

ارائه حد پایین برای مسئله زمان بندی خط تولید کارگاهی همراه با یک مرحله مونتاژ موازی با استفاده از رویکرد رنگ آمیزی گراف

فاطمه دانش آموز^۱، جواد بهنامیان^{۲*}

۱. دانشجوی دکتری، گروه مهندسی صنایع، دانشکده مهندسی، دانشگاه بوعلی سینا

۲. دانشیار، گروه مهندسی صنایع، دانشکده مهندسی، دانشگاه بوعلی سینا

(تاریخ دریافت: ۹۵/۰۸/۲۰، تاریخ دریافت روایت اصلاح شده: ۹۷/۰۳/۰۳، تاریخ تصویب: ۹۷/۰۷/۰۳)

چکیده

زمان بندی یکی از مسائل پر کاربرد در صنعت است که پژوهشگران در سال های اخیر توجهی ویژه به آن داشته اند. تلاش برای کاهش شکاف میان روش های تئوری و کاربردی در مسائل زمان بندی، از ضرورت های دنیای امروز محسوب می شود. یکی از اقدامات صورت گرفته در این زمینه، توجه هم زمان دو مرحله مونتاژ و پردازش است. در این پژوهش با توجه به اهمیت توجه هم زمان به مراحل مختلف تولید در صنعت و همچنین به منظور کاربردی تر کردن مسئله، به بررسی مسئله زمان بندی تولید کارگاهی با در نظر گرفتن یک مرحله مونتاژ موازی با هدف حداقل کردن زمان تکمیل محصولات پرداخته ایم. بدین منظور ابتدا مسئله مدنظر به مسئله رنگ آمیزی گراف تقلیل داده شد. با توجه به اینکه مسئله مدنظر و رنگ آمیزی گراف مربوط از نوع NP-hard هستند، برای حل آن ها در ابعاد متوسط و بزرگ از الگوریتم ترکیبی ژنتیک و بهینه سازی انبوه ذرات استفاده شد. همچنین به منظور بررسی صحت عملکرد و میزان کارایی الگوریتم در این پژوهش، حدود بالا و پایینی با استفاده از مسئله رنگ آمیزی گراف ارائه شد.

واژه های کلیدی: تولید کارگاهی، رنگ آمیزی گراف، زمان بندی، مونتاژ موازی.

مقدمه

یکی از مهم ترین وظایف سیستم های تولیدی و خدماتی چگونگی زمان بندی کارها بر روی ماشین هاست. یافتن بهترین زمان بندی با توجه به محیط تولید و محدودیت های آن می تواند کاری آسان یا بسیار سخت باشد. در مسائل زمان بندی کلاسیک هر کار مستقل از کاری دیگر در نظر گرفته می شود، اما این محدودیت در واقعیت همواره برقرار نیست. در بسیاری از کارگاه ها و سیستم های تولیدی پس از اینکه قطعات و مجموعه ها آماده شدند، باید با یکدیگر مونتاژ شوند تا به محصولی کامل تبدیل شوند؛ بنابراین این محدودیت با اضافه کردن یک مرحله مونتاژ به مسئله زمان بندی کلاسیک برطرف می شود. با توجه به اهمیت مسائل زمان بندی که در آن ها مراحل ساخت قطعات و مونتاژ محصولات هم زمان با هم در نظر گرفته می شود، همچنین توجه به کاربردهای فراوان آن در صنایع تولیدی، پژوهشگران حوزه زمان بندی توجه زیادی به این مسائل نشان دادند؛ زیرا بهینه کردن توالی انجام دادن کارها در یک مرحله از تولید، بدون در نظر گرفتن پارامترها و

ظرفیت های مرحله بعدی، تضمینی برای کوتاه شدن زمان کل تولید و استفاده بهینه از ماشین آلات و سایر ظرفیت های موجود نخواهد بود. از سوی دیگر، در بسیاری از سیستم های تولید با توجه به امکانات ممکن است مرحله مونتاژ منحصربه فرد نباشد و ایستگاه های موازی برای آن در نظر گرفته شود.

