

زمان بندی پروژه با منابع محدود به کمک الگوریتم ژنتیک

حامد رضا طارقیان، محمدهادی فراهی و مژگان معرب

گروه ریاضی - دانشگاه فردوسی مشهد

پست الکترونیکی: farahi@math.um.ac.ir

چکیده

در هر پروژه تعدادی فعالیت وابسته به هم وجود دارند. اجرای هر یک از این فعالیت‌ها نیازمند منابع متفاوتی است که بعضاً محدود می‌باشند. هر فعالیت پروژه می‌تواند در چندین حالت مختلف اجرا شود که اجرای هر حالت مستلزم زمان و به کارگیری منابع معین است. هدف در مسأله برنامه‌ریزی پروژه با منابع محدود که یک مسأله NP-hard است تعیین زمان شروع و حالت اجرای هر فعالیت به گونه‌ای است که زمان اجرای پروژه را کمینه نماید. در حل این مسأله از الگوریتم ژنتیک استفاده شده است. به منظور تولید جواب‌های موجه اولیه، کروموزوم‌ها به وسیله ۹ قاعده متفاوت اولویت‌بندی کدگذاری شده‌اند. علاوه بر آن، کروموزوم‌های هر نسل بیان‌گر بهترین جواب حاصل از انجام هر فعالیت پروژه در سه وضعیت؛ (الف) بدون استفاده از تأخیر مجاز آن‌ها، (ب) انجام هر فعالیت پروژه با استفاده از کل تأخیر مجاز آن‌ها، و (پ) انجام هر فعالیت با استفاده از بخشی تصادفی از تأخیر مجاز آن‌ها می‌باشند. به منظور تشریح عملکرد این روش، یک مثال عددی ارائه شده است.

واژه‌های کلیدی: برنامه‌ریزی و کنترل پروژه، الگوریتم ژنتیک، بهینه‌سازی

مقدمه

یک پروژه مجموعه‌ای از فعالیت‌هایی است که از طریق روابط منطقی متفاوتی که حاکم بر آنهاست به یکدیگر ارتباط پیدا می‌کنند. به این معنا که هر چند تعدادی از این فعالیت‌ها می‌توانند همزمان و به صورت موازی انجام شوند، اما شروع یا خاتمه تعدادی در گرو انجام یک یا چند فعالیت پیش‌نیاز آنها می‌باشند. این روابط با استفاده از عوامل کنترل‌کننده پایان به شروع، شروع به شروع، پایان به پایان و شروع به پایان تعریف می‌شوند. روابط

منطقی بین فعالیت‌های یک پروژه می‌تواند در قالب ترکیبی از این عوامل کنترل‌کننده بیان شوند. اجرای هر فعالیت پروژه معمولاً نیازمند منابع متفاوتی از جمله زمان، سرمایه، نیروی انسانی و غیره است که بعضاً محدود می‌باشند. منابع مورد استفاده در پروژه‌ها را غالباً به دو دسته تجدیدشدنی^۱ و تجدیدنشده^۲ تقسیم می‌کنند.

1- Renewable Resources

2- Nonrenewable Resources

که در آن n تعداد فعالیت‌ها، m انواع منابع، d_i مدت زمان انجام فعالیت i ، t_i زمان شروع فعالیت i ، b_k موجودی منبع تجدیدشدنی k و A_i مجموعه فعالیت‌های انجام شده در زمان t می‌باشد.

S_i مجموعه فعالیت‌هایی که فعالیت j به آن‌ها وابسته است، r_k مقدار منبع k که مورد نیاز فعالیت i است و $mode$ نیز عبارت است از حالت انجام فعالیت که مبین یکی از چند حالت انجام هر فعالیت است.

مسئله برنامه‌ریزی پروژه با منابع محدود برای نخستین بار در سال ۱۹۶۳ مطرح شد [۲ و ۳]. طی ۱۰ سال گذشته پیشرفت‌های قابل توجهی در تدوین روش‌های کارآمد حل این مسئله صورت گرفته است. به طور کلی دو رویکرد برای حل این مسئله وجود دارد: روش‌های حل بهینه^۲ و دیگری روش‌های حل ابتکاری^۳ روش‌های حل مبتنی بر رویکرد اول مورد توجه این مقاله نیستند. برای بررسی و مروری بر این روش‌ها، می‌توان مثلاً به مراجع [۴ و ۵] در خصوص برنامه‌ریزی ریاضی، به مراجع [۶ و ۷] در خصوص روش‌های شمارشی مثل برنامه‌ریزی پویا و بالاخره به مراجع [۸ و ۹] در خصوص روش‌های شاخه و کران مراجعه کرد. شایان ذکر است که در پروژه‌های بزرگ روش‌های حل بهینه کارایی لازم را ندارند [۱۰]. از این رو اکثر مسایل واقعی برنامه‌ریزی پروژه با منابع محدود به کمک روش‌های ابتکاری حل می‌شوند.

