023503510235035102350351023503510101010111110146233432456404030001*102000										

جدول ۱: فرمت پروتکل درخواست داده وضعیت سیستم

**Send:\*1**  
**Receive: \*110235035102350351023503510235035102350351023503510235035102350351023503510101010111110146233432456404030001\*102000**

A: فرمت نما و وضعیت بصورت یک عدد 8 بیتی بصورت زیر است و به ترتیب از سنسور یک الی هفت

یکان وضعیت	نشان وضعیت	وضعیت موتور/آرزش 0 خاموش	انشار نما	یکان نما	نشان نما	سیگنال نما	علامت نما 0-0
10235035	دما 23.5 درجه موتور خاموش و وضعیت 35 درصد						

**دقت شود در پرنتکل جدید علامت بجای مثبت و منفی با 1 و صفر نشان داده میشود**

جدول ۲: فرمت نما و وضعیت

B: وضعیت سیستم یک عدد 8 بیتی با فرمت زیر

آرزش	برون شبکه	دافراست	موتور	بازری	GPS	GPRS	X
0	میر فعال	1	فعال				

C: وضعیت سنسور ما یک عدد 8 بیتی که سفر نشان دهنده ایر عمل بودن و یک نشان دهنده فعل بودن سنسور است

D: طول جغرافیایی یک عدد شش بیتی مطابق فرمت زیر

شکل ۵: فرمت داده‌های ناوگان توزیع دارویی

### ب) داده‌های طوفان آتلانتیک

داده‌های بعدی جهت مقایسه و ارزیابی الگوریتم معرفی شده در این مقاله داده‌های طوفان آتلانتیک است که اطلاعات خط سیر طوفان آتلانتیک در بازه زمانی سال ۱۹۰۰ تا ۲۰۱۲ میلادی را در برمی‌گیرد. تعداد کل رکوردها ۳۸۸۹۲ می‌باشد. فیلدهای داده‌ها شامل مختصات جغرافیایی، سرعت باد، تاریخ و ساعت وقوع، جهت باد است که دارای یک فیلد Label می‌باشد که این برچسب مشخص‌کننده وجود طوفان در آن نقطه مورد نظر با مقدار Stop و Move می‌باشد.

### ج) داده‌های GeoLife

مجموعه داده GeoLife شامل ۱۷۶۲۱ خط سیر است که توسط ۱۸۲ کاربر در طول ۵ سال (از آوریل ۲۰۰۷ تا آگوست ۲۰۱۲) در محدوده شهر پکن براساس داده‌های GPS کاربران به منظور مدل‌سازی فعالیت‌های مختلف کاربران مانند خرید، گشت‌وگذار و... جمع‌آوری شده است. داده‌ها در بازه‌های ۱ تا ۵ ثانیه یا هر ۵ تا ۱۰ متر می‌باشد.

### ۵- شبیه‌سازی

محیط پیاده‌سازی الگوریتم ما برنامه MATLAB بوده است. برای انجام این آزمایش از یک کامپیوتر با پردازنده CORE i5 CPU (2.4GHz) و حافظه اصلی 8GB استفاده شده است. سیستم عامل دستگاه Windows 10 و محیط MATLAB نسخه ۱۶ از نوع ۶۴ بیتی بوده است.

### ۵-۲- شاخص‌های ارزیابی

از دو شاخص Average Purity و Average Coverage به عنوان معادل دو شاخص دقت (precision) و صحت (Recall) [۱۵] جهت ارزیابی الگوریتم استفاده می‌شود. در این پژوهش علاوه بر این دو شاخص از متریک F\_Score که در مقاله [۲۹، ۳۰] معرفی شده است، به عنوان میانگین هارمونیک دو مقدار Average Purity و Average Coverage نیز استفاده شده است. براساس مقاله [۱۵] Average Purity و Average Coverage به صورت رابطه ۸ و رابطه ۹ تعریف می‌شوند:

$$Average\ Coverage(\psi_V, \theta_T) = \frac{100}{V} \left( \sum_{i=1}^V \arg \max_{j \in [1, T]} \frac{N_{intersects}(\psi_i, \theta_j)}{N_i} \right) \quad (8)$$

$$Average\ Purity(\theta_T, \gamma_L) = \frac{100}{T} \left( \sum_{i=1}^T \arg \max_{j \in [1, L]} \left( \frac{N_{ij}}{N_i} \right) \right) \quad (9)$$