مسئله زمان بندی تولید دومرحله ای نخستین بار در پژوهش لی و همکاران (۱۹۹۳) بررسی شد [۱] و پس از آن این دسته از مسائل مدنظر بسیاری از پژوهشگران قرار گرفت [۲-۵]. حریری و پاتس [۶] مسئله زمان بندی تولید دومرحله ای را که در آن m ماشین پردازش در مرحله ساخت و یک ماشین برای مونتاژ در نظر گرفته شده بود، بررسی کردند و برای اولین بار حد پایینی برای آن توسعه دادند و چندین رابطه چیرگی را برقرار کردند. همچنین الگوریتم شاخه و کران را در ترکیبی با حد پایین و روابط چیرگی ارائه دادند. یوکویاما و همکاران [۷] نیز سیستم تولید ترکیبی را بررسی کردند که در آن تک ماشین تولیدی موجود در مرحله اول، سری قطعات مورد نیاز هر محصول را پردازش

رنگ‌آمیزی گراف به‌طور گسترده برای مدل‌سازی انواع مختلفی از مسائل دنیای واقعی استفاده شده است. مسئله رنگ‌آمیزی رئوس، تخصیص رنگ به هر رأس به نحوی است که هیچ دو رأس مرتبط به هر یال، رنگ یکسانی نداشته باشند. حداقل کردن تعداد رنگ‌ها برای یک گراف مشخص رایج‌ترین نوع از مسئله رنگ‌آمیزی گراف است که مسئله‌ای NP-hard محسوب می‌شود [۱۱]. با این حال این مدل رنگ‌آمیزی اولیه محدودیت‌های بسیاری برای مسائل زمان‌بندی واقعی دارد؛ بنابراین لازم است مدل‌های رنگ‌آمیزی گراف در جهات مختلف گسترش یابد؛ به‌عنوان مثال، مسئله رنگ‌آمیزی رئوس یک گراف بدون جهت برای مسائل زمان‌بندی با محدودیت‌های پیش‌نیازی، محدودیت‌های بسیاری دارد. برای رفع این محدودیت‌ها ساتسکو و تانیو [۱۲] مسئله رنگ‌آمیزی گراف مختلط را که قادر به رعایت این الزامات باشد معرفی کردند. ساتسکو و همکاران [۱۳] نیز برای اولین بار مسئله $C_{\max} | p_j = 1 | j$ با زمان‌ها واحد را به مسئله رنگ‌آمیزی گراف با حداقل تعداد رنگ یعنی MINMAX تقلیل دادند. همچنین الگوریتم شاخه و کران را برای حل مسئله پیشنهاد دادند. الانزی و همکاران [۱۴] مسئله تولید کارگاهی را به مسئله رنگ‌آمیزی گراف مختلط تقلیل دادند. در مسئله مدنظر آن‌ها هدف حداقل کردن حداکثر تعداد رنگ (حداقل کردن حداکثر زمان ساخت) درنظر گرفته شد. همچنین آن‌ها به ارائه چندین حدود بالا و پایین پرداختند و مسئله را با استفاده از روش شاخه و کران حل کردند. کویدر و همکاران [۱۵] برای نخستین بار به حل مسئله با استفاده از الگوریتم فراابتکاری پرداختند و از الگوریتم جست‌وجوی ممنوع برای حل مسئله استفاده کردند. آن‌ها [۱۶] دو سال بعد به حل مسئله در سه مرحله پرداختند. ابتدا یک حد بالا و پایین جدید ارائه دادند و سپس به توسعه یک مدل ریاضی با استفاده از حدود ارائه‌شده پرداختند. همچنین از الگوریتم جست‌وجوی ممنوع با استفاده از ساختار همسایگی پویا برای حل نمونه‌های بزرگ استفاده کردند.

با توجه به مطالعات انجام‌شده، تقلیل مسئله زمان‌بندی تولید کارگاهی به مسئله رنگ‌آمیزی گراف در ادبیات زمان‌بندی بسیار محدود است. همچنین مشخص شد که تاکنون حد پایینی برای مسئله زمان‌بندی تولید کارگاهی همراه