رویکردهای ابتکاری در مراجع مختلف از جمله [۱۱] مورد بررسی قرار گرفته است. در این مرجع، به کمک ۵۳۶ مسئله آزمون استاندارد، عملکرد چهار رویکرد متفاوت ابتکاری تحلیل و با یکدیگر مقایسه شده‌اند. در میان چهار رویکرد ابتکاری بررسی شده، تنها رویکرد

هر فعالیت جدا از یک پروژه می‌تواند در چند حالت مختلف اجرا شود. البته اجرای هر حالت مستلزم زمان و به کارگیری منابع معین است. مثلاً فعالیتی می‌تواند به صورت دستی و یا به صورت مکانیزه اجرا شود که انجام هر حالت در گرو به کارگیری منابع خاص خود می‌باشد.

هدف مسئله برنامه‌ریزی پروژه با منابع محدود که یک مسئله NP-hard است [۱]، تعیین زمان شروع و حالت اجرای هر فعالیت می‌باشد به گونه‌ای که زمان اجرای پروژه را کمینه نماید. بدیهی است که پاسخ این مسئله نه تنها باید قیود مربوط به ارتباط منطقی فعالیت‌ها را تأمین کند، بلکه باید به محدودیت منابع نیز توجه داشته باشد.

مسئله برنامه‌ریزی با منابع محدود را با توجه به شرایط متفاوت، به صورت‌های مختلف می‌توان مدل‌سازی نمود. فرض کنید پروژه به صورت یک شبکه فعالیت-روی-گره^۱ مثل گراف $G = (V, E)$ که در آن V مجموعه گره‌ها و بیانگر فعالیت‌های پروژه و E مجموعه یال‌ها و بیانگر رابطه منطقی پایان به شروع است، تعریف شده باشد. فعالیت‌ها از ۱ تا n شماره‌گذاری شده‌اند. فعالیت‌های ۱ و n مجازی و نشان‌دهنده شروع و پایان پروژه می‌باشند. فعالیت‌ها پس از شروع تا پایان بدون وقفه اجرا می‌شوند. در چنین شرایطی این مسئله را می‌توان به صورت زیر مدل‌سازی کرد:

$$\begin{aligned} & \text{Min} \quad t_n \\ & \text{Subject to:} \\ & t_j - t_i \geq d_{i_{mode}} \quad \forall i \in S_j \\ & \sum_{i=1}^{|A_j|} r_{k_{mode}} \leq b_k \quad k = 1, \dots, m \\ & t_i \geq 0 \quad i, j = 1, \dots, n. \end{aligned}$$

روش حل مسأله

مسأله برنامه‌ریزی پروژه با منابع محدود مورد بررسی این مقاله دارای مشخصه‌های زیر است. در برنامه‌ریزی، اجرای فعالیت‌ها به صورت پیوسته در نظر گرفته شده‌اند. تنها عامل کنترل پایان به شروع مبین رابطه منطقی بین فعالیت‌ها می‌باشد. محدودیتی در تعداد منابع مورد نیاز و تعداد حالات انجام فعالیت‌ها در نظر گرفته نشده و از هر دو نوع منابع تجدیدشدنی و تجدیدنشدنی برای فعالیت‌ها استفاده شده است.

در پیاده‌سازی الگوریتم ژنتیک هر کروموزوم را به وسیله ۹ قاعده متفاوت اولویت کدگذاری کرده و از این طریق جواب‌های موجه اولیه تولید کرده‌ایم. از این طریق توانسته‌ایم اطلاعات مربوط به ۹ رویکرد آزمون شده اولویت‌بندی را در قالب یک شیوه زمان‌بندی خلاصه نماییم.

هر کروموزوم معرف یک راه حل موجه است. هر کروموزوم را به وسیله دو رشته ژن نمایش داده‌ایم. پس در پروژه‌ای که مثلاً دارای ۶ فعالیت است، کروموزوم زیر یک کروموزوم موجه است:

۱ ۳ ۲ ۱ ۱ ۳

۵ ۳ ۹ ۴ ۷ ۲

رشته اول که معرف حالت انجام هر فعالیت می‌باشد، مبین این است که مثلاً فعالیت دوم در حالت ۳ و فعالیت سوم در حالت ۲ اجرا می‌شود. ژن‌های رشته دوم معرف قوانین اولویت هستند که در هر دور از تخصیص منابع به فعالیت‌ها، برای انتخاب یک فعالیت از بین فعالیت‌های واجد شرایط به کار گرفته می‌شوند. این انتخاب به دلیل عدم امکان انجام همه فعالیت‌ها به طور همزمان به لحاظ محدودیت منابع صورت می‌گیرد. پس در مثال فوق رشته دوم بیان می‌دارد که مثلاً در دور دوم تخصیص منابع، از

ارائه شده توسط مرجع [۱۱] قادر به یافتن جواب موجه برای تمام ۵۳۶ مسأله آزمون بوده است. ضمناً متوسط اختلاف جواب به دست آمده با جواب بهینه در این رویکرد کمترین مقدار را داشته است. کاربرد الگوریتم ژنتیک در بررسی مسأله زمان‌بندی پروژه که با مسأله تسطیح منابع ترکیب گردیده، در مراجع [۱۲ و ۱۳] گزارش شده است. در این مراجع مسأله دو منظوره مطرح شده، بدون استفاده از قواعد اولویت‌بندی به یک مسأله بدون قید تبدیل گردیده است. در مرجع [۱۴] یک رویکرد ابتکاری روش زمان‌بندی موازی پیشنهاد گردیده که در آن فعالیت‌ها بر مبنای کمترین زمان مجموع تأخیر مجازشان انتخاب و در حالت اجرایی که کمترین زمان اجرا را دارد و از نظر تخصیص منابع نیز موجه است، زمان‌بندی می‌شوند. یک روش مبتنی بر نمونه‌برداری تصادفی از یک زمان‌بندی پشت سرهم فعالیت‌ها و با استفاده از قاعده کوتاه‌ترین زمان اجرا در [۱۵] پیشنهاد شده است.

