

زمان بندی پروژه های ساخت با منابع محدود (MRCPSP) با در نظر گرفتن عدم قطعیت در مدت فعالیت ها و تأخیرات

علی کاتبی^{۱*}، فریبرز الماسیان^۲، پیمان همای^۳

۱- استادیار، دانشگاه خوارزمی، عضو هیئت علمی گروه مهندسی عمران، گرایش مهندسی و مدیریت ساخت

۲- دانشجوی مقطع کارشناسی ارشد، دانشگاه خوارزمی، گروه مهندسی عمران، گرایش مهندسی و مدیریت ساخت

۳- استادیار، دانشگاه خوارزمی، عضو هیئت علمی گروه مهندسی عمران، گرایش مهندسی سازه

چکیده

انتخاب روش های ساخت، نحوه تخصیص منابع اهمیت زیادی در کنترل پروژه دارد. مسئله زمان بندی پروژه با منابع محدود در چندین حالت اجرایی (MRCPSP) موضوعی مطرح در مدیریت پروژه است. توسعه مدل فوق برای پروژه های ساخت، به دلیل وجود فضای غیرقطعی، امری مهم به نظر می رسد. برای نمایش عدم قطعیت مدت زمان انجام فعالیت ها و تأخیر بین آن ها، از منطق فازی استفاده شده است. در این مقاله به بررسی این مسئله پرداخته و با ارائه یک الگوریتم هوشمند ترکیب شده از مجموعه های فازی و الگوریتم ژنتیک (GA)، زمان بندی صورت گرفته است. مسئله MRCPSP فازی را می توان زمان بندی مجموعه ای از فعالیت ها با هدف یافتن یک روش اجرایی و توالی زمانی مناسب برای انجام فعالیت ها در نظر گرفت؛ به طوری که محدودیت های منابع (منابع تجدید پذیر و منابع تجدید ناپذیر) و نیز محدودیت های پیش نیازی به طور هم زمان ارضا گردند و زمان تکمیل پروژه کمینه شود. در گام اول مسئله مذکور مدل سازی ریاضی می شود و سپس، اقدام به کدنویسی مدل مسئله با استفاده از الگوریتم فراابتکاری GA در نرم افزار Matlab و حل مدل ریاضی مسئله می شود. نتایج حاصل از پیاده سازی این الگوریتم بر روی مسائل استاندارد سایت PSPLIB، در مقایسه با نرم افزار GAMS، حاکی از عملکرد موفق الگوریتم GA تلفیق شده با مجموعه های فازی است. رویکرد بکار گرفته شده در این پژوهش می تواند به سادگی قابل استفاده توسط مدیران و برنامه ریزان پروژه باشد. این امر موجب جلوگیری از خطاهای انسانی مسئول کنترل پروژه در تسطیح منابع می شود و راهی به سوی زمان بندی بهینه پروژه است.

کلمات کلیدی: زمان بندی فازی پروژه ساخت با محدودیت منابع، فعالیت های چند حالتی، تأخیرات فازی، کمینه کردن زمان تکمیل پروژه، الگوریتم فراابتکاری GA.

| شناسه دیجیتال: | | سابقه مقاله: | | | | |
|----------------|--|--------------|------------------|------------|-----------------|------------|
| doi: | 10.22065/jsce.2018.108681.1396 | چاپ | انتشار آنلاین | پذیرش | بازنگری | دریافت |
| | https://10.22065/jsce.2018.108681.1396 | ۱۳۹۸/۱۰/۰۱ | ۱۳۹۷/۰۳/۱۰ | ۱۳۹۷/۰۳/۱۰ | ۱۳۹۶/۰۱/۲۱ | ۱۳۹۶/۰۹/۱۰ |
| | | | علی کاتبی | | *نویسنده مسئول: | |
| | | | katebi@khu.ac.ir | | پست الکترونیکی: | |

Multi Mode Resource Constraint Construction Project Scheduling Problem (MRCPSP) Considering the Uncertainty in the Activities Duration and Delays

Ali Katebi^{1*}, Fariborz Almassian², Peyman Homami³

1 - Assistant Professor, Kharazmi University, Faculty Member of Civil Engineering Department, Engineering Structure and Construction Management, Tehran, Iran

2 - Graduate Student, Kharazmi University, Civil Engineering Department, Engineering Structure and Construction Management, Tehran, Iran

3 - Assistant Professor, Kharazmi University, Faculty Member of Civil Engineering Department, Structural Engineering Structure, Tehran, Iran

ABSTRACT

Choosing construction methods and how resources are allocated are important in the project control. The multi-mode resource constrained project scheduling problem (MRCPSP) is a significant subject in project management. The development of the above model for construction projects is an important issue because of uncertainty. Fuzzy logic has been used to display the uncertainty in the duration of activities and the delay between them. This paper examines this problem and schedules project with providing an intelligent algorithm combining fuzzy sets and genetic algorithms (GAs). The Fuzzy MRCPSP problem can be considered as the scheduling of a set of activities with the aim of finding an activity operation mode and the activity operation priority; so that the resource constraints (renewable resources and non-renewable resources) as well as the precedence constraints are met simultaneously and the time for completion of the project is minimal. In the first step, the above problem is modeled mathematically, and then the model is coded using GA-based algorithm in Matlab software and finally the model is solved. The results of the implementation of this algorithm on the standard instances of the PSPLIB site, in comparison with the GAMS software, indicate the successful performance of the combined GA algorithm with fuzzy sets. The approach used in this research can be easily used by project managers. This prevents human errors caused by people who are responsible for controlling the project at resource leveling phase and is a way to the optimized project scheduling.

ARTICLE INFO

Receive Date: 06 January 2018

Revise Date: 17 March 2018

Accept Date: 28 April 2018

Keywords:

Resource constraints construction projects fuzzy scheduling, fuzzy delays, multi-mode activities, minimization of project completion time, GA algorithm.

All rights reserved to Iranian Society of Structural Engineering.

doi: 10.22065/jsce.2018.108681.1396

*Corresponding author: Ali Katebi

Email address katebi@khu.ac.ir

۱- مقدمه

بحث مدیریت پروژه مربوط به برنامه‌ریزی، کنترل و مدیریت زمان، منابع و هزینه فعالیت‌های یک پروژه است که در میان آنها زمان از اهمیت بیشتری برخوردار است [۱]. برنامه‌ریزی و زمانبندی مناسب به منظور برآورد صحیح از زمان و هزینه‌های انجام پروژه و میزان منابع مورد نیاز در یک پروژه عمرانی، از جایگاه ویژه‌ای برخوردار است [۲]. در اواخر دهه ۱۹۶۰، مدل‌های زمانبندی اولیه با در نظر گرفتن محدودیت‌های منابع توسعه یافتند. از دیدگاه محاسباتی، مدل‌های زمانبندی پروژه با منابع محدود جز مسائل NP-hard^۱ محسوب می‌شود [۳]. به طوری که زمان محاسبات پیدا کردن جواب بهینه مسئله با زیاد شدن تعداد متغیرها و محدودیت‌ها، به صورت نمایی افزایش می‌یابد [۳، ۴]. اهمیت مسئله زمانبندی پروژه را می‌توان از ۳ منظر مورد بررسی قرار داد: ۱- با توجه به ابعاد بزرگ پروژه‌ها، ارائه الگوریتمی که بهبودی حتی جزئی در جواب ایجاد کند، در عمل موجب صرفه‌جویی‌های هنگفت در زمان و هزینه و کاهش ریسک پروژه‌ها و افزایش حاشیه سود مجریان خواهد گردید. ۲- آشکار است که اگر پروژه سر موعده مقرر به اتمام نرسد، هزینه‌های اضافی زیادی از جمله الف) هزینه تاخیر در بهره‌برداری یا سود از دست رفته ب) هزینه ناشی از گران شدن منابع ج) هزینه تمدید مجوزها د) افزایش هزینه منابع کاری به پروژه اعمال خواهد شد. ۳- از دیدگاه نظری نیز زمانبندی پروژه یک زمینه تحقیقاتی بسیار جذاب برای محققین می‌باشد. ضرورت این تحقیق با توجه به افزایش چشمگیر کاربرد اینگونه مسائل سخت در حوزه مدیریت و تصمیم‌گیری پروژه‌ها مشخص می‌گردد [۱، ۵-۷]. مدل‌هایی که برای تهیه برنامه زمانبندی پروژه توسعه داده شده‌اند، در ادبیات به زمانبندی پروژه با محدودیت منابع (RCPSP^۲) مشهور هستند. مسئله کلاسیک RCPSP شامل زمانبندی یک پروژه تک-حالت به منظور حداقل کردن زمان تکمیل اجرای پروژه با توجه به روابط پیشنیازی^۳ و محدودیت‌های منابع است. زمانبندی پروژه‌های با منابع محدود شامل فعالیت‌هایی تحت چندین روش اجرایی (MRCPSP^۴) فازی یک نوع توسعه یافته‌ای از مسائل استاندارد (RCPSP) هستند، که هر فعالیت می‌تواند در یکی از چندین روش اجرایی، انجام شود [۶، ۸-۱۰].

۱-۱- پیشینه پژوهش

در چند دهه گذشته، مطالعات صورت گرفته در زمینه زمانبندی پروژه تحت محدودیت منابع، نشان‌دهنده اهمیت زیاد پژوهش‌های این حوزه در دنیای امروزی است. در هر یک از مقالات، پژوهشگران مسائل متفاوتی را از نظر تابع هدف و محدودیت‌ها، مورد بررسی قرار دادند و یا به دنبال ارائه یک الگوریتم فراابتکاری کارا با هدف تسریع در حل مسئله بودند و هم‌اکنون نیز تحقیقات زیادی برای ارائه الگوریتم‌های کارا تر صورت می‌گیرد.

اولین بار پرید منطق تئوری فازی را در زمانبندی پروژه بکار گرفت [۱۱]. در سال ۲۰۰۰ هاپکه و همکاران در مقاله‌ای با فازی در نظر گرفتن مدت زمان انجام فعالیت‌ها، مسئله زمانبندی پروژه با محدودیت منابع را مورد مطالعه قرار دادند. سپس مدل ریاضی مسئله را با استفاده از الگوریتم بازپخت شبیه‌سازی شده (SA^۵)، حل نمودند [۱۲]. در سال ۲۰۰۲ لیو و هانگ شرایط عدم قطعیت در مدت زمان انجام فعالیت‌ها را در مسئله زمانبندی پروژه با محدودیت منابع در نظر گرفتند و سپس با استفاده از الگوریتم ژنتیک به حل مدل پرداختند [۱۳]. در سال ۲۰۰۵ ژانگ و همکاران به واسطه استفاده از الگوریتم فراابتکاری ازدحام ذرات، کمینه کردن زمان بهینه پروژه تحت محدودیت منابع تجدیدشدنی مورد بررسی قرار دادند و بهترین ترتیب ممکن از انجام فعالیت‌ها را با هدف حداقل نمودن زمان تکمیل پروژه بدست آوردند [۱۴]. در سال ۲۰۰۹ مندز و همکاران الگوریتم ژنتیکی برای حل مسئله زمانبندی پروژه با منابع محدود ارائه کردند ابتکار آنها در بکارگیری الگوریتم ژنتیک بر پایه کلید تصادفی استوار^۶ بود. در الگوریتم آنها از یک قانون اولویت ابتکاری^۷ نیز استفاده شده است تا کارایی الگوریتم را

