مهندسی صنایع و مدیریت شریف
دوره‌ی ۱ - ۲۶، شماره‌ی ۱، ص. ۱۰۱-۱۱۵

زمان‌بندی کارکارگاهی انعطاف‌پذیر با درنظرگرفتن محدودیت نگه‌داری و تعمیرات (نت)

سید حسام‌الدین دگردی (دانشیار)
مهدی رحیمی قشقائی (کارشناس ارشد)
دانشکده‌ی فنی و مهندسی، دانشگاه تربیت مدرس

در این نوشتار، مسئله‌ی زمان‌بندی کارکارگاهی انعطاف‌پذیر[1] با وجود انعطاف‌پذیری عملیات و در نظرگرفتن محدودیت نگه‌داری و تعمیرات (نت) مورد بررسی قرار گرفته است. در اکثر مدل‌های زمان‌بندی کارکارگاهی انعطاف‌پذیر فرض بر این است که ماشین‌ها در تمام اوقات در دسترس‌اند. اما در محیط‌های صنعتی واقعی ممکن است ماشین‌ها به‌دلایل مختلف ـ مانند نگه‌داری و تعمیرات پیشگیرانه ـ در دسترس نباشند. در این تحقیق محدودیت دسترسی به ماشین‌ها از نوع قطعی و تحت عنوان محدودیت نت به صورت دسترسی نامعین[2]، مجاز بودن انقطاع عملیات[3] و سیاست پردازش از سرگیری مجدد کار[4] بررسی شده است. براین اساس، یک مدل ریاضی از نوع برنامه‌ریزی غیر خطیِ عدد صحیح مختلط برای مسئله‌ی تحقیق ارائه شده است. نتایج محاسبات مؤید کارایی روش پیشنهادی برای حل مسئله‌ی تحقیق است.

واژگان کلیدی: زمان‌بندی کارکارگاهی انعطاف‌پذیر، دسترسی به ماشین، نگه‌داری و تعمیرات (نت)، الگوریتم ژنتیک، الگوریتم ابتکاری.

zegordi@modares.ac.ir
mahdi.rahimi.gh@gmail.com

مقدمه

امروزه محیط‌های تولید انعطاف‌پذیر، با دوره‌ی عمرکوتاه و افزایش تنوع در محصول شناخته می‌شوند. در چنین محیط‌هایی، انعطاف‌پذیری تولید به‌عنوان یک سلاح رقابتیِ کلیدی مطرح است و سبب افزایش توانایی کارخانه در پاسخ به تغییرات غیر قابل پیش‌بینی ـ به‌عنوان عامل اصلی فشار رقابتی ـ می‌شود. در سال ۱۹۸۲ انعطاف‌پذیری به‌عنوان «توانایی یک سیستم در سازگارشدن با تغییرات گوناگون» تعریف شد.[1] همچنان که در سال ۱۹۹۰ نیز انعطاف‌پذیری را توانایی یک سیستم در مقابله با تغییرات عنوان کردند.[2]

در برنامه‌ریزی کف کارگاهی، هدف اصلی یافتن زمان‌بندی ممکن است به‌گونه‌یی که هزینه‌ی تولید را با در نظرگرفتن کلیه‌ی محدودیت‌های ممکن، کمینه کند. کار کارگاهی انعطاف‌پذیر یکی از رایج‌ترین سیستم‌ها در تولید قطعات متنوع است که از ترکیب سیستم کارکارگاهی و متغیرهای انعطاف‌پذیری کف کارگاه به وجود می‌آید. زمان‌بندی برای مسئله‌ی کارکارگاهی انعطاف‌پذیر، در زمینه‌ی بهینه‌سازی برنامه‌ریزی تولید بسیار حائز اهمیت است.[3]

در تولید صنعتی، سازگاری سیستم‌های تولید هوشمند با نوسانات بازار (نظیر تقاضای تصادفی) و اختلالات درونی (مانند ازکارافتادگی ماشین‌ها) امری ضروری است. ماشین‌ها باید همزمان و به‌طور مؤثر قادر به تولید چندین نوع محصول باشند. بنابراین، برنامه‌ریزی و زمان‌بندی بهینه‌ی تولید و کنترل ماشین‌ها، برای سرمایه‌گذاران و تولیدکنندگان، بسیار اهمیت دارد. تحت این شرایط، امروزه مهم‌ترین مسئله در زمینه‌ی بهینه‌سازی سیستم‌های تولیدی سیاست‌های نگه‌داری و تعمیرات، قاعده‌ی زمان‌بندی، و تخصیص قطعات به ماشین‌ها است. در اکثر مدل‌های زمان‌بندی کار کارگاهی انعطاف‌پذیر فرض بر این است که ماشین‌ها در تمام اوقات در دسترس‌اند. اما در محیط‌های صنعتی واقعی، ممکن است ماشین به‌دلایل مختلف در طول افق برنامه‌ریزی در دسترس نباشد. به‌طور عام، عدم دسترسی به ماشین ممکن است تصادفی ـ نظیر خرابی یا ازکارافتادگی ماشین ـ یا قطعی ـ نظیر نگه‌داری و تعمیرات پیشگیرانه، تعمیرات اساسی و زمان‌بندی‌های قبلی ـ باشد.[4]

نوشتار حاضر در قالب هشت بخش تهیه شده است. پس از ذکر مقدمه و ضرورت تحقیق، ادبیات مسئله و پیشینه‌ی تحقیق در دو زمینه‌ی کار کارگاهی انعطاف‌پذیر و محدودیت دسترسی قطعی به ماشین ارائه می‌شود. سپس به تعریف مسئله‌ی تحقیق پرداخته و مدل‌سازی ریاضی آن بیان می‌شود. در ادامه، ضمن بیان رویکرد حل مسئله، روشی ترکیبی با استفاده از الگوریتم ژنتیک و روش ابتکاری برای حل آن ارائه خواهد شد. پس از آن آزمایش‌هایی عددی طراحی می‌شود و نتایج حاصل از آن به‌منظور نشان‌دادن کارایی روش پیشنهادی بررسی خواهد شد. در پایان نتایج و دستاوردهای تحقیق ذکرشده و زمینه‌های مناسب برای انجام تحقیقات آتی پیشنهاد خواهد شد.

تاریخ: دریافت ۱۳۸۷/۲/۲۹، داوری ۱۳۸۷/۱۰/۲۵، پذیرش ۱۳۸۸/۷/۱.

مرور ادبیات و پیشینه‌ی تحقیق

مسئله‌ی کارکارگاهی انعطاف‌پذیر

کارکارگاهی انعطاف‌پذیر برخلاف کارکارگاهی، از ادبیات غنی و جامعی برخوردار نیست؛ این مسئله اولین‌بار در سال ۱۹۹۰ مطرح شد. کارکارگاهی انعطاف‌پذیر، جزء سخت‌ترین مسائل بهینه‌سازی ترکیبی بوده و NP-hard است. بنابراین به‌دلیل پیچیدگی محاسباتی، دست‌یابی به جواب بهینه‌ی آن با روش‌های حل متداول بهینه‌سازی بسیار دشوار است.[۵] به‌منظور حل این دسته از مسائلِ زمان‌بندی، دو رویکرد «سلسله‌مراتبی» و «جامع» وجود دارد. در رویکرد سلسله‌مراتبی دو زیرمسئله‌ی «تخصیص ماشین» و «تعیین توالی عملیات»، به‌صورت مجزا در نظر گرفته می‌شوند. اما در رویکرد جامع این دو زیرمسئله به‌طور همزمان در نظرگرفته شده و حل می‌شوند.[۶]

در سال ۱۹۹۳ برای اولین‌بار از این روش تجزیه برای مسئله‌ی کارکارگاهی انعطاف‌پذیر استفاده شد.[۷] در سال ۲۰۰۱ نیز برای حل مسئله‌ی کارکارگاهی انعطاف‌پذیر با/بدون در نظرگرفتن انقطاع عملیات الگوریتمی چندجمله‌یی ارائه شد.[۸] پس از آن، در سال ۲۰۰۲، برای حل مسئله‌ی کارکارگاهی انعطاف‌پذیر با چرخش مجدد کار از قواعد اولویتی مرکب بهره گرفته شد.[۹] در همان سال یک الگوریتم ژنتیک ارائه شد و از آن برای حل مسئله‌ی کارکارگاهی انعطاف‌پذیر با تابع هدف یگانه و چندگانه استفاده شد.[۱۰] در سال ۲۰۰۵ یک روش سلسله‌مراتبی کارا به‌منظور حل مسئله‌ی کارکارگاهی انعطاف‌پذیر با تابع هدف چندگانه پیشنهاد شد.[۶]

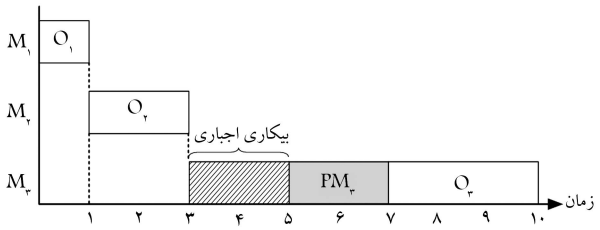
در سال ۱۹۹۴ الگوریتمی فراابتکاری برای جست‌وجوی ممنوع با روش یک‌پارچه ارائه شد.[۱۱] در سال ۲۰۰۲ نیز ضمن توسعه‌ی روش جست‌وجوی ممنوع، دو تابع همسایگی برای حل مسئله ارائه شد.[۱۲] در سال ۲۰۰۳ محققین با استفاده از روش جست‌وجوی هوش مصنوعی، روشی جدید به‌نام «الگوریتم تکاملی همزیستی[۵]» ارائه کردند.[۱۳] پس از آن در سال ۲۰۰۵، روشی برای حل مسئله‌ی زمان‌بندی انباشته اقتصادی چندمحصولی با دوره‌ی عمومی، درکارکارگاهی انعطاف‌پذیر بیان شد.[۳] در سال ۲۰۰۶ مسئله‌ی کارکارگاهی انعطاف‌پذیر با در نظرگرفتن اثر یادگیری مطرح شد و محققین به کمک الگوریتم ژنتیک به حل مسائل مطروحه پرداختند.[۱۴] در همان سال، با در نظرگرفتن زمان‌های آماده‌سازی وابسته به توالی عملیات، از الگوریتم جست‌وجوی ممنوع دومرحله‌یی به‌منظور حل مسائل استفاده شد.[۱۵] ازسوی دیگر، در این سال با در نظرگرفتن زمان‌های حمل‌ونقل و آماده‌سازی وابسته به توالی عملیات از الگوریتم بهینه‌سازی اجتماع مورچگان به‌منظور حل مسائل بهره گرفته شد.[۱۶] در سال ۲۰۰۷ برای حل مسئله‌ی کارکارگاهی انعطاف‌پذیر از الگوریتم ژنتیک با راهکارهای حل ترکیبی استفاده کردند.[۱۷] این مسئله با تابع هدف چندگانه و به‌کمک الگوریتم ژنتیک ترکیبی و روش جست‌وجوی محلی نیز حل شده است.[۱۸]

محدودیت دسترسی قطعی به ماشین

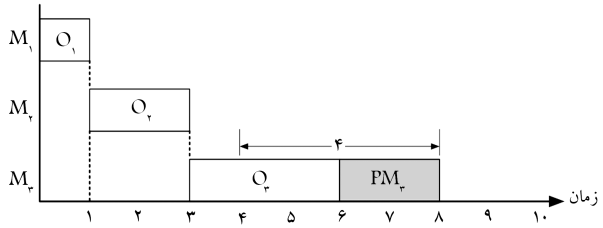
از جمله محدودیت‌هایی که در ادبیات مسائل زمان‌بندی وجود دارد، محدودیت دسترسی به ماشین‌ها است. بدین‌معنی که دسترسی مداوم به ماشین‌ها، در سراسر دوره‌ی زمان‌بندی مقدور نیست. غالباً هنگام برنامه‌ریزی و زمان‌بندی تولید، این مسئله مورد غفلت واقع شده یا به‌دلیل دشواری مسئله‌ی زمان‌بندی، ترجیح داده می‌شود محدودیت دسترسی به ماشین‌ها در زمان‌بندی اعمال نشود. اهمیت برنامه‌ریزی با در نظرگرفتن این محدودیت در محیط‌هایی آشکار می‌شود که عدم دسترسی به ماشین‌ها در آن هزینه‌های زیادی به سیستم تولید تحمیل می‌کند. بنابراین لزوم در نظرگرفتن محدودیت دسترسی در مدل‌های واقع‌گرایانه‌ی زمان‌بندی کاملاً واضح است.

محدودیت‌های دسترسی قطعی در سیستم‌های تولید را عموماً می‌توان به دو دسته تقسیم کرد: محدودیت‌های دسترسی معین[۶] و محدودیت‌های دسترسی نامعین. در محدودیت دسترسی معین دوره‌ی عدم دسترسی از یک نقطه‌ی ثابت زمانی شروع، و به یک نقطه‌ی زمانی معین ختم می‌شود. در محدودیت دسترسی نامعین، زمان شروع دوره‌ی عدم دسترسی نامعین است. به‌عبارت دیگر دوره‌ی عدم دسترسی انعطاف‌پذیر فرض می‌شود و در طول فرایند زمان‌بندی تعیین می‌گردد. البته فعالیت نگه‌داری و تعمیرات (نت) باید در پنجره‌های زمانی مشخصی انجام شود.[۱۹] در شکل ۱ فعالیت ماشین سوم «نت با محدودیت دسترسی معین» است. در این حالت امکان انجام عملیات سوم کار، پیش از انجام فعالیت نت وجود ندارد. اما در شکل ۲ ماشین سوم دارای فعالیت نت با محدودیت دسترسی نامعین است. به‌دلیل وجود پنجره زمانی انعطاف‌پذیر (با چهار واحد زمانی) برای انجام فعالیت نت، مرحله‌ی سوم کار را می‌توان پیش از انجام فعالیت نت تکمیل کرد. چنان که ملاحظه می‌شود، عملیات سوم کار به دلیل وجود بازه زمانی منعطف در حالت دوم، زودتر از حالت اول تکمیل می‌شود. این امر نشان‌گر مزیت نسبی وجود پنجره زمانی انعطاف‌پذیر است و سبب افزایش قدرت تصمیم‌گیری در حین زمان‌بندی می‌شود.

