



A Genetic Local Search Algorithm for Generalized Job Shop Scheduling Problem with Controllable Processing Times

M.R. Amin-Naseri* & A.J. Afshari

Mohammad Reza Amin-Naseri, Associate Professor Tarbiat Modares University, Industrial Engineering Dep, Tarbiat Modares University,
Ahmad J. Afshari, Assistant Professor Shomal University, Industrial Engineering Dep, Shomal University

Keywords

Flexible job shop scheduling problem,
Controllable processing times,
Alternative process plans,
Genetic local search,
Dual-resource constrained scheduling

ABSTRACT

Although incorporating complexities and flexibilities of real world manufacturing systems into classic scheduling problems results in problems with greater complexity, it has immense theoretical and practical importance due to its impressive effect on system performance. In this research, three basic assumptions of a job shop scheduling problem have been revised to develop a model with three types of flexibility which fits into many cases of real manufacturing environments; sequencing flexibility, machine flexibility, and processing time flexibility. In this paper, after describing the problem and formulating it as a mathematical program, we have proposed an efficient genetic local search algorithm to solve the problem. In the proposed algorithm first, problem-specific elements including population initialization, crossover, and mutation are designed for genetic algorithm. Then, a novel local search procedure has been developed and incorporated into genetic algorithm to enhance its search ability. In experimental study, the performance of the proposed algorithm to find optimal and near-optimal solutions has been demonstrated based on some test problems which are built by incorporating processing time flexibility into a number of problems adopted from the literature.

© 2013 IUST Publication, IJIEPM. Vol. 24, No. 1, All Rights Reserved

*
Corresponding author. Mohammad Reza Amin-Naseri
Email: amin_nas@modares.ac.ir



مدل حل مبتنی بر جستجوگر محلی ژنتیک برای مساله زمان بندی استقرار کارگاهی تعمیم یافته با زمانهای عملیات قابل کنترل

احمد جعفرزاده افشاری و محمدرضا امین ناصری*

چکیده:

افزودن پیچیدگی‌ها و انعطاف پذیری‌های موجود در سامانه‌های تولید واقعی به مفروضات مدل‌های کلاسیک زمان بندی، گرچه منجر به افزایش قابل توجه پیچیدگی مسائل می‌گردد، اما به دلیل بهبود چشمگیری که در عملکرد این سامانه‌ها ایجاد می‌کند، از دو جنبه نظری و کاربردی حائز اهمیت است. در این پژوهش سه فرض اصلی مساله زمان بندی استقرار کارگاهی مورد بازنگری قرار گرفته که حاصل آن مدلی است با سه نوع انعطاف و قابل انطباق با بسیاری از محیط‌های تولید واقعی؛ انعطاف در توالی، انعطاف در ماشین و انعطاف در زمان عملیات. پس از تعریف مساله و بیان آن در قالب یک مدل ریاضی، یک روش حل کارا مبتنی بر جستجوگر محلی ژنتیک برای آن توسعه داده شده است. به این منظور ابتدا عملگرها و رویه‌های ویژه‌ای، منطبق با مختصات مساله، شامل ایجاد جمعیت اولیه، تقاطع و جهش برای الگوریتم ژنتیک طراحی شده‌اند. سپس یک تابع جستجوگر محلی بدیع با هدف ارتقا توان جستجوی الگوریتم ژنتیک توسعه یافته و با آن تلفیق شده است. در بخش نتایج عددی، کارآمدی الگوریتم پیشنهادی در یافتن جواب‌های بهینه و نزدیک به بهینه با حل تعدادی مساله نمونه که با افزودن قابلیت انعطاف در زمان عملیات به مسائل موجود در ادبیات به دست آمده‌اند، نشان داده شده است.

کلمات کلیدی

زمان بندی استقرار کارگاهی منعطف،
زمان‌های عملیات قابل کنترل،
طرح‌های فرایند چندگانه،
جستجوی محلی ژنتیک،
زمان بندی محدود به منابع دوگانه

۱. مقدمه

افزایش قابلیت انعطاف از جمله مؤثرترین روش‌ها برای افزایش چابکی، بهره‌وری و کیفیت سامانه‌های تولیدی در مواجهه با شرایط رقابتی محیط کسب و کار و تقاضاهای پویای بازار می‌باشد.

تاریخ وصول: ۸۹/۱۱/۱۶

تاریخ تصویب: ۹۰/۸/۷

*نویسنده مسئول مقاله: دکتر محمدرضا امین ناصری، دانشیار دانشگاه تربیت مدرس، دانشگاه تربیت مدرس، دانشکده مهندسی صنایع،
amin_nas@modares.ac.ir

احمد جعفرزاده افشاری، دانشجوی دکترا دانشگاه تربیت مدرس - تهران
بزرگراه جلال آل احمد دانشگاه تربیت مدرس دانشکده مهندسی گروه مهندسی صنایع

به موازات تحقق سطوح بالاتری از انعطاف پذیری در این سامانه‌ها، لازم است مدل‌های نظری در حوزه برنامه ریزی تولید نیز ارتقا یابند. در این مقاله، یکی از مسائل مطرح در بسیاری از محیط‌های عملیاتی دنیای واقعی که تاکنون در هیچ پژوهشی بررسی نگردیده، تعریف و مدل‌سازی شده و یک الگوریتم حل کارآمد برای آن پیشنهاد شده است.

مساله زمان بندی استقرار کارگاهی^۲ یکی از مسائل دشوار در حوزه پژوهش‌های زمان بندی بوده و در دسته مسائل NP-Hard قرار می‌گیرد [۱]. در این پژوهش، برخی از مفروضات این مساله به منظور ارتقا کاربرد پذیری آن در دنیای واقعی بازنگری شده‌اند

² Job Shop Scheduling Problem (JSP)

کاربردهای بسیار در محیط‌های واقعی می‌باشد که مثال‌های جالبی از کاربرد آن در یک کارخانه فولاد و یک مرکز ارائه دهنده خدمات اینترنتی^{۱۱} در [۱۱] و [۱۲] آمده است. در اغلب پژوهش‌های قبلی، این منابع پیوسته فرض شده‌اند (پول، انرژی، سوخت) [۱۳]. اما با توجه به این که رایج‌ترین منبع منعطف مورد استفاده در مراکز تولیدی و خدماتی یعنی نیروی انسانی یک منبع گسسته است [۱۴] و مسائل CPT با منابع گسسته علاوه بر محیط‌های تولید، در حوزه‌های خدماتی از قبیل تعمیرات و نگهداری، عملیات انبار و مراکز درمانی کاربرد دارند [۱۵]، نیاز به پژوهش در این زمینه احساس می‌شود. رویکرد رایج در معدود کارهای انجام شده در این حوزه، این است که یک مدل استاندارد زمان بندی در نظر گرفته شده و به مفروضات آن امکان کنترل زمان عملیات افزوده می‌شود. در این زمینه می‌توان به مدل‌های زمان بندی بر روی ماشین‌های موازی [۱۴]، زمان بندی جریان کارگاهی [۱۵] و زمان بندی استقرار کارگاهی [۱۶] با زمان‌های قابل کنترل اشاره کرد.

نوآوری اصلی پژوهش حاضر، ارائه یک روش حل کارا برای سامانه‌های تولید انعطاف پذیر با در نظر گرفتن هم‌زمان انعطاف در ماشین، توالی عملیات و زمان‌های پردازش می‌باشد که علیرغم کاربردهای بسیار آن در محیط‌های خدماتی و تولیدی واقعی، تاکنون در هیچ پژوهشی مورد مطالعه قرار نگرفته است. به عنوان نمونه‌ای واقعی از این مساله می‌توان به مراکز بزرگ ارائه دهنده خدمات پس از فروش خودرو اشاره کرد که هر مجموعه از خدمات متجانس توسط تعدادی از ایستگاه‌های کاری قابل انجام است (انعطاف در ماشین)، در توالی عملیات‌های هر خودرو بر اساس روابط پیش‌نیاز فنی بین عملیات‌ها انعطاف وجود دارد (انعطاف در توالی عملیات) و زمان انجام عملیات در هر ایستگاه را می‌توان با افزایش تعداد پرسنل تخصیص داده شده به آن کاهش داد. در مدل‌های قبلی توسعه داده شده برای انعطاف در توالی عملیات، فرض بر این است که همه طرح‌های فرایند از قبل تولید شده و معلوم می‌باشند ([۳]، [۱۷]، [۱۸] و [۱۹]).