^۱ Non-Deterministic Polynomial-Time Hard

^۲ Resource constrained Project Scheduling Problem

^۳ Precedence Relations

^۴ Multi-Mode Resource Constrained Project Scheduling Problem

^۵ Simulation Annealing

^۶ Random Keys

بهبود بخشند [۱۵]. در سال ۲۰۱۱ هوانگ و همکاران در مقاله‌ای با فازی در نظر گرفتن مدت زمان انجام فعالیت‌ها، مسئله زمانبندی پروژه با محدودیت منابع (RCPSP) را مورد مطالعه قرار دادند. سپس مدل ریاضی مسئله را با استفاده از الگوریتم ژنتیک، حل نمودند [۱۶]. در سال ۲۰۱۳ مسمودی و هیت در مقاله‌ای با فازی در نظر گرفتن مدت زمان انجام فعالیت‌ها، مسئله زمانبندی پروژه با محدودیت منابع (RCPSP) را مورد مطالعه قرار دادند. آنها با استفاده از الگوریتم فراابتکاری، به حل مدل پرداختند و بهترین ترتیب ممکن اجرای فعالیت‌ها را بدست آوردند [۱۷]. در سال ۲۰۱۴ افشارنجفی و همکاران در مقاله‌ای، با استفاده از الگوریتم فراابتکاری ژنتیک، مسئله زمانبندی پروژه را با هدف کمینه کردن زمان اتمام پروژه، حل نمودند. آنها در مدلشان، امکان وقفه در اجرای فعالیت‌ها را مجاز دانستند و زمانی را جهت آماده‌سازی فعالیت‌ها پس از هر وقفه در نظر گرفتند. به منظور بررسی اعتبار و عملکرد الگوریتم پیشنهاد شده، مثال‌های مختلفی را در سایت PSPLIB حل نمودند که نتایج آماری نشان داد، الگوریتم پیشنهاد شده، کارایی موثری در حل این دسته از مسائل دارد [۱۸]. در سال ۲۰۱۵ کنیازا و همکاران در مقاله‌ای به بررسی مساله زمانبندی پروژه تک حالت با منابع محدود پرداختند. آنها مدت زمان انجام فعالیت‌ها را غیرقطعی در نظر گرفتند و به واسطه منطق فازی، عدم قطعیت در مدت زمان انجام فعالیت‌ها را توصیف نمودند. در نهایت آنها با استفاده از یک الگوریتم ابتکاری، اقدام به حل مدل مسئله با هدف کمینه کردن زمان اتمام پروژه، پرداختند [۱۹]. در سال ۲۰۱۶ جوی و همکاران پس از بررسی مسئله زمانبندی با محدودیت منابع را با در نظر گرفتن داده‌های قطعی و سطح متغیر در منابع در دسترس، رویکردی مبتنی بر الگوریتم PSO برای حل مدل مسئله پیشنهاد دادند [۲۰]. در سال ۲۰۱۷ گیران و همکاران به واسطه استفاده از الگوریتم فراابتکاری جستجوی هماهنگ (HSA^۱)، کمینه کردن زمان بهینه پروژه‌های ساخت تحت محدودیت منابع (RCPSP) را مورد بررسی قرار دادند و بهترین ترتیب ممکن از انجام فعالیت‌ها را با هدف حداقل نمودن زمان تکمیل پروژه بدست آوردند که نتایج حاکی از عملکرد مناسب الگوریتم پیشنهادی آنها بود [۲].

روند کلی انجام پژوهش بدین صورت است که در بخش ۲، مسئله مورد نظر تعریف شده و مدل‌سازی ریاضی می‌گردد. در ادامه مراحل الگوریتم ژنتیک تلفیق شده با مجموعه‌های فازی و نحوه پیاده‌سازی الگوریتم مذکور جهت حل مسئله بیان می‌گردد. هدف از حل مسئله استخراج بهترین روش اجرایی هر فعالیت و توالی بهینه ممکن برای انجام فعالیت‌ها است به گونه‌ای که زمان تکمیل پروژه حداقل گردد. همچنین الگوریتمی جهت محاسبه جواب موجه ابتدایی برای افزایش سرعت هر یک از الگوریتم‌ها، با استفاده از حذف فضای غیرموجه جستجو، ارائه شده است. در بخش ۳، با استفاده از الگوریتم پیشنهاد شده چند نمونه از مسائل استاندارد سایت PSPLIB حل می‌گردد. در انتها در بخش ۴، عملکرد الگوریتم پیشنهاد شده با نرم‌افزار GAMS که یک نرم‌افزار حل دقیق مسائل بهینه‌سازی است، مورد سنجش قرار می‌گیرد.

۲- روش تحقیق و ابزارها

پژوهش حاضر از نظر هدف از نوع توسعه‌ای و کاربردی، است. داده‌های مورد استفاده در تحلیل مسائل، از سایت PSPLIB استخراج شده‌است. الگوریتم GA با نرم‌افزار Matlab Ver 2013a برنامه‌نویسی شده و با لپ‌تاپ با مشخصات CPU Corei7-4720 HQ- 2.6 GHz و رم 8 GB اجرا گشته است. جهت تنظیم پارامترهای الگوریتم GA از روش تاگوچی با کمک نرم‌افزار Minitab Ver-18.1 استفاده شده‌است. برای ارزیابی عملکرد الگوریتم مذکور مدل مسئله با استفاده از نرم‌افزار GAMS Ver-24.5.6 برنامه‌نویسی شده‌است. قبل از مدل‌سازی و حل مسئله زمانبندی پروژه با محدودیت منابع ابتدا لازم است مفاهیم لازم جهت انجام تحقیق بیان شود.

۲-۱- عدم قطعیت

^۱ Heuristic Priority Rule

^۲ Harmony Search Algorithm

عدم قطعیت در پروژه موجب کاهش دقت در تخمین مناسب اهداف شده و از کارایی پروژه‌ها می‌کاهد. بنابراین، نیاز به شناخت و مدیریت ریسک در پروژه، کاملاً روشن است. پروژه‌های ساخت دارای ریسک‌های بالقوه در عملیات ساخت هستند، که می‌تواند اهداف برنامه‌ریزی پروژه را تحت تاثیر قرار دهد و پروژه را با تاخیر روبرو سازد. خطاهای انسانی، شرایط جوی، کمبود منابع و تجهیزات، عدم مهارت کافی کارگران، شرایط کاری نامناسب، تعارض، حوادث طبیعی و غیر تکراری بودن یا عدم مواجهه مدیران پروژه با فعالیت‌های مشابه در تجربیات قبلی، از جمله عواملی هستند که مدت زمان انجام فعالیت‌ها را با عدم قطعیت مواجه می‌سازند. وجود عدم قطعیت در پروژه، پیچیدگی مسئله زمانبندی پروژه را افزایش می‌دهد [۱۷]. روش‌های متعددی برای بکارگیری عدم قطعیت‌ها در تخمین مدت زمان پروژه وجود دارد که مهم‌ترین و پرکاربردترین آنها روش‌های احتمالی مانند PERT و GERT هستند. این روش‌ها از توزیع‌های احتمالی از قبیل توزیع بتا یا توزیع نرمال برای تخمین مدت زمان فعالیت‌های پروژه استفاده می‌کنند. برای استفاده از توزیع‌های احتمالی به نمونه‌های تصادفی، تکرارپذیری و استنتاج آماری نیاز است که این امر به علت جدید بودن فعالیت‌های پروژه و یا کم سابقه بودن آنها به خوبی امکان‌پذیر نیست. علاوه بر آن هنگام استفاده از یک توزیع احتمالی، پارامترهای زمانبندی پروژه به رفتار توزیع وابسته می‌شود و این موضوع باعث ایجاد محدودیت در زمانبندی پروژه می‌گردد. لذا استفاده از منطق احتمالی، گزینه مناسبی جهت برخورد با عدم قطعیت‌های موجود در پروژه نیست. بنابراین از رویکرد فازی به‌عنوان یکی از راهکارهای اساسی مواجهه با عدم قطعیت در پروژه، استفاده شده است [۲۱-۲۵].

برای تخمین مدت زمان انجام فعالیت‌ها و تاخیر مجاز بین آنها، عبارت‌هایی چون، "احتمالاً"، "تقریباً"، "کم‌وبیش" و "حدوداً" در جملات کارشناسان کنترل پروژه یافت می‌شود. این عبارت‌ها به‌وضوح نوعی از عدم قطعیت را نمایش می‌دهند که طبیعتاً منجر به محدوده‌ای از مقادیر مختلف و نه یک مقدار خاص می‌شود. به عنوان مثال جمله "فعالیت A به صورت معمول و به احتمال خیلی زیاد چهارده روز به طول می‌انجامد، اما با توجه به شرایط که غیر قابل کنترل هستند ممکن است در خوشبینانه‌ترین حالت هشت روز و در بدبینانه‌ترین حالت بیست روز طول بکشد." حال با استفاده از مفاهیم نظریه فازی و ارتباط آن با متغیرهای کلامی سعی بر آن است تا نظرات کارشناسان کنترل پروژه، در قالب مجموعه‌های فازی تعریف شوند. در همین راستا با توجه به جملات استفاده شده در گفتار کارشناسان کنترل پروژه و انطباق آن با اعداد فازی مثلثی، این اعداد به عنوان یک مجموعه فازی که مبین نظرات کارشناسان است، تعریف می‌گردد. در مورد فعالیت A، می‌توان مدت زمان انجام آن را در قالب عدد فازی مثلثی $t(A) = (8, 14, 20)$ بیان نمود. بنابراین در حالت کلی مدت زمان انجام فعالیت‌ها و تاخیر بین آنها با سه زمان "خوشبینانه، واقع‌بینانه و بدبینانه" تخمین زده و به‌صورت یک عدد فازی مثلثی در نظر گرفته می‌شود.

۲-۲- منطق فازی^۹

منطق فازی توسط دکتر لطفی‌زاده، استاد دانشگاه کالیفورنیا در برکلی، در سال ۱۹۶۵ ارائه گردید. این نظریه با در نظر گرفتن عدم قطعیت در پارامترهای تصمیم‌گیری و استفاده از مدل‌های ذهنی کارشناسان، نگرشی نو را در علوم مختلف از جمله زمانبندی پروژه ایجاد نمود. منطق فازی در واقع شکل تکامل‌یافته و عمومی شده منطق کلاسیک است. در منطق کلاسیک که منطق دو ارزشی است، هر گزاره می‌تواند درست یا نادرست باشد. در حالی که در منطق فازی که منطق چند ارزشی است، ارزش درستی هر گزاره می‌تواند عددی بین صفر و یک باشد. لذا قضاوت تقریبی و نادقیق با بکارگیری منطق فازی ممکن می‌شود [۲۶، ۲۷].

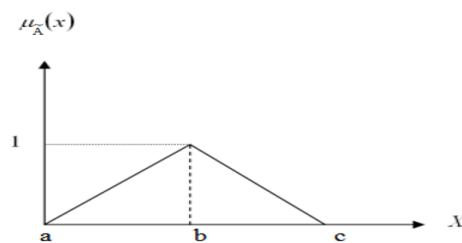
مجموعه فازی A در فضای جهانی U را می‌توان به‌صورت زوج‌های مرتبی از X و مقدار تابع تعلق آن $\mu(x)$ مطابق رابطه (۱) نمایش داد.

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) | x \in U\} \quad (1)$$

^۹ Fuzzy Logic

در این رابطه $\mu_{\tilde{A}}(x)$ نشان دهنده تابع تعلق (عضویت) یا درجه عضویت مجموعه A هست. به طوری که برد این تابع شامل اعداد حقیقی غیر منفی در فاصله بسته $[0, 1]$ است. انجام محاسبات با اعداد فازی به دلیل ساختار خاص آن‌ها بسیار زمان بر و پیچیده است لذا برای تسهیل و کاربردی نمودن اعداد فازی، معمولاً اعداد فازی خاصی به کار گرفته می‌شوند. این اعداد خاص به صورت اعداد زنگوله‌ای^{۱۰}، مثلثی^{۱۱}، دوزنقه‌ای^{۱۲} و گوسی^{۱۳} هستند که با توجه به نوع و نیاز مساله می‌توان از هر یک از آن‌ها استفاده کرد [۲۶-۲۸]. در این مقاله، از اعداد فازی مثلثی جهت برآورد مدت زمان انجام فعالیت‌ها و تاخیر مجاز بین آنها استفاده می‌گردد. یک عدد فازی مثلثی \tilde{A} به وسیله یک سه وجهی $\tilde{A} = (a, b, c)$ مطابق شکل (۱) نشان داده می‌شود و تابع عضویت آن به صورت رابطه (۲) تعریف می‌گردد:

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{(x-a)}{(b-a)}, & a \leq x \leq b \\ \frac{(c-x)}{(c-b)}, & b \leq x \leq c \\ 0, & \text{otherwise} \end{cases} \quad (2)$$



شکل ۱: عدد فازی مثلثی.