می‌کند. سپس در مرحله دوم قطعات هر محصول مونتاژ می‌شود. در مسئله مدنظر، تعدادی محصول از یک نوع باید تولید شود؛ بنابراین موضوع تعیین اندازه دسته و زمان‌های تنظیم و راه‌اندازی نیز مطرح شد. تکنیک اصلی معرفی‌شده برای حل این مسئله، استفاده از برنامه‌ریزی پویا برای یافتن پاسخ نزدیک بهینه بود. ضمن آنکه حد پایین مناسبی نیز توسعه داده شد و نتایج نشان داد پاسخ‌های حاصل از الگوریتم پیشنهادی بسیار نزدیک به بهینه است. سانگ و همکاران [۸] نیز مسئله‌ای را بررسی کردند که در آن n کار (محصول) وجود دارد که هرکدام از دو قطعه تشکیل می‌شود که یکی از قطعات را تک‌ماشین پردازش می‌کند و قطعه دیگر توسط تأمین‌کننده بیرونی تهیه می‌شود. در مرحله دوم نیز این دو قطعه پس از آماده‌بودن مونتاژ و محصول نهایی آماده می‌شوند. تابع هدف درنظر گرفته‌شده در این مسئله Makespan است و چند الگوریتم ابتکاری مبتنی بر قضیه جانسون و روش شاخه و کران به همراه سه حد پایین برای حل آن مدنظر قرار می‌گیرد. سانگ و کیم [۹] مسئله زمان‌بندی مونتاژ دومرحله‌ای چندماشین را با هدف کمینه‌کردن جمع زمان‌های تکمیل کارها درنظر گرفتند. در مدل پیشنهادی آن‌ها مرحله اول شامل دو ماشین مستقل است که هرکدام نوع خاصی از اجزا را تولید می‌کنند. مرحله دوم متشکل از دو ماشین یکسان و مستقل است که موازی با هم کار می‌کنند. آن‌ها کمینه‌کردن جمع زمان اتمام کارها در مرحله دوم را به‌عنوان معیار زمان‌بندی درنظر گرفتند سپس حد پایینی براساس روش کمترین زمان پردازش ارائه و با استفاده از آن الگوریتم شاخه و کران را پیشنهاد دادند. همچنین الگوریتم ابتکاری کارایی را نیز توسعه دادند. فتاحی و همکاران [۱۰] مسئله زمان‌بندی جریان کارگاهی ترکیبی مونتاژ دومرحله‌ای را مطالعه کردند. در مسئله مورد نظر مرحله اول، جریان کارگاهی ترکیبی و مرحله دوم یک ماشین مونتاژ و هدف درنظر گرفته‌شده، کمینه‌سازی زمان تکمیل تمام محصولات است. همچنین آن‌ها با درنظرگرفتن گلوگاه در مراحل مختلف چندین حد بالا و پایین ارائه و با استفاده از آن‌ها یک الگوریتم شاخه و کران پیشنهاد دادند. بررسی پژوهش‌های گذشته نشان می‌دهد ارائه حدود پایین برای مسئله زمان‌بندی تولید کارگاهی همراه با یک مرحله مونتاژ موازی تاکنون مطالعه نشده است.

| | |
|---|-----------------|
| تعداد محصولات ($p=1,2,\dots,P$) | P |
| تعداد قطعات محصول ($j=1,2,\dots,n_p$) | n_p |
| تعداد ماشین‌ها در مرحله تولید کارگاهی ($i=1,2,\dots,m$) | m |
| تعداد ماشین‌های موازی در مرحله مونتاژ ($i'=1,2,\dots,m'$) | m' |
| تعداد عملیات اختصاص یافته به ماشین i ($k=1,2,\dots,K_i$) | K |
| تعداد عملیات قطعه j از محصول p ($h=1,2,\dots,H_{j,p}$) | $H_{j,p}$ |
| عملیات h از قطعه j از محصول p | $O_{j,p,h}$ |
| زمان پردازش عملیات $O_{j,p,h}$ | $pt_{j,p,h}$ |
| زمان مونتاژ محصول p | A_p |
| یک عدد بزرگ | L |
| یک متغیر صفر و ۱ است؛ به طوری که اگر عملیات $O_{j,p,h}$ روی ماشین i انجام شود، برابر ۱ و در غیر این صورت صفر است. | $a_{i,j,p,h}$ |
| متغیرهای این مدل عبارت‌اند از: | |
| زمان حداکثر دوره ساخت | C_{max} |
| زمان ساخت محصول p در مرحله اول | F_p |
| زمان تکمیل نهایی محصول p | Ct_p |
| زمان شروع عملیات $O_{j,p,h}$ | $t_{j,p,h}$ |
| زمان شروع عملیات مونتاژ | S_p |
| زمان شروع به کار ماشین i در نوبت k در مرحله ۱ | $Tm_{i,k}$ |
| زمان شروع به کار ماشین i' در نوبت k' در مرحله ۲ | $Sm_{i',k'}$ |
| یک متغیر صفر و ۱ است؛ به طوری که اگر عملیات $O_{j,p,h}$ روی ماشین i در نوبت k انجام شود، برابر ۱ و در غیر این صورت صفر است. | $x_{i,j,p,h,k}$ |
| یک متغیر صفر و ۱ است؛ به طوری که اگر k' در نوبت i' بر روی ماشین p محصول برابر ۱ و در غیر این صورت صفر است. | $Z_{i',p,k'}$ |

با داشتن پارامترهای بیان شده، مسئله به صورت زیر مدل سازی می شود:

$$\text{Min}Z = C_{\max} \quad (۱)$$

Subject to:

$$C_{\max} \geq Ct_p \quad (۲)$$

$$t_{j,p,h} + pt_{j,p,h} \leq t_{j,p,h+1} \quad (۳)$$

با یک مرحله مونتاژ موازی ارائه نشده است؛ بنابراین در این مقاله برای اولین بار مسئله مورد نظر به مسئله رنگ آمیزی گراف تقلیل می یابد و سپس با توجه به آن حدود بالا و پایین ارائه می شود. همچنین از آنجا که مسئله رنگ آمیزی گراف NP-hard است، از میان الگوریتم های فراابتکاری، الگوریتم ژنتیک به طور گسترده برای حل آن استفاده شده است [۱۷]. اما با توجه به دانش ما هیچ الگوریتم ترکیبی برای حل مسئله رنگ آمیزی گراف مختلط استفاده نشده است؛ بنابراین در این مقاله از الگوریتم ترکیبی ژنتیک و بهینه سازی انبوه ذرات برای حل مسئله استفاده شد.