در ادبیات مسائل زمان‌بندی با در نظرگرفتن محدودیت دسترسی به ماشین‌ها، کارها ممکن است ازویژگی‌های خاصی برخوردار باشند و در حین پردازش وضعیت متفاوتی بروز دهند. در حالت کلی و با توجه به این محدودیت، برای انجام یک عملیات سه شیوه وجود دارد: مجاز بودن انقطاع عملیات، انقطاع جزئی عملیات[۷] و عدم مجازبودن انقطاع عملیات[۸]. مجاز بودن انقطاع عملیات بدین معناست که امکان قطع‌کردن عملیات پردازش روی ماشین، به‌منظور انجام عملیاتی دیگر ـ یا نگه‌داری و تعمیرات (نت) ـ وجود دارد و عملیات می‌تواند بدون هیچ جریمه‌یی بعداً ادامه یابد.[۲۰] در انقطاع جزئی عملیات چنانچه کار در یک بازه زمانی معین انجام شود انقطاع مجاز نیست، ولی اگر در خارج از این بازه زمانی انجام شود ممکن است انقطاع داشته باشد.[۲۱] در حالت مجازنبودن انقطاع عملیات، عملیات شروع شده روی یک ماشین باید تا زمان تکمیل، بدون وقفه، ادامه یابد و شروع عملیاتی دیگر یا



شکل ۱. وجود محدودیت دسترسی معین برای ماشین سوم.



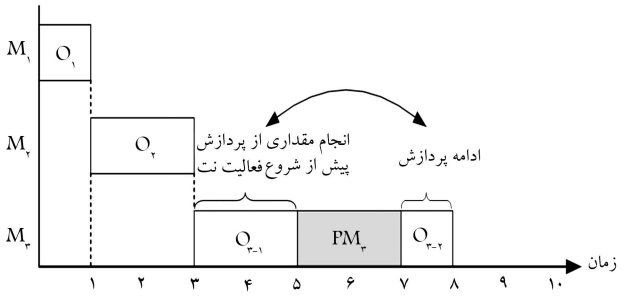
شکل ۲. وجود محدودیت دسترسی نامعین برای ماشین سوم.

انجام فعالیت نت روی آن ماشین، پیش از تکمیل این عملیات مجاز نیست.[۲۲]

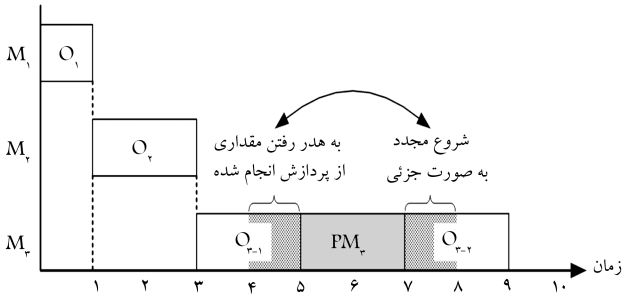
اگر انقطاع کل عملیات یا انقطاع جزئی از آن مجاز باشد و پردازش کار بر روی ماشین، پیش از شروع دوره‌ی عدم دسترسی ماشین پایان نیابد، به‌منظور ادامه‌ی پردازش سه سیاست وجود خواهد داشت: ادامه‌ی پردازش کار[۹]، ادامه‌ی پردازش جزئی کار[۱۰]، پردازش از سرگیری مجدد کار. در ادامه‌ی پردازش کار، پس از این که ماشین بدون اتلاف زمان یا جریمه‌ی از سرگیری مجدد کار در دسترس قرار گرفت، پردازش کار ادامه می‌یابد. این نوع شیوه‌ی پردازش در شکل ۳ نشان داده شده است.

در ادامه‌ی پردازش جزئی کار، هنگامی که ماشین در دسترس قرار گرفت، کار پردازش به‌صورت جزئی (از میانه و نه از ابتدا) شروع می‌شود.[۲۳] بدین معنا که می‌بایست درصدی از پردازش‌هایی که پیش از دوره‌ی عدم دسترسی ماشین بر روی کار انجام شده بود مجدداً انجام شود. این امر سبب تضییع مقداری از پردازش‌های قبلی خواهد شد (شکل ۴). در پردازش از سرگیری مجدد کار، هنگامی که ماشین در دسترس قرار گرفت، کار پردازش را از ابتدا شروع می‌کند و همه‌ی پردازش‌های قبلی به هدر می‌رود.[۴] در شکل ۵ امکان انقطاع عملیات و از سرگیری مجدد کار پس از فعالیت نشان داده شده است.

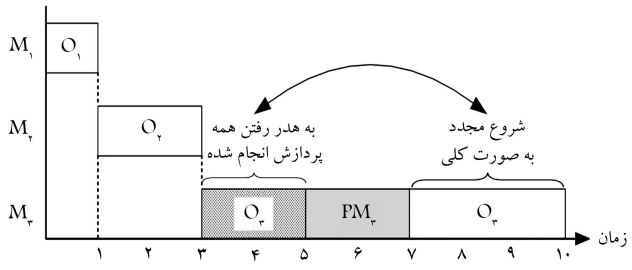
مسئله‌ی زمان‌بندی با در نظرگرفتن محدودیت دسترسی به ماشین‌ها اولین بار در سال ۱۹۸۴ مطرح، و مسئله‌ی زمان‌بندی ماشین‌های موازی با دوره‌های عدم دسترسی مختلف به ماشین‌ها و انقطاع عملیات بررسی شد.[۲۴] در سال ۱۹۸۹ محققین نشان دادند که مسئله‌ی زمان‌بندی تک‌ماشینه با مشخص‌بودن دوره‌ی عدم دسترسی به ماشین و سیاست تولید نوع N-R جزء مسائل NP-hard است.[۲۵] سپس در سال ۱۹۹۱، مسئله‌ی زمان‌بندی ماشین‌های موازی با وجود حداکثر یک دوره عدم دسترسی برای هر ماشین مطرح شد.[۲۶] در سال ۱۹۹۹ مسئله‌ی زمان‌بندی جریان کارگاهی دوماشینه با در نظرگرفتن محدودیت دسترسی به ماشین‌ها و سیاست تولید نوع R بیان شد.[۲۷] در سال ۲۰۰۰ مسائل زمان‌بندی تک‌ماشینه، ماشین‌های موازی، و جریان کارگاهی با در نظرگرفتن محدودیت دسترسی به ماشین‌ها توسعه یافت.[۲۸] در سال ۲۰۰۱ مسئله‌ی زمان‌بندی جریان کارگاهی دوماشینه با در نظر



شکل ۳. مجازبودن انقطاع عملیات و سیاست ادامه‌ی پردازش کار.



شکل ۴. مجازبودن انقطاع عملیات و ادامه‌ی پردازش جزئی کار.



شکل ۵. مجازبودن انقطاع عملیات و پردازش از سرگیری مجدد کار.

گرفتن عدم انتظارکار و محدودیت دسترسی به ماشین‌ها با سیاست تولید نوع R و N-R ارائه شد. محققین برای حل مسئله‌ی خود از یک الگوریتم تقریبی استفاده کردند.[۲۹] آنان در سال ۲۰۰۲ مسئله‌ی زمان‌بندی جریان کارگاهی انعطاف‌پذیر دومرحله‌یی را با وجود ماشین‌های موازی و با در نظرگرفتن محدودیت دسترسی به ماشین‌ها و سیاست تولید نوع R ارائه کردند و به‌منظور حل مسئله‌ی خود از روش شاخه و کران بهره گرفتند.[۳۰] در سال ۲۰۰۳ مسئله‌ی زمان‌بندی جریان کارگاهی با در نظرگرفتن عدم انتظارکار و اعمال محدودیت روی هر دو ماشین ارائه شد.[۳۱] پس از آن در سال ۲۰۰۴، مسئله‌ی زمان‌بندی جریان کارگاهی دومرحله‌یی با ماشین‌های موازی را ارائه کردند که در آن حداکثر یک دوره‌ی عدم دسترسی برای هر ماشین وجود داشت و سیاست تولید نوع R و N-R بود. آنها مسئله‌ی خود را با دو روش شاخه و کران و الگوریتم ترکیبی ابتکاری حل کردند.[۴] در همان سال محققین مسئله‌ی عمومی زمان‌بندی جریان کارگاهی را با اعمال محدودیت روی هر یک از ماشین‌ها بررسی کردند.[۲۰] در سال ۲۰۰۵ مسئله‌ی زمان‌بندی جریان کارگاهی دوماشینه با در نظرگرفتن عدم انتظارکار و اعمال محدودیت روی هر ماشین ارائه، و به‌کمک یک الگوریتم تقریبی حل شد.[۳۲] در همان سال مسئله‌ی زمان‌بندی جریان کارگاهی (دومرحله‌یی) با ماشین‌های موازی و با در نظرگرفتن محدودیت دسترسی به ماشین‌ها و سیاست تولید نوع R ارائه شد و برای حل آن از الگوریتم تقریبی استفاده شد.[۳۳] در سال ۲۰۰۶ زمان‌بندی تک‌ماشینه با در نظرگرفتن اثر یادگیری و محدودیت دسترسی به ماشین‌ها با سیاست تولید نوع R و N-R مطرح شد. محققین برای حل مسئله‌ی خود از الگوریتم ابتکاری و روش شاخه و کران استفاده کردند.[۳۴] در سال ۲۰۰۷ مسئله‌ی زمان‌بندی کارگاهی با ماشین‌های چندمنظوره و وجود محدودیت دسترسی به ماشین‌ها با سیاست تولید نوع R ارائه شد که در آن برای حل مسئله از الگوریتم ابتکاری و جست‌وجوی ممنوع استفاده شد.[۳۵]

در کارهای معدودی، مسئله‌ی زمان‌بندی با محدودیت دسترسی نامعین بیان شده است. در سال ۱۹۹۹ محققین نشان دادند که مسئله‌ی تک‌ماشینه با وجود بازه‌های عدم دسترسی متعدد NP-hard است. آنها مسئله‌ی خود را با سیاست تولید نوع N-R و به‌کمک روش شاخه و کران حل کردند.[۳۶] در همان سال مسئله‌ی تک‌ماشینه با دوره‌ی عدم دسترسی نامعین و رژیم تولید نوع S-R نیز مورد مطالعه قرار گرفت.[۳۷] پس از آن در سال ۲۰۰۰، مسئله‌ی زمان‌بندی ماشین‌های موازی با سیاست تولید نوع N-R بررسی شد که در آن برای حل مسئله از الگوریتم چندجمله‌یی، شبه چندجمله‌یی و شاخه و کران بهره گرفته شد.[۳۸] در سال ۲۰۰۴ محققین مسائل زمان‌بندی تولید و نت را در محیط جریان کارگاهی ترکیب کردند.[۲۲] در سال ۲۰۰۶ مسئله‌ی زمان‌بندی تک‌ماشینه با در نظرگرفتن محدودیت دسترسی به ماشین‌ها و سیاست تولید نوع N-R بیان شد و برای حل آن از الگوریتم تقریبی بهره گرفته شد.[۳۹] در همان سال، محققین مسئله‌ی زمان‌بندی کارگاهی را با

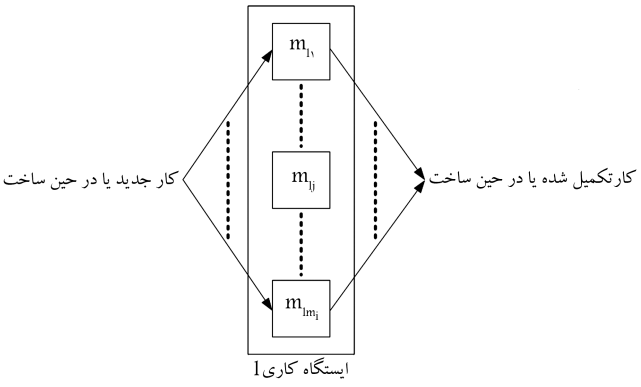
امکان یک دوره عدم دسترسی برای هر ماشین و رژیم تولید نوع N-R ارائه کردند. آنها برای حل این مسئله یک الگوریتم ژنتیک ترکیبی با دو نوع همسایگی بر پایه‌ی مفهوم مسیر بحرانی تعریف کردند.[۱۹]

مسئله‌ی تحقیق

تعریف مسئله

مسئله‌ی تحقیق شامل زمان‌بندی کارکارگاهی انعطاف‌پذیر در حالت وجود انعطاف‌پذیری عملیات با در نظرگرفتن محدودیت نت است. انعطاف‌پذیری عملیات، انجام یک عملیات با زمان پردازش و هزینه‌ی متفاوت بر روی ماشین‌های مختلف را ممکن می‌سازد. این نوع انعطاف‌پذیری را «انعطاف‌پذیری مسیر» نیز می‌نامند. در این نوع انعطاف‌پذیری، ماشین‌ها و توالی عملیاتی مختلفی وجود دارد، اما عملیات‌هایی که باید انجام شوند معین و ثابت‌اند.[۱۳] البته انعطاف‌پذیری که در این تحقیق مورد مطالعه قرار می‌گیرد، حالت خاصی از انعطاف‌پذیری عملیات است. این حالت شامل تعدادی ایستگاه کاری است که در آن، حداقل یک ایستگاه دارای بیش از یک ماشین موازی یکسان[۱۱] است. این مسئله، تعمیم‌یافته‌ی دو نظام کلاسیک کارکارگاهی و کارگاه ماشین‌های موازی است. در این محیط، امکان انجام عملیات فقط در یک ایستگاه کاری مشخص وجود دارد و در سایر ایستگاه‌ها انجام آن ممکن نیست. همچنین عملیات مورد نظر، ممکن است توسط هر ماشین از مجموعه ماشین‌های در دسترس ایستگاه کاری از پیش تعیین شده انجام شود. در این محیط، هر کار بر پایه‌ی مسیر پردازش خود، نیازمند تعیین توالی عملیات در ایستگاه‌های کاری مختلف است و نباید در هر مرحله روی بیش از یک ماشین پردازش شود. همچنین، بعضی از کارها می‌توانند از بعضی ایستگاه‌ها عبور نکنند. در شکل ۶ گردش کارها روی ماشین‌های یک ایستگاه کاری در کارگاه نشان داده شده است.