برای دو عدد فازی مثلثی $\tilde{A} = (a, b, c)$ و $\tilde{B} = (a', b', c')$ همانند اعداد حقیقی می‌توان عملگرهای جمع، تفریق، ضرب، تقسیم، ماکسیمم، مینیمم و ... را تعریف نمود که در اینجا بر حسب کاربرد و نیاز فقط عملگرهای جمع، تفریق، ماکسیمم و مینیمم معرفی می‌گردد [۲۶، ۲۷].

$$\tilde{C} = \tilde{A} + \tilde{B} = (a + a', b + b', c + c') \quad \mu_{\tilde{C}}(Z) = \mu_{\tilde{A} + \tilde{B}}(Z) \quad (3)$$

$$\tilde{C} = \tilde{A} - \tilde{B} = (a - c', b - b', c - a') \quad \mu_{\tilde{C}}(Z) = \mu_{\tilde{A} - \tilde{B}}(Z) \quad (4)$$

$$M\tilde{A}X(\tilde{A}, \tilde{B}) = (\max(a, a'), \max(b, b'), \max(c, c')) \quad (5)$$

$$M\tilde{I}N(\tilde{A}, \tilde{B}) = (\min(a, a'), \min(b, b'), \min(c, c')) \quad (6)$$

^{۱۰} Bell

^{۱۱} Trianglie

^{۱۲} Trapezoidal

^{۱۳} Gaussian

با توجه به بررسی ادبیات مرتبط، رتبه بندی اعداد فازی بر دو گونه است رتبه بندی ضعیف و رتبه بندی قوی اعداد فازی که در ادامه به طور مختصر تشریح می گردند.

در حالت رتبه بندی قوی اعداد فازی (SCR^{14})، برای مقایسه دو عدد فازی \tilde{A} و \tilde{B} رابطه (۷) وجود دارد: [۲۸, ۲۹]

$$\tilde{A} \gg \tilde{B} \Leftrightarrow \max(\tilde{A}, \tilde{B}) = \tilde{A} \quad (7)$$

چنگ یک روش مبتنی بر فاصله را برای مقایسه دو عدد فازی مبنی بر فاصله بین مرکز مختصا و مرکز سطح عدد فازی، توسعه داد. مرکز سطح عدد فازی \tilde{A} به صورت روابط (۸) و (۹) محاسبه می شود [۲۸, ۳۰].

$$\bar{x}(\tilde{A}) = \frac{\int x dA}{A} \quad (8)$$

$$\bar{y}(\tilde{A}) = \frac{\int y dA}{A} \quad (9)$$

برای عدد فازی مثلثی $\tilde{A} = (a, b, c)$ ، فرمول های فوق را می توان به صورت روابط (۱۰) و (۱۱) نوشت:

$$\bar{x}_o(\tilde{A}) = \frac{(a + b + c)}{3} \quad (10)$$

$$\bar{y}(\tilde{A}) = \frac{1}{3} \quad (11)$$

با استفاده از فرمول های فوق، فاصله عدد فازی مثلثی \tilde{A} از مرکز مختصا طبق رابطه (۱۲) محاسبه می شود:

$$R(\tilde{A}) = \sqrt{(\bar{x}_o(\tilde{A}))^2 + (\bar{y}(\tilde{A}))^2} \quad (12)$$

در نهایت رابطه (۱۳) برای مقایسه دو عدد فازی \tilde{A}_1 و \tilde{A}_r به روش فاصله ای بکار برده می شود:

$$\begin{aligned} R(\tilde{A}_1) > R(\tilde{A}_r) &\Rightarrow \tilde{A}_1 > \tilde{A}_r \\ R(\tilde{A}_1) < R(\tilde{A}_r) &\Rightarrow \tilde{A}_1 < \tilde{A}_r \\ R(\tilde{A}_1) = R(\tilde{A}_r) &\Rightarrow \tilde{A}_1 = \tilde{A}_r \end{aligned} \quad (13)$$

برای تبدیل یک عدد فازی به یک مقدار شفاف قابل قبول، روش های مختلفی وجود دارد. در این پایان نامه از روش مرکز مختصا به علت عدم نیاز به قضاوت فردی آنالیست استفاده شده است. بنابراین عدد فازی مثلثی \tilde{A} با استفاده از رابطه (۱۴) فازی زدایی می گردد [۳۰].

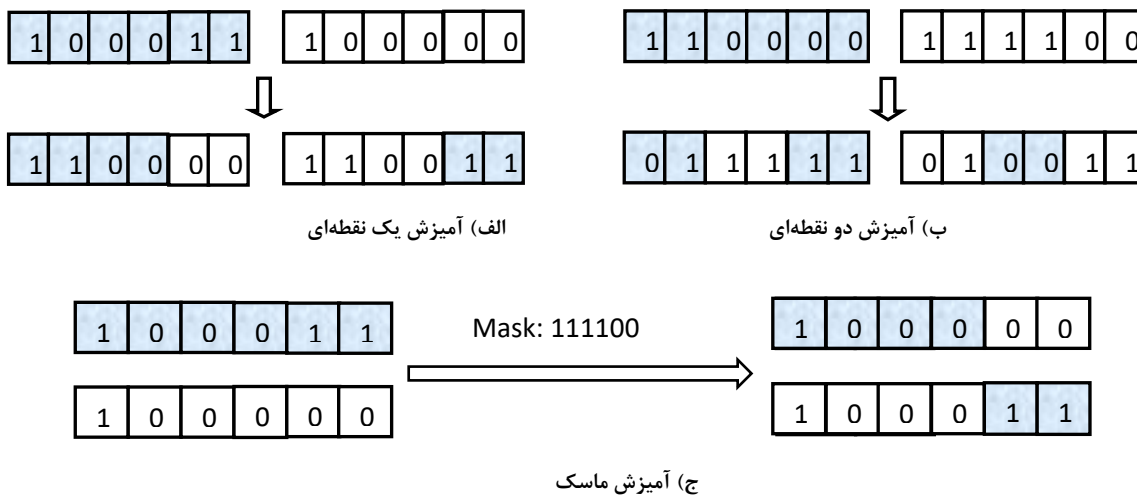
¹⁴ Strong Comparison Rule

$$dyfuzzy(\tilde{A}) = \frac{(a + b + c)}{3} \quad (14)$$

۲-۳- الگوریتم GA

الگوریتم ژنتیک (Genetic Algorithm) یکی از روش‌های تصادفی بهینه‌یابی بر پایه جمعیت بوده که توسط جان هالند در سال ۱۹۶۷ ابداع و در سال ۱۹۷۵ مبانی ریاضی آن در کتابی توسط هالند با نام "تطابق در سیستم‌های طبیعی و مصنوعی" منتشر شد. این روش الهام گرفته از علم ژنتیک و نظریه تکامل داروین است که از تکنیک‌های زیست‌شناسی مانند وراثت و جهش استفاده می‌کند [31].

در مرحله اجرای الگوریتم، یک نقطه از فضای جستجو به صورت یک کروموزوم متشکل از تعداد ثابتی ژن (متغیر) معرفی شده که معمولاً به صورت یک رشته کدگذاری (رشته‌های بیتی) نشان داده می‌شود و در طول چندین نسل تکامل می‌یابد. در هر نسل کروموزوم‌ها ارزیابی شده و متناسب با ارزش خود از امکان بقا و تکثیر برخوردار می‌گردند. تولید نسل جدید با تاثیر عملگرهای آمیزش (Crossover) و جهش (Mutation) بر روی مجموعه کروموزوم‌ها (جمعیت) صورت می‌گیرد. معمولاً جمعیت جدید برآزندگی بیشتری دارد این بدان معناست که از نسلی به نسل دیگر جمعیت بهبود می‌آید. آمیزش تک نقطه‌ای^{۱۵}، آمیزش دو نقطه‌ای^{۱۶} و آمیزش ماسک^{۱۷} از متداول‌ترین روش‌های آمیزش در الگوریتم ژنتیک است. در آمیزش تک نقطه‌ای که در شکل (الف) نشان داده شده‌است، یک موقعیت تصادفی در طول کروموزوم والد در نظر گرفته می‌شود. سپس تمامی ژن‌های طرف راست یا طرف چپ این موقعیت در کروموزوم‌های والد با یکدیگر جابجا شده تا کروموزوم‌های فرزند بدست آیند. در آمیزش دو نقطه‌ای که در شکل (ب) نشان داده شده‌است، دو موقعیت در طول کروموزوم والد به صورت تصادفی انتخاب می‌شود و تمامی ژن‌های بین این دو موقعیت در کروموزوم‌های والد با یکدیگر جابجا می‌شوند. در آمیزش ماسک فرزندان بر اساس یک ماسک ایجاد شده، تولید می‌شوند.



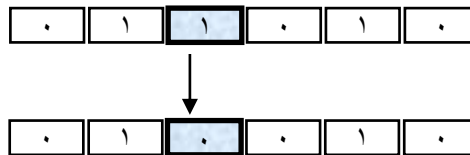
شکل ۲: انواع آمیزش‌های متداول در الگوریتم ژنتیک.

^{۱۵} One-point Crossover

^{۱۶} Two-point Crossover

^{۱۷} Mask Crossover

پس از اتمام عمل ادغام، مطابق شکل (۳) عملگر جهش بر روی کروموزومها اثر داده می شود. این عملگر یک ژن از یک کروموزوم را به طور تصادفی انتخاب نموده و سپس محتوای آن ژن را تغییر می دهد. جهش باعث جستجو در فضاهای دست نخورده مسئله می شود می توان استنباط کرد که مهمترین وظیفه جهش اجتناب از همگرایی به بهینه محلی است. جهش نباید زیاد صورت بگیرد زیرا در این صورت الگوریتم ژنتیک به جستوی کاملاً تصادفی تبدیل خواهد شد.



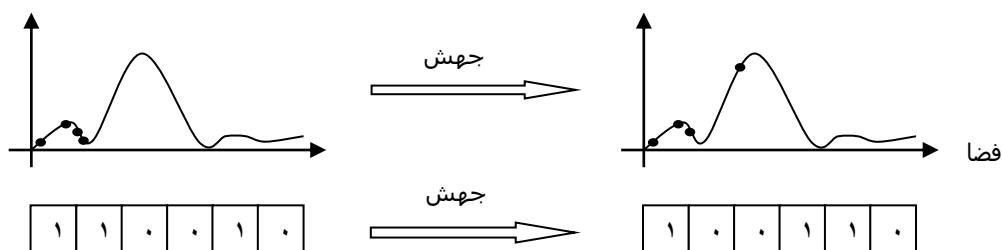
شکل ۳: شبیه سازی جهش به کمک نمودار.