الگوریتم ژنتیک در حل مسایلی که دارای ماهیت ترکیباتی هستند، همانند مسأله زمان‌بندی به طور مؤثر مورد استفاده قرار گرفته است [۱۶ و ۱۷]. یک مطالعه مروری در زمینه کاربرد الگوریتم ژنتیک در حل مسائل زمان‌بندی توسط [۱۸] ارائه گردیده که در آن ادغام این الگوریتم با روش‌های مختلف جستجو بررسی شده است. ما نیز در این مقاله برای حل مسأله زمان‌بندی پروژه با منابع محدود از یک طراحی ویژه الگوریتم ژنتیک که در آن کروموزوم‌ها با اطلاعات ناشی از ماهیت مسأله و روش حل سرشته شده‌اند، استفاده نموده‌ایم. در این طراحی هر کروموزوم حاوی دو رشته ژن است. رشته ژن اول بیان‌گر حالت اجرای هر فعالیت و رشته ژن دوم مبین قاعده اولویت‌بندی است که بر اساس آن فعالیت مربوطه زمان‌بندی می‌شود.

پنجره زمانی خود زمان‌بندی می‌شوند. در رویکرد سوم به نام زمان‌بندی میان-پنجره‌ای^۴، همه فعالیت‌ها از قسمتی از پنجره زمانی به جز est و $d_A - lft$ (d_A زمان انجام فعالیت مثلاً A است) که به تصادف انتخاب می‌شود، زمان‌بندی می‌گردند. انجام فعالیت در قسمت‌های متفاوتی از پنجره زمانی باعث تعدیل اوج نیاز به منابع گردیده و از این رو موجب می‌شود تا زمینه برای زمان‌بندی همزمان فعالیت‌های بیشتری فراهم گردیده و در نهایت به کاهش زمانی پروژه منجر گردد.

در هر سه رویکرد پیش‌گفته و در هر دور از تخصیص منابع به فعالیت‌ها، فعالیت‌های واجد شرایط را مشخص می‌کنیم. آن‌گاه با استفاده از قوانین اولویت‌کد شده در رشته دوم کروموزوم‌ها، یکی از فعالیت‌ها را انتخاب کرده و سپس زمان شروع و پایان آن را مشخص می‌کنیم. این فرایند تا تهی شدن مجموعه فعالیت‌های واجد شرایط ادامه می‌یابد. سپس زمان شروع دور بعدی تخصیص منابع به فعالیت‌ها، به کوچک‌ترین زمان پایان در میان فعالیت‌های برنامه‌ریزی شده تغییر می‌یابد. این روند تا اتمام تخصیص منابع به آخرین فعالیت پروژه ادامه می‌یابد.

همان‌طور که گفته شد جواب متناظر با هر کروموزوم پس از اعمال سه شیوه پیش‌گفته و تعیین کوتاه‌ترین زمان اجرای پروژه به ازای آن‌ها به دست می‌آید. پس از تعیین مدت زمان اجرای پروژه برای کروموزوم مثلاً k ، احتمال تکثیر آن را به صورت زیر به دست می‌آوریم:

$$p(k) = \frac{(\sqrt{t_k})^2}{\sum_{k \in S} (\sqrt{t_k})^2}$$

که در آن t_k زمان اجرای پروژه متناظر با کروموزوم k و S جمعیت نسل اول است. شکل خاص رابطه احتمالی

بین فعالیت‌های واجد شرایط، یک فعالیت بر طبق قانون اولویت ۳ انتخاب و برنامه‌ریزی می‌شود. فعالیت‌های واجد شرایط، آنهایی هستند که اولاً فعالیت‌های پیش‌نیاز آن‌ها انجام شده‌اند و ثانیاً میزان نیاز آن‌ها به منابع تجدیدنشدنی از مقدار منابع تجدیدنشدنی موجود در شروع دوره زمانی تخصیص منابع به فعالیت‌ها، بیشتر نباشد. قوانین اولویت که در کدگذاری رشته دوم به کار گرفته شده‌اند، از بین بهترین قوانین اولویت انتخاب شده‌اند [۱۹ و ۲۰].