یکی از عواملی که ممکن است سبب محدودیت دسترسی به ماشین‌ها در طول افق برنامه‌ریزی شود، وجود فعالیت‌های نت است. در این تحقیق محدودیت دسترسی به ماشین‌ها از نوع قطعی و تحت عنوان محدودیت نت در نظر گرفته می‌شود. محدودیت نت در حالت دسترسی نامعین بررسی شده و دوره‌ی عدم دسترسی در این شرایط انعطاف‌پذیر است. البته هر فعالیت نت می‌بایست در پنجره‌ی زمانی از پیش تعیین شده خود انجام شود و قطع‌کردن آن مجاز نیست. خصوصیات کارها با وجود این محدودیت، در حالت مجاز بودن انقطاع عملیات و رژیم پردازش از سرگیری مجدد کار مورد بررسی قرار می‌گیرد. همچنین تابع هدف این مسئله، کمینه‌سازی معیار عملکرد بیشترین زمان تکمیل کلیه کارها به‌منظور افزایش بهره‌برداری از ماشین‌ها است.



شکل ۶. گردش کارها بر روی ماشین‌های یک ایستگاه کاری در محیط کارکارگاهی انعطاف‌پذیر.

با توجه به ویژگی‌های مسئله‌ی تحقیق، مسئله‌ی زمان‌بندی کار کارگاهی انعطاف‌پذیر با در نظرگرفتن محدودیت نت براساس روش علامت‌گذاری به‌صورت FJS‖nr,nfa‖CMax نشان داده می‌شود.[۴۰]

علاوه بر فرضیات پایه‌ی مسائل زمان‌بندی، عمده‌ی فرضیات خاص این تحقیق عبارت‌اند از:

۱. قطع کردن عملیات مجاز است.

۲. دسترسی پیوسته به ماشین‌ها در طول دوره‌ی زمان‌بندی ممکن نیست.

۳. محدودیت نت از نوع قطعی و نامعین است.

۴. سیاست پردازش، از سرگیری مجدد کار است.

۵. قطع کردن فعالیت نت مجاز نیست.

۶. امکان انجام عملیات فقط در یک ایستگاه کاری مشخص وجود دارد.

۷. ماشین‌های موازی در هر ایستگاه کاری یکسان‌اند.

ساختار مسئله

مسئله‌ی زمان‌بندی کار کارگاهی انعطاف‌پذیر با در نظرگرفتن محدودیت نت به سه زیرمسئله‌ی مسیریابی، تعیین توالی و زمان‌بندی نت تقسیم می‌شود. زیرمسئله‌ی مسیریابی شامل تخصیص عملیات به یک ماشین از مجموعه ماشین‌های در دسترس ایستگاه کاری از پیش تعیین شده است. زیرمسئله‌ی تعیین توالی، توالی عملیات تخصیص‌یافته به ماشین‌ها را تعیین می‌کند و زیرمسئله‌ی زمان‌بندی نت، زمان تکمیل هریک از فعالیت‌های نت را مشخص می‌کند.

در سال ۱۹۷۶ محققین نشان دادند که مسئله‌ی زمان‌بندی کارکارگاهی NP-hard است.[۴۱] از آنجا که مسئله‌ی زمان‌بندی کارکارگاهی انعطاف‌پذیر با در نظر گرفتن ماشین‌های موازی، توسعه‌یافته‌ی کارکارگاهی است، مسئله‌ی مذکور نیز NP-hard خواهد بود. همچنین در سال ۱۹۹۹ نشان داده شد که مسئله‌ی تک‌ماشینه با وجود دوره‌های عدم دسترسی متعدد، NP-hard است.[۳۶] بنابراین مسئله‌ی زمان‌بندی کارکارگاهی انعطاف‌پذیر مورد مطالعه که ترکیب دو مسئله‌ی فوق است، نیز NP-hard خواهد بود.

مدل ریاضی مسئله

در این تحقیق یک مدل ریاضی از نوع برنامه‌ریزی غیرخطی عدد صحیح مختلط برای مسئله‌ی مورد مطالعه ارائه می‌شود. در این مدل، تعداد n کار و L ایستگاه کاری وجود دارد. کار iام دارای o_i عملیات با توالی معین و ایستگاه کاری lام دارای m_l ماشین موازی یکسان است. عملیات o_{ik} توسط ماشین m_{lj} از مجموعه ماشین‌های در دسترس M_{ikl} و در ایستگاه کاری از پیش تعیین شده W_l، به‌مدت p_{iklj} واحد زمانی پردازش می‌شود. ماشین m_{lj} دارای R_{lj} فعالیت نت است که می‌بایست در طول برنامه‌ریزی انجام شوند. همچنین فعالیت نت PM_{ljr} به‌مدت t_{ljr} واحد زمانی انجام شده و در پنجره‌ی زمانی مشخصی تکمیل می‌شود. در ادامه اندیس‌ها، پارامترها و متغیرهای تصمیم معرفی، و سپس مدل ریاضی مسئله ارائه خواهد شد.

اندیس‌ها:

کارها	$i, h = ۱, ۲, ..., n$
ماشین‌های موازی در ایستگاه کاری lام	$j = ۱, ۲, ..., m_l$
توالی عملیات	$k, q = ۱, ۲, ..., o_i$
فعالیت‌های نت	$r = ۱, ۲, ..., R_{lj}$
ایستگاه‌های کاری	$l = ۱, ۲, ..., L$

پارامترها:

n:	تعداد کل کارها
M_{ikl}:	مجموعه ماشین‌های در دسترس در ایستگاه کاری lام جهت انجام o_{ik}
m:	تعداد کل ماشین‌ها
o_i:	تعداد عملیات کار iام
$t_{m_{lj}}$:	زمان در دسترس بودن m_{lj}
o_{ik}:	عملیات kام از کار iام
R_{lj}:	تعداد فعالیت‌های نت بر روی m_{lj}
L:	تعداد کل ایستگاه‌های کاری
PM_{ljr}:	فعالیت نت rام بر روی m_{lj}
W_l:	ایستگاه کاری lام
t_{ljr}:	مدت زمان انجام PM_{ljr}
m_l:	تعداد ماشین‌های موازی در ایستگاه کاری lام
U^E_{ljr}:	زودترین زمان تکمیل PM_{ljr}
m_{lj}:	ماشین jام از ایستگاه کاری lام
U^L_{ljr}:	دیرترین زمان تکمیل PM_{ljr}
p_{iklj}:	مدت زمان پردازش o_{ik} بر روی m_{lj}
$[U^E_{ljr}, U^L_{ljr}]$:	پنجره زمانی جهت تکمیلPM_{ljr}

متغیرهای تصمیم:

c_{ik}:	زمان تکمیل o_{ik}
u_{ljr}:	زمان تکمیل PM_{ljr}

$$x_{iklj} = \begin{cases} ۱ & \text{اگر } m_{ij} \text{ برای انجام } o_{ik} \text{ انتخاب شود} \\ ۰ & \text{درغیراین‌صورت} \end{cases}$$

$$y_{ikhq} = \begin{cases} ۱ & \text{اگر } o_{ik} \text{ پیش از } o_{hq} \text{ انتخاب شود} \\ ۰ & \text{درغیراین‌صورت} \end{cases}$$

$$\nu_{ikljr} = \begin{cases} ۱ & \text{اگر } o_{ik} \text{ پیش از } PM_{ljr} \text{ انتخاب شود} \\ ۰ & \text{درغیراین‌صورت} \end{cases}$$

اگر تابع هدف را کمینه‌سازی بیشترین زمان تکمیل کلیه‌ی کارها فرض کنیم، مسئله تحقیق به‌صورت زیر فرمول‌بندی می‌شود:

$$Min: \ c_{Max} = \underset{۱\le i\le n}{Max}\{c_{io_i}\} \quad (۱)$$

$$S.t. \ c_{ik} - c_{i,k-۱} \ge p_{iklj} \cdot x_{iklj} \quad k = ۲, ۳, ..., o_i; \ \forall i, l, j \quad (۲)$$

$$(c_{ik} - c_{hq} - p_{iklj}) \cdot x_{iklj} \cdot x_{hqlj} \cdot (۱ - y_{ikhq}) \ge ۰$$
$$\forall \{i, h \,|i \ne h\}, k, q, l, j \quad (۳)$$

$$(c_{hq} - c_{ik} - p_{hqlj}) \cdot x_{iklj} \cdot x_{hqlj} \cdot y_{ikhq} \ge ۰$$
$$\forall \{i, h \,|i \ne h\}, k, q, l, j \quad (۴)$$

$$(c_{ik} - u_{ljr} - p_{iklj}) \cdot x_{iklj} \cdot (۱ - v_{ikljr}) \ge ۰ \quad \forall i, k, l, j, r \quad (۵)$$

$$(u_{ljr} - c_{ik} - t_{ljr}) \cdot x_{iklj} \cdot v_{ikljr} \ge ۰ \quad \forall i, k, l, j, r \quad (۶)$$

$$\sum_{l=۱}^{L} \sum_{j \in M_{ikl}} x_{iklj} = ۱ \quad \forall i, k \quad (۷)$$

$$U^E_{ljr} \le u_{ljr} \le U^L_{ljr} \quad \forall l, j, r \quad (۸)$$

$$c_{ik} \ge ۰ \quad \forall i, k \quad (۹)$$

$$u_{ljr} \ge ۰ \quad \forall l, j, r \quad (۱۰)$$

$$x_{iklj}, y_{ikhq}, v_{ikljr} \in \{۰, ۱\} \quad \forall \{i, h \,|i \ne h\}, k, q, l, j, r \quad (۱۱)$$

در معادله‌ی ۱ هدف کمینه‌سازی معیار عملکرد بیشترین زمان تکمیل کلیه کارها است. نامعادله‌ی ۲ بیان‌گر محدودیت تقدم بین عملیات مختلف یک کار بر روی ماشین‌ها است و طبق آن عملیات $(k-۱)$ام از کار iام پیش از عملیات kام آن انجام می‌شود. نامعادلات ۳ و ۴ بیان‌گر محدودیت ناهمزمانیِ انجام عملیات کارهای مختلف روی یک ماشین هستند. به‌عبارت دیگر دو عملیات مختلف، همزمان به یک ماشین تخصیص داده نمی‌شوند. باید توجه داشت که این دو محدودیت از نوع محدودیت‌های انفصالی‌اند. یعنی برای هر ماشین، یکی از این محدودیت‌ها خود به خود ارضاء، و از مدل حذف می‌شود. نامعادلات ۵ و ۶ بیان‌گر محدودیت ناهمزمانی انجام فعالیت‌های نت و عملیات کارهای مختلف بر روی یک ماشین هستند. به‌عبارت دیگر، در هر لحظه فقط امکان تخصیص یک فعالیت نت یا عملیات به هر ماشین وجود دارد. این دو محدودیت نیز از نوع محدودیت‌های انفصالی‌اند. معادله‌ی ۷ نشان می‌دهد که می‌بایست یک ماشین از مجموعه ماشین‌های دردسترس برای انجام هر عملیات انتخاب شود و نامعادله‌ی ۸ بیان می‌کند که فعالیت‌های نت می‌بایست در پنجره زمانی مشخصی تکمیل شوند.

رویکرد حل مسئله

بسیاری از مسائل بهینه‌سازی در سیستم‌های تولید، خصوصاً مسائل بهینه‌سازی زمان‌بندی در کارگاه، بسیار پیچیده بوده و حل آنها با روش‌های بهینه‌سازی معمولی امکان‌پذیر نیست. از سال ۱۹۶۰، الگوریتم‌های تکاملی به‌عنوان دسته‌یی از الگوریتم‌های بهینه‌سازی توجه بسیاری از محققان را برای حل این‌گونه مسائل پیچیده به خود جلب کرده‌اند. این الگوریتم‌ها که مبنایی تصادفی دارند، به‌منظور جست‌وجوی نقطه‌ی بهینه‌ی کلی، در فضاهای پیچیده بسیار مناسب‌اند و کم‌تر در نقاط بهینه‌ی محلی گرفتار می‌شوند.[۱۳و۶] با توجه به ادبیات مسئله‌ی تحقیق و روش‌های حل آن که در بخش ۲ مرور شد، باید اشاره کرد که الگوریتم ژنتیک یکی از شناخته‌شده‌ترین الگوریتم‌های تکاملی است و روشی کارا برای حل مسئله‌ی کارکارگاهی انعطاف‌پذیر تلقی می‌شود. این الگوریتم تکاملی اولین‌بار در سال ۱۹۷۵ مطرح شد و سپس توسط محققین بعدی توسعه یافت. این روش که بر پایه‌ی جست‌وجوی تصادفی و تکامل استوار شده، در حل مسائل بهینه‌سازی ترکیبی عملکردی مؤثر دارد. بدین‌ترتیب که با استفاده از اطلاعات مربوط به نسل‌های گذشته، فرایند جست‌وجو در جمعیت انجام می‌شود، و با استفاده از سازوکار توارث، خصوصیات افراد جمعیت از نسلی به نسل دیگر بهبود می‌یابد.[۲۲]

چنان که پیش‌تر اشاره شد، مسئله‌ی اصلی به سه زیرمسئله‌ی مسیریابی، تعیین توالی و زمان‌بندی نت تقسیم می‌شود. در تحقیق حاضر به‌منظور حل مسئله از یک روش ترکیبی استفاده می‌شود. بدین‌ترتیب که دو زیرمسئله‌ی مسیریابی و تعیین توالی به‌کمک الگوریتم ژنتیک حل می‌شوند. حل این دو زیرمسئله، با رویکردی یک‌پارچه انجام می‌گیرد. به‌عبارت دیگر دو زیرمسئله‌ی مسیریابی و تعیین توالی به‌صورت همزمان در نظرگرفته شده و حل می‌شوند. همچنین برای حل زیرمسئله‌ی زمان‌بندی نت از یک الگوریتم ابتکاری استفاده می‌شود.

یک روش بسیار پرکاربرد در مواجهه با مسائل بهینه‌سازی با انواع مختلف متغیرهای تصمیم این است که تعدادی از متغیرهای تصمیم در فرایند کدگذاری کروموزوم‌ها در نظر گرفته نشده و در فرایند کدگشایی به‌وسیله‌ی روش‌های ابتکاری تعیین می‌شوند. چنانچه اطلاعات این نوع از متغیرهای تصمیم در کروموزوم‌ها وجود نداشته باشد، طول کروموزوم‌ها نسبت به زمان وجود آنها کوتاه‌تر خواهد بود. از این رو فضای جست‌وجوی ژنتیک کاهش یافته و انجام عملیات تقاطع و جهش به‌دلیل کاهش تعداد ژن‌ها در هر کروموزوم آسان می‌شود و درنتیجه، کارایی الگوریتم ژنتیک بهبود خواهد یافت.[۱۹] در این تحقیق نیز در برخورد با متغیرهای تصمیم مربوط به زمان‌بندی نت (u_{ljr}, v_{iklir})، از واردکردن اطلاعات آن متغیرها در کروموزوم صرف نظر شده و مقادیر آنها با استفاده از یک الگوریتم ابتکاری و در فرایند کدگشایی بر پایه‌ی اولویت در توالی تعیین می‌شود و بدین‌ترتیب زیرمسئله‌ی زمان‌بندی نت نیز حل خواهد شد.