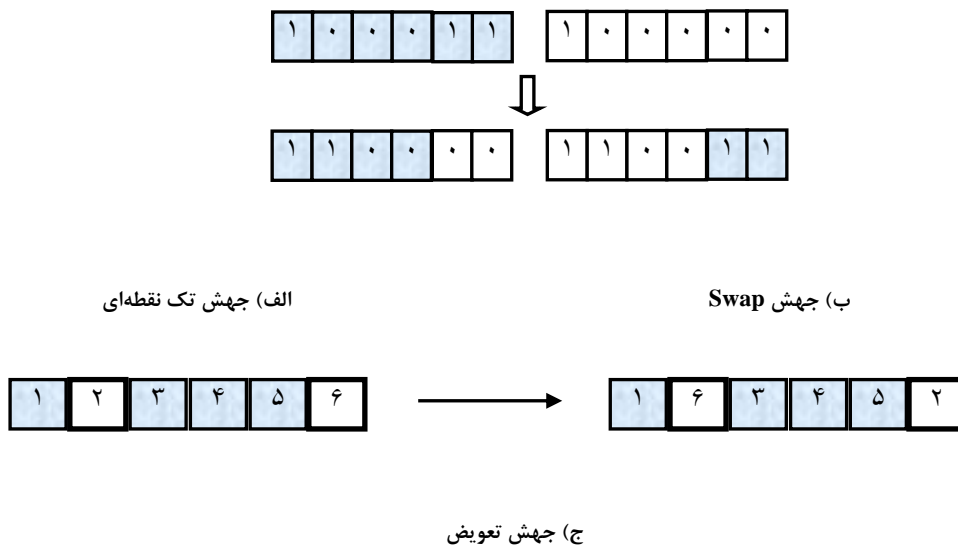
احتمال جهش را می توان ثابت در نظر گرفت یا اینکه آن را متناسب با تعداد نسل های تولید شده قرار داد. به عنوان مثال رابطه (۱۵) شدت کاهش جهش در هر نسل را نشان می دهد.

$$P_m = P_m - K_{pm} \frac{t}{T} \quad (15)$$

که در آن P_m احتمال جهش در هر نسل است K_{pm} یک ضریب ثابت است که شدت کاهش احتمال جهش را مشخص می کند این کار باعث می شود در ابتدا جهش با احتمال زیادی انجام شده و فضای جستجو گسترش یابد سپس احتمال جهش کاهش یافته و جستجو در قسمت هایی در نتایج یافت شده متمرکز گردد و سرعت همگرایی افزایش یابد.

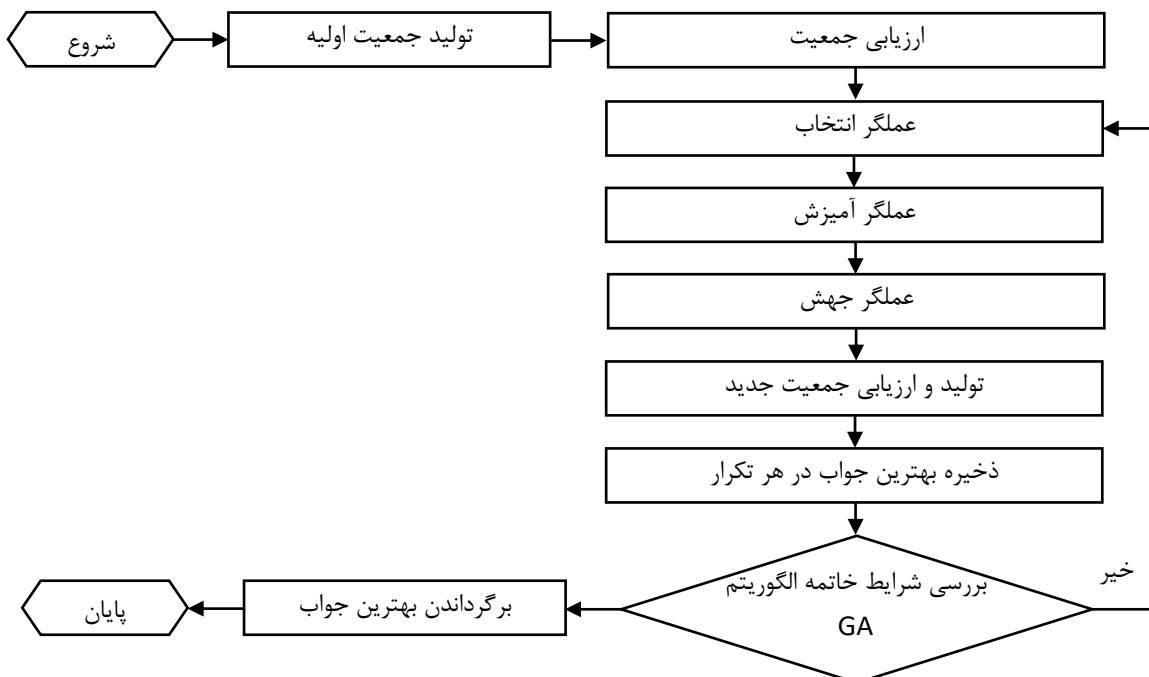
جهش نقطه ای، جهش Swap و جهش تعویض از متداول ترین روش های جهش در الگوریتم ژنتیک است. در جهش نقطه ای مطابق شکل (۴ الف)، یک ژن از یک کروموزوم به طور تصادفی انتخاب شده و مقدار آن تغییر می کند. در جهش Swap مطابق شکل (۴ ب)، تبادل ژن بین دو کروموزوم صورت می گیرد. از جهش تعویض مطابق شکل (۴ ج)، مخصوصاً در الگوریتم هایی استفاده می شود که کد گذاری بر اساس مقدار باشد البته در دیگر کد گذاری ها مثل کد گذاری باینری هم می توان این جهش را بکار برد. در این جهش، محل قرارگیری دو ژنی که می خواهد جهش بیابند در کروموزوم تعویض می شود





شکل ۴: انواع جهش‌های متداول در الگوریتم GA.

روند کلی اجرای الگوریتم ژنتیک به صورت شکل (۵) است:



شکل ۵: روند کلی اجرای الگوریتم ژنتیک.

۴-۲- مسئله MRCPSF فازی به همراه تاخیر بین فعالیت‌ها

در این بخش مسئله بررسی شده در پژوهش جاری با هدف کمینه کردن زمان تکمیل پروژه معرفی شده و مدل ریاضی آن توسعه داده می‌شود. همانطور که واضح است در عمل گاه یک فعالیت به بیش از یک روش قابل انجام است که هر روش اجراء، مقدار منبع و مدت زمان انجام خاص خود را طلب می‌کند. به عبارت دیگر می‌توان به عنوان نمونه با صرف منابع بیشتر و یا روش دیگر اجراء، آن فعالیت را به

اتمام رسانید. به این علت دسته‌های دیگر از مسائل، تحت عنوان زمانبندی پروژه با محدودیت منابع چند حالتی بوجود آمده است. در ادبیات موضوع مسئله RCPSP چندحالتی به صورت MRCPSM نمایش داده می‌شود.

مسئله زمانبندی فازی پروژه‌های ساخت با منابع محدود در چندین روش اجرایی (MRCPSM) را می‌توان زمانبندی مجموعه‌ای از فعالیت‌ها $V=\{0,1,\dots,n+1\}$ با هدف یافتن بهترین روش اجرایی برای هر فعالیت و بهینه‌ترین توالی زمانی مناسب برای انجام فعالیت‌ها در نظر گرفت به طوری که زمان تکمیل پروژه کمینه شود و محدودیت‌های منابع (منابع تجدیدپذیر و منابع تجدیدنپذیر) و نیز محدودیت‌های پیشنیازی به طور همزمان ارضا گردند. مدت زمان انجام فعالیت‌ها و تاخیر مجاز بین آنها به صورت فازی در نظر گرفته شده است.

فعالیت‌های ۰ و $n+1$ فعالیت‌های مجازی ابتدا و انتهای پروژه با زمان صفر و بدون احتیاج به منابع هستند. هر فعالیت j با حالت اجرایی m_j آغاز می‌شود و باید در همان حالت اجرایی m_j پایان پذیرد. \bar{d}_{jm_j} بیان کننده مدت زمان اجرای فعالیت j در حالت اجرایی m_j با در نظر گرفتن عدم قطعیت در برآورد دقیق آن، است. هر یک از روش‌های اجرایی به یک یا چند نوع از مجموعه منابع تجدیدپذیر (RR) و مجموعه منابع تجدیدنپذیر (NR) نیاز دارند. منبع تجدیدپذیر نوع k دارای ظرفیت محدود Q_k^{RR} در هر دوره زمانی و به همین ترتیب منبع تجدیدنپذیر نوع u دارای ظرفیت محدود Q_u^{NR} در کل زمان پروژه است. اگر فعالیت j در حالت اجرایی m_j انجام شود، به اندازه $r_{jm_j,k}^{RR}$ واحد از نوع $k \in RR$ منابع تجدیدپذیر در هر پریود از زمان و به اندازه $r_{jm_j,u}^{NR}$ واحد از نوع $u \in NR$ منابع تجدیدنپذیر نیاز دارد. همانطور که گفته شد در این مسئله دو نوع منبع وجود دارد. ۱- منابع تجدیدپذیر از جمله منابع انسانی، ابزار و تجهیزات که پس از هر بار استفاده در یک فعالیت، مجدداً قابل استفاده در یک فعالیت دیگر هستند. ۲- منابع تجدیدنپذیر از جمله منابع مصرفی که در طول افق زمانبندی پروژه محدود هستند و محدودیت آنها خاص طول یک پریود زمانی نیست. بعضی از منابع مانند بودجه تخصیص داده شده به پروژه، علاوه بر اینکه در یک پریود زمانی محدوداند، در افق زمانبندی پروژه نیز محدودیت دارند. این دسته از منابع، به منابع با محدودیت دوگانه^{۱۸} مشهور هستند که می‌توان آنها را به صورت ترکیبی از منابع تجدیدپذیر و منابع تجدیدنپذیر نشان داد [۱۹، ۳۴-۳۲].

جهت انجام پژوهش جاری مفروضاتی در نظر گرفته شده است که این شرایط و فرضیات مسئله را می‌توان به شرح زیر بیان نمود:

- ۱- یک پروژه مد نظر است. ۲- روابط پیشنیازی از نوع (FS)^{۱۹} است. ۳- فضای انجام پژوهش به صورت فازی در نظر گرفته شده است به عبارت دیگر مدت زمان انجام فعالیت‌ها و تاخیر مجاز^{۲۰} بین آنها غیرقطعی است. ۴- نیازمندی هر فعالیت به منابع مشخص و قطعی است.
- ۵- منابع از دو نوع تجدیدپذیر و تجدیدنپذیر هستند. ۶- ظرفیت منابع تجدیدپذیر در هر دوره زمانی مشخص، قطعی و ثابت است. ۸-
- ظرفیت منابع تجدیدنپذیر در کل پروژه مشخص و قطعی است. ۹- هر فعالیت دارای چندین حالت اجرایی است که فعالیت تحت یک روش اجرایی شروع شده و با همان روش اجرایی پایان می‌یابد. ۱۰- هر فعالیت فقط یکبار می‌تواند شروع شده و به صورت پیوسته و بدون انقطاع تا پایان ادامه یابد. ۱۱- هر فعالیت در روزهایی می‌تواند انجام شود که منابع مورد نیازش موجود باشد. ۱۲- تسطیح منابع، فقط با جابجایی فعالیت‌ها انجام می‌گیرد و روش‌های دیگری مانند: کاهش بار کاری منابع با افزایش مدت زمان انجام فعالیت‌ها، اضافه کاری منابع و شکستن فعالیت‌ها مد نظر قرار داده نمی‌شوند.