روش حل با ایجاد اولین جمعیت از کروموزوم‌ها آغاز می‌گردد. ژن‌های مربوط به کروموزوم‌ها در جمعیت اول به طور تصادفی ایجاد می‌شوند. به ازای هر منبع تجدیدنشدنی مجموع نیاز فعالیت‌های پروژه به آن منبع را محاسبه کرده، از این طریق موجه بودن هر کروموزوم را نسبت به منابع تجدیدنشدنی بررسی می‌کنیم و کروموزوم‌های موجه را در جمعیت قرار می‌دهیم. در شروع زمان‌بندی و به کمک روش مسیر بحرانی^۱، زودترین زمان شروع (est) و دیرترین زمان پایان (lft) هر یک از فعالیت‌های پروژه را به دست می‌آوریم. بازه‌ای که ابتدای آن زودترین زمان شروع و انتهای آن دیرترین زمان پایان فعالیت است را پنجره زمانی فعالیت می‌نامند و مدت اجرای هر فعالیت می‌تواند در آن جابجا شده و به اصطلاح شناور باشد.

جواب متناظر با هر کروموزوم بهترین جواب حاصل از اعمال سه رویکرد متفاوت در زمان‌بندی پروژه (تخصیص منابع به فعالیت‌ها) است. در رویکرد اول به نام زمان‌بندی پیشرو^۲، همه فعالیت‌ها در زودترین زمان از پنجره زمانی خود زمان‌بندی می‌شوند. در رویکرد دوم به نام زمان‌بندی پسرو^۳، همه فعالیت‌ها در دیرترین زمان از

1- Critical Path Method-CPM

2- Forward

3- Backward

4- Within-Window

[۲۴] یک مسأله برنامه‌ریزی پروژه با منابع محدود با تعداد ۱۲ فعالیت تولید کرده‌ایم.

جدول ۱ - فرایند زمان‌بندی پروژه بر اساس روش پیشرو

زمان شروع	فعالیت(های) واجد شرایط	مرحله	قانون اولویت	فعالیتی که برنامه‌ریزی می‌شود	زمان پایان
۰	۱	۱	۵	۱	۰
۰	۲۳۴۵	۲	۸	۲	۲
۰	۳۵	۳	۳	۳	۲
۰	-				
۲	۴۵۶۷۸	۴	۴	۶	۳
۲	۴۷۸	۵	۳	۴	۶
۲	۷۸	۶	۹	۸	۷
۲	۷	۷	۱	۷	۷
۲	-				
۳	-				
۶	-				
۷	۵۱۰۱۱	۸	۲	۱۱	۱۴
۷	۱۰	۹	۴	۱۰	۱۶
۷	-				
۱۴	۵	۱۰	۷	۵	۱۸
۱۴	-				
۱۶	-				
۱۸	۹۱۲	۱۱	۳	۹	۱۹
۱۸	۱۲	۱۲	۱	۱۲	۲۷
۱۸	-				
۱۹	۱۳	۱۳	۳	۱۳	۲۰
۱۹	-				
۲۰	-				
۲۷	۱۴	۱۴	۴	۱۴	۲۷

هر فعالیت حداکثر در سه حالت قابل اجراست. برای انجام هر فعالیت به دو منبع تجدیدشدنی و دو منبع تجدیدشدنی نیاز است. سایر داده‌های اولیه این مسأله در

فوق جهت نخبه‌پروری^۱ و لذا دسترسی به همگرایی سریع‌تر است. به تعداد جمعیت نسل اول، نمونه تصادفی (کروموزوم والد^۲) برداشته و برای انتقال به نسل بعدی در نظر می‌گیریم [۲۱]. تکثیر کروموزوم عملاً تغییر ترتیب ژن‌های آن است و کروموزوم حاصل فرزند^۳ نامیده می‌شود. پس از انتخاب والد (والدین) برای تکثیر، تنها یکی از اعمال جهش^۴ یا تزویج^۵ را به تصادف بر کروموزوم والد اعمال و از این طریق کروموزوم فرزند به وجود می‌آید. هرچند انواع مختلفی از اعمال جهش و تزویج وجود دارند، ما در این مقاله از جهش تعویضی و تزویج دو نقطه‌ای استفاده کرده‌ایم [۲۲]. در عمل جهش و تزویج، شرط موجه بودن کروموزوم را به لحاظ منابع تجدیدشدنی منظور می‌کنیم. به این ترتیب عمل تکثیر صورت می‌گیرد. به طور کلی در الگوریتم ژنتیک شرط توقف می‌تواند بر اساس تابع فاصله [۲۲] و یا تعداد نسل‌ها [۱۹] باشد. در مقاله حاضر، با توجه به آزمون‌های متعددی که بر روی ۴۳۶ مسأله آزمون استاندارد انجام داده‌ایم و با در نظر گرفتن شیوه به کار رفته در [۱۹] و ضمن رعایت نخبه‌پروری، تعداد نسل‌ها را به عنوان شرط پایان الگوریتم در نظر گرفته‌ایم. در آخرین نسل، کروموزومی که دارای کم‌ترین زمان است، به عنوان جواب نهایی انتخاب می‌شود.