الگوریتم ژنتیک پیشنهادی

چنان که گفته شد، ساختار کلی الگوریتم ژنتیک پیشنهادی دربرگیرنده‌ی طراحی کروموزوم، تولید جمعیت اولیه، انجام عملیات تقاطع و جهش، استفاده از الگوریتم ابتکاری برای حل زیرمسئله‌ی نت، و انجام فرایندهای ارزیابی و انتخاب است. رویه‌ی کلی الگوریتم ژنتیک پیشنهادی در شکل ۷ نشان داده شده است.

طراحی کروموزوم

یکی از مهم‌ترین عوامل تعیین‌کننده‌ی کارایی الگوریتم ژنتیک، طراحی سیستم کدگذاری برای نمایش جواب‌ها و تأثیر متقابل کدگذاری و عملگرهای ژنتیک است.[۱۹] در تحقیق حاضر با توجه به ماهیت مسئله، یک کروموزوم دورشته‌یی برای نمایش هر جواب ارائه شده که در آن رشته‌ی اول نشان‌گر تخصیص ماشین و رشته‌ی دوم نشان‌گر توالی عملیات است. طول کروموزوم نیز برابر تعداد کل عملیات ($\sum_{i=1}^{n} o_i$) است. هر ژن در کروموزوم دارای مکان مشخصی است که نشان‌دهنده‌ی عملیات مربوط به یک کار و ایستگاه کاری پردازش‌گر آن عملیات است. همچنین رشته‌ی توالی عملیات به روش نمایش جایگشت[۱۲] نشان داده شده است. به‌منظور درک بهتر سیستم کدگذاری، مثال ارائه‌شده با چهار کار و سه ایستگاه کاری را در نظر بگیرید که در آن، هر کار دارای سه عملیات و هر ایستگاه به‌ترتیب دارای سه، دو و دو ماشین موازی یکسان است. کروموزوم متناظر با جواب دارای دو رشته به‌طول دوازده است و مطابق شکل ۸ کدگذاری می‌شود. مفهوم این نحوه‌ی نمایش کروموزوم چنین بیان می‌شود: براساس رشته‌ی توالی عملیات، نخستین عملیات به‌منظور پردازش همان عملیات اول کار یک است که با توجه به رشته‌ی تخصیص ماشین، بر روی ماشین اول ایستگاه کاری یک انجام می‌شود. سایر مکان‌های کروموزوم نیز به‌همین صورت قابل تفسیرند.

تولید جمعیت اولیه

در الگوریتم ژنتیک پیشنهادی، جمعیت اولیه (Pop Size) به‌صورت تصادفی و بدون استفاده از روش‌های ابتکاری تولید شده و فاقد هرگونه اطلاعات در رابطه با مسئله است. این امر سبب حفظ تنوع کروموزوم‌ها در جمعیت و کاهش احتمال هم‌گرایی سریع و پیش از موعد و به دام‌افتادن در نقاط بهینه‌ی محلی خواهد شد.[۱۹]

عملیات تقاطع

در الگوریتم ژنتیک، عملیات تقاطع سبب ترکیب دو کروموزوم والد به‌منظور تولید کروموزوم‌های فرزند، که ویژگی‌های والدین خود را دارند، می‌شود.[۱۹] در الگوریتم ژنتیک پیشنهادی، از دو عملیات تقاطع مکان‌محور[۱۳] و ترتیب‌محور[۱۴] به‌صورت ترکیبی و با نرخ P_c استفاده شده است. عملگر تقاطع مکان‌محور بر روی رشته‌ی تخصیص ماشین و عملگر تقاطع ترتیب‌محور بر روی رشته توالی عملیات اعمال می‌شوند. در عملیات تقاطع مکان‌محور، تعدادی از ژن‌های کروموزوم P_1 به‌طور تصادفی انتخاب شده و دقیقاً در همان مکان از کروموزوم O_1 قرار می‌گیرند. درصورتی که یک ژن از کروموزوم P_1 در عملیات مشارکت نداشته باشد، ژنِ هم‌مکان آن از کروموزوم P_2 در عملیات شرکت کرده و به همان مکان از O_1 منتقل خواهد شد. در عملیات تقاطع ترتیب‌محور، ابتدا تعدادی از ژن‌های کروموزوم P_1 به‌طور تصادفی انتخاب، و سپس ژن‌های منتخب کروموزوم P_1 از P_2 حذف می‌شوند. آنگاه ژن‌های باقی‌مانده در کروموزوم P_2 دقیقاً در همان مکان از کروموزوم O_1 قرار می‌گیرند. برای تکمیل کروموزوم O_1 ژن‌های منتخب کروموزوم P_1 با حفظ ترتیب از چپ به راست در مکان‌های خالی کروموزوم O_1 قرار خواهند گرفت. در شکل‌های ۹ و ۱۰ نحوه‌ی انجام عملیات تقاطع مکان‌محور و ترتیب‌محور نشان داده شده است.

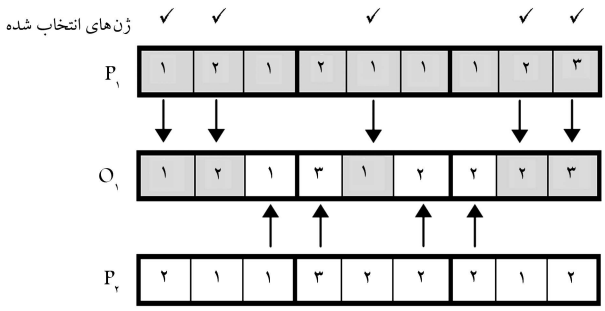
کروموزوم O_2 نیز طبق فرایند فوق، و با تعویض جای دو کروموزوم P_1 و P_2

```
Procedure: Genetic Algorithm
Begin
  t←0
  Initialize P(t);
  While (Not termination condition) do
    Crossover P(t) to yield O(t);
    Mutation P(t) to yield O(t);
    Heuristic Algorithm to solve maintenance scheduling sub-problem
    Evaluate P(t) and O(t);
     Select P(t+1) form P(t) and O(t);
     t← t + 1
   End
End.
```

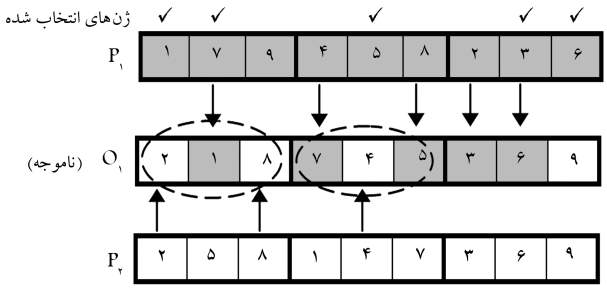
شکل ۷. رویه‌ی کلی الگوریتم ژنتیک پیشنهادی.

مکان	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲
اندیس عملیات	O_{11}	O_{12}	O_{13}	O_{21}	O_{22}	O_{23}	O_{31}	O_{32}	O_{33}	O_{41}	O_{42}	O_{43}
اندیس ایستگاه کاری	W_1	W_2	W_3	W_1	W_3	W_2	W_2	W_3	W_1	W_2	W_1	W_3
رشته تخصیص ماشین	۱	۲	۱	۲	۱	۱	۱	۲	۳	۲	۳	۲
رشته توالی عملیات	۱	۸	۱۰	۵	۶	۹	۲	۳	۱۲	۴	۷	۱۱

شکل ۸. نمایش کروموزوم متناظر با جواب مسئله.



شکل ۹. عملیات تقاطع مکان‌محور بر روی رشته‌ی تخصیص ماشین.



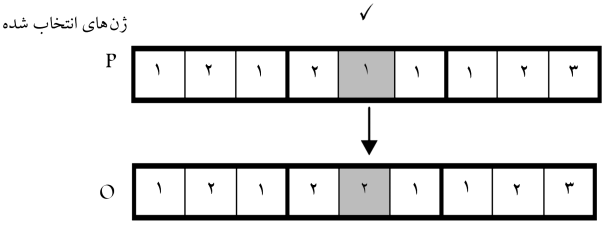
شکل ۱۰. عملیات تقاطع ترتیب‌محور بر روی رشته‌ی توالی عملیات.

تولید خواهد شد. چنان‌که ملاحظه می‌شود با انجام این عملیات تقاطع، امکان ایجاد کروموزوم غیرموجه وجود دارد. در ادامه به نحوه‌ی حل این مشکل اشاره خواهد شد.

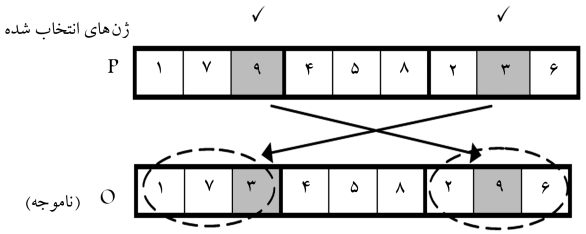
عملیات جهش

عملیات جهش در الگوریتم ژنتیک با ایجاد تغییرات تصادفی برنامه‌ریزی نشده در یک کروموزوم، جست‌وجوی قسمت‌های وسیع‌تری از فضای جواب را ممکن می‌سازد و از هم‌گرایی پیش از موعد الگوریتم جلوگیری می‌کند.[۴۲] در الگوریتم ژنتیک پیشنهادی، از دو عملگر جهش یکنواخت[۱۵] و تبادل دوطرفه[۱۶] به‌ترتیب با نرخ‌های P_{m_1} و P_{m_2} استفاده می‌شود. عملگر جهش یکنواخت بر رشته‌ی تخصیص ماشین، و عملگر جهش تبادل دوطرفه بر رشته‌ی توالی عملیات اعمال می‌شود. در عملیات جهش یکنواخت، یک ژن از کروموزوم P به‌طور تصادفی انتخاب، و مقدار آن نیز به‌صورت تصادفی به عدد دیگری در دامنه‌ی ژن تبدیل می‌شود. این عملیات در الگوریتم ژنتیک پیشنهادی بدین‌صورت انجام می‌شود که ماشینی از رشته‌ی تخصیص ماشین کروموزوم P، به‌طور تصادفی انتخاب، و سپس یک ماشین در دسترس جدید از همان ایستگاه کاری جایگزین آن می‌شود. در شکل ۱۱ ماشین اول از یک ایستگاه کاری (با سه ماشین موازی) به‌منظور عملیات جهش انتخاب، و ماشین دوم همان ایستگاه جایگزین آن می‌شود.

در عملیات جهش تبادل دوطرفه، دو ژن از کروموزوم P به‌طور تصادفی انتخاب،



شکل ۱۱. عملیات جهش یکنواخت بر رشته‌ی تخصیص ماشین.



شکل ۱۲. عملیات جهش تبادل دوطرفه بر رشته‌ی توالی عملیات.

و سپس جای آنها با یکدیگر تعویض می‌شود. این عملیات در الگوریتم ژنتیک پیشنهادی بدین‌صورت انجام می‌شود که دو عملیات از دو کار مختلف در رشته‌ی توالی عملیات کروموزوم P به‌طور تصادفی انتخاب، و سپس جای آنها با یکدیگر عوض می‌شود. در شکل ۱۲، عملیات سوم از کار اول و عملیات دوم از کار سوم برای عملیات جهش انتخاب، و جای آنها با یکدیگر تعویض می‌شود. با انجام این عملیات جهش، ایجاد کروموزوم غیرموجه ممکن خواهد شد.

الگوریتم ابتکاری به‌منظور حل زیرمسئله‌ی نت

چنان‌که پیش‌تر اشاره شد، ضمن چشم‌پوشی از واردکردن متغیرهای تصمیم مربوط به زمان‌بندی نت در کروموزوم، برای حل زیرمسئله‌ی زمان‌بندی نت و تعیین مقادیر متغیرهای تصمیم آن از الگوریتم ابتکاری در فرایند کدگشایی بر پایه‌ی اولویت در توالی، استفاده می‌شود.

در سال ۲۰۰۶، به‌منظور حل زیرمسئله‌ی زمان‌بندی نت برای مسئله‌ی زمان‌بندی کارگاهی انعطاف‌پذیر، یک الگوریتم ابتکاری ارائه شد.[۱۹] در این تحقیق الگوریتم برای حل زیرمسئله‌ی زمان‌بندی نت و اعمال بر روی ماشین‌های موازی در حالت وجود بیش از یک فعالیت نت برای هر ماشین، توسعه داده شده است. گام‌های این الگوریتم ابتکاری به‌شرح زیر هستند:

- گام ۱: فعالیت‌های نت کلیه‌ی ماشین‌های ایستگاه‌های کاری مختلف را در انتهای پنجره‌ی زمانی خود، چنین زمان‌بندی کنید: $u_{ljr} = U^L_{ljr}$.
- گام ۲: عملیات کلیه‌ی کارها را براساس اولویت در توالی، به اولین ماشین در دسترس ایستگاه کاری از پیش تعیین شده تخصیص دهید؛ این کار را تا رسیدن به نزدیک‌ترین پنجره زمانی انجام فعالیت نت هر ماشین ادامه دهید.
- گام ۳: در بازه زمانی انجام فعالیت نت، درصورت در دسترس بودن ماشین و امکان انجام عملیات پیش از فعالیت نت، عملیات را به ماشین مورد نظر تخصیص داده و به گام ۵ بروید؛ در غیر این‌صورت فعالیت نت را تا آنجا که امکان دارد به‌صورت زیر به سمت چپ جابه‌جا کرده و به ابتدای پنجره زمانی خود نزدیک کنید: $u_{ljr} = Max\left\{t_{m_{lj}} + t_{ljr}, U^E_{ljr}\right\}$.
- گام ۴: درصورت در دسترس بودن ماشین و امکان انجام عملیات پس از فعالیت نت، آن عملیات را به ماشین تخصیص دهید.
- گام ۵: چنانچه عملیات کلیه‌ی کارها زمان‌بندی شده‌اند به الگوریتم خاتمه دهید؛ در غیر این‌صورت به گام ۲ بازگردید.

تابع برازندگی

در الگوریتم ژنتیک، برای ارزیابی کروموزوم‌ها از شاخص معینی به‌نام «تابع برازندگی» استفاده می‌شود. این تابع در مسائل بهینه‌سازی، همان تابع هدف مسئله است.