مسئله مذکور، به صورت زیر مدل‌سازی می‌شود: [۳۵]

$$\text{Min} \sum_{t=\bar{e}_{n+1}}^{\bar{s}_{n+1}} t \cdot x_{(n+1) \square t} \quad (16)$$

^{۱۸} Doubly constrained resource

^{۱۹} Finish to Start

^{۲۰} Time Lag

Subject to:

$$\tilde{t}_i = \cdot \quad (17)$$

$$\sum_{m_j=1}^{M_j} \sum_{t=\tilde{e}s_j}^{\tilde{l}s_j} x_{jm_jt} = 1 \quad j = 1, \dots, n \quad (18)$$

$$\sum_{m_i=1}^{M_i} \sum_{t=\tilde{e}s_i}^{\tilde{l}s_i} (t + \tilde{d}_{im_i} + L\tilde{a}g_{ij})x_{im_it} \leq \sum_{m_j=1}^{M_j} \sum_{t=\tilde{e}s_j}^{\tilde{l}s_j} t \cdot x_{jm_jt} \quad (19)$$

$$\forall i \in P_j; j = 2, \dots, n+1$$

$$\sum_{j=1}^n \sum_{m_j=1}^{M_j} r_{jm_jk}^{RR} \sum_{s=\max\{t-\tilde{d}_{jm_j+1}, \tilde{e}s_j\}}^{\min\{t, \tilde{l}s_j\}} x_{jm_js} \leq Q_k^{RR} \quad (20)$$

$$\forall k = 1, \dots, RR; t = 1, \dots, T$$

$$\sum_{j=1}^n \sum_{m_j=1}^{M_j} r_{jm_ju}^{NR} \sum_{s=\tilde{e}s_j}^{\tilde{l}s_j} x_{jm_js} \leq Q_u^{NR} \quad \forall u = 1, \dots, NR \quad (21)$$

$$T \leq 1.5 \sum_{j=1}^n \sum_{m_j=1}^{M_j} \sum_{t=\tilde{e}s_j}^{\tilde{l}s_j} \tilde{d}_{jm_j} \cdot x_{jm_jt} \quad (22)$$

$$\tilde{d}_i = \cdot \quad (23)$$

$$\tilde{d}_{n+1} = \cdot \quad (24)$$

$$x_{jm_jt} = \begin{cases} 1 & \text{if } j \text{ in mode } m \text{ started at } t \\ \cdot & \text{otherwise} \end{cases} \quad (25)$$

$$j = 1, \dots, n; m_j = 1, 2, \dots, M_j; t = \tilde{e}s_j, \dots, \tilde{l}s_j \quad (26)$$

پارامترهای مورد استفاده در مدل، به صورت جدول (۱) تعریف شده اند:

جدول ۱: پارامترهای مورد استفاده در مدل MRCPSP فازی.

| | |
|--|------------------------|
| مجموعه فعالیت‌های پیشنهادی فعالیت j | P_j |
| تعداد کل فعالیت‌های پروژه | n |
| مجموعه فعالیت‌های پروژه | $\{0, 1, \dots, n+1\}$ |
| مدت زمان انجام فعالیت j در روش اجرایی m_j به صورت عدد فازی مثلثی | \tilde{d}_{jm_j} |
| زودترین زمان شروع فعالیت j بر اساس روش‌های اجرایی و با کمترین زمان ممکن برای اجرا شدن به صورت عدد فازی مثلثی | $\tilde{e}s_j$ |
| زودترین زمان پایان فعالیت j بر اساس روش‌های اجرایی و با کمترین زمان ممکن برای پایان یافتن به صورت عدد فازی مثلثی | $\tilde{l}s_j$ |
| تاخیر مجاز بین انجام فعالیت j و فعالیت پیشنهادش i به صورت عدد فازی مثلثی | $\tilde{L}ag_{ij}$ |
| تعداد کل روش‌های اجرایی فعالیت j | M_j |
| شمارنده روش‌های اجرایی برای هر فعالیت j | m_j |
| شمارنده منابع تجدیدپذیر | k |
| تعداد منابع تجدیدپذیر | RR |
| شمارنده منابع تجدیدناپذیر | u |
| تعداد منابع تجدیدناپذیر | NR |
| مقدار لازم از منبع تجدیدپذیر نوع k برای انجام فعالیت j در روش اجرایی m_j | $r_{jm_jk}^{RR}$ |
| مقدار لازم از منبع تجدیدناپذیر نوع u برای انجام فعالیت j در روش اجرایی m_j | $r_{jm_ju}^{NR}$ |
| حداکثر مقدار موجود از منبع تجدیدپذیر نوع k در هر دوره زمانی | Q_k^{RR} |
| حداکثر مقدار موجود از منبع تجدیدناپذیر نوع u در کل پروژه | Q_u^{NR} |
| افق زمانبندی پروژه | T |
| متغیر تصمیم | $x_{jm_j\tilde{t}}$ |

رابطه (۱۶) تابع هدف مسئله، زمان اتمام فعالیت انتهایی پروژه را کمینه می‌کند؛ رابطه (۱۷) بیان کننده زمان شروع پروژه است؛ رابطه (۱۸) مشخص می‌کند که هر فعالیت تنها یک روش اجرایی و یک زمان شروع دارد و با هر روش اجرایی که شروع شود با همان روش اجرا باید خاتمه یابد؛ رابطه (۱۹) بر محدودیت روابط پیشنهادی بین فعالیت‌های پروژه تاکید دارد. در این محدودیت تاخیر مجاز بین فعالیت‌ها نیز مد نظر قرار داده شده است؛ رابطه (۲۰) مربوط به محدودیت منابع تجدیدپذیر در دسترس برای انجام فعالیت‌ها در هر بازه زمانی t است؛ رابطه (۲۱) مربوط به محدودیت منابع تجدیدناپذیر در دسترس برای انجام فعالیت‌ها در کل پروژه است؛ رابطه (۲۲) افق زمانبندی پروژه را نشان می‌دهد؛ رابطه (۲۳) و (۲۴) بیان می‌کند که مدت زمان اجرای فعالیت‌های مجازی ابتدا و انتهای پروژه صفر است؛ رابطه (۲۵) متغیر تصمیم است که این متغیر، باینری بوده و در صورتی که فعالیت j در روش اجرایی m_j در زمان t شروع به انجام گیرد، مقدار آن یک و در غیر این صورت مقدار صفر را به خود می‌گیرد؛ رابطه (۲۶) محدوده مقادیر پارامترهای موجود در مدل را نشان می‌دهد.

۵-۲- تبدیل فازی به کلاسیک

می‌توان برای مسئله زمانبندی پروژه، مدت زمان تکمیل پروژه را به دو صورت ۱- مدت تکمیل مورد انتظار با استفاده از روش مرکز سطح عدد فازی ۲- مدت تکمیل مورد انتظار با استفاده از روش ضریب سطح برش و اندیس خوشبینی مدیریت پروژه مدل‌سازی کرد.

۳- تئوری و محاسبات

۱-۳- الگوریتم ژنتیک برای حل مدل مسئله

پس از مدل‌سازی مسئله، اطلاعات اولیه لازم برای مسئله در نرم‌افزار Matlab وارد می‌شود. در ادامه کاهش فضای جستجو با حذف روش‌های اجرایی ناکارآمد و غیرقابل اجرا و منابع تجدیدناپذیر زائد با توجه به فرآیند Preprocessing صورت می‌گیرد.

اسپرچر و همکاران (۱۹۹۷) برای کاهش فضای جستجو از روشی به نام Preprocessing بهره گرفتند. در روش پیشنهادی آنها روش‌های اجرایی ناکارآمد و غیرقابل اجرا و منابع تجدیدناپذیر زائد حذف می‌شوند. این فرآیند در سه گام به شرح زیر مسئله را پیش از بکارگیری الگوریتم فراابتکاری، اصلاح می‌کند:

گام ۱) حذف روش‌های اجرایی غیرقابل اجرا: مد m_j غیر اجرایی است اگر رابطه (۲۷) و یا (۲۸) برقرار باشد.

$$r_{jm,k}^{RR} > Q_k^{RR} \quad k \in RR \quad (27)$$

$$\sum_{\substack{i=1 \\ i \neq j}}^N r_{miniu}^{NR} + r_{jmju}^{NR} > Q_u^{NR} \quad u \in NR \quad (28)$$

گام ۲) حذف روش‌های اجرایی ناکارآمد: مد m_j ناکارآمد است اگر مد اجرایی دیگری مانند m'_j وجود داشته باشد به طوری که رابطه (۲۹) برقرار باشد.

$$d_{jm_j} \geq d_{jm'_j}, \quad r_{jm_j,k}^{RR} \geq r_{jm'_j,k}^{RR}, \quad r_{jm_j,u}^{NR} \geq r_{jm'_j,u}^{NR} \quad (29)$$

گام ۳) حذف منابع تجدیدناپذیر زائد: منبع تجدیدناپذیر u زائد است اگر رابطه (۳۰) برقرار باشد.

$$\sum_{j=1}^N r_{maxju}^{NR} \leq Q_u^{NR} \quad (30)$$

۱-۳-۱- نحوه نمایش جواب‌ها

در روش پیشنهادی به‌منظور حل مدل، هر جواب به‌صورت دو فهرست n عنصری که n برابر با تعداد فعالیت‌های پروژه است، نشان داده می‌شود. فهرست اول روش اجرایی فعالیت‌ها بوده که فهرست روش نامیده می‌شود. فهرست دوم یک توالی ممکن برای انجام فعالیت‌ها است که فهرست فعالیت نامیده می‌شود. از آنجا که مسئله زمانبندی پروژه از نوع مسائل گسسته جایگشتی است. لذا الگوریتم مورد استفاده جهت حل مسئله، بایستی با این فضای گسسته تطبیق داده شوند. البته می‌توان مسئله را با الگوریتم‌های مذکور به‌صورت گسسته نیز حل کرد ولی در این پژوهش رویکرد حل، پیوسته در نظر گرفته شده‌است. بنابراین بدین صورت عمل می‌شود که با استفاده از

روش کلید تصادفی^{۲۱} یکسری اعداد حقیقی تصادفی بین صفر و یک، تولید و فهرست فعالیت‌ها و فهرست روش‌ها به وسیله آن کدگذاری می‌شوند.

۲-۱-۳- تنظیم پارامترهای الگوریتم ژنتیک

پس از استخراج اطلاعات اولیه، الگوریتم GA در متلب کدنویسی می‌شود. این الگوریتم به پارامترهایی نیاز دارد که تنظیم مناسب آنها در کیفیت جواب الگوریتم تاثیر بسزایی دارد؛ از روش تاگوچی جهت تنظیم پارامترهای الگوریتم GA استفاده می‌شود. تاگوچی طراحی آزمایش‌ها را به گونه‌ای ایجاد کرد که در آن تعداد آزمایش‌ها کاهش یافت. بر اساس این روش زمانیکه مسئله از دسته بهینه‌سازی و نوع حداقل‌سازی باشد از رابطه (۳۱) استفاده می‌شود.

$$SN = -10 \cdot \log \frac{1}{n} \sum_{i=1}^n y_i^2 \quad (31)$$

آزمایشات جهت تنظیم پارامترهای الگوریتم GA در چهار سطح انجام شد. محدوده تغییر این پارامترها با مرور مقالات محققان پیشین در حوزه زمانبندی پروژه تعیین شد. پارامترهای مذکور در جدول (۲) پیشنهاد داده شده‌اند.

جدول ۲: پارامترهای پیشنهاد شده الگوریتم ژنتیک در ۴ سطح.

| سطوح پارامترها | | | | پارامتر | الگوریتم |
|----------------|------|------|------|-----------|----------|
| ۴ | ۳ | ۲ | ۱ | | |
| ۰/۹ | ۰/۸ | ۰/۷ | ۰/۶ | p_c | GA |
| ۰/۲۵ | ۰/۲ | ۰/۱۵ | ۰/۱ | p_m | |
| ۰/۱ | ۰/۰۹ | ۰/۰۷ | ۰/۰۵ | K_{p_m} | |
| ۳۰ | ۲۰ | ۱۵ | ۱۰ | nPop | |
| ۱۵۰ | ۱۰۰ | ۶۰ | ۴۰ | MaxIt | |

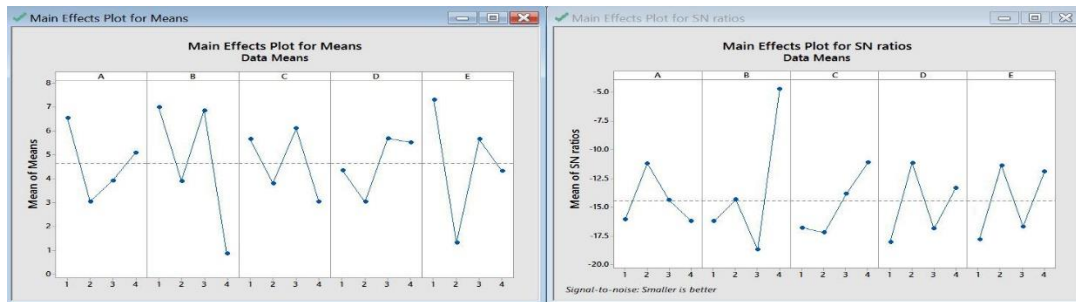
مسائل مختلف موجود در ادبیات موضوع با روش تاگوچی به تعداد ۱۶ آزمایش (L۱۶) برای الگوریتم GA آزموده شد که نتیجه در جدول (۳) آمده است.