میانگین هارمونیک با در نظر گرفتن Average Coverage با عنوان C و Average Purity با عنوان P به صورت تعریف می‌شود [۲۹]:

$$Harmonic\_Mean = \frac{2 * P * C}{P + C} \quad (10)$$

## الف) ارزیابی معیارهای Average Purity و Average Coverage و Harmonic Mean

در این بخش از پیاده‌سازی، داده‌های ناوگان توزیع دارو با استفاده از الگوریتم SEGA بخش‌بندی شده و با توجه به معیار MDL (رابطه ۳) بهترین سگمنت‌ها مشخص می‌شوند و نقاط نماینده در حافظه ذخیره می‌گردند. جهت ارزیابی الگوریتم، از روابط ۸، ۹ و ۱۰ بهره گرفته‌ایم. این نتایج در جدول ۱ نشان داده شده است.

جدول ۱: تنظیمات پارامترهای ورودی الگوریتم SEGA (نرخ تقاطع و نرخ پرش) و نتایج معیارهای ارزیابی الگوریتم SEGA بر روی مجموعه داده ناوگان توزیع دارو

Harmonic mean	Average Coverage	Average Purity	Mutation	Cross over	تعداد داده‌های ورودی
۰.۹۱	۰.۸۹	۰.۹۴	۰.۷	۰.۵	۴۰۰۰

همان‌طور که از جدول ۱ مشخص است، الگوریتم معرفی شده، بخش‌بندی داده‌های ناوگان توزیع دارو - با تعداد ۴۰۰۰ ورودی - را به درستی و با دقت حدود ۹۱ درصد انجام داده است. این معیارها به ترتیب نشان می‌دهد که دقت و صحت عملکرد الگوریتم پیشنهادی در مورد داده‌های واقعی - داده‌های ناوگان توزیع دارویی - به دلیل اینکه این داده‌ها دارای برچسب بوده و نتایج مشخص است، به درستی عمل کرده است. همچنین با توجه به اینکه دو شاخص Average Coverage و Average Purity به صورت متعادل<sup>۱</sup> بوده و افزایش یکی با کاهش دیگری مواجه است، میزان میانگین هارمونیک هرچقدر به ۱ نزدیک باشد بهتر خواهد بود؛ که در این پیاده‌سازی حدود ۰.۹۱ درصد می‌باشد.

## ب) مقایسه نتایج الگوریتم SEGA در مقابل الگوریتم ژنتیک مرسوم [۳۱]

در فاز بعدی، الگوریتم معرفی شده در این مقاله (SEGA) را با نتایج الگوریتم ژنتیک مرسوم بر روی داده‌های ناوگان توزیع دارو، داده طوفان آتلانتیک و داده GeoLife پیاده‌سازی کرده و نتایج را در جدول ۲ نشان داده‌ایم. قابل ذکر است که مقادیر صفات<sup>۲</sup> مورد استفاده در داده‌های فوق با توجه به رابطه ۱۱ نرمال‌سازی شده‌اند تا مقادیر آن‌ها بین "صفر" تا "یک" قرار گیرد [۳۲].

<sup>1</sup> Orthogonal

<sup>2</sup> Feature

$$normalize(value) = \frac{value - \min}{\max - \min} \quad (11)$$

نتایج پیاده‌سازی الگوریتم SEGA و الگوریتم ژنتیک مرسوم بر روی داده‌های مذکور در جدول ۲ با توجه به معیارهای زمان، میانگین هارمونیک، تعداد نقاطی که بعد از بخش‌بندی به دست می‌آید و باید در پایگاه داده ذخیره شود، نشان داده شده است.