مثال عددی

برای نمایش قابلیت الگوریتم طراحی شده، آن را در حل یک مثال به کار می‌گیریم. به کمک نرم‌افزار Progen

- 1- Elitism
- 2- Parent
- 3- Offspring
- 4- Mutation
- 5- Crossover

جدول ۱۱ (پیوستار مقاله) آورده شده است. برای شروع، رشته کروموزم حاوی ژن‌های مبین حالت اجرا و قاعده اولویت را به صورت تصادفی زیر در نظر می‌گیریم:

حالت : ۳ ۲ ۱ ۳ ۲ ۲ ۱ ۱ ۲ ۳ ۱ ۲
اولویت : ۸ ۳ ۴ ۳ ۹ ۱ ۲ ۴ ۷ ۳ ۱ ۳

نیاز کروموزوم بالا به منبع تجدیدنشدنی ۱ و ۲ از ۳۰ واحد تجاوز نمی‌کند.

$$(0+4+0+5+0+7+2+0+7+0+1+0+2+0=28)$$

از این رو به زمان‌بندی پروژه برحسب اطلاعات ژن‌های کروموزوم فوق، مبادرت می‌کنیم. فرایند زمان‌بندی پروژه بر اساس روش پیشرو در جدول ۱ آورده شده است. با توجه به جدول ۲ ملاحظه می‌شود که چنانچه پروژه بر مبنای روش پیشرو زمان‌بندی شود در مدت ۲۷ روز قابل تحویل است. حال محاسبات را به ازای همین کروموزوم بر مبنای روش پسرو دنبال می‌کنیم. ابتدا زودترین زمان شروع و دیرترین زمان پایان هر یک از فعالیت‌ها را به دست می‌آوریم (جدول ۲ را ملاحظه کنید).

جدول ۲- زودترین زمان شروع و دیرترین زمان پایان فعالیت‌ها

فعالیت	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴
<i>est</i>	۲۷	۲۵	۲۵	۲۱	۹	۲۴	۲۰	۲۰	۸	۱۱	۱۳	۰	۷	۰
<i>lft</i>	۲۷	۲۷	۲۷	۲۵	۱۳	۲۵	۲۵	۲۵	۹	۲۰	۲۰	۹	۸	۰

برنامه زمان‌بندی شده پروژه بر اساس روش پسرو در جدول ۳ آورده شده است. با توجه به این جدول ملاحظه می‌شود که در روش پسرو پروژه در مدت ۲۴ روز قابل تحویل است.

در جدول ۴ که زمان‌بندی را بر اساس شیوه میان-پنجره‌ای نشان می‌دهد، ملاحظه می‌شود که پروژه در مدت ۲۳ روز قابل تحویل است. بنابراین جواب متناظر با کروموزوم مفروض ۲۳ است.

الگوریتم ژنتیک را در مورد این مسئله و با در نظر گرفتن ۳۰ برای جمعیت، ۲۰ برای تعداد نسل‌ها، نرخ تزویج ۰/۹ و نرخ جهش ۰/۱ اعمال کردیم. جواب حاصل نشان داد که پروژه ۱۹ روزه قابل تحویل است. از آنجا که در منزل‌گاه مربوط به نرم‌افزار Progen و در موارد ممکن، جواب بهینه مسئله نیز وجود دارد، یادآور می‌شویم

که ۱۹، جواب بهینه این مسئله است. جدول ۵ زمان‌بندی این مسئله را نشان می‌دهد. به منظور بررسی جامع‌تر تاثیر استفاده از روش‌های پیشرو، پسرو و میان-پنجره‌ای در کیفیت عملکرد الگوریتم، از مسائل آزمون استاندارد استفاده کرده‌ایم. تعداد ۴۶۴ مسئله از ۵۳۶ مسئله آزمون استاندارد مرجع [۲۴] را با چهار شیوه به شرح زیر حل کرده‌ایم. در شیوه اول (پیشرو) جواب حاصل از اعمال شیوه پیشرو را برای کروموزوم در نظر گرفته‌ایم. در شیوه دوم (پسرو) جواب حاصل از اعمال شیوه پسرو را برای کروموزوم در نظر گرفته‌ایم. در شیوه سوم (پیشرو-پسرو) بهترین جواب حاصل از اعمال شیوه پیشرو و پسرو را برای کروموزوم در نظر گرفته‌ایم، و بالاخره در شیوه چهارم (میان-پنجره‌ای) بهترین جواب حاصل از اعمال شیوه

جدول ۴- فرایند زمان‌بندی پروژه بر اساس روش میان پنجره‌ای

زمان شروع	فعالیت(های) واجد شرایط	مرحله	قانون اولویت	فعالیتی که برنامه‌ریزی می‌شود	زمان پایان
۰	۱	۱	۵	۱	۰
۰	۳	۲	۸	۳	۲
۰	-	-	-	-	۰
۲	۲۵۷	۳	۳	۲	۴
۲	۵۷	۴	۴	۷	۷
۲	۵	۵	۳	۵	۶
۴	۸	۶	۹	۸	۹
۴	-	-	-	-	۴
۶	۴۶	۷	۱	۴	۱۰
۶	۶	۸	۲	۶	۷
۷	۱۱	۹	۴	۱۱	۱۴
۷	-	-	-	-	۷
۹	۱۰	۱۰	۷	۱۰	۱۹
۱۰	-	-	-	-	۱۰
۱۴	۱۲	۱۱	۳	۱۲	۲۳
۱۴	-	-	-	-	۱۴
۱۹	۹	۱۲	۱	۱۲	۲۰
۱۹	-	-	-	-	۱۹
۲۰	۱۳	۱۳	۳	۱۳	۲۱
۲۰	-	-	-	-	۲۰
۲۱	۱۴	۱۴	۴	۱۴	۲۳

پیشرو، پسرو و میان- پنجره‌ای را برای کروموزم در نظر گرفته‌ایم. در حل این مسائل تعداد ۲۰ نسل ۳۰ کروموزومی را در نظر گرفته و برای تکثیر از عملگرهای تزویج با نرخ ۰/۹ و جهش با نرخ ۰/۱ استفاده نموده‌ایم.