^{۲۱} Random Key Representation

جدول ۳: تنظیم پارامترهای الگوریتم GA به روش تاگوچی برای مسائل مختلف.

| مسائل | | | | | پارامتر | الگوریتم |
|--------------|--------------|--------------|--------------|--------------|----------|----------|
| j3027_1 فازی | j2044_7 فازی | j1822_4 فازی | j1433_7 فازی | j1010_1 فازی | | |
| ۰/۷ | ۰/۸ | ۰/۷ | ۰/۷ | ۰/۷ | p_c | GA |
| ۰/۲ | ۰/۲۵ | ۰/۲ | ۰/۱ | ۰/۲۵ | p_m | |
| ۰/۰۷ | ۰/۰۷ | ۰/۰۷ | ۰/۰۹ | ۰/۱ | K_{pm} | |
| ۲۰ | ۲۰ | ۲۰ | ۱۵ | ۱۵ | nPop | |
| ۱۰۰ | ۱۰۰ | ۱۰۰ | ۶۰ | ۶۰ | MaxIt | |

به عنوان نمونه نتیجه حاصل از آزمایش تاگوچی برای تنظیم پارامترهای مسئله j1010_1 در شکل ۶ آورده شده است.



شکل ۶: تنظیم پارامترهای مسئله j1010_1 با روش تاگوچی.

۳-۱-۳- تولید جمعیت اولیه

در ابتدا با استفاده از روش کلید تصادفی یک فهرست روش اجرایی به صورت تصادفی برای هر عضو از جمعیت ایجاد می‌شود. فهرست روش ایجاد شده ممکن است از لحاظ میزان مصرف منابع تجدیدناپذیر امکان پذیر نباشد. به عبارت دیگر روش‌های اجرایی تخصیص داده شده به فعالیت‌ها به گونه‌ای باشد که میزان مصرف منابع تجدیدناپذیر در کل پروژه بیشتر از ظرفیت مجاز آنها باشد ($\sum_{j=1}^N r_{jm}^{NR} - Q_u^{NR} > 0$)، که این موضوع امری نشدنی است. در بسیاری از مواقع تابع هدف با تابع برازش در یک زمان اتفاق می‌افتد چرا که تخمین دقیقی از چگونگی شایستگی افراد در اختیار می‌گذارد. برای مسئله RCPSP، زمان اتمام پروژه طبیعتاً تابع برازش خوبی به‌شمار می‌رود. اما برای مسائل MRCPSP، نمی‌تواند تابع برازش مناسبی باشد؛ به دلیل اینکه برخی از افراد هر جمعیتی می‌توانند در ارتباط با منابع تجدیدناپذیر، عملی نباشند. و چنانچه روش‌های اجرایی این افراد، امکان‌پذیر نگردد، لذا ممکن است پس از تعیین برازندگی افراد مذکور، مقدار برازندگی آنها (افراد غیر عملی و ناممکن) بهتر از افراد ممکن و عملی بشود. این بدان معنا است که افراد ناممکن که در اصل زمانبندی را غیر عملی می‌سازند باید در فرآیند انتخاب جریمه^{۲۲} شوند. جهت برخورد با امکان‌ناپذیری مربوط به روش‌های اجرایی، دو رویکرد وجود دارد: (۱) از تابع جریمه استفاده کرد و به جواب‌های امکان‌ناپذیر جریمه‌ای، اختصاص یابد. توابع جریمه مختلفی متناسب با هر مسئله از سوی محققین پیشنهاد شده است. [۳۶] اختلاف بین تابع برازش را با در نظر گرفتن تابع جریمه و بدون تابع جریمه مورد سنجش

²² Penalize

قرار دادند و نشان دادند که تابع برازش با داشتن تابع جریمه کارایی بهتری را دارا می باشد. ساده ترین تابع تناسب (بrazش) مطابق رابطه (۳۵) است. می توان بر اساس تابع برازش لوا و همکاران [۳۷] که برای مسائل MRCPSP قطعی ارائه شده است، تابع برازشی به صورت رابطه (۳۶) برای مسائل فازی در نظر گرفت.

$$f(i) = \begin{cases} C_{max} & ERR = 0 \\ T + ERR(i) & otherwise \end{cases} \quad (32)$$

$$f(i) = \begin{cases} 1 - \frac{E(\max_feas_C_{max}) - E(C_{max})}{E(\max - feas - C_{max})} & ERR = 0 \\ 1 + \frac{E(C_{max}) - E(CP^{min})}{E(C_{max})} + normal(ERR) & otherwise \end{cases} \quad (33)$$

در رابطه (۳۲) تابع جریمه در صورت امکان پذیر بودن لیست روش های اجرایی، تغییری در جواب مسئله اعمال نکرده و مقدار C_{max} ، یعنی زمان تکمیل پروژه را برمی گرداند اما در صورت امکان ناپذیری، مقدار بدست آمده از تست امکان پذیری را به T (حد بالای مسئله) می افزاید و بدین ترتیب آن را به جوابی نامطلوب بدل می کند. در رابطه (۳۳)، $E(C_{max})$ زمان مورد انتظار تکمیل پروژه است. $E(\max_feas_C_{max})$ بیشترین مدت تکمیل مورد انتظار زمانبند عملی در ارتباط با افراد نسل جاری است. $E(CP^{min})$ مسیر بحرانی^{۲۳} است که از کمترین مدت تکمیل مورد انتظار هر فعالیت استفاده می کند. همچنین $normal(ERR)$ مقدار نرمال ERR است که از رابطه (۳۸) حاصل می شود. در تابع برازش (۳۳) فردی که ممکن و عملی باشد و بیشترین مدت تکمیل را داشته باشد، یک مقدار برازش برابر ۱ خواهد داشت. در حالیکه بهترین فرد مقدار برازندگی نزدیک به صفر دارد. همچنین مقدار تابع برازش برای فرد ناممکن و نشدنی همیشه بزرگتر از یک است. لذا مقدار برازندگی تمام افراد نشدنی، بزرگتر از افراد شدنی است و فرصت کمتری را در فرآیند انتخاب دارند. (۲) در رویکرد دوم که در پژوهش جاری نیز از آن استفاده شده است، با استفاده از یک الگوریتم جستجوی محلی، می توان لیست روش های اجرایی را امکان پذیر ساخت تا اصلا روش اجرایی امکان ناپذیر ایجاد نشود. در این الگوریتم برای هر منبع تجدیدنپذیر مقدار شاخص امکان پذیری (ERR) مطابق با رابطه (۳۴) محاسبه می شود.

$$ERR(u) = \max \left\{ 0, \sum_{j=1}^N r_{jm_ju}^{NR} - Q_u^{NR} \right\} \quad (34)$$

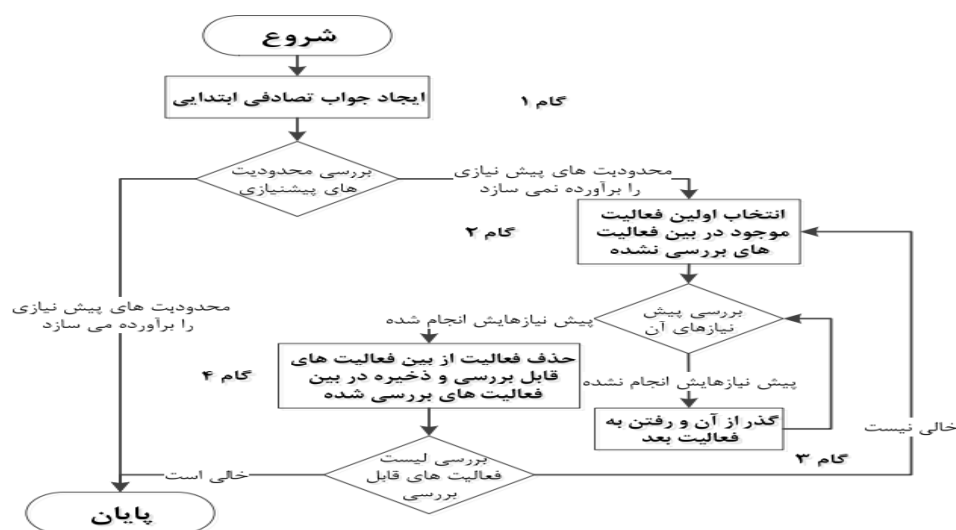
$$normal(ERR(u)) = \sum_{u=1}^{NR} \left(\max \left\{ 0, \frac{\sum_{i=1}^N (r_{jm_ju}^{NR}) - Q_u^{NR}}{Q_u^{NR}} \right\} \right) \quad (35)$$

در صورتیکه $ERR(u) = 0, \forall u = 1, \dots, NR$ باشد آنگاه لیست روش های اجرایی شدنی بوده و با توجه به روش های اجرایی انتخاب شده در لیست، برای هر فعالیت اطلاعاتی شامل: مدت زمان اجرا و مقدار نیازمندی هر فعالیت به منابع تجدیدپذیر، استخراج می گردد. اما در صورتیکه $ERR(u) > 0, \forall u = 1, \dots, NR$ باشد آنگاه لیست روش های اجرایی نشدنی بوده و بایستی اصلاح گردد تا به یک لیست شدنی تبدیل شود. جهت اصلاح این لیست، روش اجرایی یک و یا دو فعالیت چند روشه را به طور تصادفی تغییر داده و مقدار $ERR(u)$ محاسبه می گردد. این روند تا برقراری شرط $ERR(u) = 0, \forall u = 1, \dots, NR$ ادامه می یابد.

²³ Critical Path

بعد از ایجاد فهرست روش‌های اجرایی برای هر عضو از جمعیت و استخراج اطلاعات اولیه برای هر فعالیت، فهرست توالی ممکن برای انجام فعالیت‌ها برای هر عضو متناظر با فهرست روش اجرایی آن عضو، ایجاد می‌گردد تا فرآیند تولید جمعیت اولیه تکمیل شود. جهت تولید برنامه زمانبندی دو روش اصلی وجود دارد: ۱- تولید برنامه زمانبندی به روش سری [۳۸] و موازی [۳۹]. در این پژوهش از روش سری استفاده می‌شود. جهت تولید برنامه زمانبندی، ابتدا با استفاده از روش کلید تصادفی به هر فعالیت پروژه یک عدد حقیقی تصادفی بین صفر و یک، تخصیص داده می‌شود. سپس این اعداد مرتب شده و در لیست q ذخیره می‌گردند، ترتیب اعداد تخصیص داده شده به فعالیت‌ها، در واقع نقش قاعده اولویت را برای هر فعالیت بازی می‌کند. جایگشت حاصل از مرتب شدن این اعداد تصادفی، که نشان‌دهنده توالی انجام فعالیت‌های پروژه است، به عنوان جواب در نظر گرفته می‌شود.