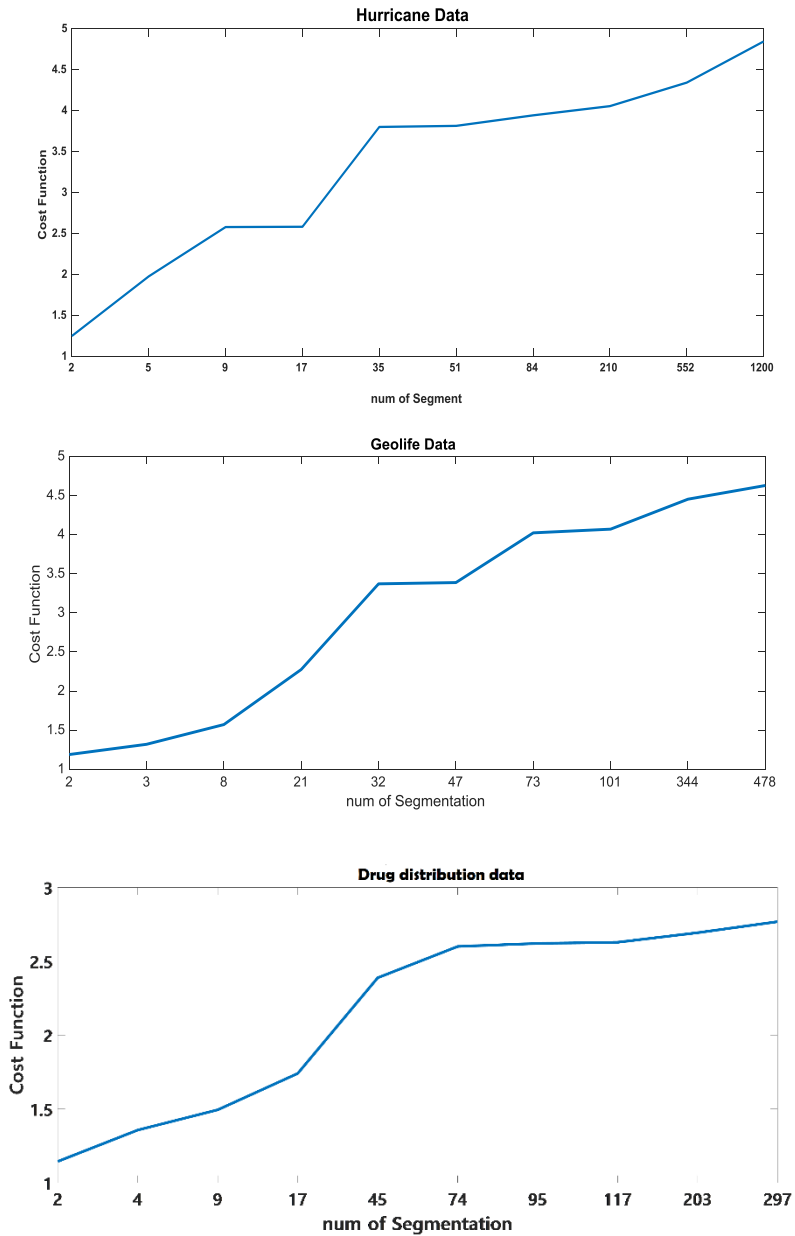
جدول ۲: نتایج پیاده‌سازی الگوریتم SEGA و الگوریتم ژنتیک مرسوم بر روی داده‌های مذکور با توجه به معیارهای زمان، میانگین هارمونیک، تعداد نقاطی که بعد از بخش‌بندی به دست می‌آید و باید در پایگاه داده ذخیره شود

میانگین هارمونیک در الگوریتم ژنتیک مرسوم	میانگین هارمونیک در الگوریتم SEGA	زمان اجرای الگوریتم ژنتیک مرسوم (ثانیه)	زمان اجرای الگوریتم SEGA (ثانیه)	تعداد نقاط نماینده جهت ذخیره در پایگاه داده بعد از اعمال الگوریتم مرسوم	تعداد نقاط نماینده جهت ذخیره در پایگاه داده بعد از اعمال الگوریتم SEGA	تعداد داده‌ها	داده‌های ورودی
۰.۸۳	۰.۸۸	۴۰۹۸	۹۲۳	۲۱۱۰۹	۱۵۴۴۸	۳۸۸۹۲	طوفان آتلانتیک
۰.۸۹	۰.۹۰	۳۴۶۷	۴۱۱	۱۱۴۵۱	۶۸۱۵	۱۷۶۲۱	GeoLife
۰.۸۸	۰.۹۲	۱۷۱۳۱	۸۱۷۵	۳۰۰۱۲۱	۱۲۱۷۷۱	۴۵۰۰۰۰	توزیع دارو