جدول ۳- فرایند زمان‌بندی پروژه بر اساس روش پسرو

زمان شروع	فعالیت(های) واجد شرایط	مرحله	قانون اولویت	فعالیتی که برنامه‌ریزی می‌شود	زمان پایان
۰	۱۴	۱	۴	۱۴	۰
۰	۶۱۰۱۲۱۳	۲	۳	۶	۱
۰	۱۰۱۲۱۳	۳	۱	۱۳	۱
۰	۱۰۱۲	۴	۳	۱۲	۹
۰	-	-	-	-	۰
۱	۲۹۱۰	۵	۷	۹	۲
۱	-	-	-	-	۱
۲	۲۱۰	۶	۴	۲	۴
۲	-	-	-	-	۲
۴	۱۰	۷	۲	۱۰	۱۳
۴	-	-	-	-	۴
۹	۵۱۱	۸	۱	۱۱	۱۶
۹	-	-	-	-	۹
۱۳	۴۸	۹	۹	۴	۱۷
۱۳	۸	۱۰	۳	۸	۱۸
۱۳	-	-	-	-	۱۳
۱۶	۷	۱۱	۴	۷	۲۱
۱۶	-	-	-	-	۱۶
۱۷	-	-	-	-	۱۷
۱۸	۵	۱۲	۳	۵	۲۲
۱۸	-	-	-	-	۱۸
۲۱	-	-	-	-	۲۱
۲۲	۳	۱۳	۸	۳	۲۴
۲۲	-	-	-	-	۲۲
۲۴	۱	۱۴	۵	۱	۲۴

جدول ۵- زمان‌بندی بهینه پروژه

فعالیت	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴
حالت	۱	۳	۲	۱	۱	۲	۱	۱	۱	۱	۳	۱۲	۲	۱
زمان شروع	۰	۰	۰	۳	۲	۷	۲	۲	۱۰	۷	۳	۱۰	۱۱	۱۹

همانند جدول قبلی، در این نوع مسائل بزرگ و نسبتاً واقعی نیز استفاده از شیوه میان- پنجره‌ای به بهبود قابل توجه الگوریتم ژنتیک طراحی شده منجر شده است. در واقع شیوه میان- پنجره‌ای توانسته است عملکرد الگوریتم را در مقایسه با روش‌های پیشرو، پسرو و پیشرو- پسرو به ترتیب به میزان ۳۱۲٪، ۲۶/۶٪ و ۱۴/۳٪ بهبود ببخشد. در ادامه مقاله به بررسی تاثیر برخی از پارامترهای الگوریتم ژنتیک می‌پردازیم.

جدول ۸- تأثیر اندازه جمعیت در کیفیت عملکرد چهار

شیوه حل

حجم جمعیت	۳۰	۶۰	۱۰۰
M_1	۳/۳۱	۲/۶۵	۱/۹۹
M_2	۰/۴	۰/۵۴	۰/۷۱
N_0	۴۱	۷۰	۱۰۳

N_0 : تعداد مسائلی است که الگوریتم توانسته است

آن‌ها را به صورت بهینه حل کند.

با استفاده از همان ۴۶۴ مسأله آزمون استاندارد مرجع [۲۴] به بررسی تاثیر برخی از پارامترهای الگوریتم پرداخته‌ایم. در این مسائل از ترویج دو نقطه‌ای و جهش تعویضی استفاده شده است. جدول ۸ تاثیر اندازه جمعیت (S) را بر نتایج حاصل از اجرای الگوریتم نشان می‌دهد. در این آزمایشات تعداد نسل‌ها ۲۰، نرخ ترویج ۰/۹ و نرخ جهش ۰/۱ در نظر گرفته شده‌اند.

جدول ۶- کیفیت عملکرد چهار شیوه حل در بررسی

۴۶۴ مسأله آزمون استاندارد

شیوه حل	پیشرو	پسرو	پیشرو- پسرو	میان- پنجره‌ای
M_1	۴/۰۲	۳/۶۹	۳/۳۱	۲/۷۵
M_2	۰/۲۷	۰/۲۵	۰/۴۰	۰/۵۶

M_1 : میانگین اختلاف جواب‌های حاصل از الگوریتم ژنتیک با

جواب‌های بهینه

M_2 : میانگین زمان صرف شده پردازشگر برای اجرای الگوریتم

با توجه به جدول ۶ ملاحظه می‌شود که با استفاده از شیوه میان- پنجره‌ای کیفیت جواب‌های حاصل ارتقاء پیدا کرده و مثلاً در مقایسه با شیوه پیشرو تقریباً ۳۲٪ و در مقایسه با شیوه پیشرو-پسرو حدوداً ۱۷٪ بهبود یافته است. برای بررسی بیشتر این وضعیت، به کمک برنامه Progen ۳۰ مسأله ۹۰ فعالیتی با ۲ منبع تجدیدشدنی و ۲ منبع تجدیدنشده که در آن هر فعالیت ممکن است در یکی از ۲ حالت انجام شود، در نظر گرفته‌ایم (سایر پارامترها مثل بررسی قبلی است). نتیجه بررسی این مسایل را در جدول ۷ آورده‌ایم.