با توجه به تعدد طرح‌های فرایند موجه در مسائل واقعی، تولید صریح و در نظر گرفتن آن‌ها به عنوان داده‌های مساله، منجر به بزرگ شدن ابعاد مساله می‌گردد. در روش حل توسعه یافته در این مقاله، تنها کافی است که روابط پیش‌نیاز بین عملیات‌ها معلوم باشد و عملگرهای ژنتیک و جستجوگر محلی به گونه‌ای طراحی شده‌اند که همه طرح‌های فرایند ممکنه به صورت ضمنی در نظر گرفته شده و طرح فرایند بهینه در حین حل مساله تعیین می‌گردد. حل یک مساله FJSP با زمان‌های عملیات قابل کنترل

که حاصل یک مدل JSP با سه نوع انعطاف می‌باشد: انعطاف در ماشین، انعطاف در توالی و انعطاف در زمان‌های انجام عملیات. در ادبیات زمان بندی، انعطاف در ماشین به مفهوم امکان انجام یک عملیات بر روی چند ماشین می‌باشد. بنابراین مساله زمان بندی استقرار کارگاهی منعطف^۱ عبارتست از یک مساله JSP که در آن هر عملیات با ماشین‌های مختلف قابل انجام است. در این حالت مسیریابی روی ماشین‌ها به ابعاد مساله زمان بندی اضافه می‌شود [۲]. مساله FJSP یک مساله شمارشی بسیار دشوار است که برای حل آن الگوریتم‌های مختلفی ارائه شده است. در مقاله یزدانی و همکاران، روش‌های حل توسعه داده شده برای این مساله مرور شده‌اند [۲].

انعطاف در توالی^۲ یعنی امکان جابجا کردن ترتیب عملیات‌های مربوط به یک کار که در نتیجه آن یک محصول می‌تواند دارای طرح‌های فرایند چندگانه^۳ باشد [۳]. طرح‌ریزی فرایند، یکی از وظایف اصلی سامانه‌های تولیدی است که رویه‌های عملیاتی جهت تبدیل یک طرح مهندسی به محصول نهایی را تعیین می‌کند [۴]. یکپارچه سازی تصمیمات مربوط به طرح‌ریزی فرایند و زمان بندی، منجر به بهبودهایی چشمگیر در عملکرد مراکز تولیدی می‌شود که انعطاف در توالی عملیات از جمله آن‌هاست [۵].

از زمانی که چریسلوریس و همکارانش برای اولین بار ایده ادغام طرح‌ریزی فرایند و زمان بندی^۴ را مطرح کردند تاکنون کارهای زیادی در این مورد انجام شده است [۶]. اغلب روش‌های حل پیشنهادی، روش‌های ابتکاری و فراابتکاری مبتنی بر هوش مصنوعی مانند الگوریتم ژنتیک^۵ [۷]، تبرید شبیه‌سازی شده^۶ [۸]، بهینه سازی گروه ذرات^۷ [۴]، سامانه چندعامله^۸ [۹]، بهینه سازی کولونی مورچگان^۹ [۱۰] می‌باشند. لی و همکارانش مرور جامعی بر کارهای انجام شده در حوزه IPPS انجام داده‌اند [۵]. انعطاف پذیری زمان‌های عملیات، ویژگی دیگر مساله مورد بررسی می‌باشد.

به مسائلی که در آن‌ها می‌توان با تخصیص منابعی منعطف به منابع ایستا (ایستگاه‌های کاری) زمان عملیات را کاهش داد، مدل‌های زمان بندی با زمان‌های قابل کنترل^{۱۰} گفته می‌شود. کنترل زمان‌های عملیات با تخصیص منابع بیشتر، دارای

¹ Flexible Job Shop Scheduling Problem (FJSP)

² Sequencing Flexibility (SF)

³ Alternative Process Plans (APP)

⁴ Integrated Process Planning & Scheduling (IPPS)

⁵ Genetic Algorithm (GA)

⁶ Simulated Annealing (SA)

⁷ Particle Swarm Optimization (PSO)

⁸ Multi-Agent System (MAS)

⁹ Ant Colony Optimization (ACO)

¹⁰ Controllable Processing Time (CPT)

¹¹ Internet Service Provider (ISP)

سایر مفروضات FJSP-APP-CPT همانند مسأله FJSP می باشد [۲]. در شکل (۱) ابعاد تصمیم در یک مسأله FJSP-APP-CPT و داده های یک مسأله نمونه که در تشریح الگوریتم پیشنهادی استفاده خواهد شد، ارائه گردیده است. مسأله از سه کار و مجموعاً ۹ عملیات که باید بر روی ۴ ماشین (ایستگاه کاری) زمان بندی شوند تشکیل شده و ۱۲ منبع منقطع برای تخصیص به این ۴ ماشین در دسترس است.

۲-۲. مدل ریاضی مسأله

در این بخش مدل ریاضی مسأله FJSP-APP-CPT توسعه داده شده است. این مدل برای یافتن جواب بهینه مسائل کوچک و بررسی عملکرد جستجوگر محلی ژنتیک بکار گرفته خواهد شد.

پارامترهای مدل:

N مجموعه کارها؛ $n = |N|$

M مجموعه ماشینها؛ $m = |M|$

O_i مجموعه عملیاتهای کار i ؛ $n_i = |O_i|$

o_{ij} عملیات i ام از کار i ام

$M_{ij} \subseteq M$ ؛ o_{ij} مجموعه ماشینهای ممکنه برای o_{ij}

P_i مجموعه زوجهای مرتب که نشان دهنده روابط پیش نیازی

مربوط به کار i ام می باشد؛ $\langle j, j' \rangle \in P_i$ اگر و تنها اگر

o_{ij} پیش نیاز $o_{ij'}$ باشد.

Q_i مجموعه زوجهای مرتب شامل عملیاتهایی از کار i ام که با

هم رابطه پیش نیازی ندارند؛ $\langle j, j' \rangle \in Q_i$ اگر و تنها

اگر o_{ij} و $o_{ij'}$ با هم رابطه پیش نیازی نداشته باشند

R تعداد کل منابع منقطع

α_k کمینه منبع منقطع مورد نیاز در ایستگاه کاری k ام

β_k بیشینه منبع منقطع قابل تخصیص به ایستگاه کاری k ام

q_{kl} ضریب کاهش زمان عملیات در ایستگاه k ام در صورت

استفاده از l منبع منقطع در این ایستگاه - به این مقدار نرخ

تسریع^۵ گفته می شود [۱۴]

pt_{ijk} زمان پایه انجام o_{ij} اگر بر روی ماشین k انجام گیرد.

H عددی مثبت و به اندازه کافی بزرگ

متغیرهای تصمیم:

st_{ijk} زمان شروع o_{ij} بر روی ماشین k

ct_{ijk} زمان پایان o_{ij} بر روی ماشین k

x_{ijk} اگر o_{ij} بر روی ماشین k انجام گیرد؛ در غیر این صورت

یکی از خروجیهای جانبی این پژوهش می باشد که قبلاً در هیچ پژوهشی بدان پرداخته نشده است. در مسأله مورد مطالعه کمینه سازی دوره ساخت به عنوان تابع هدف در نظر گرفته شده است. این تابع هدف در مسائل دربرگیرنده منابع منقطع حائز اهمیت است، زیرا اتمام کارها را در زودترین زمان ممکن الزامی ساخته و بارکاری را به طور یکنواخت روی منابع توزیع می کند [۲۰].