در مرحله‌ی ارزیابی، پس از کدگشایی هر کروموزوم به یک جواب و قرار دادن مقادیر آن در تابع هدف، میزان برازندگی آن کروموزوم مشخص می‌شود.[۴۲] از آنجا که تابع هدف مسئله‌ی تحقیق «کمینه‌سازی c_{Max}» است، پس هر جوابی که مقدار تابع هدف کم‌تری داشته باشد برازنده‌تر است. بنابراین به‌منظو تطابق مفهوم تابع هدف با برازش، تابع برازندگی کروموزوم kام، چنین تعریف می‌شود:

$$f(k) = \frac{۱}{c_{Max}(k)}.$$

فرایند انتخاب

فضای نمونه‌گیری مورد استفاده در الگوریتم ژنتیک پیشنهادی، فضای نمونه‌گیری توسعه‌یافته است که در آن کروموزوم‌های والد و فرزند، هر دو، برای بقاء شانس یکسان دارند.

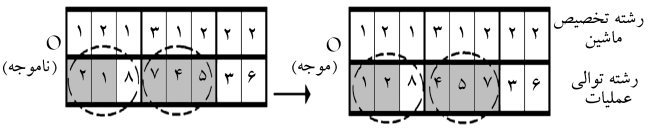
سازوکار نمونه‌گیری، ترکیبی از دو روش انتخاب قطعی $(\mu + \lambda)$ و انتخاب تصادفی چرخ رولت بوده و تعمیم‌یافته‌ی روش Bäck وHoffmeister است. بر اساس روش انتخاب $(\mu + \lambda)$ تعداد μ' کروموزوم متفاوت، از بهترین کروموزوم‌های فضای نمونه‌گیری انتخاب شده و به نسل بعد منتقل می‌شوند. سپس به‌کمک روش چرخ رولت، تعداد $\mu - \mu'$ کروموزوم باقی‌مانده برای تکمیل نسل جدید انتخاب می‌شوند. همچنین با جلوگیری از ورود کروموزوم‌های تکراری به نسل جدید، مانع از تسلط اَبَرکروموزوم‌ها[۱۷] بر جمعیت شده و از هم‌گرایی سریع به جواب بهینه‌ی محلی جلوگیری می‌شود.

معیار توقف

شرط توقف الگوریتم ممکن است عدم تغییر جواب در چند تکرار متوالی، سپری شدن مدت زمان معینی از حل، یا هر شرط خاص دیگری باشد.[۴۲] در الگوریتم ژنتیک پیشنهادی، شرط توقف همانا تولید تعداد معینی نسل (Max_Gen) از جمعیت اولیه است.

نحوه‌ی اعمال محدودیت‌ها

در مسائل بهینه‌سازی نحوه‌ی اعمال محدودیت‌ها در الگوریتم ژنتیک دارای اهمیت ویژه‌یی است، زیرا ممکن است عملگرهای ژنتیک کروموزوم‌های غیرموجه تولید کنند. در مواجهه با این مشکل، روش‌های مختلفی نظیر استراتژی ردکردن[۱۸]، اصلاحی[۱۹]، جریمه‌یی[۲۰] و اصلاح عملگرهای ژنتیک[۲۱] ارائه شده است.[۴۲] در الگوریتم ژنتیک پیشنهادی برای اعمال محدودیت‌های مسئله از استراتژی اصلاحی استفاده می‌شود. در این روش، هرکروموزوم غیرموجه به‌کمک یک فرایند اصلاح مشخص به کروموزوم موجه تبدیل می‌شود. در روش حل ارائه‌شده، طراحی کروموزوم و انتخاب عملگرهای ژنتیک به‌گونه‌یی صورت‌گرفته که کم‌ترین مشکل در برخورد با محدودیت‌ها و تولید کروموزوم غیرموجه ایجاد شود. بدین‌صورت که رشته‌ی تخصیص ماشین همواره موجه خواهد بود. در رشته‌ی توالی عملیات نیز فقط امکان نقض محدودیت تقدم توالی بین عملیات یک کار وجود دارد که این مشکل با مرتب‌سازی صعودی توالی عملیات هر کار در رشته‌ی توالی عملیات، حل خواهد شد. نحوه‌ی انجام عملیات اصلاح بر روی رشته‌ی توالی عملیات، در شکل ۱۳ نشان داده شده است.



شکل ۱۳. نحوه انجام عملیات اصلاح بر روی رشته توالی عملیات.

طراحی آزمایش‌های عددی

روش‌های مورد مقایسه

در این تحقیق عملکرد روش ترکیبی پیشنهادی در دو حالت ارزیابی خواهد شد. از آنجا که مسئله‌ی کارکارگاهی انعطاف‌پذیر، با وجود انعطاف‌پذیری عملیات حالت ساده‌شده از مسئله‌ی تحقیق است و در آن تعداد دوره‌های عدم دسترسی برای ماشین‌ها صفر $(R = ۰)$ در نظرگرفته می‌شود. لذا در این حالت الگوریتم ژنتیک پیشنهادی با روش فرا ابتکاری که در سال ۲۰۰۴ برای حل مسئله‌ی جریان کارگاهی انعطاف‌پذیر ارائه شد، مقایسه می‌شود. مسئله‌ی جریان کارگاهی انعطاف‌پذیر بیشترین شباهت را با مسئله‌ی تحقیق دارد. همچنین روش حل مسئله‌ی مذکور به‌سهولت و بدون از دست‌دادن کلیات الگوریتم به‌منظور حل مسئله‌ی تحقیق کاربرد دارد. محققین برای حل مسئله‌ی خود از روش فرا ابتکاریِ RKGA در زمان‌بندی عملیات اول کارها و از الگوریتم‌های ابتکاری SPT چرخشی و قاعده‌ی جانسون برای زمان‌بندی عملیاتِ بعدی آن کارها استفاده کردند.[۴۳] در این تحقیق نیز از روش آنها برای حل مسئله استفاده می‌شود. سپس نشان داده خواهد شد که کارایی روش پیشنهادی برای حل مسئله‌ی تحقیق از روش حل آنها بیشتر است. در حالت دوم، مسئله‌ی تحقیق که تعمیم‌یافته‌ی مسئله‌ی کارکارگاهی انعطاف‌پذیر است به‌کمک روش پیشنهادی و در نظرگرفتن به‌ترتیب تعداد یک، دو و سه فعالیت نت بر روی هر ماشین $(R = ۱, ۲, ۳)$ مورد مطالعه قرارگرفته و نتایج حاصل از آن با حالت اول $(R = ۰)$ مقایسه می‌شود. در این حالت کارایی روش حل پیشنهادی به‌کمک شاخص مورد استفاده‌یی که در سال ۲۰۰۶ محققین برای حل مسئله‌ی کارکارگاهی انعطاف‌پذیر با یک دوره‌ی عدم دسترسی برای هر ماشین ارائه کرده بودند، بررسی می‌شود. آنها با مقایسه‌ی مقدار افزایش تابع هدف با بیشترین مدت زمان انجام فعالیت‌های نت بر روی ماشین‌ها نشان دادند که این مقدار افزایش در تابع هدف ناچیز است.[۱۹] در این تحقیق نیز از همین روش به‌منظور نشان‌دادن کارایی روش حل ارائه‌شده استفاده می‌شود و نشان داده خواهد شد که این مقدار افزایش در تابع هدف مسئله ناچیز است.

تولید داده‌های تصادفی

داده‌های مورد نیاز برای حل مسئله شامل تعداد کارها، تعداد ایستگاه‌های کاری، تعداد ماشین‌ها در هر ایستگاه، مدت زمان پردازش، همچنین تعداد فعالیت‌های نت هر ماشین، مدت زمان انجام، زودترین و دیرترین زمان تکمیل هر فعالیت نت، و زمان بین انجام دو فعالیت نت (زمان بین پایان یک فعالیت نت و شروع فعالیت نت بعدی) است. در این تحقیق، برای تولید چهار پارامتر نخست، از داده‌های مسئله‌ی کروز و آسکین استفاده شده است.[۴۳] چهار پارامترِ مدت زمان انجام، زودترین زمان تکمیل، دیرترین زمان تکمیل، و زمان بین انجام دو فعالیت نت به‌ترتیب دارای توزیع‌های یکنواخت $U[۱۰۰, ۱۲۰]$، $U[۱۴۰, ۱۶۰]$، $U[۲۶۰, ۲۸۰]$ و $U[۹۰۰, ۱۱۰۰]$ هستند. همچنین مسئله‌ی تحقیق درصورت وجود به‌ترتیب ۰، ۱، ۲، ۳ فعالیت نت برای هر ماشین مورد بررسی قرارگرفته است.

به‌منظور اعتبارسنجی روش حل پیشنهادی، از تمامی ترکیبات داده‌های تصادفی

جدول ۱. تولید داده‌های تصادفی.

پارامتر	مقادیر	حالات
تعداد کارها (n)	۶ ,۳۰ ,۱۰۰	۳
تعداد ایستگاه‌ها (L)	۲ ,۴ ,۸	۳
توزیع ماشین‌ها	متغیر عدد ثابت	۲
تعداد ماشین ها (m_l)	$U[1,...,4], U[1,...,n]$ ۲,n	۲
زمان پردازش (p_{ik})	$U[50,70], U[20,100]$	۲
تعداد فعالیت‌های نت (R)	۰ ,۱ ,۲ ,۳	۴
زمان فعالیت های نت (t_r)	$U[100,120]$	۱
زودترین زمان تکمیل فعالیت نت (U_r^E)	$U[140,160]$	۱
(U_r^L)	$U[240,260]$	۱
زمان مابین فعالیت‌های نت	$U[900,1100]$	۱
تعداد کل سناریوها		۴ × ۷۲

استفاده شده است. برای انجام آزمایش‌های عددی، تعداد ۲۸۸ سناریو ممکن وجود دارد که از هر سناریو تعداد ۱۰ مثال تولید شده است. بنابراین درمجموع تعداد ۲۸۸۰ اجرا در آزمایش‌های عددی وجود دارد. الگوریتم ژنتیک پیشنهادی به زبان C++ کد نویسی شده و آزمایش‌های عددی فوق بر روی رایانه‌ی شخصی Pentium IV با مشخصات CPU ۳٫۲GHz وRAM ۱ GB انجام گرفته است. در جدول ۱ پارامترهای مدل و مقادیر مربوط به آنها، و نیز تعداد حالات ممکن برای تولید داده‌های تصادفی نشان داده شده است.

تنظیم پارامترهای الگوریتم ژنتیک پیشنهادی

تنظیم پارامترهای یک الگوریتم ژنتیک، عملکرد آن را به‌شدت تحت تأثیر قرار می‌دهد.[۱۹] این امر بر کیفیت جواب‌های حاصل از الگوریتم بسیار مؤثر است. توجه به این نکته ضروری است که الگوریتم‌های ژنتیک برای حل مسائل مختلف در رویکرد (به‌علت تفاوت در اجزاء مسئله) و ساختار (به‌دلیل وجود تفاوت در نوع عملگرهای ژنتیک، نحوه‌ی ارزیابی فضا، سازوکار نمونه‌گیری و...) تفاوت‌هایی دارند. به‌همین دلیل تنظیم پارامترهای الگوریتم ژنتیک ضروری به نظر می‌رسد.

به‌منظور تنظیم پارامترهای الگوریتم ژنتیک روش‌های مختلفی (نظیر روش‌های آماری طراحی آزمایش‌ها، شبیه‌سازی و روش‌های تجربی) وجود دارد. یادآور می‌شود که هر پارامتر می‌تواند مقادیر متنوعی داشته باشد و این امر سبب ایجاد حالات و ترکیبات بی‌شماری از پارامترها در الگوریتم می‌شود. بنابراین دست‌یابی به مقدار بهینه‌ی پارامترها در الگوریتم ژنتیک دشواری‌های خاص خود را دارد و به‌سادگی دست‌یافتنی نیست. در این تحقیق از روش شبیه‌سازی برای تنظیم پارامترهای الگوریتم استفاده شده است. به‌منظور انجام شبیه‌سازی پارامترها، ابتدا حوزه‌ی تغییر هر یک از پارامترها مشخص می‌شود. سپس با ثابت نگه‌داشتن سایر پارامترها در کم‌ترین مقدارشان، اولین پارامتر در حوزه‌ی تغییر خود مقداردهی شده و تأثیر نتایج آزمایش‌های حاصل از شبیه‌سازی بر تابع هدف بررسی می‌شود. به‌عبارت دیگر به‌ازای هر مقداری که بهبود بیشتری در جواب حاصل کند، پارامتر اول تنظیم می‌شود.

جدول ۲. مقادیر پارامترهای الگوریتم ژنتیک پس از تنظیم.

پارامتر	Pop_Size	Max_Gen	μ'	P_c	P_{m_1}	P_{m_2}
مقادیر	۱۰۰	۱۲۵	۲۰	۰٫۷	۰٫۰۲	۰٫۰۴

پس از تنظیم پارامتر نخست، مقدار این پارامتر در حوزه‌ی تغییرش تثبیت شده و سایر پارامترهای الگوریتم نیز به‌همین ترتیب تنظیم می‌شوند.

پارامترهای الگوریتم ژنتیک پیشنهادی عبارت‌اند از: اندازه جمعیت، تعداد نسل‌ها، نرخ تقاطع ترکیبی، نرخ‌های جهش یکنواخت و تبادل دوطرفه، تعداد بهترین کروموزوم‌های هر نسل. به‌منظور انجام شبیه‌سازی برای تنظیم پارامترهای الگوریتم ژنتیک پیشنهادی، تعداد ۲۸۸ سناریو ممکن آزمایش شده و از هر سناریو یک مثال تولید شد. با توجه به نتایج حاصل از شبیه‌سازی، مناسب‌ترین مقادیر برای پارامترهای الگوریتم در جدول ۲ نشان داده شده‌اند.