جدول ۷- کیفیت عملکرد چهار شیوه حل در بررسی ۳۰

مسأله ۹۰ فعالیتی

شیوه حل	پیشرو	پسرو	پیشرو-پسرو	میان-پنجره‌ای
M_1	۳۲/۳۳	۳۰/۳۴	۲۵/۹۶	۲۲/۲۶
M_2	۲۱/۱۳	۲۱/۹۲	۳۹/۱۷	۴۳/۷۸

نتیجه گیری

در این مقاله یک روش حل مبتنی بر الگوریتم ژنتیک برای یکی از مسایل پیچیده زمان بندی ارائه گردیده است. در این روش کروموزومها به منظور تولید جوابهای موجه اولیه کدبندی شدهاند. سه روش متفاوت برای حصول جواب بهتر به ازای هر کروموزوم طراحی شده است. با استفاده از مسایل آزمون استاندارد تاثیر هر یک از این سه روش در بهسازی عملکرد الگوریتم، بررسی شده است. نتایج حاصل نشان می دهد که روش میان- پنجره ای نسبت به روش های پیشرو، پسرو و پیشرو- پسرو عملکرد بهتری دارد.

جدول ۱۰- تأثیر نرخ تکثیر و توزیع در کیفیت عملکرد

چهار شیوه حل

نرخ	۰/۰۰۵	۰/۰۱۰	۰/۰۵۰	۰/۱۰۰	۰/۱۰۰	۰/۱۰۰
توزیع و جهش	و	و	و	و	و	و
	۰/۹۰۰	۰/۹۰۰	۰/۹۰۰	۰/۸۰۰	۰/۷۰۰	۰/۷۰۰
M_1	۴/۶۸	۴/۱۶	۴/۰۷	۳/۳۱	۳/۲۱	۳/۱۱
M_2	۰/۱۸	۰/۲۲	۰/۳۸	۰/۴۰	۰/۳۷	۰/۴۶

قدردانی

مؤلفین از داوران محترم که با نظرات سازنده خود موجب ارتقای کیفیت مقاله گردیده اند قدردانی می نمایند.

پیوستار

در این پروژه از منبع تجدیدشدنی ۱، ۸ واحد در هر واحد زمان و از منبع تجدیدشدنی ۲، ۶ واحد در هر واحد زمان در دسترس است. ضمناً از منبع تجدیدنشده ۱، جمعاً ۳۰ واحد، و از منبع تجدیدشدنی ۲، کلاً ۳۰ واحد در اختیار پروژه است.

با توجه به جدول ۸، همان طور که انتظار می رفت با افزایش جمعیت هم کیفیت جوابها ارتقاء پیدا کرده و هم جواب بهینه تعداد بیشتری از مسائل آزمون به دست آمده اند. در واقع وقتی حجم جمعیت از ۳۰ به ۱۰۰ افزایش می یابد، بهبود ۴۰٪ در کیفیت جوابها مشاهده می شود (۱/۹۹ در مقایسه با ۳/۳۱) و همچنین تعداد جوابهای بهینه حاصل شده از ۴۱ به ۱۰۳ ارتقاء می یابد. تاثیر تعداد نسلها در کیفیت عملکرد الگوریتم را در جدول ۹ بررسی کرده ایم. حجم جمعیت را ۳۰ در نظر گرفته و سایر پارامترها را مثل جدول ۸ فرض کرده ایم.

جدول ۹- تأثیر تعداد نسلها در کیفیت عملکرد چهار

شیوه حل

تعداد نسلها	۲۰	۳۰	۴۰
M_1	۳/۳۱	۲/۷۰	۱/۹۵
M_2	۰/۴	۰/۴۱	۰/۵۹
N_0	۴۱	۴۵	۵۵

همان طور که از جدول ۹ پیداست با افزایش تعداد نسلها جوابها نیز بهبود می یابد، هرچند نرخ بهینه سازی در مقایسه با افزایش حجم جمعیت از شتاب کمتری برخوردار است.

حال به بررسی تأثیر نرخ به کارگیری هر یک از اعمال تکثیر یعنی توزیع و جهش می پردازیم. در جدول ۱۰ فقط قسمتی از نتایج به دست آمده را گزارش کرده ایم. از این جدول پیداست که در این الگوریتم با افزایش نرخ توزیع وقتی نرخ جهش ثابت در نظر گرفته می شود جوابها بهبود می یابند. از طرفی همین وضعیت با کاهش نرخ جهش مشاهده می شود.

resource constraints: Classification and complexity. Discrete Applied Mathematics, 5 (1983) 11-24.