در ادامه مقاله در بخش ۲، مسأله تعریف شده و یک مدل ریاضی برای آن ارائه می گردد. برای حل مسأله، یک الگوریتم ژنتیک و یک جستجوگر محلی توسعه داده شده و با هم تلفیق شده اند. حاصل یک جستجوگر محلی ژنتیک^۱ است که در بخش ۳ تشریح شده است. در بخش ۴ عملکرد الگوریتم پیشنهادی بر اساس تعدادی مسأله نمونه ارزیابی می شود. بالاخره بخش ۵، شامل جمع بندی و پیشنهادهایی برای پژوهشهای آتی است.

۲. تعریف مسأله

۲-۱. مفروضات مسأله

مسأله مورد بررسی در این پژوهش، عبارتست از یک مسأله زمان بندی استقرار کارگاهی منقطع با طرحهای فرایند چندگانه و زمانهای عملیات قابل کنترل^۲. این مسأله تعمیمی است از مسأله زمان بندی استقرار کارگاهی منقطع با طرحهای فرایند چندگانه^۳ که خود تعمیمی است از مسأله FJSP. مسأله FJSP-APP از n کار و هر کار از تعدادی عملیات تشکیل گردیده که باید بر روی m ماشین زمان بندی شوند. هر عملیات بر روی زیرمجموعه ای از این ماشینها قابل انجام است (انعطاف در ماشین). همچنین برای عملیاتهای هر کار چندین توالی شدنی وجود دارد (انعطاف در توالی عملیات).

توالیهای ممکنه (طرحهای فرایند موجه) برای هر کار را می توان با استفاده از روابط پیش نیازی بین عملیاتها تولید نمود. در مسأله FJSP-APP-CPT زمان انجام عملیات در یک ایستگاه کاری تابعی است از منابع منقطع تخصیصی به آن ایستگاه (انعطاف در زمان عملیات). منابع منقطع (ثانویه)، هزینههای شناور^۴ فرض می شوند؛ یعنی منابع تجدید پذیری که مجموع تعداد در دسترس از آنها مقداری است ثابت [۱۵]. تخصیص منابع ثانویه به صورت ایستا انجام می شود، یعنی تعداد منابع منقطع در هر ایستگاه کاری در طول افق زمان بندی ثابت است.

¹ Genetic Local Search (GLS)

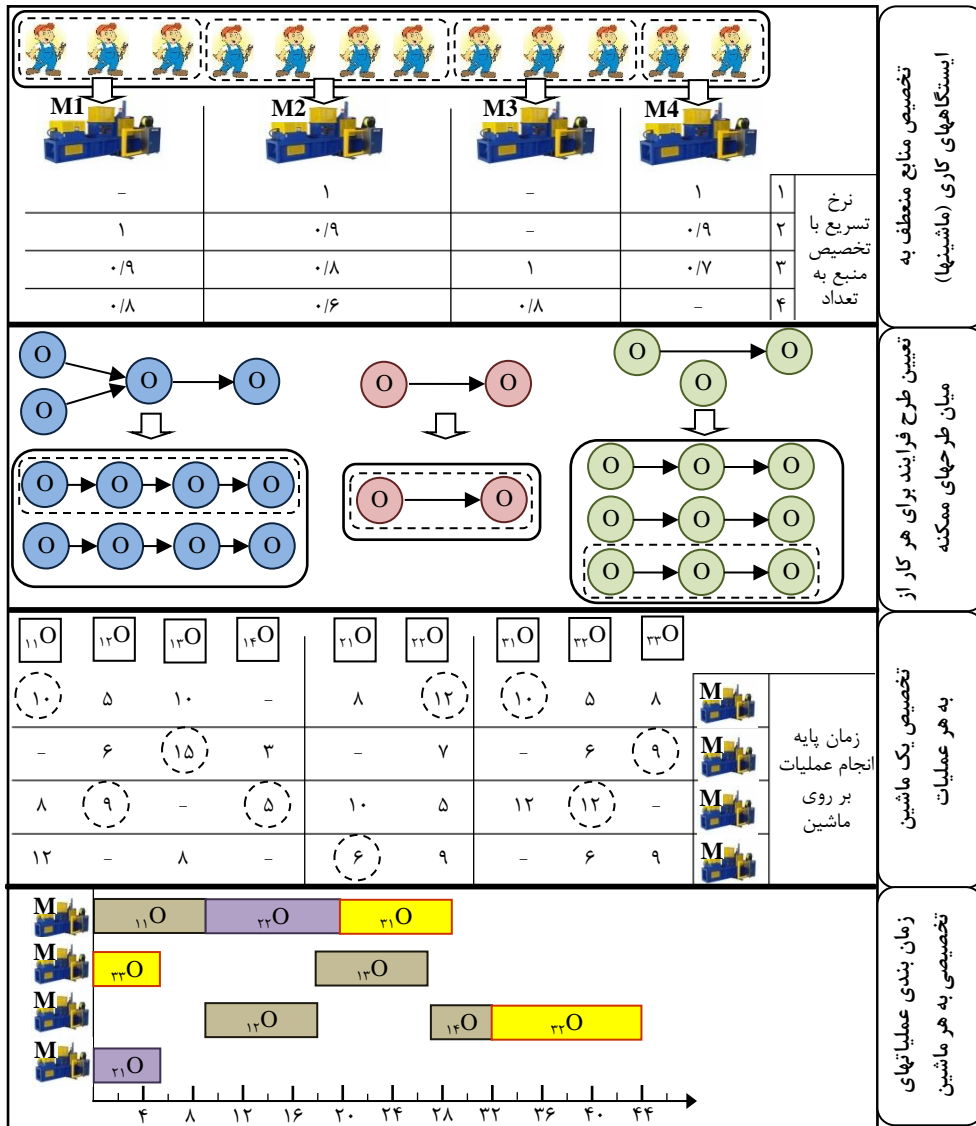
² Flexible Job Shop Scheduling Problem with Alternative Process Plans and Controllable Processing Times (FJSP-APP-CPT)

³ Flexible Job Shop Scheduling Problem with Alternative Process Plans (FJSP-APP)

⁴ sunk cost

⁵ speed rate

اگر $u_{ij}^k = 1$ اگر o_{ij} قبل از o_{ij}' انجام گیرد در حالی که هر دو روی ماشین k انجام می گیرند؛ ۰ در غیر این صورت
 اگر $v_{ij} = 1$ اگر o_{ij}' قبل از o_{ij} انجام شود؛ ۰ در غیر این صورت
 اگر $z_{kl} = 1$ اگر منبع منقطع به ایستگاه کاری k تخصیص یابد؛ ۰ در غیر این صورت:



شکل ۱. ابعاد تصمیم در یک مساله FJSP-APP-CPT

$$ct_{ijk} \leq H \cdot x_{ijk} \quad \forall i \in N \quad \forall j \in O_i \quad \forall k \in M_{ij} \quad (4)$$

$$z_{k0} + \sum_{l=\alpha_k}^{\beta_k} z_{kl} = 1 \quad \forall k \in M \quad (5)$$

$$\sum_{k \in M} \sum_{l=\alpha_k}^{\beta_k} l \cdot z_{kl} \leq R \quad (6)$$

$$ct_{ijk} - st_{ijk} = pt_{ijk} \cdot \left(\sum_{l=\alpha_k}^{\beta_k} q_{kl} \cdot z_{kl} + H \cdot z_{k0} \right) \cdot x_{ijk} \quad \forall i \in N \quad \forall j \in O_i \quad \forall k \in M_{ij} \quad (7)$$

C_{max} دوره ساخت^۱

تابع هدف و محدودیت ها:

$$\min C_{max} \quad (1)$$

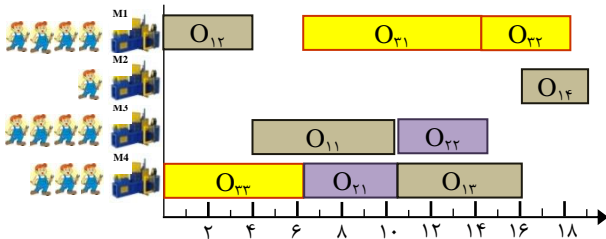
subject to:

$$C_{max} \geq \sum_{k \in M_{ij}} ct_{ijk} \quad \forall i \in N \quad \forall j \in O_i \quad (2)$$

$$\sum_{k \in M_{ij}} x_{ijk} = 1 \quad \forall i \in N \quad \forall j \in O_i \quad (3)$$