همان‌طور که از نتایج جدول ۲ مشخص است، اعمال الگوریتم ژنتیک مرسوم که به صورت عادی از عملگر برش و پرش استفاده کرده و هیچ تغییری در عملیات آن ندارد، باعث افزایش زمان بخش‌بندی خواهد شد و در مقابل تعداد نقاط نماینده جهت ذخیره در پایگاه داده نیز زیاد خواهد بود. اما استفاده از الگوریتم SEGA به دلیل اینکه هیوریستیک‌ی به عملگر پرش اضافه می‌کند و ترتیب استفاده از عملگر برش و پرش در این الگوریتم تغییر یافته است سبب کاهش زمان بخش‌بندی خواهد شد و همین امر سبب می‌شود که تعداد نقاط نماینده قابل ذخیره در پایگاه داده - که حاوی اطلاعات اصلی تمام نقاط داده‌ها می‌باشد- بهینه گردد. این بهینگی از میزان میانگین هارمونیک برای هر یک از داده‌ها و در صورت استفاده از هر الگوریتم در دو ستون آخر جدول ۲ به وضوح مشخص است.

### ج) رابطه میان مقدار تابع هزینه و تعداد سگمنت‌های به دست آمده

بخش زیر رابطه بین مقدار تابع هزینه (رابطه ۳) و تعداد بخش‌های به دست آمده پس از اجرای الگوریتم SEGA برای سه داده را نشان می‌دهد: طوفان آتلانتیک، GeoLife و مجموعه داده‌های ناوگان توزیع دارو.



شکل ۶: رابطه بین تابع هزینه و تعداد سگمنتها، خروجی روش SEGA برای الف) داده های طوفان آتلانتیک ب) داده های Geolife ج) داده های ناوگان توزیع دارو

شکل ۶ رابطه بین تابع هزینه (رابطه ۳) و تعداد سگمنت‌ها را پس از اجرای الگوریتم SEGA نشان می‌دهد. مشخص است که وقتی تعداد سگمنت‌ها کم است، مقدار تابع هزینه (رابطه ۳) نیز کم خواهد بود. ما به دنبال کاهش (بهینه‌سازی) تابع هزینه هستیم و شکل ۶ نشان می‌دهد که مسیر<sup>۱</sup> با کمترین بخش دارای کمترین تابع هزینه است. با توجه به دو شاخص برای ارزیابی راه‌حل (Average Purity و Average Coverage)، زمانی که تعداد سگمنت‌ها کم باشد، این دو شاخص مقدار مناسبی را نشان نمی‌دهند، اما اگر راه‌حلی با تابع هزینه متوسط در نظر بگیریم، میزان شاخص‌ها برای آن مدل بخش‌بندی مناسب و صحیح خواهد بود.

## ۶- نتیجه‌گیری

در سیستم‌های لجستیک دارویی، حفظ امنیت و کیفیت زنجیره تأمین داروها، به‌ویژه در شرایط بحرانی، از اهمیت ویژه‌ای برخوردار است. اختلالات در حمل‌ونقل و تأخیرهای ناشی از ترافیک، سوانح حمل‌ونقل و خطاهای انسانی می‌تواند تأثیرات مخربی بر زنجیره تأمین داروها داشته باشد. به‌علاوه، نگهداری و حمل صحیح داروها و مواد بیولوژیکی حساس به دما نیازمند سیستم‌های پیشرفته نظارتی و کنترلی است تا از قاچاق دارو، فساد و از بین رفتن این مواد ارزشمند جلوگیری شود. در این راستا، استفاده از الگوریتم‌های بهینه‌سازی مانند SEGA نقش حیاتی در حفظ و مدیریت این فرآیندها ایفا می‌کند.