ساختار مقاله به ترتیب زیر است: در بخش دوم بررسی مسئله مورد نظر و در بخش سوم تحلیل مسئله رنگ آمیزی گراف و تشابه آن با مسئله مورد نظر صورت گرفت. سپس حدود بالا و پایینی برای مسئله ارائه شد. در بخش چهارم الگوریتم پیشنهادی و در بخش پنجم آزمایش های عددی و تجزیه و تحلیل و سپس نتیجه گیری و پیشنهاد پژوهش های آتی در بخش ششم ارائه شد.

۲. تعریف مسئله و مدل ریاضی

به طور کلی می توان تولید محصولات در سیستم تولید دومرحله ای مدنظر در این پژوهش را به صورت مدل سازی و تعیین توالی انجام دادن کارها در سیستم تولیدی تعریف کرد که مرحله اول آن تولید کارگاهی و مرحله دوم آن یک ایستگاه با خط مونتاژ موازی است. بدیهی است مونتاژ هر محصول در مرحله دوم زمانی می تواند آغاز شود که تمام قطعات مورد نیاز آن در مرحله اول پردازش شده و آماده باشند. در این قسمت از مدل ریاضی ارائه شده توسط دانش آموز و همکاران [۱۸] به منظور شناسایی و استفاده از آن برای حل مسئله کاربرد دارد. مرحله اول، مسئله زمان بندی تولید کارگاهی است که به صورت زمان بندی J قطعه $(J_1, J_2, \dots, J_{n_p})$ از P محصول (P_1, P_2, \dots, P_p) روی مجموعه M از ماشین ها که m ماشین (M_1, M_2, \dots, M_m) را شامل می شود قرار گرفته است. مرحله دوم، مسئله زمان بندی مونتاژ موازی است که به صورت زمان بندی P محصول با زمان مونتاژ A_p روی مجموعه M' از ماشین ها که شامل m' ماشین $(M'_1, M'_2, \dots, M'_{m'})$ است به صورت موازی قرار گرفته اند. پارامترهای این مدل عبارت‌اند از:

بیان می‌کند. محدودیت ۱۱ بیانگر زمان تکمیل محصولات است. با توجه به محدودیت ۱۲ در صورتی ماشین می‌تواند به کار در نوبت $k+1$ پردازد که کار آن ماشین در نوبت k به اتمام رسیده باشد. با توجه به محدودیت ۱۳ و ۱۴ در صورتی عملیاتی می‌تواند شروع شود که هم محدودیت پیش‌نیازی (عملیات قبلی آن به اتمام رسیده باشد) و هم محدودیت ماشین (ماشین مورد نظر بیکار باشد) رعایت شده باشند. محدودیت‌های ۱۵ و ۱۶ موجب می‌شوند که به هر ماشین در هر نوبت تنها یک عملیات از یک کار اختصاص یابد. محدودیت‌های ۱۷ و ۱۸ متغیرهای مسئله را به صورت صفر و ۱ تعریف می‌کنند. محدودیت ۱۹ نیز این الزام را به وجود می‌آورد که متغیرها باید مثبت باشند.

۳. تقلیل مسئله

تقلیل مسئله رنگ آمیزی به تولید کارگاهی

در یک گراف مختلط $G=(V, A, E)$ ، چنانچه عدد رنگی $\varphi(v_i) \in N$ تخصیص داده شده به ترتیب برای رئوس کمان $(v_i, v_j) \in A$ و یال‌های $[v_i, v_j] \in E$ با توجه به محدودیت‌های زیر باشد:

$$\varphi(v_i) < \varphi(v_j) \quad (20)$$

$$\varphi(v_i) \neq \varphi(v_j) \quad (21)$$

هدف رنگ‌آمیزی یک گراف مختلط $G=(V, A, E)$ خواهد بود؛ به طوری که تعداد رنگ‌های یکسان حداقل می‌شود.