^۱ makespan

موجود تجاوز نکند. محدودیت (۷) بیانگر رابطه بین زمان شروع و پایان هر عملیات است. محدودیت‌های انفصالی^۱ (۸) و (۹) مانع از انجام هم‌زمان دو عملیات مربوط به دو کار متفاوت بر روی یک ماشین می‌شوند. محدودیت‌های انفصالی^{۱۰} و (۱۱) نیز مانع از انجام هم‌زمان دو عملیات مربوط به یک کار می‌شوند. محدودیت (۱۲) باعث می‌شود که عملیات j' پس از j زمان بندی گردد اگر j' پیش نیاز j باشد. محدودیت‌های (۱۳) تا (۱۹) نیز برای تعریف متغیرهای دودویی و متغیرهای غیر منفی استفاده شده‌اند. مدل ریاضی مساله ارائه شده در شکل (۱) بر اساس روابط (۱) تا (۱۹) شامل ۱۶۶ متغیر و ۲۴۰ محدودیت خواهد بود که جواب بهینه حاصل از حل آن در شکل (۲) آمده است.



شکل ۲. جواب بهینه مساله شکل (۱) با $C_{max} = 19/1$

۳. الگوریتم جستجوی محلی ژنتیک

در این بخش ابتدا مؤلفه‌های الگوریتم ژنتیک شامل طرح بازنمایی پاسخ‌ها، تولید جمعیت ابتدایی، تابع برازندگی، عملگرهای تقاطع و جهش و سپس رویه جستجوی محلی و تابع همسایگی بکار برده شده در آن تشریح می‌شود. در طراحی این عملگرها و توابع، تلفیق آن‌ها با دانش مساله مدنظر بوده که بنا بر شواهد نظری و تجربی، منجر به ارتقا عملکرد الگوریتم می‌گردد [۲۱].

۳-۱ الگوریتم ژنتیک

۳-۱-۱ بازنمایی و کدینگ جواب‌ها

در الگوریتم پیشنهادی جهت رمز کردن جواب‌ها، روش فهرست توالی وظایف^۲ بسط داده شده است. هر جواب از دو کروموزوم با طول‌های متفاوت تشکیل شده است؛ کروموزوم A شامل NO سه‌گانه مرتب می‌باشد که NO عبارتست از مجموع تعداد عملیات‌ها. سه‌گانه مرتب (i, j, k) بیانگر تخصیص ماشین k به عملیات O_{ij} می‌باشد. عملیات ظاهر شده در مکان h دارای اولویت h می‌باشد. تعداد ژن‌های کروموزوم B برابر است با تعداد

$$st_{ij'k} - ct_{ijk} + H.u_{ij'j'k} \geq 0 \quad \forall i, i' \in N, i < i' \quad \forall j \in O_i \quad \forall j' \in O_{i'} \quad \forall k \in M_{ij} \cap M_{ij'} \quad (8)$$

$$st_{ijk} - ct_{ij'k} + H.(1 - u_{ij'j'k}) \geq 0 \quad \forall i, i' \in N, i < i' \quad \forall j \in O_i \quad \forall j' \in O_{i'} \quad \forall k \in M_{ij} \cap M_{ij'} \quad (9)$$

$$\sum_{k' \in M_{ij'}} st_{ij'k'} - \sum_{k \in M_{ij}} ct_{ijk} + H.v_{ijj'} \geq 0 \quad \forall i \in N \quad \forall \langle j, j' \rangle \in Q_i \quad (10)$$

$$\sum_{k \in M_{ij}} st_{ijk} - \sum_{k' \in M_{ij'}} ct_{ij'k'} + H.(1 - v_{ijj'}) \geq 0 \quad \forall i \in N \quad \forall \langle j, j' \rangle \in Q_i \quad (11)$$

$$\sum_{k' \in M_{ij'}} st_{ij'k'} - \sum_{k \in M_{ij}} ct_{ijk} \geq 0 \quad \forall i \in N \quad \forall \langle j, j' \rangle \in P_i \quad (12)$$

$$ct_{ijk} \geq 0 \quad \forall i \in N \quad \forall j \in O_i \quad \forall k \in M_{ij} \quad (13)$$

$$st_{ijk} \geq 0 \quad \forall i \in N \quad \forall j \in O_i \quad \forall k \in M_{ij} \quad (14)$$

$$x_{ijk} \in \{0,1\} \quad \forall i \in N \quad \forall j \in O_i \quad \forall k \in M_{ij} \quad (15)$$

$$z_{k0} \in \{0,1\} \quad \forall k \in M \quad (16)$$

$$z_{kl} \in \{0,1\} \quad \forall k \in M \quad \alpha_k \leq l \leq \beta_k \quad (17)$$

$$u_{ij'j'k} \in \{0,1\} \quad \forall i, i' \in N, i < i' \quad \forall j \in O_i \quad \forall j' \in O_{i'} \quad \forall k \in M_{ij} \cap M_{ij'} \quad (18)$$

$$v_{ijj'} \in \{0,1\} \quad \forall i \in N \quad \forall \langle j, j' \rangle \in Q_i \quad (19)$$

در مدل ریاضی رابطه (۱) بیانگر تابع هدف یعنی دوره ساخت می‌باشد. محدودیت (۲) تعیین کننده دوره ساخت است. محدودیت (۳) دقیقاً یک ماشین به هر عملیات تخصیص می‌دهد. محدودیت (۴) زمان پایان عملیات‌های مربوطه به ماشین‌های تخصیص نیافته را صفر می‌کند. رابطه (۵) تعداد منابع منقطع تخصیصی به هر ایستگاه کاری را تعیین می‌کند. در یک ایستگاه کاری ممکن است هیچ عملیاتی انجام نشود و منبع ثانویه تخصیص داده شده به آن صفر باشد. محدودیت (۶) باعث می‌شود که مجموع تعداد منابع ثانویه تخصیصی از تعداد کل منابع ثانویه

¹ disjunctive

² Task sequencing list

است با مجموع بارکاری ماشین k . مقادیر r_k ، برای مابقی اعضای جمعیت به روش تصادفی تولید می شود.

$$r_k = WL_k / \sum_{k' \in M} WL_{k'} \quad (21)$$

۳-۱-۳. تابع برازندگی

چون هدف مسأله کمینه سازی دوره ساخت می باشد، هر جواب مستقیماً بر اساس این مقدار ارزیابی می شود. با توجه به این که دوره ساخت یک معیار عادی^۵ می باشد که مقدار بهینه آن حتماً در مجموعه برنامه های زمانی فعال^۶ قرار دارد [۲۳]، در این پژوهش رویه ای برای تبدیل هر جواب به یک برنامه زمانی فعال توسعه داده شده است. گام های این رویه که تعمیمی است از الگوریتم گیفلر و تامپسون [۲۳] در شکل (۴) آمده است.