جواب مورد نظر ممکن است به علت رعایت نکردن پیش‌نیازها، نشدنی باشد، لذا جایگشت حاصله با استفاده از الگوریتم محاسبه جواب موجه ابتدایی، اصلاح شده و به جوابی امکان‌پذیر تبدیل می‌گردد. الگوریتم تولید جواب موجه ابتدایی بر مبنای روابط پیش‌نیازی بین فعالیت‌ها بنا نهاده شده است. در شکل (۷) روند اجرای الگوریتم مذکور نشان داده شده است [۴۰]. گام‌های این الگوریتم به صورت زیر است: گام (۱) یک جواب تصادفی ابتدایی که اولویت اجرای فعالیت‌ها را به دست می‌دهد، ایجاد می‌شود؛ بدین صورت که از بین $N!$ طریق ممکن در بین اولویت‌های اجرای فعالیت‌ها بدون توجه به محدودیت‌های مسئله یک جواب تصادفی انتخاب می‌شود. در صورتیکه جواب تصادفی بدست آمده از لحاظ محدودیت‌های پیش‌نیازی صحیح باشد. جواب موجه ابتدایی حاصل شده است. در غیر این صورت به گام ۲ می‌رود؛ گام (۲) اولین فعالیت در بین فعالیت‌های بررسی نشده را در نظر می‌گیرد؛ در صورتیکه تمامی پیش‌نیازی‌های آن انجام گرفته باشد به گام ۴ و در غیر این صورت به گام ۳ می‌رود. در نخستین مرحله، تنها فعالیتی که می‌تواند انتخاب شود، فعالیت مجازی شروع و در آخرین مرحله هم فعالیت مجازی پایان خواهد بود؛ گام (۳) فعالیتی را که پیش‌نیاز آن انجام نگرفته رها می‌کند و به گام ۲ باز می‌گردد. از بین دیگر فعالیت‌های بررسی نشده موجود، فعالیت بعدی را انتخاب می‌کند؛ گام (۴) فعالیت بررسی شده در گام ۲ را از فهرست فعالیت‌های قابل بررسی حذف می‌کند و به فهرست فعالیت‌های بررسی شده انتقال می‌دهد. در صورتیکه هنوز فعالیت بررسی نشده‌ای باقی مانده باشد به گام ۲ باز می‌گردد. در غیر این صورت فعالیت‌های ذخیره شده در فهرست فعالیت‌های بررسی شده را به ترتیب ذخیره شدن مرتب می‌کند و به عنوان جواب ابتدایی اصلاح شده یا جواب موجه نشان می‌دهد. پس از انتخاب و زمانبندی کلیه فعالیت‌ها زمان مجازی پایان به عنوان زمان پایان پروژه گزارش می‌گردد. از آنجایی که مدت زمان انجام فعالیت‌ها فازی است، بنابراین عملیات کمینه و بیشینه و عملگر جمع در تولید برنامه زمانبندی سری نیز باید فازی باشد.



شکل ۷: روند اجرای الگوریتم تولید جواب موجه ابتدایی.

۴-۱-۳- ارزیابی جمعیت اولیه و ذخیره بهترین جواب

هنگامی که الگوریتم GA بکار گرفته می‌شود، لازم است که یک تابع برازش متناسب با تابع هدف تعریف شود تا احتمال بقای افراد در نسل بعدی را تعیین کند. در این مرحله برای هر توالی ممکن ایجاد شده با توجه به روش اجرایی آن، ارزیابی صورت می‌گیرد تا زمان تکمیل پروژه (C_{max}) بدست آید. پس از مرتب‌سازی جمعیت اولیه بر حسب C_{max} های بدست‌آمده، بهترین جواب متشکل از بهترین روش اجرایی هر فعالیت، توالی بهینه ممکن برای انجام فعالیت‌ها در جمعیت اولیه، زمان شروع و پایان آنها، زمان تکمیل پروژه و منابع تجدیدپذیر استفاده شده و باقیمانده در هر دوره زمانی، ذخیره می‌گردد.

۵-۱-۳- تولید جمعیت جدید

در الگوریتم GA، جمعیت جدید با استفاده از عملگرهای ادغام و جهش ایجاد می‌شود. برای عملیات ادغام، دو عضو از جمعیت به‌طور تصادفی با استفاده از چرخ رولت، به عنوان والد انتخاب می‌گردد. سپس به تصادف یکی از سه روش ۱- ادغام تک نقطه‌ای ۲- ادغام دو نقطه‌ای ۳- ادغام با ماسک برگزیده می‌شود. عمل ادغام هم در فهرست روش‌های اجرایی و هم در فهرست توالی انجام فعالیت‌ها به‌طور مجزا صورت می‌گیرد.

برای عملیات جهش در الگوریتم GA نیز سه استراتژی ۱- جهش تک نقطه‌ای ۲- روش Swap ۳- روش تعویض در نظر گرفته شده‌است که با احتمال کاهنده مطابق رابطه (۱۵) یک استراتژی در هر بار تکرار انتخاب و عمل جهش انجام می‌شود. عملیات جهش بر روی فهرست توالی انجام فعالیت‌ها انجام می‌گردد و می‌توان آن را برای فهرست روش‌های اجرایی انجام نداد.

از آنجایی که از یک طرف الگوریتم ژنتیک نیاز به ارزیابی راه‌حل‌ها دارد و از طرف دیگر مدت زمان انجام و تاخیر مجاز بین فعالیت‌ها و در نهایت مدت زمان تکمیل پروژه برای هر راه‌حل یک عدد فازی است؛ لذا مدت تکمیل فازی هر راه‌حل با یک مقدار نماینده یعنی مقدار مورد انتظار حقیقی جایگزین شده و سپس عمل ارزیابی بین راه‌حل‌ها صورت می‌گیرد. جایگزینی جمعیت فعلی با نسل جدید در حالت کلی به دو روش انجام می‌پذیرد. ۱- سیاست کلی یا ساده^{۲۴} که در آن همه افراد جمعیت فعلی جایگزین افراد نسل جدید با اندازه جمعیت یکسان می‌شود. ۲- سیاست وضعیت پایدار^{۲۵} که در آن برخی از افراد جمعیت فعلی جایگزین افراد نسل جدید می‌شوند. دو عامل اصلی در روند الگوریتم GA حائز اهمیت است. ۱- گوناگونی و پراکندگی جمعیت ۲- فشار انتخابی. به‌منظور داشتن جستجوی موثر، لازم است تا یک معیار جستجو یا همان تابع برازش و یک فشار انتخاب در نظر گرفته شود. بدین ترتیب به افرادی که ارزش بیشتری دارند، شانس بیشتری در بقا و ترکیب می‌دهد. بدون فشار انتخاب، فرآیند جستجو تصادفی می‌شود و نواحی نوید بخش فضای مسئله، مساعدتی در دیگر نواحی نخواهند کرد. البته گوناگونی جمعیت بخش مهمی در قابلیت الگوریتم ژنتیک برای ادامه اکتشاف فضای مسئله است. فشار انتخاب و گوناگونی جمعیت با یکدیگر رابطه عکس دارند. هر چقدر فشار انتخاب بیشتر گردد، گوناگونی جمعیت کاسته می‌شود و بالعکس. این دو فاکتور مفید، بایستی به‌طور هم‌زمان کنترل گردند تا نواحی نوید بخش بیشتری از فضای مسئله حاصل شود. در نهایت بهترین جواب متشکل از اطلاعاتی از قبیل: توالی بهینه ممکن برای انجام فعالیت‌ها، زمان شروع و پایان آنها، زمان تکمیل پروژه و منابع تجدیدپذیر باقی‌مانده و استفاده شده در هر دوره زمانی، در هر تکرار در لیستی به نام BestSolMode ذخیره می‌گردد.

۶-۱-۳- اجرای مجدد الگوریتم‌ها با ورودی BestSolMode تا رسیدن به یک تکرار معین

در این مرحله بهترین روش اجرایی و توالی متناظر با آن به عنوان ورودی در نظر گرفته می‌شوند. سپس هر یک از الگوریتم‌های فراابتکاری تا رسیدن به یک تکرار معین، به‌منظور یافتن توالی بهینه برای روش اجرایی اختصاص‌یافته، اجرا می‌شوند.

^{۲۴} Generational or Simple

^{۲۵} Steady State

۷-۱-۳- استخراج جواب پایانی تحت نام BestSol

در مرحله آخر، جواب پایانی تحت نام BestSol برگردانده می‌شود. جواب نهایی اطلاعاتی شامل موارد زیر را در اختیار می‌گذارد:

- ۱- بهترین روش اجرایی برای هر فعالیت؛ ۲- بهینه‌ترین توالی ممکن برای انجام فعالیت‌ها؛ ۳- زمان تکمیل پروژه (C_{max})؛ ۴- زمان شروع و پایان هر فعالیت؛ ۵- میزان منابع تجدیدناپذیر استفاده شده و باقیمانده در پروژه؛ ۶- میزان منابع تجدیدپذیر استفاده شده و باقیمانده در هر دوره زمانی.

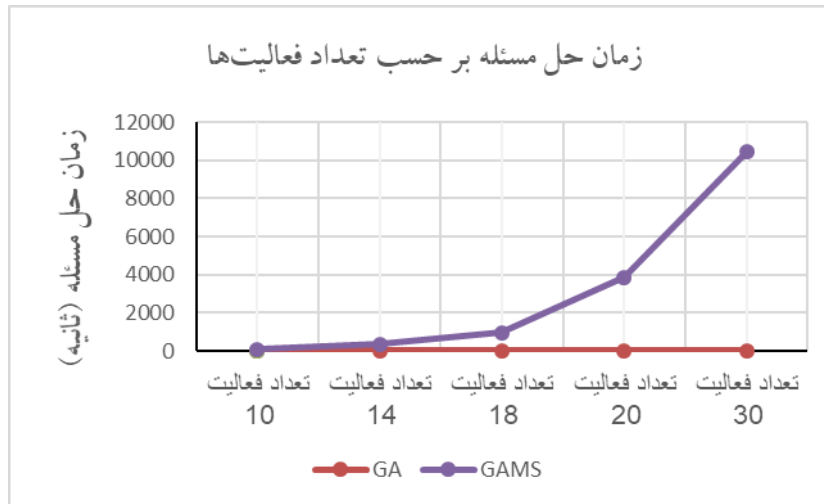
۴- بحث و نتایج

به‌منظور اثبات کارایی الگوریتم GA تلفیق‌شده با مجموعه‌های فازی، نتایج حل این الگوریتم با نتایج حل به‌وسیله نرم‌افزار GAMS که یک نرم‌افزار بهینه‌سازی دقیق است، مقایسه می‌شود. برای این امر چند نمونه مسئله استاندارد زمانبندی، به‌طور تصادفی از سایت PSPLIB انتخاب می‌شود. از مجموعه‌های J10، J14، J18، J20، J30، که از نظر تعداد فعالیت‌ها با یکدیگر متفاوتند، مسائل نمونه دارای ۲ منبع تجدیدپذیر و ۲ منبع تجدیدناپذیر هستند. از آنجا که مسائل موجود در سایت PSPLIB برای RCPSP پایه و MRCPSP پایه در محیط قطعی ارائه شده‌اند. در ابتدا با اعمال تغییراتی، مسائل مورد نظر در کتابخانه مسائل استاندارد MRCPSP پایه متناسب با پژوهش جاری به‌صورت مسائل فازی با تاخیر مجاز بین فعالیت‌ها در می‌آیند. جهت ارزیابی الگوریتم GA با نرم‌افزار GAMS، سه معیار ۱- کمترین C_{max} در ۱۰۰۰ تکرار الگوریتم GA ۲- میانگین زمان حل مسئله (Average CPU Time) -۳ \overline{APD} در ۱۰۰۰ تکرار الگوریتم GA تعریف شده‌است. الگوریتم GA برای هر مسئله ۱۰۰۰ بار اجرا می‌شود و شاخص‌های ۲ و ۳ به‌صورت میانگین محاسبه می‌گردد. نتایج حاصل از حل در جدول (۴) نشان داده شده‌است.

جدول ۴: نتایج حاصل از حل مسائل منتخب استاندارد با GA و GAMS.

| مسئله | روش حل | کمترین C_{max} در ۱۰۰۰ تکرار | Average CPU Time (s) | \overline{APD} (%) |
|---------|--------|--------------------------------|----------------------|----------------------|
| j1010_1 | GA | ۱۹ | ۹/۰۳ | ۰/۷۶ |
| | GAMS | ۱۹ | ۹۱/۰۳ | - |
| j1433_7 | GA | ۳۹ | ۱۴/۵۵ | ۱/۸۷ |
| | GAMS | ۳۹ | ۳۴۷ | - |
| j1822_4 | GA | ۵۸/۳۳۳ | ۱۶/۵۹ | ۳/۴۴ |
| | GAMS | ۵۷/۳۳۳ | ۹۴۷ | - |
| j2044_7 | GA | ۵۰/۶۶۷ | ۱۹/۹۶ | ۴/۳۷ |
| | GAMS | ۴۹/۳۳۳ | ۳۸۷۸ | - |
| j3027_1 | GA | ۵۱/۳۳۳ | ۲۹/۲۹ | ۹/۸۷ |
| | GAMS | ۴۹ | ۱۰۵۶ | - |

مطابق شکل (۸) هر چه تعداد فعالیت‌های مسئله افزایش یابد، زمان حل مسئله برای الگوریتم GA با یک آهنگ ثابت افزایش می‌یابد. اما با افزایش تعداد فعالیت‌ها، زمان حل مسئله با استفاده نرم‌افزار GAMS، با آهنگ زیادی، افزایش می‌یابد. حتی ممکن است مسئله به‌قدری پیچیده شود که نرم‌افزار GAMS نتواند جوابی را در اختیار بگذارد.