رنگ‌آمیزی گراف مختلط را می‌توان به تولید کارگاهی تبدیل کرد که هدف آن حداقل کردن زمان تکمیل n کار روی m ماشین با زمان پردازش‌های متفاوت است. این مسئله می‌تواند به صورت رنگ‌آمیزی یک گراف مختلط $G=(V, A, E)$ با مجموعه رئوس V که نشان‌دهنده مجموعه عملیات است فرموله شود. در این صورت مجموعه‌ای از یال E می‌تواند نشان‌دهنده عملیات پردازش شده از سوی ماشین‌های یکسان باشد؛ از این رو (V^k, \emptyset, E_k) مجموعه عملیات V^k پردازش شده توسط ماشین یکسان M_k است و یک مجموعه کمان A می‌تواند نشان‌دهنده مجموعه عملیات روی کاری مشخص باشد؛ بنابراین (V_i, A_i, \emptyset) مجموعه عملیات V_i برای پردازش کار J_i است. همچنین محدودیت‌های ۲۰ و ۲۱

$$Tm_{i,k} + pt_{j,p,h} \cdot x_{i,j,p,h,k} \leq Tm_{i,k+1} \quad (4)$$

$$Tm_{i,k} \leq t_{j,p,h} + (1 - x_{i,j,p,h,k}) \cdot L \quad (5)$$

$$Tm_{i,k} + (1 - x_{i,j,p,h,k}) \cdot L \geq t_{j,p,h} \quad (6)$$

$$\sum_p \sum_j \sum_h x_{i,j,p,h,k} = 1 \quad (7)$$

$$\sum_k x_{i,j,p,h,k} = a_{i,j,p,h} \quad (8)$$

$$t_{j,p,h} + pt_{j,p,h} \leq F_p \quad (9)$$

$$F_p \leq S_p \quad (10)$$

$$A_p + S_p \leq Ct_p \quad (11)$$

$$Sm_{i,k'} + A_p \cdot Z_{i,p,k'} \leq Sm_{i,k'+1} \quad (12)$$

$$Sm_{i,k'} \leq S_p + (1 - Z_{i,p,k'}) \cdot L \quad (13)$$

$$Sm_{i,k'} + (1 - Z_{i,p,k'}) \cdot L \geq S_p \quad (14)$$

$$\sum_i \sum_k Z_{i,p,k'} = 1 \quad (15)$$

$$\sum_p Z_{i,p,k'} = 1 \quad (16)$$

$$x_{i,j,p,h,k} \in \{0,1\} \quad (17)$$

$$Z_{i,p,k'} \in \{0,1\} \quad (18)$$

$$S_p, F_p, t_{j,p,h}, Tm_{i,k}, Sm_{i,k'}, Ct_p \geq 0 \quad (19)$$

معادله ۱ تابع هدف مدل را نشان می‌دهد که حداقل‌سازی زمان تکمیل کارهاست. محدودیت ۲ مقدار تابع هدف را تعریف می‌کند. در محدودیت ۳ محدودیت‌های پیش‌نیازی معرفی می‌شود. این محدودیت زمان شروع هر عملیات را به اتمام عملیات قبلی از همان کار محدود می‌کند. محدودیت ۴، این الزام را به وجود می‌آورد که ماشینی می‌تواند به کار در نوبت $k+1$ پردازد که کار آن ماشین در نوبت k به اتمام رسیده باشد. محدودیت ۵ و ۶ نشان می‌دهند که یک عملیات در صورتی می‌تواند شروع شود که هم محدودیت پیش‌نیازی (عملیات قبلی آن به اتمام رسیده باشد) و هم محدودیت ماشین (ماشین مورد نظر بیکار باشد) رعایت شده باشند. محدودیت ۷ سبب می‌شود که به هر ماشین در هر نوبت، تنها یک عملیات از یک کار اختصاص یابد. با توجه به محدودیت ۸، برای اجرای یک عملیات از ماشینی اختصاص یافته به آن عملیات استفاده می‌شود. محدودیت ۹ بزرگ‌ترین زمان پردازش قطعات یک محصول را نشان می‌دهد. محدودیت ۱۰ حداقل زمان شروع مرحله مونتاژ را

هم زمان بر روی یک ماشین انجام شود؛ بنابراین چنانچه زمان همه عملیات برابر یا یک واحد باشد، رنگ آمیزی توالی شدنی عملیات V در فواصل زمانی واحد $[0, 1], (1, 2], (2, 3], \dots, (t-1, t]$ را تعیین می کند. در زیر روابط مسئله تولید کارگاهی و گراف مختلط شرح داده شده است:

| مسئله تولید کارگاهی | گراف مختلط |
|---|---|
| { عملیات v_j } | { رأس v_j } |
| { مجموعه عملیات V_i برای پردازش کار J_i } | { مجموعه رئوس V_i از مسیر (V_i, A_i, \emptyset) } |
| { مجموعه عملیات V^k پردازش شده توسط ماشین یکسان M_k } | { مجموعه رئوس V^k از گروه (V^k, \emptyset, E_k) } |
| { زمان بندی φ } | { رنگ آمیزی φ } |
| { زمان بندی بهینه } | { رنگ آمیزی بهینه } |