نمادهای بکار رفته در الگوریتم:

$SO(k)$: مجموعه عملیاتهای قابل زمان بندی در گام k

$CO(k)$: مجموعه عملیاتهای کاندید زمان بندی در مرحله k

$s_{ij} \in SO(k)$: زودترین زمان شروع ممکنه برای عملیات o_{ij}

$f_{ij} \in SO(k)$: زودترین زمان پایان ممکنه برای عملیات o_{ij}

m_{ij} : ماشین تخصیصی به عملیات o_{ij}

گامهای الگوریتم:

گام ۱: قرار دهید $k:=0$ و تشکیل دهید $SO(k)$ را با همه عملیاتهای بدون پیش نیاز

گام ۲: تعیین کنید $f^+ := \min \{f_{ij} | o_{ij} \in SO(k)\}$ ؛ اگر o_{ij}^+ عملیات متناظر با f^+ باشد، ماشین تخصیصی و کار مربوط به o_{ij}^+ را با m^+ و j^+ نشان می دهیم

گام ۳: مجموعه $CO(k)$ را تشکیل دهید

$$CO(k) := \left\{ o_{ij} \in SO(k) \mid \left(s_{ij} < f^+ \right) \wedge \left(m_{ij} = m^+ \vee j = j^+ \right) \right\}$$

گام ۴: از $CO(k)$ عملیات o_{ij}^* را که دارای بالاترین اولویت است انتخاب و شروع آن را در s_{ij}^* زمان بندی کنید

گام ۵: با حذف o_{ij}^* از $SO(k)$ و افزودن پس نیازهای o_{ij}^* به آن، $SO(k+1)$ را تشکیل دهید

گام ۶: اگر $k:=k+1$ ؛ اگر $SO(k) \neq \emptyset$ به گام ۲ بروید، در غیر این صورت توقف

شکل ۴. الگوریتم تولید برنامه زمانی فعال

ماشین ها، محتوای هر ژن مقداری است بین صفر تا یک و مجموع آن ها برابر با یک می باشد. تعداد منابع ثانویه تخصیصی به هر ماشین بر اساس رابطه (۲۰) تعیین می شود که در آن FR_k تعداد منبع منعطف تخصیصی به ماشین k و مقدار ژن k ام است. بر اساس رابطه (۲۰) ابتدا به هر ماشین به تعداد حداقل مورد نیاز (α_k) منبع منعطف تخصیص می یابد و منابع باقی مانده متناسب با مقدار r_k بین ماشین ها تقسیم می شوند. مثالی از نحوه بازنمایی جواب ها در شکل (۳) آمده است. اولین ژن کروموزوم A، (۴)، ۱، ۲ بیان می کند که o_{21} باید با بالاترین اولویت بر روی M4 انجام گیرد. بر اساس مقادیر کروموزوم B و داده های مسأله نمونه در شکل (۱)، منابع منعطف تخصیصی به ماشین های ۱ تا ۴ به ترتیب ۳، ۴، ۳ و ۲ بدست می آیند.

$$FR_k = \min \left\{ \text{round} \left[r_k \cdot \left(R - \sum_{k \in M} \alpha_k \right) \right], (\beta_k - \alpha_k) \right\} + \alpha_k \quad (20)$$



شکل ۳. چگونگی بازنمایی جوابها

۳-۱-۲. تولید جمعیت اولیه

در جمعیت اولیه، برای تولید هر جواب لازم است که اولویت هر عملیات، ماشین تخصیصی به عملیات و تعداد منابع منعطف برای هر ماشین تعیین شود. جهت تعیین اولویت هر عملیات دو قاعده ابتکاری شناخته شده بکار گرفته شده اند؛ اولویت برای P_{SPT} درصد از جمعیت به روش کوتاه ترین زمان پردازش^۱، برای P_{MWR} درصد به روش بیشترین کار باقی مانده^۲ و برای مابقی به صورت تصادفی تعیین می شود. برای تعیین ماشین تخصیصی به هر عملیات، رویکرد محلی سازی^۳ که توسط کاسم و همکاران پیشنهاد شده، بکار گرفته شده است [۲۲]. در این روش تخصیص ماشین به گونه ای انجام می گیرد که زمان انجام عملیات ها و بارکاری ماشین ها هم زمان کمینه گردد. تخصیص ماشین برای P_{AL} درصد از جوابها به روش AL^۴ و برای بقیه به صورت تصادفی انجام می شود. مقدار تعیین کننده تعداد منابع منعطف هر ماشین (r_k)، برای P_{MWL} درصد از جمعیت بر اساس بارکاری آن ماشین^۴ و طبق رابطه (۲۱) محاسبه می شود. در رابطه (۲۱)، WL_k برابر

¹ Shortest Processing Time (SPT)

² Most Work Remaining (MWR)

³ Approach by Localization (AL)

⁴ Machine Work Load (MWL)

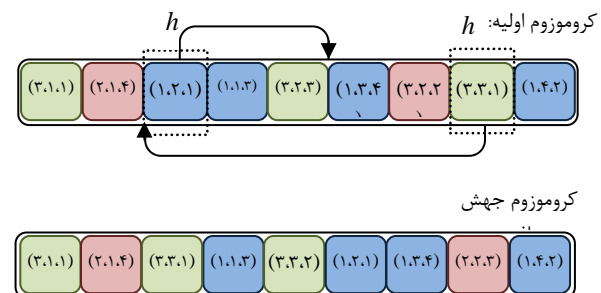
⁵ regular

⁶ Active Schedules

۳-۱-۴. عملگرهای ژنتیک

گزینش، تقاطع و جهش عملگرهای ژنتیک می‌باشند که جهت تولید یک جمعیت بهبود یافته در هر نسل اعمال می‌شوند. عملگر گزینش، کروموزوم‌های مولد نسل بعد را انتخاب می‌کند. در الگوریتم پیشنهادی، روش گزینش مسابقه دوتایی^۱ به دلیل توان آن در ایجاد توازن بین قدرت اکتشاف و استخراج GA استفاده شده است [۱۸].

به دلیل آن که عملگرهای تقاطع رایج برای بازنمایی جایگشتی مانند نگاشت بخشی^۲، تقاطع دوره ای^۳ و تقاطع مبتنی بر ترتیب^۴ می‌توانند جوابهای ناموجه تولید کنند، در الگوریتم پیشنهادی عملگر تقاطع ویژه‌ای طراحی شده که جوابها را همواره موجه نگه می‌دارد. در این عملگر که بر روی کروموزوم A اعمال می‌شود، ابتدا دو نقطه برش تصادفی، کروموزوم را به سه قسمت تقسیم می‌کنند. عناصر قطعات ابتدایی و انتهایی والد اول (P1) در مکان‌های مشابه در فرزند اول (O1) کپی می‌شوند. برای عناصر قطعه میانی O1، والد دوم (P2) از چپ به راست پیمایش شده و عملیاتی که در O1 وجود ندارند، با همان ترتیب ظاهر شده در P2، در O1 کپی می‌شوند. فرزند دوم (O2) با تعویض نقش P1 و P2 بدست می‌آید. در الگوریتم پیشنهادی سه عملگر جهش بکار گرفته می‌شوند. در عملگر اول که افزودن مضاعف^۵ نام گذاری شده، اولویت عملیات‌ها بدون آن که جواب از حالت موجه خارج شود، تغییر می‌کند. در این عملگر، دو عنصر e_1 و e_2 که در مکان‌های h_1 و h_2 کروموزوم A قرار دارند ($h_1 < h_2$) به صورت تصادفی انتخاب می‌گردند. ابتدا e_2 در مکانی موجه (مکانی پس از تمام پیش نیازها و قبل از تمام پس نیازهای این عملیات) در منتهالیه سمت چپ بازه h_1 تا h_2 افزوده می‌شود. سپس، e_2 در مکانی موجه در منتهالیه سمت راست بازه h_1 تا h_2 قرار داده می‌شود. مثالی از عملیات جهش به روش افزودن مضاعف در شکل (۵) آمده است.



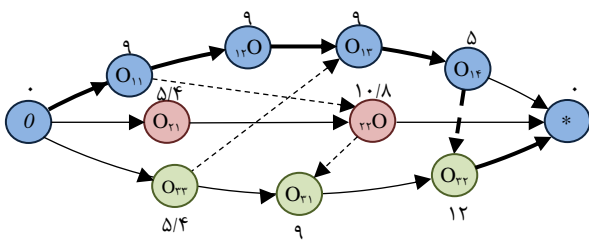
شکل ۵. فرایند جهش به روش افزودن مضاعف

¹ binary tournament
² Partially-Mapped Crossover (PMX)
³ Cycle Crossover (CX)
⁴ Order-Based Crossover (OBX)
⁵ Double insertion

عملگر جهش دوم، جهش روی ماشین است که در طی آن، یک ژن از کروموزوم A به تصادف انتخاب شده و ماشین تخصیص داده شده به آن به صورت تصادفی به یکی دیگر از ماشین‌های ممکنه تغییر می‌یابد. در جهش نوع سوم، تعداد منابع منعطف ماشین‌ها تغییر می‌کند، به این صورت که دو ژن به تصادف از کروموزوم B انتخاب گردیده و مقادیر r_k آن‌ها با هم عوض می‌شود. سپس با استفاده از رابطه (۲۰) تعداد منابع منعطف برای ماشین‌ها محاسبه می‌گردد.