- [2] Wiest, J.D., The scheduling of large projects with limited resources. Ph.D. dissertation, Carnegie Institute of Technology, (1963).
- [3] Kelly, J.E., Jr. The critical path method: Resource planning and scheduling. Ch. 21 in Industrial Scheduling , J.F. Muth and G.L. Thompson (eds.), Prentice-Hall, Englewood Cliffs, NJ, 347-365.
- [4] Deckro, R.F., Winkofsky, E.P., Hebert, J.E. and Gagnon, R., A decomposition approach to multi-project scheduling. European Journal of O.R., 51 (1991) 110-118.
- [5] Icmeli, O. and Rom, W.O., Solving the resource-constrained project scheduling problem with optimization subroutine library. Computers and Operations Research, 23 (1996) 801-817.
- [6] Carruthers, J.A. and Battersby, A., Advances in critical path methods. Operational Research Quarterly, 17 (1966) 359-380.
- [7] Petrovic, R., Optimisation of resource allocation in project planning. Operations Research, 16 (1968) 559-586.
- [8] Demeulemeester, E. and Herroelen, W., New benchmark results for the resource-constrained project scheduling problem. Management Science, 43 (1997) 1485-1492.

جدول ۱۱- داده‌های اولیه مثال عددی

فعالیت	پیش‌نیاز(ها)	حالت	زمان	R_1	R_2	NR_1	NR_2
۱	-	۰	۰	۰	۰	۰	۰
۲	۱	۲	۲	۴	۰	۰	۰
۳	۱	۲	۲	۰	۳	۰	۰
۴	۱	۱	۴	۵	۰	۰	۰
۵	۱	۲	۲	۰	۱	۳	۰
۶	۲	۲	۱	۰	۷	۰	۰
۷	۳	۲	۵	۰	۳	۱	۰
۸	۳	۱	۵	۰	۱	۲	۰
۹	۴ ۸ ۱۱	۱	۱	۰	۷	۳	۰
۱۰	۴ ۸	۲	۹	۰	۲	۲	۰
۱۱	۷	۲	۱۱	۰	۰	۱	۱۴
۱۲	۵ ۱۱	۱	۹	۰	۲	۳	۰
۱۳	۹	۲	۱	۰	۳	۲	۰
۱۴	۶ ۱۰ ۱۲ ۱۳	۱	۰	۰	۰	۰	۰

R_1 : نیاز فعالیت به منبع تجدیدشدنی ۱

R_2 : نیاز فعالیت به منبع تجدیدشدنی ۲

NR_1 : نیاز فعالیت به منبع تجدیدنشدنی ۱

NR_2 : نیاز فعالیت به منبع تجدیدنشدنی ۲

مراجع

- [1] Blazewics, J., Lenstra, J.K. and Rinnooy Kan, A.H.G., Scheduling projects to

- [17]Hermann, J.W. and Lee, C.Y., Solving a class scheduling problem with a genetic algorithm. Dept. Ind. Syst. Eng. Univ. Florida, (1993).
- [18]Portmann, M.C., Genetic algorithms and scheduling: A state of the art and some propositions. In proceedings Workshop Production Planning Contr., Mons., (1996).
- [19]Ozdamar, L., A Genetic Algorithm Approach to General Category Project Scheduling Problem. IEEE Transactions on Systems and Cybernetics, 29, 1 (1999).
- [20]Ulusoy, G. and Ozdamar, L. A constraint based perspective in resource constrained project scheduling. Int. J. Prod. Res., 32 (1994) 693-705.
- [21]Ross, M., S. Simulation. Academic Press, (1996).
- [22]Mitsuo, G. and Renwe, C., Genetic Algorithms and Engineering Optimization. New York: John Wiley, (2000).
- [23]Bartschi Wall, M., A Genetic Algorithm for Resource-constrained Scheduling. PhD Thesis submitted to the department of Mechanical Engineering. MIT. June, (1996).
- [24]Kolisch, R., Sprecher, A. and Drexl, A., Characterization and Generation of a General Class of Resource-constrained Project Scheduling Problems. Management Science, 41 (1995) 1693-1703.
- [9] Brucker, P., Schoo, A. and Thiele, O., A branch and bound algorithm for the resource constrained project scheduling problem. European Journal of O.R., 17 (1998) 143-158.
- [10]Herroelen, W., De Reyck, B. and Demeulemeester, E., Resource-constrained project scheduling: A survey of recent developments. Computers Ops Res., 25, 4 (1998) 279-302.
- [11]Kolisch, R. and Drexl, A., Local search for nonpreemptive multimode resource-constrained project scheduling. Christian-Albrechts Universitat zu Kiel, Kiel, Germany, (1994).
- [12]Chan, W. and Chua, D., Construction Resource Scheduling with Genetic Algorithms. Journal of Construction Engineering and Management, 122, 2 (1996) 125-132.
- [13]Senonci, A.B. and Eldin, N.N., Use of genetic algorithm in resource scheduling of construction projects. Journal of Construction Engineering and Management. (2004) 869-877.
- [14]Boctor, F.F., Some efficient multi-heuristic procedures for resource-constrained project scheduling. European Journal of O.R., 49 (1990) 3-13.
- [15]Drexl, A. and Grunewald, J., Nonpreemptive multi-mode resource constrained project scheduling. IIE Trans., 25 (1993) 74-81.
- [16]Biegel, J.,E. and Davern, J.J., Genetic algorithms and job-shop scheduling. Comput. Ind. Eng., 19 (1990) 81-91.