نتایج آزمایش‌های عددی

در مرحله‌ی نخست، کارایی الگوریتم ژنتیک پیشنهادی در حالت عدم وجود فعالیت نت بر روی ماشین‌ها ($R = 0$) با روش فرااِبتکاری RKGA مقایسه شده است. برای مقایسه‌ی این دو روش حل، از شاخص میانگین تابع هدف به‌عنوان شاخص اصلی و از شاخص‌های کم‌ترین مقدار تابع هدف، میانگین زمان حل، میزان و تعداد دفعات بهبود به‌عنوان شاخص‌های کمکی در ابعاد مختلف مسئله استفاده شده است. چنان که در جدول ۳ ملاحظه می‌شود، روش پیشنهادی با توجه به شاخص میانگین تابع هدف در ابعاد کوچک، متوسط و بزرگ به‌ترتیب با میانگین ۰٫۹۰٪، ۱٫۱۴٪ و ۱٫۸۱٪ بهبود و درکل با میانگین ۱٫۵۹٪ بهبود، عملکرد بهتری نسبت به RKGA دارد. همچنین الگوریتم ژنتیک ارائه‌شده براساس شاخص کم‌ترین مقدار تابع هدف در ابعاد کوچک، متوسط و بزرگ به‌ترتیب با میانگین ۰٫۲۴٪، ۱٫۰۰٪ و ۱٫۱۲٪ بهبود و در کل با میانگین ۰٫۹۱٪ بهبود، برتری محسوسی نسبت به

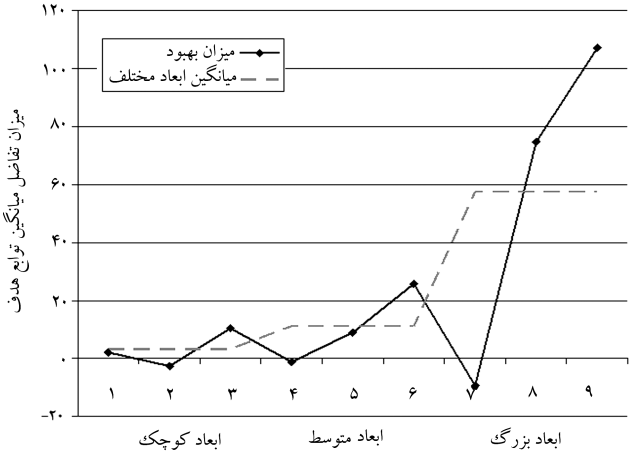
جدول ۳. مقایسه‌ی نتایج.

ابعاد مسئله	تعداد کار	تعداد مراحل	الگوریتم ژنتیک پیشنهادی (R=۰)			روش RKGA			میزان بهبود (درصد)		تعداد دفعات بهبود	
			کمترین تابع هدف (ثانیه)	میانگین تابع هدف (ثانیه)	میانگین زمان حل (ثانیه)	کمترین تابع هدف (ثانیه)	میانگین تابع هدف (ثانیه)	میانگین زمان حل (ثانیه)	کمترین تابع هدف	میانگین تابع هدف	کمترین تابع هدف	میانگین تابع هدف
کوچک	۶	۲	۱۲۰	۲۱۰٫۳۳	۰٫۱۲	۱۲۲	۲۱۲٫۴۵	۰٫۰۴	٪۱٫۶۴	٪۱٫۰۰	۶۵	۶۳
		۴	۲۴۶	۳۲۴٫۷۴	۰٫۳۰	۲۳۶	۳۲۲٫۱۷	۰٫۰۴	٪-۴٫۲۴	٪-۰٫۸۰	۴۶	۵۸
		۸	۴۸۱	۵۶۷٫۹۰	۰٫۹۹	۴۹۱	۵۷۸٫۳۸	۰٫۰۵	٪۲٫۰۴	٪۱٫۸۱	۷۲	۶۹
	میانگین بعد کوچک		۲۸۲٫۳۳	۳۶۷٫۶۵	۰٫۴۷	۲۸۳٫۰۰	۳۷۱٫۰۰	۰٫۰۴	٪۰٫۲۴	٪۰٫۹۰	۶۱	۶۳
متوسط	۳۰	۲	۲۲۴	۸۱۱٫۵۰	۱٫۵۱	۲۱۸	۸۱۰٫۳۱	۰٫۲۸	٪-۲٫۷۵	٪-۰٫۱۵	۵۵	۶۱
		۴	۳۰۸	۹۰۶٫۱۸	۳٫۵۹	۳۲۰	۹۱۵٫۱۸	۰٫۲۵	٪۳٫۷۵	٪۰٫۹۸	۷۷	۶۹
		۸	۵۵۶	۱۲۰۲٫۷۶	۲۰٫۶۶	۵۶۱	۱۲۲۸٫۵۳	۰٫۲۴	٪۰٫۸۹	٪۲٫۱۰	۶۲	۷۷
	میانگین بعد متوسط		۳۶۲٫۶۷	۹۷۳٫۴۸	۸٫۵۹	۳۶۶٫۳۳	۹۸۴٫۶۷	۰٫۲۶	٪۱٫۰۰	٪۱٫۱۴	۶۵	۶۹
بزرگ	۱۰۰	۲	۶۱۰	۲۷۹۱٫۷۹	۱۳٫۷۳	۶۲۴	۲۷۸۲٫۴۷	۱٫۶۷	٪۲٫۲۴	٪-۰٫۳۳	۷۶	۵۷
		۴	۷۰۵	۲۸۹۱٫۲۴	۴۰٫۳۳	۷۰۱	۲۹۶۵٫۹۸	۲٫۵۳	٪-۰٫۵۷	٪۲٫۵۲	۵۹	۷۳
		۸	۹۷۶	۳۶۵۰٫۳۶	۲۵۱٫۶۷	۹۹۲	۳۷۵۷٫۴۶	۲٫۹۳	٪۱٫۶۱	٪۲٫۸۵	۶۸	۷۵
	میانگین بعد بزرگ		۷۶۳٫۶۷	۳۱۱۱٫۱۳	۱۰۱٫۹۱	۷۷۲٫۳۳	۳۱۶۸٫۶۴	۲٫۳۸	٪۱٫۱۲	٪۱٫۸۱	۶۸	۶۸
میانگین کلیه ابعاد			۴۶۹٫۵۶	۱۴۸۴٫۰۹	۳۶٫۹۹	۴۷۳٫۸۹	۱۵۰۸٫۱۰	۰٫۸۹	٪۰٫۹۱	٪۱٫۵۹	۶۴	۶۷
کل تعداد دفعات بهبود											۵۸۰	۶۰۲
درصد کل دفعات بهبود											٪۸۰٫۵۶	٪۸۳٫۶۱

* در جدول فوق اعداد منفی نشان‌دهنده برتری روش RKGA می‌باشد.

RKGA دارد. کارایی روش پیشنهادی، با توجه به شاخص میانگین تابع هدف در ابعاد کوچک، متوسط و بزرگ به‌طور میانگین و به‌ترتیب با ۶۹، ۶۳ و ۶۸ بار بهبود و درکل با میانگین ۶۷ بار و در مجموع با ۶۰۲ بار (از ۷۲۰ اجرا) یا ٪۸۳٫۶۱ بهبود از روش فراابتکاری RKGA بیشتر است.

الگوریتم ژنتیک ارائه‌شده برای حل مسئله در مقایسه با RKGA نیازمند زمان حل بیشتری است، اما با توجه به میزان بهبود حاصله این افزایش زمان قابل توجیه است. بنابراین با توجه به نتایج حاصل از آزمایش‌های عددی، در مجموع کارایی روش حل ارائه‌شده نسبت به روش RKGA تأیید می‌شود. در نمودار ۱ میزان تفاضل میانگین توابع هدف الگوریتم ژنتیک پیشنهادی در حالت ($R=0$) با روش فراابتکاری RKGA برای ابعاد مختلف مسئله رسم شده است.



نمودار ۱. مقایسه تفاضل میانگین توابع هدف در حالت (R=۰) با روش فراابتکاری RKGA.

آزمایش‌های الگوریتم ژنتیک پیشنهادی و روش RKGA

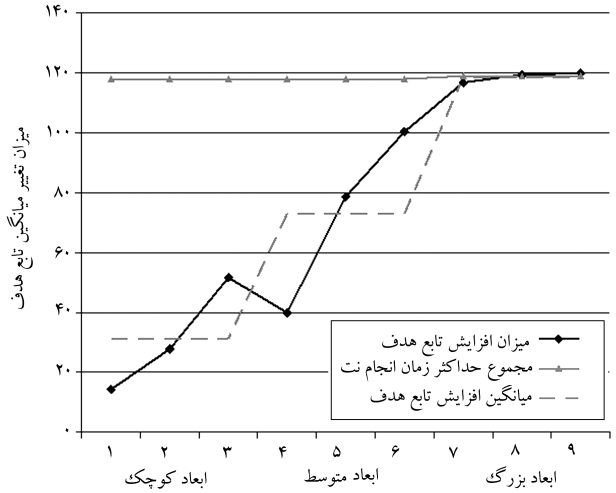
در این مرحله تأثیر اعمال محدودیت نت بر عملکرد روش حل ارائه‌شده به‌ترتیب با در نظرگرفتن تعداد یک، دو و سه فعالیت نت بر روی هر ماشین ($R = 1, 2, 3$) بررسی شده و نتایج حاصل از این آزمایش‌ها با حالت عدم وجود فعالیت نت مقایسه می‌شود. در این حالت کارایی روش حل پیشنهادی به‌کمک شاخص بیشترین مدت‌زمان انجام فعالیت‌های نت روی ماشین‌ها که در سال ۲۰۰۶ برای حل مسئله‌ی کارگاهی انعطاف‌پذیر با یک دوره عدم دسترسی برای هر ماشین ارائه شده بود، بررسی می‌شود.[۱۹]

روشن است که با در نظرگرفتن فعالیت نت برای ماشین‌ها، زمان در دسترس

جدول ۴. مقایسه‌ی نتایج آزمایش‌های الگوریتم ژنتیک پیشنهادی در حالت R=۰ با R=۱.

ابعاد مسئله	تعداد کار	تعداد مراحل	روش پیشنهادی در حالت R=۰			روش پیشنهادی در حالت R=۱			میزان افزایش (درصد)		شاخص کارایی الگوریتم ابتکاری نت	
			کمترین تابع هدف (ثانیه)	میانگین تابع هدف (ثانیه)	میانگین زمان حل (ثانیه)	کمترین تابع هدف (ثانیه)	میانگین تابع هدف (ثانیه)	میانگین زمان حل (ثانیه)	کمترین تابع هدف	میانگین تابع هدف	میزان افزایش میانگین تابع هدف	مجموع حداکثر مدت زمان انجام فعالیت‌های نت
کوچک	۶	۲	۱۲۰	۲۱۰٫۳۳	۰٫۱۲	۱۲۳	۲۲۴٫۶۳	۰٫۱۳	٪۲٫۵۰	٪۶٫۸۰	۱۴٫۳۱	۱۱۶
		۴	۲۴۶	۳۲۴٫۷۴	۰٫۳۰	۲۷۰	۳۵۲٫۵۰	۰٫۳۶	٪۹٫۷۶	٪۸٫۵۵	۲۷٫۷۶	۱۱۹
		۸	۴۸۱	۵۶۷٫۹۰	۰٫۹۹	۵۰۷	۶۱۹٫۵۵	۱٫۲۱	٪۵٫۴۱	٪۹٫۰۹	۵۱٫۶۵	۱۱۹
	میانگین بعد کوچک		۲۸۲٫۳۳	۳۶۷٫۶۵	۰٫۴۷	۳۰۰٫۰۰	۳۹۸٫۸۹	۰٫۵۷	٪۶٫۲۶	٪۸٫۵۰	۳۱٫۲۴	۱۱۸
متوسط	۳۰	۲	۲۲۴	۸۱۱٫۵۰	۱٫۵۱	۲۵۴	۸۵۱٫۴۳	۱٫۷۷	٪۱۳٫۳۹	٪۴٫۹۲	۳۹٫۹۳	۱۱۸
		۴	۳۰۸	۹۰۶٫۱۸	۳٫۵۹	۳۸۳	۹۸۴٫۸۸	۳٫۶۳	٪۲۴٫۳۵	٪۸٫۶۹	۷۸٫۷۱	۱۱۹
		۸	۵۵۶	۱۲۰۲٫۷۶	۲۰٫۶۶	۶۲۳	۱۳۰۳٫۱۸	۲۱٫۷۸	٪۱۲٫۰۵	٪۸٫۳۵	۱۰۰٫۴۱	۱۱۷
	میانگین بعد متوسط		۳۶۲٫۶۷	۹۷۳٫۴۸	۸٫۵۹	۴۲۰٫۰۰	۱۰۴۶٫۵۰	۹٫۰۶	٪۱۵٫۸۱	٪۷٫۵۰	۷۳٫۰۲	۱۱۸
بزرگ	۱۰۰	۲	۶۱۰	۲۷۹۱٫۷۹	۱۳٫۷۳	۷۱۵	۲۹۰۸٫۶۱	۱۳٫۰۶	٪۱۷٫۲۱	٪۴٫۱۸	۱۱۶٫۸۲	۱۱۷
		۴	۷۰۵	۲۸۹۱٫۲۴	۴۰٫۳۳	۸۲۱	۳۰۱۰٫۶۸	۴۱٫۲۱	٪۱۶٫۴۵	٪۴٫۱۳	۱۱۹٫۴۴	۱۱۹
		۸	۹۷۶	۳۶۵۰٫۳۶	۲۵۱٫۶۷	۱۱۲۵	۳۷۷۰٫۳۱	۲۵۲٫۴۵	٪۱۵٫۲۷	٪۳٫۲۹	۱۱۹٫۹۵	۱۲۰
	میانگین بعد بزرگ		۷۶۳٫۶۷	۳۱۱۱٫۱۳	۱۰۱٫۹۱	۸۸۷٫۰۰	۳۲۲۹٫۸۷	۱۰۲٫۲۴	٪۱۶٫۱۵	٪۳٫۸۲	۱۱۸٫۷۴	۱۱۹
میانگین کلیه ابعاد			۴۶۹٫۵۶	۱۴۸۴٫۰۹	۳۶٫۹۹	۵۳۵٫۶۷	۱۵۵۸٫۴۲	۳۷٫۲۹	٪۱۴٫۰۸	٪۵٫۰۱	۷۴٫۳۳	۱۱۸