۳-۲. جستجوی محلی

چون در تعریف تابع همسایگی از گراف انفصالی^۶ استفاده شده است، ابتدا این گراف توضیح داده می‌شود. گراف انفصالی یک تکنیک تصویری برای بازنمایی جواب‌های موجه مسایل زمان از $G=(N, A, E)$ بندگی می‌باشد [۲۴]. یک گراف انفصالی و تعدادی کمان (A)، تعدادی کمان اتصالی (N) تعدادی گره تشکیل شده است. متناظر با هر عملیات، یک گره (E) انفصالی به عنوان * و 0 در گراف ایجاد می‌شود، به اضافه دو گره مجازی گره‌های شروع و پایان. وزن هر گره برابر است با زمان پردازش عملیات مربوطه. کمان‌های اتصالی، دو عملیات متوالی مربوط به یک کار را بر اساس توالی پردازشی آن به هم متصل می‌کنند و کمان‌های انفصالی ترتیب انجام عملیات بر روی هر ماشین را مسیر بحرانی * به نمایش می‌دهند. طولانی‌ترین مسیر از دوره ساخت. عملیات‌های تشکیل نامیده می‌شود که برابر است با دهنده مسیر بحرانی، عملیات‌های بحرانی می‌باشند. گراف انفصالی متناظر با برنامه زمانی ارائه شده در شکل (۱) در شکل (۶) آمده است.



شکل ۶. یک گراف انفصالی

تابع همسایگی مورد استفاده در جستجوی محلی بر اساس جابجا کردن یک عملیات تعریف می‌شود. برای یک جواب مفروض با گراف انفصالی G، هر همسایگی با حذف یک گره دلخواه v و افزودن آن در مکان موجه دیگری از گراف بدست می‌آید. چون جابجا کردن عملیات‌های غیربحرانی نمی‌تواند منجر به بهبود جواب گردد [۲۴]، تنها همسایگی‌های حاصل از جابجا کردن

⁶ Disjunctive Graph

۵۰	درصدی از جمعیت اولیه که منابع منعطف با قاعده MWL تخصیص می یابند	P_{MWL}
٪۹۰	احتمال تقاطع	P_c
٪۱۰	احتمال جهش به روش افزودن مضاعف	P_m
٪۰/۵	احتمال جهش روی ماشین	P_{mm}
٪۱	احتمال جهش روی منبع منعطف	P_{FR}
٪۵۰	احتمال اعمال جستجوی محلی به یک جواب	P_{LS}
٪۵	بیشینه انحراف از بهترین جواب یافت شده جهت اعمال جستجوی محلی	P_{DEV}

نمادهای بکار رفته در الگوریتم:

$M(v)$: مجموعه ماشینهای ممکنه برای عملیات v
$p_{v,k}$: زمان عملیات v روی ماشین k
$J(v)$: کار مربوط به عملیات v
$C_{max}(G)$: طول مسیر بحرانی در گراف G
$CP(G)$: مجموعه عملیتهای بحرانی در گراف G
$G^-(v)$: گراف حاصل از حذف عملیات v از گراف G
$SQ^v(k)$: مجموعه عملیتهایی در $G^-(v)$ که بر روی ماشین k انجام می شوند به اضافه گره مجازی 0
RT^v : مجموعه عملیتهایی از $J(v)$ که v در $G^-(v)$ می تواند پس از آنها قرار گیرد به اضافه گره مجازی 0 وقتی v پیش نیاز ندارد
$JS^v(w)$: عملیات پس از w مربوط به کار $J(w)$ در $G^-(v)$
$MS^v(u)$: عملیات پس از u روی ماشین تخصیصی به u در $G^-(v)$
$E^v(w)$: طولانی ترین مسیر از گره 0 تا w شامل w در $G^-(v)$
$L^v(w)$: طولانی ترین مسیر از گره w تا 0 شامل w در $G^-(v)$
$G'(v, k, w, u)$: گراف حاصل از افزودن v به $G^-(v)$ ، بر روی ماشین k ، پس از u و w
C'_{max} : کوتاهترین مقدار بدست آمده برای دوره ساخت تاکنون

گامهای الگوریتم:

گام ۱: قرار دهید $C'_{max} := C_{max}(G)$
گام ۲: انجام دهید برای همه عملیتهای $v \in CP(G)$
گام ۳: انجام دهید برای همه عملیتهای $w \in RT^v$
گام ۴: انجام دهید برای همه ماشینهای $k \in M(v)$
گام ۵: انجام دهید برای همه عملیتهای $u \in SQ^v(k)$
گام ۶: در صورت برقراری رابطه
$\max\{E^v(w), E^v(u)\} + \max\{L^v(JS^v(w)), L^v(MS^v(u))\} + p_{v,k} < C'_{max}$
مقادیر v, w, k و u را ذخیره نموده و
$C'_{max} := C_{max}(G'(v, k, w, u))$
تکرار گام ۵ برای عملیات u بعدی
تکرار گام ۴ برای ماشین k بعدی
تکرار گام ۳ برای عملیات w بعدی
تکرار گام ۲ برای عملیات v بعدی
گام ۷: ارائه $G'(v, k, w, u)$ به عنوان بهترین همسایگی و توقف

عملیاتهای بحرانی بررسی می شود. الگوریتم جستجوی محلی در شکل (۷) آمده است. بر اساس این الگوریتم ابتدا برای هر عملیات بحرانی v ، گراف $G^-(v)$ با حذف v از G بدست می آید. سپس، افزودن مجدد v در محل های موجه دیگر بررسی می شود. پروسه یافتن یک محل موجه برای v شامل سه تصمیم است: تعیین یک مکان در توالی پردازشی کاری که v به آن تعلق دارد جهت قرار دادن v در آن مکان، تخصیص یک ماشین به v ، و تعیین مکانی در توالی عملیاتی این ماشین جهت قرار دادن v در آن مکان. برای هر عملیات بحرانی همه گزینه های ممکن برای این سه تصمیم به صورت سلسله مراتبی در نظر گرفته می شود؛ یعنی ابتدا مکانی در توالی پردازشی کار، سپس ماشین تخصیصی و در نهایت مکانی در توالی عملیاتی ماشین. مجموعه مکان های ممکنه جهت افزودن v در توالی پردازشی کار مربوطه در $G^-(v)$ ، با مجموعه عملیتهایی که v می تواند پس از آنها قرار گیرد (RT^v) معرفی می شود. به گونه ای مشابه، پس از گزینش یک ماشین از مجموعه $M(v)$ ، مکان های ممکنه بر روی این ماشین برای افزودن v با مجموعه عملیتهایی که v می تواند پس از آنها قرار گیرد $(SQ^v(k))$ معرفی می شود. گرچه ترکیب گزینه های ممکنه برای این تصمیم های سه گانه می تواند بسیار زیاد باشد، رابطه مورد استفاده در گام ۶ الگوریتم، بسیاری از این حالات را از فضای جستجو حذف می کند. بر اساس این رابطه تنها مکان هایی از $G^-(v)$ برای افزودن v بررسی می شوند که در گراف حاصله طولانی ترین مسیری که از v می گذرد از کم ترین مقدار دوره ساخت بدست آمده تاکنون کوتاه تر باشد.

۴. تجزیه و تحلیل عددی

۴-۱. تنظیم پارامترهای الگوریتم جستجوی محلی ژنتیک

مقدار بدست آمده برای پارامترهای الگوریتم پیشنهادی به روش تنظیم پارامترها^۱ [۲۵]، همراه با شرح آنها در جدول (۱) آمده است.