نتایج این پژوهش به وضوح نشان می‌دهد که بهره‌گیری از الگوریتم‌های ژنتیک می‌تواند به نحو قابل توجهی در مدیریت و بهینه‌سازی داده‌های حجیم مربوط به سامانه‌های توزیع دارویی نقش آفرینی کند. این الگوریتم‌ها با استفاده از رویکردهای پیشرفته و نوآورانه قادر به کاهش حجم داده‌های ذخیره‌شده، افزایش سرعت تحلیل و بهبود کارایی سامانه‌های توزیع دارویی هستند، به طوری که ضمن حفظ جامعیت و صحت داده‌ها، اطلاعات حیاتی و حساس خط سیر به صورت کارآمدتر پردازش و نگهداری می‌شوند.

در نهایت، استفاده از الگوریتم‌های SEGA در سامانه‌های توزیع دارویی نه تنها به بهبود کارایی و عملکرد این سیستم‌ها منجر می‌شود بلکه به عنوان یک رویکرد نوآورانه در پدافند غیرعامل، می‌تواند به حفاظت و امنیت بیشتر در زنجیره تأمین داروها و مواد بیولوژیکی

<sup>1</sup> Trajectory



حساس کمک کند. این الگوریتم‌ها با فراهم آوردن امکان نظارت پیوسته و بهینه‌سازی فرآیندهای حمل‌ونقل، نقش مؤثری در مدیریت چالش‌های لجستیک دارویی ایفا می‌کنند و به دولت‌ها و نهادهای مرتبط این امکان را می‌دهند که در شرایط بحرانی، به نحو مطلوب‌تری از منابع دارویی کشور حفاظت و بهره‌برداری نمایند.

## ۷- تشکر و قدردانی

با تشکر از دانشگاه پیام نور مرکز شیراز، که امکانات آزمایشگاهی را فراهم نموده و در انجام این پروژه نهایت همکاری را داشته‌اند

## ۸- تعارض منافع

نویسنده(گان) اعلام می‌دارند که در مورد انتشار این مقاله تضاد منافع وجود ندارد. علاوه بر این، موضوعات اخلاقی شامل سرقت ادبی، رضایت آگاهانه، سوء رفتار، جعل داده‌ها، انتشار و ارسال مجدد و مکرر توسط نویسندگان رعایت شده است.

## ۹- دسترسی آزاد

این نشریه دارای دسترسی باز است و اجازه اشتراک (تکثیر و بازآرایی محتوا به هر شکل) و انطباق (بازترکیب، تغییر شکل و بازسازی بر اساس محتوا) را می‌دهد.

## منابع

۱. Krempf, G., et al., *Open challenges for data stream mining research*. SIGKDD Explor. Newsl., 2014. 16(1): p. 1-10.
۲. Esheiba, L., H. Mokhtar, and M. E. El-Sharkawi, *Spatio-Temporal Queries for Moving Objects Data Warehousing*. Vol. 5. 2013.
۳. Gali, Z., et al., *Distributed processing of big mobility data as spatio-temporal data streams*. Geoinformatica, 2017. 21(2): p. 263-291.
۴. Feng, Z. and Y. Zhu, *A Survey on Trajectory Data Mining: Techniques and Applications*. IEEE Access, 2016. 4: p. 2056-2067.
۵. Damiani, M.L. and F. Hachem, *Segmentation techniques for the summarization of individual mobility data*. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2017. 7(6): p. n/a-n/a.
۶. Madurika, H.K.G.M., *Gis Based Analysis For Suitability Location Finding In The Residential Development Areas Of Greater Matara Region*. 2015.
۷. Zheng, Y., Zhou, Xiaofang, *Computing with Spatial Trajectories*. 2011: New York: Springer.