$|V_i|$ اندازه مجموعه i یا تعداد رئوس مجموعه i است:

$$LB_{11} = \max_{i=1}^n |V_i| \quad (22)$$

در بهترین حالت در گراف (V, \emptyset, E) ، یال های بدون جهت به صورتی جهت دار می شوند که عملیاتی از کار با بیشترین تعداد عملیات (i) ، کمترین زمان (عدد رنگی) را به خود بگیرند. به عبارت دیگر ابتدای کمان جهت دار با آن ها تلاقی داشته باشد که این مقدار با بزرگترین تعداد عملیات برابر است.

در بدترین حالت در گراف (V, \emptyset, E) ، یال های بدون جهت به صورتی جهت دار می شوند که عملیاتی از کار با بیشترین تعداد عملیات (i) ، بیشترین زمان (بزرگترین عدد رنگی) را به خود بگیرند. به عبارت دیگر انتهای کمان جهت دار با آن ها تلاقی داشته باشد.

$$UB_{11} = \varphi(V_i) = \max_m \varphi(V^m) \quad (23)$$

همچنین در مسئله مورد بررسی چنانچه تعداد کارها بر روی هر ماشین مشخص شود و حداقل تعداد عملیات تا رسیدن به آن ماشین با h_{\min}^m ($h_{\min}^m = \min_i (j-1: M_{ij} = M_k)$) و حداقل تعداد عملیات ممکن بعد از آن ماشین برابر با s_{\min}^m ($s_{\min}^m = \min_i (|V_i| - j: M_{ij} = M_k)$) باشد حد پایینی به صورت زیر ارائه می شود [۱۳]:

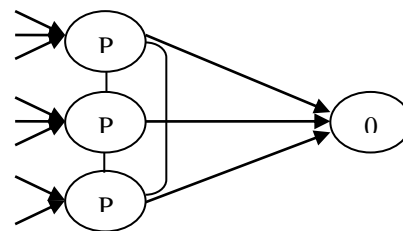
$$LB_{12} = \max_m (h_{\min}^m + |V^m| + s_{\min}^m) \quad (24)$$

که در آن $|V^m|$ تعداد کارها بر روی ماشین m که نشان دهنده تعداد تغییر رنگ است قرار دارد و حداکثر

شرایط پیش نیاز مسئله تولید کارگاهی را نشان می دهند. براساس محدودیت ۲۰ در یک کمان جهت دار، عدد رنگی v_j باید بزرگتر از عدد رنگی v_i باشد. به عبارت دیگر، عملیات بعدی در صورتی که عملیات قبلی آن ها انجام شده باشند قابل اجرا هستند. در یک یال بدون جهت نیز تنها باید عددهای رنگی رئوس مرتبط با هم مخالف یکدیگر باشند (محدودیت ۲۱). به عبارت دیگر کارها نباید

تقلیل مسئله رنگ آمیزی به مسئله مونتاژ

می توان مسئله رنگ آمیزی گراف مختلط را به مسئله مونتاژ تبدیل کرد که هدف آن حداقل کردن زمان تکمیل محصول بر روی ماشین مونتاژ است. چنانچه زمان لازم برای تمام محصولات یکسان و برابر ۱ باشد، می توان گرافی به شکل زیر برای مسئله مونتاژ رسم کرد:



شکل ۱. تقلیل مسئله رنگ آمیزی گراف به مسئله مونتاژ

در این مسئله به تعداد کل محصولات، گره در نظر گرفته می شود و همه گره ها به کمک یال های بدون جهت به یکدیگر متصل می شوند که این امر نشان دهنده عملیات مونتاژ هر محصول بر روی ماشین مونتاژ است. اکنون می توان مسئله را به طریق مشابه به مسئله مونتاژ موازی تبدیل کرد؛ به طوری که حداکثر مجاز به داشتن $m-1$ همسایه با رنگ مشابه باشد.