بودن آنها کاهش یافته و تکمیل عملیات کارهای مختلف دیرتر انجام می‌شود و درنتیجه به افزایش مقدار تابع هدف می‌انجامد. حال این سؤال مطرح می‌شود که افزایش مقدار تابع هدف تا چه‌اندازه‌یی معقول است و کارایی روش حل را تأیید می‌کند. برای پاسخ به این پرسش، باید گفت که شاخص بیشترین مدت زمان انجام فعالیت‌های نت، نشان‌دهنده‌ی بیشینه زمان عدم دسترسی به ماشین‌ها است. این شاخص مستقیماً بر تابع هدف تأثیر داشته و به‌نوعی نشان‌گر حد مجاز افزایش معقول آن است. محققین این شاخص را معیاری مناسب برای تشخیص معقول بودن افزایش مقدار تابع هدف دانسته و اظهار داشتند که چنانچه میزان افزایش تابع هدف نسبت به بیشترین مدت زمان نت، کم‌تر (یا نزدیک به آن) باشد نشان‌گر ناچیزبودن افزایش تابع هدف و کارایی الگوریتم ابتکاری است.[۱۹] اما از آنجا که آنها مسئله‌ی خود را تنها در یک دوره عدم دسترسی بررسی کردند ولی مسئله تحقیق در حالت‌های وجود دو و سه دوره عدم دسترسی نیز بررسی می‌شود، لازم است شاخص مورد استفاده‌شان را برای اعتبارسنجی روش پیشنهادی تعمیم داده، و به‌صورت مجموع بیشینه زمان انجام فعالیت‌های نت تعریف شود. بنابراین میزان افزایش تابع هدف با مجموع بیشینه زمان انجام فعالیت‌های نت روی ماشین‌ها مقایسه شده و به‌کمک آن کارایی روش ترکیبی بررسی می‌شود. در جدول ۴ نتایج حاصل از مقایسه‌ی حالت عدم وجود فعالیت نت با حالت وجود یک فعالیت نت نشان داده شده است. چنان‌که ملاحظه می‌شود، متوسط میزان افزایش میانگین تابع هدف در مقایسه با مجموع بیشینه زمان انجام فعالیت‌های نت در ابعاد کوچک، متوسط و بزرگ، دارای مقادیر کم‌تری است. پس با توجه به نتایج محاسبات، این امر دلالت بر ناچیزبودن افزایش تابع هدف دارد و نشان‌گر کارایی روش ترکیبی است. بنابراین روش حل پیشنهادی براساس شاخص میانگین تابع هدف در ابعاد کوچک، متوسط و بزرگ به‌ترتیب با میانگین ۸٫۵۰٪،



نمودار ۲. مقایسه‌ی متوسط میزان افزایش میانگین تابع هدف با مجموع بیشینه مدت زمان انجام فعالیت‌های نت در حالت‌های ($R = ۰$) و ($R = ۱$).

۷٫۵۰٪ و ۳٫۸۲٪ افزایش و در کل با میانگین ۵٫۰۱٪ درصد افزایش، عملکرد مناسبی دارد. همچنین با در نظرگرفتن شاخص میانگین زمان حل، تفاوت چندانی بین وجود یک فعالیت نت در مقایسه با حالت عدم وجود فعالیت نت ملاحظه نمی‌شود. بنابراین در حضور فعالیت‌های نت، روش حل پیشنهادی دارای میانگین زمان حل قابل قبولی است.

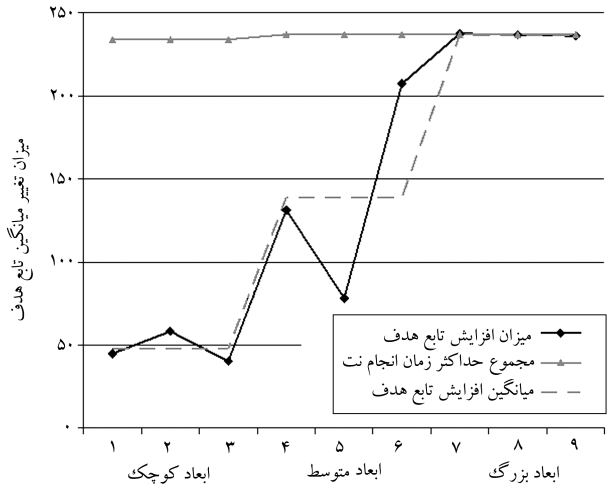
در نمودار ۲ متوسط میزان افزایش میانگین تابع هدف الگوریتم ژنتیک پیشنهادی

جدول ۵. مقایسه‌ی نتایج آزمایش‌های الگوریتم ژنتیک پیشنهادی در حالت R=۰ با R=۲.

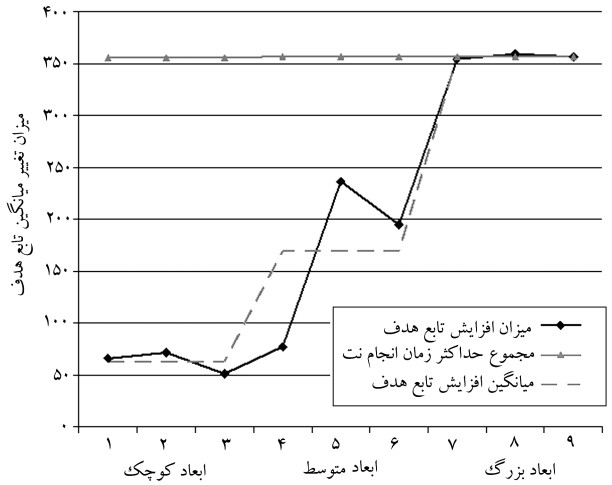
ابعاد مسئله	تعداد کار	تعداد مراحل	روش پیشنهادی در حالت R=۰			روش پیشنهادی در حالت R=۲			میزان افزایش (درصد)		شاخص کارایی الگوریتم ابتکاری نت	
			کمترین تابع هدف (ثانیه)	میانگین تابع هدف (ثانیه)	میانگین زمان حل (ثانیه)	کمترین تابع هدف (ثانیه)	میانگین تابع هدف (ثانیه)	میانگین زمان حل (ثانیه)	کمترین تابع هدف	میانگین تابع هدف	میزان افزایش میانگین تابع هدف	مجموع حداکثر مدت زمان انجام فعالیت‌های نت
کوچک	۶	۲	۱۲۰	۲۱۰٫۳۳	۰٫۱۲	۱۲۴	۲۵۵٫۱۰	۰٫۱۳	٪۳٫۳۳	٪۲۱٫۲۹	۴۴٫۷۸	۲۳۶
		۴	۲۴۶	۳۲۴٫۷۴	۰٫۳۰	۲۶۵	۳۸۲٫۹۸	۰٫۳۶	٪۷٫۷۲	٪۱۷٫۹۳	۵۸٫۲۴	۲۳۴
		۸	۴۸۱	۵۶۷٫۹۰	۰٫۹۹	۵۱۶	۶۰۸٫۱۵	۱٫۲۲	٪۷٫۲۸	٪۷٫۰۹	۴۰٫۲۵	۲۳۳
	میانگین بعد کوچک		۲۸۲٫۳۳	۳۶۷٫۶۵	۰٫۴۷	۳۰۱٫۱۶۷	۴۱۵٫۴۱	۰٫۵۷	٪۶٫۸۵	٪۱۲٫۹۹	۴۷٫۷۵	۲۳۴
متوسط	۳۰	۲	۲۲۴	۸۱۱٫۵۰	۱٫۵۱	۳۱۲	۹۴۲٫۸۰	۱٫۷۴	٪۳۹٫۲۹	٪۱۶٫۱۸	۱۳۱٫۳۰	۲۳۷
		۴	۳۰۸	۹۰۶٫۱۸	۳٫۵۹	۴۰۷	۹۸۴٫۴۳	۳٫۶۵	٪۳۲٫۱۴	٪۸٫۶۴	۷۸٫۲۶	۲۳۴
		۸	۵۵۶	۱۲۰۲٫۷۶	۲۰٫۶۶	۶۱۷	۱۴۱۰٫۰۸	۲۱٫۹۲	٪۱۰٫۹۷	٪۱۷٫۲۴	۲۰۷٫۳۱	۲۴۰
	میانگین بعد متوسط		۳۶۲٫۶۷	۹۷۳٫۴۸	۸٫۵۹	۴۴۵٫۳۳	۱۱۱۲٫۴۴	۹٫۱۰	٪۲۲٫۷۹	٪۱۴٫۲۷	۱۲۸٫۹۶	۲۳۷
بزرگ	۱۰۰	۲	۶۱۰	۲۷۹۱٫۷۹	۱۳٫۷۳	۷۳۴	۳۰۲۹٫۴۳	۱۳٫۰۱	٪۲۰٫۳۳	٪۸٫۵۱	۲۳۷٫۶۴	۲۳۷
		۴	۷۰۵	۲۸۹۱٫۲۴	۴۰٫۳۳	۷۹۷	۳۱۲۸٫۱۸	۴۱٫۲۷	٪۱۳٫۰۵	٪۸٫۲۰	۲۳۶٫۹۴	۲۳۶
		۸	۹۷۶	۳۶۵۰٫۳۶	۲۵۱٫۶۷	۱۱۲۰	۳۸۸۶٫۴۷	۲۵۲٫۶۱	٪۱۴٫۷۵	٪۶٫۴۷	۲۳۶٫۱۱	۲۳۹
	میانگین بعد بزرگ		۷۶۳٫۶۷	۳۱۱۱٫۱۳	۱۰۱٫۹۱	۸۸۳٫۶۷	۳۳۴۸٫۰۳	۱۰۲٫۳۰	٪۱۵٫۷۱	٪۷٫۶۱	۲۳۶٫۹۰	۲۳۷
میانگین کلیه ابعاد			۴۶۹٫۵۶	۱۴۸۴٫۰۹	۳۶٫۹۹	۵۴۳٫۵۶	۱۶۲۵٫۲۹	۳۷٫۳۲	٪۱۵٫۷۶	٪۹٫۵۱	۱۴۱٫۲۰	۲۳۶

جدول ۶. مقایسه‌ی نتایج آزمایش‌های الگوریتم ژنتیک پیشنهادی در حالت R=۰ با R=۳.

ابعاد مسئله	تعداد کار	تعداد مراحل	روش پیشنهادی در حالت R=۰			روش پیشنهادی در حالت R=۳			میزان افزایش (درصد)		شاخص کارایی الگوریتم ابتکاری نت	
			کمترین تابع هدف (ثانیه)	میانگین تابع هدف (ثانیه)	میانگین زمان حل (ثانیه)	کمترین تابع هدف (ثانیه)	میانگین تابع هدف (ثانیه)	میانگین زمان حل (ثانیه)	کمترین تابع هدف	میانگین تابع هدف	میزان افزایش میانگین تابع هدف	مجموع حداکثر مدت زمان انجام فعالیت‌های نت
کوچک	۶	۲	۱۲۰	۲۱۰٫۳۳	۰٫۱۲	۱۲۴	۲۷۶٫۲۳	۰٫۱۳	٪۳٫۳۳	٪۳۱٫۳۳	۰٫۸۱	۱
		۴	۲۴۶	۳۲۴٫۷۴	۰٫۳۰	۲۷۵	۳۹۶٫۴۵	۰٫۳۱	٪۱۱٫۷۹	٪۲۲٫۰۸	۰٫۵۸	۱
		۸	۴۸۱	۵۶۷٫۹۰	۰٫۹۹	۵۰۹	۶۱۹٫۱۵	۱٫۰۴	٪۵٫۸۲	٪۹٫۰۲	۰٫۹۰	۱
	میانگین بعد کوچک		۲۸۲٫۳۳	۳۶۷٫۶۵	۰٫۴۷	۳۰۲٫۶۷	۴۳۰٫۶۱	۰٫۴۹	٪۷٫۲۰	٪۱۷٫۱۲	۰٫۷۶	۱
متوسط	۳۰	۲	۲۲۴	۸۱۱٫۵۰	۱٫۵۱	۳۱۶	۸۸۸٫۷۵	۱٫۴۹	٪۴۱٫۰۷	٪۹٫۵۲	۰٫۶۹	۱
		۴	۳۰۸	۹۰۶٫۱۸	۳٫۵۹	۴۱۲	۱۱۴۲٫۹۳	۳٫۹۳	٪۳۳٫۷۷	٪۲۶٫۱۳	۰٫۹۶	۱
		۸	۵۵۶	۱۲۰۲٫۷۶	۲۰٫۶۶	۶۲۵	۱۳۹۷٫۳۷	۲۱٫۴۴	٪۱۲٫۴۱	٪۱۶٫۲۱	۰٫۷۸	۱
	میانگین بعد متوسط		۳۶۲٫۶۷	۹۷۳٫۴۸	۸٫۵۹	۴۵۱٫۰۰	۱۱۴۳٫۱۳	۸٫۹۵	٪۲۴٫۳۶	٪۱۷٫۴۳	۰٫۸۱	۱
بزرگ	۱۰۰	۲	۶۱۰	۲۷۹۱٫۷۹	۱۳٫۷۳	۷۲۷	۳۱۴۶٫۳۹	۱۴٫۴۹	٪۱۹٫۱۸	٪۱۲٫۷۰	۰٫۹۵	۱
		۴	۷۰۵	۲۸۹۱٫۲۴	۴۰٫۳۳	۸۰۵	۳۲۵۰٫۷۵	۴۰٫۴۶	٪۱۴٫۱۸	٪۱۲٫۴۳	۰٫۷۴	۱
		۸	۹۷۶	۳۶۵۰٫۳۶	۲۵۱٫۶۷	۱۱۱۵	۴۰۰۷٫۲۳	۲۵۳٫۰۱	٪۱۴٫۲۴	٪۹٫۷۸	۰٫۸۵	۱
	میانگین بعد بزرگ		۷۶۳٫۶۷	۳۱۱۱٫۱۳	۱۰۱٫۹۱	۸۸۲٫۳۳	۳۴۶۸٫۱۲	۱۰۲٫۶۵	٪۱۵٫۵۴	٪۱۱٫۴۷	۰٫۸۵	۱
میانگین کلیه ابعاد			۴۶۹٫۵۶	۱۴۸۴٫۰۹	۳۶٫۹۹	۵۴۵٫۳۳	۱۶۸۰٫۶۲	۳۷٫۳۷	٪۱۶٫۱۴	٪۱۳٫۲۴	۰٫۸۱	۱



نمودار ۳. مقایسه‌ی متوسط میزان افزایش میانگین تابع هدف با مجموع بیشینه زمان انجام فعالیت‌های نت در حالت‌های ($R = ۰$) و ($R = ۲$).



نمودار ۴. مقایسه‌ی متوسط میزان افزایش میانگین تابع هدف با مجموع بیشینه زمان انجام فعالیت‌های نت در حالت‌های ($R = ۰$) و ($R = ۳$).

با مجموع بیشینه مدت زمان انجام فعالیت‌های نت بر روی ماشین‌ها در حالت‌های ($R = ۰$) و ($R = ۱$) برای ابعاد مختلف مسئله مورد مقایسه قرار گرفته‌اند.