- .۸ Vose, M.D., *The Simple Genetic Algorithm: Foundations and Theory*. 1998 : MIT Press. 251.
- .۹ Mitchell, M., *An Introduction to Genetic Algorithms*. 1998: MIT Press. 209.
- .۱۰ Panagiotakis, C., et al., *Segmentation and Sampling of Moving Object Trajectories Based on Representativeness*. IEEE Transactions on Knowledge and Data Engineering, 2012. 24(7): p. 1328-1343.
- .۱۱ Resheff, Y.S. *Online trajectory segmentation and summary with applications to visualization and retrieval*. in *2016 IEEE International Conference on Big Data (Big Data)*. 2016.
- .۱۲ Palma, A.T., et al., *A clustering-based approach for discovering interesting places in trajectories*, in *Proceedings of the 2008 ACM symposium on Applied computing*. 2008, ACM: Fortaleza, Ceara, Brazil. p. 863-868.
- .۱۳ Zheng, Y., *Trajectory Data Mining: An Overview*. ACM Trans. Intell. Syst. Technol., 2015. 6(3): p. 1-41.
- .۱۴ Junior, A.S., et al. *A Semi-Supervised Approach for the Semantic Segmentation of Trajectories*. in *2018 19th IEEE International Conference on Mobile Data Management (MDM)*. 2018.
- .۱۵ Soares Júnior, A., et al., *GRASP-UTS: an algorithm for unsupervised trajectory segmentation*. International Journal of Geographical Information Science, 2015. 29(1): p. 46-68.
- .۱۶ Etemad, M., et al., *SWS: an unsupervised trajectory segmentation algorithm based on change detection with interpolation kernels*. GeoInformatica, 2020.
- .۱۷ Grnwald, P.D., I.J. Myung, and M.A. Pitt, *Advances in Minimum Description Length: Theory and Applications (Neural Information Processing)*. 2005: The MIT Press.
- .۱۸ Talbi, E.-G., *Metaheuristics: From Design to Implementation*. 2009: Wiley Publishing. 593.
- .۱۹ Feo, T.A. and M.G.C. Resende, *Greedy Randomized Adaptive Search Procedures*. Journal of Global Optimization, 1995. 6(2): p. 109-133.
- .۲۰ Karbowska-Chilińska, J. and P. Zabielski. *A Genetic Algorithm vs. Local Search Methods for Solving the Orienteering Problem in Large Networks*. 2013. Berlin, Heidelberg: Springer Berlin Heidelberg.
- .۲۱ Roy, W.B.P.B.V., *Approximate Dynamic Programming for High-Dimensional Resource Allocation Problems*, in *Handbook of Learning and Approximate Dynamic Programming*. 2012.
- .۲۲ Oliveto, P.S., J. He, and X. Yao, *Time complexity of evolutionary algorithms for combinatorial optimization: A decade of results*. International Journal of Automation and Computing, 2007. 4(3): p. 281-293.
- .۲۳ Maulik ,U. and S. Bandyopadhyay, *Genetic algorithm-based clustering technique*. Pattern Recognition, 2000. 33(9): p. 1455-1465.
- .۲۴ Holland, J.H., *Genetic Algorithms*. Scientific American, 1992. 267(1): p. 66-73.
- .۲۵ Piszcz, A. and T. Soule, *Genetic programming: optimal population sizes for varying complexity problems*, in *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. 2006, ACM: Seattle, Washington, USA. p. 953-954.

- .۲۶ Pelikan, M., D.E. Goldberg, and E. Cantu-Paz, *Bayesian optimization algorithm, population sizing, and time to convergence*, in *Proceedings of the 2nd Annual Conference on Genetic and Evolutionary Computation*. 2000, Morgan Kaufmann Publishers Inc.: Las Vegas, Nevada. p. 275-282.
- .۲۷ Lobo, F.G. and D.E. Goldberg, *The parameter-less genetic algorithm in practice*. Inf. Sci., 2004. 167(1-4): p. 217-232.
- .۲۸ Lobo, F.G., et al., *A review of adaptive population sizing schemes in genetic algorithms*, in *Proceedings of the 7th annual workshop on Genetic and evolutionary computation*. 2005, ACM: Washington, D.C. p. 228-234.
- .۲۹ Etemad, M., et al. *A Trajectory Segmentation Algorithm Based on Interpolation-based Change Detection Strategies*. in *EDBT/ICDT Workshops*. 2019.
- .۳۰ Chicco, D. and G. Jurman, *The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation*. BMC genomics, 2020. 21(1): p. 6-6.
- .۳۱ Mirjalili, S., *Genetic Algorithm*, in *Evolutionary Algorithms and Neural Networks: Theory and Applications*, S. Mirjalili, Editor. 2019, Springer International Publishing: Cham. p. 43-55.
- .۳۲ Adeyemo, A., *Effects of Normalization Techniques on Logistic Regression in Data Science*. Proceedings of the Conference on Information Systems Applied Research, 2018. 11.