ارائه حد پایین و بالا با استفاده از مسئله رنگ آمیزی

برای گراف (V, A, \emptyset) بدون در نظر گرفتن یال های بدون جهت، ماکزیمم تعداد عملیات با فرض برابری زمان انجام عملیات $(pt=1)$ به صورت زیر به دست خواهد آمد که

به دست می‌آید. در نهایت با استفاده از حدود ارائه شده از مسئله تولید کارگاهی و مسئله ماشین موازی می‌توان به حدود بالا و پایین برای کل مسئله دست یافت. علاوه بر این با توجه به حداقل رنگ آمیزی در رابطه ۲۲ خواهیم داشت:

$$LB_5 = \min(\varphi(V_{ij})) + \left\lceil \frac{\sum_{i=1}^P |V_i'|}{m'} \right\rceil \quad (31)$$

همچنین می‌توان حدود بالا و پایین برای مسئله مورد نظر را با در نظر گرفتن زمان متفاوت برای کارها، با توجه به خاصیت مقیاس پذیری مسائل زمان بندی به ترتیب از طریق حاصل ضرب بزرگ‌ترین و کوچک‌ترین زمان پردازش (pt_{\min} و pt_{\max}) در حدود بالا و پایین حاصل از زمان پردازش یکسان به دست آورد.

۴. الگوریتم ترکیبی ژنتیک و بهینه‌سازی انبوه ذرات (HGAPSO)

با توجه به مدل ریاضی ارائه شده در بخش گذشته و محدودیت‌های متعدد آن، حل مسئله به وسیله روش‌های حل دقیق و نرم‌افزار بهینه‌سازی موجود، بسیار سخت به نظر می‌رسد؛ بنابراین به منظور صرفه جویی در زمان و برای حل مسائل اندازه متوسط و بزرگ از الگوریتم فراابتکاری استفاده شده است.

الگوریتم ژنتیک (GA) روش مبتنی بر جست‌وجوی تصادفی فراگیر و برگرفته از علم ژنتیک و نظریه تکامل داروین است. این الگوریتم براساس بقای بهترین‌ها با انتخاب طبیعی استوار است و در آن جمعیت از افراد براساس میزان مطلوبیت در محیط باقی می‌مانند. افرادی با قابلیت‌های برتر، شانس انتخاب بیشتری را خواهند یافت؛ بنابراین پس از چند نسل فرزندان با کارایی بهتر به وجود می‌آیند [۱۹]. کائو و زاهارا [۲۰] ادعا کردند که در میان الگوریتم‌های تکاملی موجود، بهترین شاخه شناخته شده الگوریتم ژنتیک است. همچنین ابرهات و کندی [۲۱] الگوریتم بهینه‌سازی انبوه ذرات (PSO) را ارائه کردند که از مزایای تعامل بین اجزا بهره می‌برد که ذرات نامیده می‌شوند. الگوریتم بهینه‌سازی انبوه ذرات از روش‌های محاسبات تکاملی است که با تقلید از پرواز پرندگان و تبادل مطالعات میان آن‌ها ابداع شده است [۲۲]. محاسبات الگوریتم PSO ساده است و ادغام این روش به الگوریتم ژنتیک تنها بار محاسباتی کمی به مسئله اضافه

مقدار به دست آمده بر روی تمامی ماشین‌ها حد پایینی را برای مسئله مورد نظر ایجاد می‌کند.

علاوه بر این چنانچه در هر کار حداکثر میزان تغییرات رنگ بر روی هر عملیات با $m(j)$ مشخص شود، حد پایین دیگری به صورت زیر حاصل می‌شود که در آن $|V_i|$ تعداد عملیات هر کار است.

$$LB_{13} = \min_i (\sum_j m(j) + |V_i|) \quad (25)$$

همچنین اگر تعداد رنگ‌های حاصل بدون در نظر گرفتن توالی ماشین‌ها با $n(j)$ و تعداد ماشین‌هایی که عملیاتی که با رنگ j را انجام می‌دهند با $m(j)$ نشان داده شود، $n(j) - m(j)$ تعداد کل تغییر رنگ بر روی رنگ j خواهد بود، که براساس آن:

$$LB_{14} = \max_i |V_i| + \min_j (n(j) - m(j)) \quad (26)$$

به منظور ارائه حدود بالا در مسئله مورد بررسی در صورتی که در هر مسیر وزن عملیات j از کار i به صورت $W_{ij} = \frac{pt_{ij}}{\sum_j pt_{ij}} = \frac{1}{|V_i|} \forall i, j$ بوده و جهت‌دهی به کمان‌ها از

بین عملیات مربوط به یک ماشین با عملیاتی با وزن کمتر باشد، حد بالا جدیدی به صورت زیر به دست خواهد آمد:

$$UB_{12} = \max_m \varphi(V^m) \quad (27)$$

علاوه بر این با توجه به میزان کار انجام شده در هر مرحله و جهت‌دهی به کمان‌ها از میان عملیات مربوط به یک ماشین با عملیاتی با وزن بیشتر، حد بالایی با توجه به روابط زیر به دست خواهد آمد:

$$W_{ij}' = \frac{\sum_{k=1}^{j-1} pt_{ik}}{\sum_j pt_{ij}} \forall i \quad \text{میزان کار انجام شده}$$

$$UB_{13} = \max_m \varphi(V^m) \quad (28)$$

همچنین برای به دست آوردن حد بالا و پایین برای مسئله مونتاژ با در نظر گرفتن حداکثر $m' - 1$ همسایه با رنگ مشابه خواهیم داشت:

$$UB_{21} = \max(\varphi(V_{ij})) + \left\lceil \frac{\sum_{i=1}^P |V_i'|}{m'} \right\rceil \quad (29)$$

$$LB_{21} = \max(\varphi(V_{ij})) + 1 \quad (30)$$

در رابطه ۲۹ و ۳۰ با استفاده از حدود بالا و پایین ارائه شده در مرحله تولید کارگاهی