شکل ۸: نمودار زمان حل مسئله بر حسب تعداد فعالیت‌ها برای هر سه الگوریتم.

مطابق جدول ۴، با افزایش تعداد فعالیت‌ها، پارامتر متوسط درصد انحراف مطلق (\overline{APD}) برای الگوریتم GA با آهنگ کمی، افزایش می‌یابد این افزایش پارامتر \overline{APD} با افزایش تعداد فعالیت‌ها ناشی از افزایش فضای جستجوی مسئله (فضای جستجوی روش‌های اجرایی و فضای جستجوی توالی ممکن برای انجام فعالیت‌ها) است. این امر باعث افزایش مینیمم‌های محلی مسئله می‌شود. با اینکه تدابیری برای جلوگیری از احتمال گرفتار شدن الگوریتم مذکور در مینیمم‌های محلی اخذ شده‌است ولی با این حال افزایش فضای جستجوی مسئله، باعث افزایش احتمال گرفتار شدن الگوریتم مذکور در مینیمم‌های محلی می‌شود.

۵- نتیجه‌گیری

زمانبندی بهینه پروژه می‌تواند از اعمال هزینه‌های اضافی به پروژه، هدر رفت منابع جلوگیری کند با زمانبندی بهینه پروژه می‌توان در مورد روش اجرایی مناسب هر فعالیت تصمیم صحیحی گرفت. به‌طور کلی با توجه به ابعاد بزرگ پروژه‌های ساخت، در عمل زمانبندی بهینه پروژه موجب صرفه‌جویی‌های هنگفت در زمان، هزینه و کاهش ریسک پروژه‌ها و افزایش حاشیه سود مجریان خواهد گردید. در این پژوهش مسئله زمانبندی پروژه‌های ساخت با منابع محدود چندگانه در چندین حالت اجرایی (MRCPS) و در نظر گرفتن عدم قطعیت در مدت زمان فعالیت‌ها و تاخیر مجاز بین انجام آنها مورد مطالعه قرار گرفت. پس از مدل‌سازی ریاضی مسئله، با استفاده از الگوریتم GA تلفیق شده با مجموعه‌های فازی، مسئله مذکور حل گردید. از مزیت‌های مدل ارائه‌شده در این پژوهش در نظر گرفتن عدم قطعیت در مدت زمان انجام فعالیت‌ها و تاخیر مجاز بین آنها است تا در هنگام زمانبندی پروژه مبحث احتمال وقوع ریسک در انجام فعالیت‌های پروژه نیز در نظر گرفته شده‌باشد. در انتها عملکرد الگوریتم، با نرم‌افزار GAMS مورد سنجش قرار گرفت. نتایج بدست‌آمده حاکی از عملکرد موفق الگوریتم ذکر شده از لحاظ سه معیار ۱- کمترین C_{max} در ۱۰۰۰ تکرار الگوریتم GA ۲- میانگین زمان حل مسئله (Average CPU Time) - ۳ \overline{APD} در ۱۰۰۰ تکرار الگوریتم GA در زمانبندی بهینه پروژه است. بنابراین رویکرد بکار گرفته‌شده در این پژوهش می‌تواند به‌سادگی قابل استفاده توسط مدیران و برنامه‌ریزان پروژه باشد. این امر موجب جلوگیری از خطاهای انسانی مسئول کنترل پروژه در تسطیح منابع می‌شود و راهی به‌سوی زمانبندی بهینه پروژه است. در نرم افزارهای MSP و Primavera، مسئول کنترل پروژه باید با تجربه شخصی و به‌صورت دستی و با صرف زمان و انرژی بیشتری به تسطیح منابع بپردازد و هیچ‌گونه الگویی جهت مقایسه

برنامه زمانبندی با زمانبندی بهینه پروژه ندارد به عبارت دیگر برنامه ریزی با نرم افزارهای کنترل پروژه، بهینه سازی زمانبندی را تضمین نمی کند. اما با استفاده از رویکرد بکار گرفته در این پژوهش، مسئول کنترل پروژه می تواند با صرف زمان و انرژی کمتر، پروژه را به صورت بهینه زمانبندی کند. بنابراین الگوریتم های فراابتکاری می توانند به عنوان ابزار مفیدی در کنار نرم افزارهای کنترل پروژه مورد استفاده قرار گیرند.

مراجع

- [1] Sabzeparvar, M. (2012). *Project control step-by-step*. Fourteenth ed. Tehran: Terme.
- [2] Giran, O. Temur R. and Bekdaş, G. (2017). *Resource constrained project scheduling by harmony search algorithm*. KSCE Journal of Civil Engineering, 21(2), 479-487.
- [3] Blazewicz, J. Lenstra, J.K. and A Kan, A.H.G.R. (1983). *Scheduling subject to resource constraints: classification and complexity*. Discrete Applied Mathematics, 5(1), 11-24.
- [4] Artigues, C. Demasse, S. and Néron, E. (2008). *Resource-constrained project scheduling : models, algorithms, extensions and applications*. ed. F. Sourd, USA: John Wiley & Sons, Inc.
- [5] Shirmohamadi, A.H. (2009). *Project management and control*. 2nd ed. Isfahan: Jihad University of Isfahan University of Technology.
- [6] Neumann, K. and Zimmermann, J. (2003). *Project Scheduling with Time Windows and Scarce Resources*. 2 ed. Berlin: Springer-Verlag Berlin Heidelberg.
- [7] Hartmann, S. and Briskorn, D. (2010). *A survey of variants and extensions of the resource-constrained project scheduling problem*. European Journal of Operational Research. 207(1), 1-14.
- [8] Weglarz, J. (2009). *Project Scheduling Recent Models, Algorithms And Applications*. 1 ed. Springer US.
- [9] Liu, B. (2009). *Theory and Practice of Uncertain Programming*. Studies in Fuzziness and Soft Computing. Springer-Verlag Berlin Heidelberg.
- [10] Geiger, M.J. (2017) *A multi-threaded local search algorithm and computer implementation for the multi-mode, resource-constrained multi-project scheduling problem*. European Journal of Operational Research, 256(3), 729-741.
- [11] Prade, H. (1979). *Using fuzzy set theory in a scheduling problem: A case study*. Fuzzy Sets and Systems, 2(2), 153-165.
- [12] Hapke, M. Jaszkiwicz, A. and Słowiński, R. (2000). *Pareto Simulated Annealing for Fuzzy Multi-Objective Combinatorial Optimization*. Journal of Heuristics, 6(3), 329-345.
- [13] Leu, S.S. and Hung, T.H. (2002). *A genetic algorithm-based optimal resource-constrained scheduling simulation model*. Construction Management and Economics, 20(2), 131-141.
- [14] Zhang, H. et al. (2005). *Particle swarm optimization-based schemes for resource-constrained project scheduling*. Automation in Construction, 14(3), 393-404.
- [15] Mendes, J.J.M. Gonçalves, J.F. and Resende, M.G.C. (2009). *A random key based genetic algorithm for the resource constrained project scheduling problem*. Computers & Operations Research, 36(1), 92-109.
- [16] Huang, Y. Shou, Y. and Zhang, L. (2011). *Genetic algorithm for the project scheduling problem with fuzzy time parameters*. In: 2011 IEEE International Conference on Industrial Engineering and Engineering Management.
- [17] Masmoudi, M. and Häit, A. (2013). *Project scheduling under uncertainty using fuzzy modelling and solving techniques*. Engineering Applications of Artificial Intelligence, 26(1), 135-149.
- [18] Afshar-Nadjafi, B. and Majlesi, M. (2014). *Resource constrained project scheduling problem with setup times after preemptive processes*. Computers & Chemical Engineering, 69(3), 16-25.
- [19] Knyazeva, M., Bozhenyuk, A. and Rozenberg, I. (2015). *Resource-constrained Project Scheduling Approach Under Fuzzy Conditions*. Procedia Computer Science, 77(1), 56-64.
- [20] Joy, J. Rajeev, S. and Narayanan, V. (2016). *Particle Swarm Optimization for Resource Constrained-project Scheduling Problem with Varying Resource Levels*. Procedia Technology, 25(1), 948-954.
- [21] Badri, A., Nadeau S. and Gbodossou, A. (2012). *Proposal of a risk-factor-based analytical approach for integrating occupational health and safety into project risk evaluation*. Accident Analysis & Prevention, 48(1), 223-234.
- [22] Mulholland, B. and Christian, J. (1999). *Risk Assessment in Construction Schedules*. Journal of Construction Engineering and Management, 125(1), 8-15.
- [23] Pinto, A., Nunes, I.L. and Ribeiro, R.A. *Occupational risk assessment in construction industry – Overview and reflection*. Safety Science, 49(5), 616-624.
- [24] Daneshpayeh, H. Haswanpoor, H. (2013). *Project scheduling with fuzzy data using refrigeration simulation algorithm*. Production management, 9(2) 57-74.

- [25] Gavareshki, M.K. (2004). *New fuzzy GERT method for research projects scheduling*. In: *Engineering Management Conference*, IEEE.
- [26] Zadeh, L.A. (1965). *Fuzzy sets*. *Information and Control*, 8(3), 338-353.
- [27] Zadeh, L.A. (1978). *Fuzzy sets as a basis for a theory of possibility*. *Fuzzy sets and systems*, 1(1), 3-28.
- [28] Hamzedanesh P. and Hossienali P. (2013). *Project scheduling with fuzzy data using refrigeration simulation algorithm*. *Production and Operations Management*, 5(2), 57-74.
- [29] Liu, S. Yung, K.L. (2007). *Genetic local search for resource-constrained project scheduling under uncertainty*. *International Journal of Information and Management Sciences*, 18(4), 347-356.
- [30] Cheng, C.H. (1998). *A new approach for ranking fuzzy numbers by distance method*. *Fuzzy sets and systems*, 95(3), 307-317.
- [31] Holland, J.H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press, Ann Arbor.
- [32] Coelho, J. and Vanhoucke, M. (2011). *Multi-mode resource-constrained project scheduling using RCPSP and SAT solvers*. *European Journal of Operational Research*, 213(1), 73-82.
- [33] Xu, J. and Feng, C. (2014). *Multimode Resource-Constrained Multiple Project Scheduling Problem under Fuzzy Random Environment and Its Application to a Large Scale Hydropower Construction Project*. *The Scientific World Journal*, 2014(1), 1-20.
- [34] Menesi, W. and Hegazy, T. (2015). *Multimode Resource-Constrained Scheduling and Leveling for Practical-Size Projects*. *Journal of Management in Engineering*, 31(6), 1-7.
- [35] Atli, O. and Kahraman, C. (2012). *Fuzzy resource-constrained project scheduling using taboo search algorithm*. *International Journal of Intelligent Systems*, 27(10), 873-907.
- [36] Józefowska, J. et al. (2001). *Simulated Annealing for Multi-Mode Resource-Constrained Project Scheduling*. *Annals of Operations Research*, 102(1), 137-155.
- [37] Lova, A. et al. (2009). *An efficient hybrid genetic algorithm for scheduling projects with resource constraints and multiple execution modes*. *International Journal of Production Economics*, 117(2), 302-316.
- [38] Kelley, J.E. (1963). *The critical-path method: Resources planning and scheduling*. *Industrial scheduling*, 347-365.
- [39] Bedworth, D.D. and Bailey, J.E. (1987). *Integrated Production Control Systems: Management*. Analysis, Design, 2nd ed. New York: John Wiley & Sons, Inc.
- [40] Safari, H. (2015). *solving Resource Constrained-project Scheduling Problem (RCPSP) using the modified Imperialist Competitive Algorithm (DICA)*. *Industrial Management*, 17(7), 333-364.