به‌همین ترتیب در مقایسه‌ی حالت عدم وجود فعالیت نت با حالت وجود دو و سه فعالیت نت، و با توجه به جداول ۵ و ۶ ملاحظه می‌شود که متوسط میزان افزایش میانگین تابع هدف در مقایسه با مجموع بیشینه زمان انجام فعالیت‌های نت در ابعاد کوچک، متوسط و بزرگ مقادیر کم‌تری دارد و نتایج محاسبات دال بر ناچیزبودن افزایش تابع هدف در حالت وجود دو و سه فعالیت نت است و کارایی روش ترکیبی را تأیید می‌کند. به‌همین ترتیب با در نظر گرفتن شاخص میانگین زمان حل تفاوت چندانی بین وجود دو و سه فعالیت نت در مقایسه با حالت عدم وجود فعالیت نت ملاحظه نمی‌شود. بنابراین در حضور فعالیت‌های نت، روش حل پیشنهادی دارای میانگین زمان حل قابل قبولی است.

در نمودارهای ۳ و ۴، متوسط میزان افزایش میانگین تابع هدف الگوریتم ژنتیک پیشنهادی با مجموع بیشینه زمان انجام فعالیت‌های نت بر روی ماشین‌ها به‌ترتیب در حالت‌های ($R = ۰$) و ($R = ۲$) و نیز ($R = ۰$) و ($R = ۳$) برای ابعاد مختلف مسئله مورد مقایسه قرار گرفته‌اند.

با توجه به نتایج محاسباتی فوق، روش ارائه‌شده براساس شاخص میانگین تابع هدف در حالت وجود یک، دو و سه فعالیت نت به‌ترتیب درکل با میانگین ۵٫۰۱٪، ۹٫۵۱٪ و ۱۳٫۲۴٪ درصد افزایش از عملکرد مطلوبی برخوردار است.

نتیجه‌گیری

در این نوشتار مسئله‌یی مرکب از زمان‌بندی تولید کارکارگاهی انعطاف‌پذیر با در نظر گرفتن امکان وجود ماشین‌های موازی در هر ایستگاه کاری و محدودیت نت برای ماشین‌ها معرفی شد. سپس یک مدل ریاضی از نوع برنامه‌ریزی غیرخطی عدد صحیح مختلط برای مسئله ارائه شد. در ادامه، روشی ترکیبی برای حل مسئله با به‌کارگیری الگوریتم ژنتیک به‌منظور حل دو زیرمسئله‌ی مسیریابی و تعیین توالی با رویکردی یک‌پارچه و الگوریتم ابتکاری کارا برای حل زیرمسئله‌ی زمان‌بندی نت مورد استفاده قرار گرفت. طراحی کروموزوم و انتخاب عملگرهای ژنتیک به‌گونه‌یی صورت گرفت که کم‌ترین مشکل در برخورد با محدودیت‌ها و تولید کروموزوم غیرموجه ایجاد شود. در الگوریتم ژنتیک پیشنهادی، جمعیت اولیه به‌صورت تصادفی تولید شد و فاقد هرگونه اطلاعات در رابطه با مسئله بود. این امر سبب حفظ تنوع کروموزوم‌ها در جمعیت، و کاهش احتمال هم‌گرایی سریع و پیش از موعد شد. همچنین با جلوگیری از ورود کروموزوم‌های تکراری به نسل جدید، مانع از تسلط ابرکروموزوم‌ها بر جمعیت شده و از هم‌گرایی سریع به جواب بهینه‌ی محلی جلوگیری می‌شود. در این تحقیق، اثربخشی و کارایی الگوریتم ژنتیک پیشنهادی در دو حالت ارزیابی شد. در حالت اول، روش ارائه‌شده در حالت عدم انجام فعالیت نت بر روی ماشین‌ها ($R = ۰$) با روش فرابتکاری RKGA مقایسه و نشان داده شد که کارایی روش پیشنهادی برای حل مسئله‌ی تحقیق از روش RKGA بیشتر است.

در حالت دوم، مسئله‌ی تحقیق به‌ترتیب با در نظر گرفتن تعداد یک، دو و سه فعالیت نت بر روی هر ماشین ($R = ۱, ۲, ۳$) به‌کمک الگوریتم ژنتیک پیشنهادی حل، و نتایج حاصل از آن با حالت اول ($R = ۰$) مقایسه شد. در این حالت متوسط میزان افزایش میانگین تابع هدف در مقایسه با شاخص مجموع بیشینه زمان انجام فعالیت‌های نت در ابعاد کوچک، متوسط و بزرگ مقادیر کم‌تری دارد. این امر نشان‌دهنده‌ی تأثیر اندک وجود فعالیت نت درکیفیت جواب، و کارایی روش ترکیبی است. بنابراین، بررسی‌های انجام‌گرفته بیان‌گر این مطلب است که روش حل ارائه‌شده، روشی کارا برای حل مسئله‌ی زمان‌بندی کارکارگاهی انعطاف‌پذیر با در نظر گرفتن محدودیت نت است.

با توجه به نتایج تحقیق حاضر و شناسایی خلاءهای تحقیقاتی موجود، زمینه‌هایی همچون در نظر گرفتن ماشین‌های موازی یکنواخت[۲۲] و غیرمرتبط[۲۳] در هر ایستگاه کاری، وجود پنجره‌های زمانی نرم[۲۴] به‌منظور انجام فعالیت‌های نت، در نظر گرفتن تغییر در سرعت ماشین‌ها پس از انجام فعالیت‌های نت، اعمال هم‌زمان محدودیت دسترسی به‌صورت قطعی و تصادفی، استفاده از راهکار جریمه‌یی در الگوریتم ژنتیک برای جست‌وجوی قسمت‌های بیشتری از ناحیه‌ی جواب به‌منظور انجام تحقیقات آتی پیشنهاد می‌شوند.

پانوشت

1. flexible job shop (FJS)
2. non-fixed availability
3. preemption
4. no resumption (N-R)
5. symbiotic evolutionary algorithm
6. fixed availability constraint
7. semi preemption
8. No-preemption
9. resumption (R)
10. semi resumption (S-R)
11. identical parallel machines
12. permutation presentation
13. position based crossover
14. order based crossover
15. uniform mutation
16. reciprocal exchange mutation
17. super chromosome
18. rejecting strategy
19. repairing strategy
20. penalizing strategy
21. modifying genetic operations strategy
22. uniform parallel machines
23. unrelated parallel machines
24. soft time windows

منابع

1. Zelenovic, D. "Flexibility - a condition for effective manufacturing system", *International Journal of Production Research,* **20**(3), pp. 319-337 (1982).
2. Mandelbaum, M. "Flexibility and decision-making", *European Journal of Operational Research,* **44**, pp. 17-27 (1990).
3. Torabi, S.A.; Karimi, B., and Fatemi-Ghomi, S.M.T. "The common cycle economic lot scheduling in flexible job shops: The finite horizon case", *International Journal of Foundations of Production Economics,* **97**, pp. 52-65 (2005).
4. Allaoui, H., and Artiba, A. "Scheduling two-stage hybrid flow shop with availability constraints", *Computers & Operations Research,* **33**, pp. 1399-1419 (2004).
5. Bruker, P., and Schlie, R. "Job-shop scheduling with multi-purpose machines", *Computing,* **45**, pp. 369-375 (1990).
6. Xia, W., and Wu, Z. "An effective hybrid optimization approach for multi-objective flexible job-shop scheduling problems", *Computers & Industrial Engineering,* **48**, pp. 409-425 (2005).
7. Brandimarte P. "Routing and scheduling in a flexible job shop by taboo search", *Annals of Operations Research,* **41**, pp. 157-183 (1993).
8. Jansen K. Mostrollili, M., and Solis-Oba, R. "Approximation algorithms for fexible job-shop problem", *International Journal of Foundations of Computer Science,* (16), pp.361-379 (2005).
9. Ho, N.B., and Tay, J.C. "Evolving dispatching rules for solving the flexible job shop problem", Congress on Evolutionery Compution, pp.2848-2855 (2005).
10. Kacem, I.; Hammadi, S., and Borne P. "Approach by localization and multiobjective evolutionary optimization for flexible job-shop scheduling problems", *IEEE Transactions on Systems, Man, and Cybernetics,* **32**, pp. 1-13 (2002).
11. Hurink, E.; Jurisch, B., and Thole, M. "Tabu search for the job shop scheduling problem with multi-purpose machines", *Operations Research Spectrum,* **15**, pp. 205-215 (1994).
12. Mastrolilli, M., and Gambardella, L.M. "Effective neighborhood functions for the flexible job shop problem", *Journal of Scheduling,* **3**, pp. 3-20 (2002).
13. Kim, Y.K.; Park, K., and Ko, J. "A symbiotic evolutionary algorithm for the integration of process planning and job shop scheduling", *Computers & Operations Research,* **30**, pp. 1151-1171 (2003).
14. Ho, N.B.; Tay, J.C., and Lai, E.M. "An effective architecture for learning and evolving flexible job-shop schedules", *European Journal of Operational Research,* **179**, pp. 316-333 (2006).
15. Saidi-Mehrabad, M., and Fattahi, P. "Flexible job shop scheduling with tabu search algorithms", *Int Journal of Advanced Manufacturing Technology,* **32**, pp. 563-570 (2006).
16. Rossi, A., and Dini, G. "Flexible job-shop scheduling with routing flexibility and separable setup times using ant colony optimisation method", *Robotics and Computer-Integrated Manufacturing,* **23**, pp. 503-516 (2006).
17. Pezzella, F.; Morganti, G., and Ciaschetti, G. "A genetic algorithm for the flexible Job-shop scheduling problem", *Computers & Operations Research,* pp. 1-11 (2007).
18. Gao, J.; Sun, L. and Gen, M. "A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems", *Computers & Operations Research,* pp. 1-16 (2007).
19. Gao, J.; Gen, M., and Sun, L. "Scheduling jobs and maintenances in flexible job shop with a hybrid genetic algorithm", *Journal of Intelligent Manufacturing,* **17**, pp. 493-507 (2006).
20. Aggoune, R., and Portmann, M.C. "Flow shop scheduling problem with limited machine availability A heuristic approach", *Int Journal of Production Economics,* **99**, pp. 4-15 (2005).
21. Gharbi, A., and Haouari, M. "Optimal parallel machines scheduling with availability constraints", *Discrete Applied Mathematics,* **148**, pp. 63-87 (2005).
22. Aggoune, R. "Minimizing the makespan for the flow shop scheduling problem with availability constraints", *European Journal of Operational Research,* **153**, pp. 534-543 (2004).
23. Lee C. Y. "Two-machine flowshop scheduling with availability constraints", *European Journal of Operational Research,* **114**, pp. 420-429 (1997).
24. Schmidt G. "Scheduling on semi-identical processors", *Zeitschrift fur Operations Research,* **28**, pp.153-162 (1984).

25. Adiri, I.; Bruno, J.; Frostig, E., and Rinnooy-Kan, A.H.G. "Single machine flow-time scheduling with a single breakdown", *Acta Informatica,* **26**, pp. 679-696 (1989).

26. Lee, C.Y. "Parallel machines scheduling with non-simultaneous machine available time", *Discrete Applied Mathematics and Computers in Simulation,* **30**, pp. 53-61 (1991).

27. Lee, C.Y. "Two-machine flowshop scheduling with availability constraints", *European Journal of Operational Research,* **114**, pp. 420-429 (1999).

28. Schmidt, G. "Scheduling with limited machine availability", *European Journal of Operational Research,* **121**, pp. 1-15 (2000).

29. Espinouse, M.L.; Formanowicz, P., and Penz, B. "Complexity results on and approximation algorithms for the two machine no-wait flow shop with limited machine availability", *Journal of Operational Research Society,* **52**, pp. 116-121 (2001).

30. Xie, J., and Wang, X. "Branch and bound algorithm for flexible flowshop with limited machine availability", *Asian Information Science Life,* **1**, pp. 241-248 (2002).

31. Cheng, T.C.E., and Liu, Z. "Approximability of two-machine flowshop scheduling with availability constraints", *Operations Research Letters,* **31**, pp. 319-322 (2003).

32. Kubzin, M.A., and Strusevich, V.A. "Two-machine flow shop no-wait scheduling with machine maintenance", *A Quarterly Journal of Operations Research,* **3**, pp. 303-313 (2005).

33. Xie, J., and Wang, X. "Complexity and algorithms for two-stage flexible flowshop scheduling with availability constraints", *Computers and Mathematics with Applications,* **50**, pp. 1629-1638 (2005).

34. Wu, C.C., and Lee, W.C. "A note on single-machine scheduling with learning effect and an availability constraint", *Int. J. of Advance Manufacturing Technology,* (33), pp.540-544 (2007).

35. Zribi, N.; Kamel, A.E., and Borne, P. "Minimizing the makespan for the MPM job-shop with availability constraints", *Int Journal of Production Economics,* (112),pp. 151-160 (2008).

36. Qi, X.; Chen, T., and Tu, F. "Scheduling the maintenance on a single machine", *Journal of the Operational Research Society,* **50**, pp. 1071-1078 (1999).

37. Graves, G.H., and Lee, C.Y. "Scheduling maintenance and semiresumable jobs on a single machine", *Naval Research Logistics,* **46**, pp. 845-863 (1999).

38. Lee, C.Y., and Chen, Z.L. "Scheduling of jobs and maintenance activities on parallel machines", *Naval Research Logistics,* **47**, pp. 145-165 (2000).

39. Breit, J. "Improved approximation for non-preemptive single machine flow-time scheduling with an availability constraint", *European Journal of Operational Research,* (183), pp.516-524 (2007).

40. Graham, R.L.; Lawler, E.L.; Lenstra, J.K., and Kan, A.H.K.R. "Optimization and approximation in deterministic sequencing and scheduling: A Survey", *Annuals of discrete mathematics,* **5**, pp. 287-326, (1979).

41. Garey, M.R.; Johnson, D.S., and Sethi R. "The complexity of flowshop and job-shop scheduling", *Mathematics of Operations Research Letters,* **1**, pp. 117-129 (1976).

42. Gen, M., and Cheng R. *Genetic algorithms and engineering design,* John Wiley & sons (1997).

43. Kurz, M.E., and Askin, R.G. "Scheduling flexible flow lines with sequence-dependent setup times", *European Journal of Operational Research,* **159**, pp. 66-82 (2004).