جدول ۱. پارامترهای الگوریتم جستجوی محلی ژنتیک

پارامتر	شرح	مقدار
Pop_size	اندازه جمعیت	۲۰۰
Max_gen	بیشینه تعداد نسل های تولیدی	۵۰
P_{SPT}	درصدی از جمعیت اولیه که تعیین اولویت با قاعده SPT انجام می شود	۱۰
P_{MWR}	درصدی از جمعیت اولیه که تعیین اولویت با قاعده MWR انجام می شود	۳۰
P_{AL}	درصدی از جمعیت اولیه که تخصیص ماشین با رویکرد AL انجام می شود	۴۰

^۱ parameter tuning method

مشخصات مسائل حاصله و جواب بدست آمده به روش دقیق (با استفاده از مدل ریاضی توسعه داده شده در بخش ۲-۲) و جواب حاصل از الگوریتم پیشنهادی در جدول (۲) آمده است. ملاحظه می گردد که در همه مسائل، الگوریتم پیشنهادی در زمانی بسیار کوتاه تر به جواب بهینه رسیده است.

جدول ۲. نتایج عددی برای مسائل نمونه

مساله	P1	P2	P3	P4	P5
ماخذ	[۲۶]	[۳]	[۱۹]	[۱۷]	[۷]
تعداد کار	۵	۶	۵	۵	۸
تعداد ماشین	۵	۵	۶	۵	۶
تعداد منابع منعطف	۱۵	۱۵	۱۸	۱۵	۱۸
تعداد کل عملیات	۲۰	۱۸	۲۱	۱۳	۲۰
مقدار جواب به روش دقیق	۲۱/۶۳	۱۶/۷۱	۱۸/۰۱	۹/۰۹	۱۳/۸۰
مقدار جواب به روش پیشنهادی	۲۱/۶۳	۱۶/۷۱	۱۸/۰۱	۹/۰۹	۱۳/۸۰
زمان حل به روش دقیق (s)	۷۱۴۱	۳۱۹۶۰	۳۵۴۷۶	۵۱۷	۱۰۸۹۵
زمان حل به روش پیشنهادی (s)	۳/۳	۳/۴	۴/۸	۲/۷	۵/۱

جهت بررسی عملکرد الگوریتم پیشنهادی در مسائل با ابعاد بزرگ، دو مورد از مسائل فوق (P4 و P5) انتخاب و با تغییر تقاضا برای هر سفارش، شش مسأله جدید ایجاد شده است. مشخصات و نتایج عددی برای این مسائل در جدول (۳) آمده است. جهت کاهش اثرات تصادفی، الگوریتم پیشنهادی برای هر مسأله ده بار اجرا شده و میانگین و بهترین مقدار بدست آمده گزارش شده است. همچنین برای نشان دادن بهبود حاصل از کاربرد منابع ثانویه، جواب های بدست آمده با بهترین جواب های موجود در حالتی که منابع ثانویه وجود ندارند، مقایسه شده است.

جدول ۳. نتایج عددی برای مسائل بزرگ

مساله تعمیم یافته	EP1	EP2	EP3	EP4	EP5	EP6
مساله پایه	P4	P4	P4	P5	P5	P5
تعداد تقاضا	۵	۱۰	۲۰	۵	۱۰	۲۰
تعداد کار	۲۵	۵۰	۱۰۰	۴۰	۸۰	۱۶۰
تعداد ماشین	۵	۵	۵	۶	۶	۶
تعداد منابع منعطف	۱۵	۱۵	۱۵	۱۸	۱۸	۱۸
تعداد کل عملیات	۶۵	۱۳۰	۲۶۰	۱۰۰	۲۰۰	۴۰۰
بهترین جواب	۳۸/۷۵	۷۷/۵۰	۱۵۴/۴۶	۶۵/۶۳	۱۳۱/۲۵	۲۵۸/۰۰
میانگین جوابها	۳۹/۰۴	۷۸/۱۵	۱۵۵/۳۳	۶۶/۱۰	۱۳۲/۶۰	۲۶۳/۸۱
زمان حل (s)	۳۶	۱۲۷	۲۷۱	۸۲	۱۵۳	۳۰۱
جواب بدون منابع ثانویه	۶۲/۰۰	۱۲۴/۰۰	۲۴۷/۰۰	۱۰۵/۰۰	۲۰۸/۰۰	۴۱۶/۰۰
بهبود ناشی از منابع ثانویه (%)	۳۷/۵۰	۳۷/۴۶	۳۷/۴۹	۳۶/۹۰	۳۶/۹۰	۳۷/۹۸

به منظور تعدیل حجم محاسبات جستجوگر محلی و پیشگیری از همگرایی زودرس، دو سازوکار بکار رفته است. اول آن که جستجوگر محلی به صورت احتمالی به جواب ها اعمال می شود. مقدار این احتمال برای نسل اول صفر است و به صورت خطی تا مقدار P_{LS} برای نسل آخر افزایش می یابد. از طریق این سازوکار، در نسل های اولیه، عملگرهای ژنتیک بازیگر غالب بوده و مناطق امیدبخش را در فضای جواب پیدا می کنند. در نسل های متأخر، این مناطق توسط جستجوگر محلی که با تواتر بیشتری انجام می شود، مورد کاوش عمیق تر قرار می گیرند. سازوکار دوم به کارگیری راهبرد انطباق مبتنی بر برازندگی^۱ یعنی سو گیری گزینش گر جستجوی محلی به سمت جواب های دارای برازندگی بیشتر می باشد. بر این اساس عملگر جستجوی محلی در صورتی به یک جواب اعمال می شود که نامساوی (۲۲) برقرار باشد:

$$\frac{f(i,t) - f^*(t-1)}{f^*(t-1)} < P_{DEV} \cdot idle(t-1) \quad (22)$$

در رابطه (۲۲)، $f(i,t)$ مقدار برازندگی جواب i ام در جمعیت t ، $f^*(t-1)$ بهترین مقدار برازندگی یافت شده تا نسل $t-1$ ، $idle(t-1)$ تعداد نسل های متوالی تا نسل $t-1$ که بهبود در جواب را تجربه نکرده اند و P_{DEV} انحراف نسبی مجاز می باشد. الگوریتم پیشنهادی در محیط ویژوال بیسیک کاربردی^۲ برنامه نویسی و بر روی یک کامپیوتر پنتیوم ۴ با واحد پردازنده مرکزی ۲/۴۰ پیاده سازی شده است.

۲-۴. نتایج عددی

برای حل مسأله FJSP-APP-CPT تاکنون هیچ پژوهشی انجام نشده است و نزدیک ترین مسأله به آن در ادبیات، مسأله FJSP-APP می باشد. به منظور ارزیابی عملکرد الگوریتم پیشنهادی، ۵ مسأله FJSP-APP از ادبیات انتخاب و به روش پیشنهادی روی تورس و همکاران تبدیل به FJSP-APP-CPT شده اند [۱۴]. بر اساس روش مذکور، تعداد منابع منعطف سه برابر تعداد ماشین ها ($R/m=3$)، کمینه و بیشینه منابع منعطف برای همه ماشین ها برابر با $R/m \pm 2$ (یعنی $\alpha_k=1$ و $\beta_k=5$) در نظر گرفته شده و نرخ تسریع نیز از رابطه (۲۳) بدست می آید. ρ و χ پارامترهای رابطه (۲۳) می باشند که مقدار آن ها ۰/۵ و صفر در نظر گرفته شده است.

$$q_{kl} = 1 - \rho \left(1 - 1 / (l - \alpha_k + 1)^{\chi} \right) \text{ for } l = \alpha_k \text{ to } \beta_k \quad (23)$$

¹ Fitness-based adaptation

² Visual Basic for Application (VBA)

- [5] Li, X., Gao, L., Zhang, C., Shao, X., "A review on Integrated Process Planning and Scheduling", International Journal of Manufacturing Research, Vol. 5, No. 2, 2010, pp. 161-180.
- [6] Chryssolouris, G., Chan, S., Cobb, W., "Decision Making on the Factory Floor: An Integrated Approach to Process Planning and Scheduling", Robotics and Computer-Integrated Manufacturing, Vol. 1, No. 3/4, 1985, pp. 315-319.
- [7] Lee, H.L., Jeong, C.S., Moon, C., "Advanced Planning and Scheduling with Outsourcing in Manufacturing Supply Chain", Computers & Industrial Engineering, Vol. 43, No. 1/2, 2002, pp. 351-374.
- [8] Li, W.D., McMahon, C.A., "A Simulated Annealing-Based Optimization Approach for Integrated Process Planning and Scheduling", International Journal of Computer Integrated Manufacturing, Vol. 20, No. 1, 2007, pp. 80-95.
- [9] Shen, W., Wang, L., Hao, Q., "Agent-Based Distributed Manufacturing Process Planning and Scheduling: a State-of-the-Art Survey", IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews, Vol. 36: No. 4, 2006, pp. 563-577.
- [10] Leung, C.W., Wong, T.N., Maka, K.L., Fung, R.Y.K., "Integrated Process Planning and Scheduling by an Agent-Based Ant Colony Optimization", Computers & Industrial Engineering, Vol. 59, No. 1, 2010, pp. 166-180.
- [11] Janiak, A., "Minimization of the Blooming Mill Standstills - Mathematical Model, Suboptimal Algorithms", Mechanika, Vol. 8, No. 2, 1989, pp. 37-49.
- [12] Lee, C.Y., Lei, L., "Multiple-Project Scheduling with Controllable Project Duration and Hard Resource Constraint: Some Solvable Cases", Annals of Operations Research, Vol. 102, No. 1/4, 2001, pp. 287-307.
- [13] Shabtay D., & Steiner G., "A survey of scheduling with controllable processing times" Discrete Applied Mathematics, Vol. 155, No. 13, 2007, PP. 1643-1666.
- [14] Ruiz-Torres, A.J., Lopez, F.J., Ho, J.C., "Scheduling Uniform Parallel Machines Subject to a Secondary Resource to Minimize the Number of Tardy Jobs", European Journal of Operational Research, Vol. 179, No. 2, 2007, pp. 302-315.
- [15] Ruiz-Torres, A.J., Centeno, G., "Minimizing the Number of Late Jobs for the Permutation Flowshop Problem with Secondary Resources", Computers & Operations Research, Vol. 35, No. 4, 2008, pp. 1227-1249.
- [16] Jansen, K., Mastrolilli, M., Solis-Oba, R., "Approximation Schemes for Job Shop Scheduling Problems with Controllable Processing Times", European Journal of Operational Research, Vol. 167, No. 2, 2005, pp. 297-319.

۵. جمع بندی

در این پژوهش یک مسأله زمان بندی با انعطاف در ماشین، توالی عملیات و زمان‌های عملیات که دارای کاربردهای بسیاری در سامانه‌های تولیدی نوین می‌باشد مورد بررسی قرار گرفت. ابتدا مسأله در قالب یک مدل برنامه ریزی غیرخطی عدد صحیح فرموله شد. علیرغم آن که با استفاده از مسائل نمونه نشان داده شد که مدل ریاضی در بدست آوردن جواب دقیق مسائل توانمند است، اما به دلیل ماهیت پیچیده مسأله، با بزرگ شده ابعاد آن بدست آوردن جواب دقیق غیرممکن می‌گردد. به همین دلیل یک الگوریتم ژنتیک ترکیب شده با جستجوی محلی که می‌تواند جواب‌هایی با کیفیت را در زمان قابل قبول بدست آورد برای حل مسأله توسعه داده شد. از جمله مختصات بارز الگوریتم پیشنهادی به‌کارگیری عملگرهای ژنتیک طراحی شده بر اساس مختصات مسأله، تولید جمعیت اولیه با کیفیت بالا بر اساس روش‌های ابتکاری، طراحی یک تابع جستجوی محلی بدیع و کارآمد و اعمال گزینشی جستجوگر محلی با اتخاذ راهبردهای انطباقی ویژه می‌باشد. توسعه مدل از طریق ادغام آن با تصمیمات مربوط به شبکه توزیع، حمل و نقل و ارسال، بازنگری در مدل جهت بهینه سازی معیارهای دیگری از قبیل مجموع هزینه، موعد تحویل، بارکاری ماشین‌ها و زمان‌های دیرکرد، توسعه راه حل‌هایی جهت حل مسأله در حالت پویا و نیز شرایط احتمالی، در نظر گرفتن سایر شرایط حاکم در دنیای واقعی از قبیل زمان‌های راه اندازی و جابجایی بین ایستگاه‌ها، امکان پردازش دسته‌ای و انجام هم‌زمان عملیات‌های یک سفارش از جمله حوزه‌های پیشنهادی برای پژوهش‌های آتی می‌باشد.

منابع

- [1] Garey, M.R., Johnson, D.S., Sethi, R., "The Complexity of Flow Shop and Job Shop Scheduling" Mathematics of Operational Research, Vol. 1, No. 2, 1976, pp. 117-129.
- [2] Yazdani, M., Amiri, M., Zandieh, M., "Flexible Job-Shop Scheduling with Parallel Variable Neighborhood Search Algorithm", Expert Systems with Applications, vol. 37, No. 1, 2010, pp. 678-687.
- [3] Li, X., Gao, L., Shao, X., Zhang, C., Wang, C., "Mathematical Modeling and Evolutionary Algorithm-Based Approach for Integrated Process Planning and Scheduling", Computers & Operations Research, Vol. 37, No. 4, 2010, pp. 656-667.
- [4] Guo, Y.W., Li, W.D., Mileham, A.R., Owen, G.W., "Applications of Particle Swarm Optimisation in Integrated Process Planning and Scheduling", Robotics and Computer-Integrated Manufacturing, Vol. 25, No. 2, 2009, pp. 280-288.

- [17] Moon, C., Lee, Y.H., Jeong, C.S., Yun, Y.S., "Integrated Process Planning and Scheduling in a Supply Chain", *Computer & Industrial Engineering*, Vol. 54, No. 4, 2008, pp. 1048-1061.
- [18] Li, X., Shao, X., Gao, L., Qian, W., "An Effective Hybrid Algorithm for Integrated Process Planning and Scheduling", *International Journal of Production Economy*, Vol. 126, No. 3, 2010, pp. 289-298.
- [19] Shao, X., Li, X., Gao, L., Zhang, C., "Integration of Process Planning and Scheduling - A Modified Genetic Algorithm-Based Approach", *Computer & Operation Research*, Vol. 36, No. 6, 2009, pp. 2082-2096.
- [20] Moon, C., Seo, Y., Yun, Y., Gen, M., "Adaptive Genetic Algorithm for Advanced Planning in Manufacturing Supply Chain", *Journal of Intelligent Manufacturing*, Vol. 17, No. 5, 2006, pp. 509-522.
- [21] Hart, W.E., Krasnogor, N., Smith, J.E., "Memetic Evolutionary Algorithms" In: Hart W.E., Krasnogor N., & Smith J.E. (eds): *Recent advances in memetic algorithms*, Springer, Berlin, 2005, pp. 3-27.
- [22] Kacem, I., Hammadi, S., Borne, P., "Approach by localization and Multiobjective Evolutionary Optimization for Flexible Job-Shop Scheduling Problems", *IEEE Transactions on System & Man and Cybernetics: Part C, Application Review*, Vol. 32, No. 1, 2002, pp. 1-13.
- [23] Baker, K.R., Trietsch, D., *Principles of Sequencing and Scheduling*, Wiley, New York, 2009
- [24] Gao, J., Sun, L., Gen, M., "A Hybrid Genetic and Variable Neighborhood Descent Algorithm for Flexible Job Shop Scheduling Problems", *Computer & Operation Research*, Vol. 35, No. 9, 2008, pp. 2892-2907.
- [25] Franca, P.M., Mendes, A., Moscato, P., "A Memetic Algorithm for the Total Tardiness Single Machine Scheduling Problem", *European Journal of Operation Research*, Vol. 132, No. 1, 2001, pp. 224-242.
- [26] Sundaramm R.M., Fu, S.S., "Process Planning and Scheduling—a Method of Integration for Productivity Improvement", *Computer & Industrial Engineering*, Vol. 15, No. 1/4, 1998, pp. 296-301.