گره‌ها تشکیل می‌شود که سطر اول اعداد مجاز رنگی و سطر دوم اعداد تصادفی تولید شده است. در هر مرحله، پس از تولید اعداد تصادفی رنگ‌های مربوط به ترتیب از چپ به راست به تعداد مجموع کل عملیات در هر دو مرحله تولید و مونتاژ انتخاب می‌شود و به عملیات تخصیص می‌یابند؛ برای مثال با در نظر گرفتن حداکثر عدد رنگی ۲ و مجموع کل عملیات ۵، ماتریس به صورت جدول ۱ ایجاد خواهد شد که سطر دوم آن اعداد تصادفی تولید شده است. سپس با توجه به این اعداد، اعداد رنگی تخصیص داده شده به هر درایه مشخص می‌شود. در نهایت با توجه به اینکه مجموع عملیات ۵ است، ۵ عدد رنگی به ترتیب از چپ به راست انتخاب و به عملیات تخصیص می‌یابد. در این حالت جواب تولید شده ممکن است شدنی نباشد؛ از این رو در هر مرحله میزان تخطی در رنگ آمیزی برای تمامی گره‌ها محاسبه و به عنوان جریمه در تابع هدف در نظر گرفته می‌شود.

تطبيق الگوریتم انبوه ذرات با فضای گسسته

در این حالت ابتدا اعداد به صورت نزولی مرتب می‌شوند، سپس جایگاه مربوط به هر درایه به عنوان عدد تصادفی مدنظر قرار می‌گیرد و در ادامه، بقیه مراحل مشابه قسمت قبل انجام می‌شود. جدول ۲ نحوه تبدیل جواب پیوسته به جواب گسسته را نشان می‌دهد.

شکل گیری و ترکیب جمعیت جدید

در این مسئله ترکیب شامل دو قسمت است. ابتدا برای قسمت اول کروموزوم که توالی عملیات زیرمسئله تولید کارگاهی را نشان می‌دهد، عددی تصادفی در بازه $[1, H]$ که H مجموع تعداد عملیات زیرمسئله است، تولید و A نامیده می‌شود. از میان عملیات زیرمسئله، A عملیات به براساس والد دوم تغییر می‌یابد و قسمت اول فرزند اول را تولید می‌کند. همچنین ترتیب این عملیات در والد دوم

می‌کند. همچنین الگوریتم PSO و GA هر دو با یک جمعیت از راه‌حل‌ها کار می‌کنند و ابزارهای الگوریتم ژنتیک مانند ادغام می‌تواند در بهبود قابلیت جست‌وجوی محلی PSO مؤثر باشد. ترکیب توانایی جست‌وجوی هر دو روش رویکردی خوب و کارا به نظر می‌رسد؛ بنابراین در این مقاله از الگوریتم ترکیبی ژنتیک و بهینه‌سازی انبوه ذرات برای حل مسئله استفاده شد. در مسئله مورد نظر در ابتدا هر عضو از جمعیت تولید می‌شود. این اعضا ممکن است به عنوان کروموزوم در GA یا به‌منزله ذرات در PSO تولید شوند. پس از آنکه مقادیر شایستگی برای همه راه‌حل‌ها محاسبه شد، نیمی از راه‌حل‌ها که بهترین تابع شایستگی را دارند، انتخاب و به عنوان نخبه ذخیره می‌شوند. مجموعه اعضای نخبه به صورت یک جامعه و هریک از نخبه‌ها یک ذره در نظر گرفته می‌شود. سپس اعضای نخبه با استفاده از الگوریتم PSO بهبود داده می‌شوند؛ به طوری که با استفاده از نخبه‌های بهبود یافته، نسل بعدی معمولاً بهتر از نسلی خواهند بود که به طور مستقیم از نخبه‌ها ایجاد می‌شوند. به این ترتیب نیمی از راه‌حل‌های نسل بعد، از راه‌حل‌های بهبود یافته PSO و باقی از تقاطع به دست خواهند آمد. پس از عمل تقاطع و جهش، جهت گزینش و انتقال کروموزوم‌ها به نسل بعد از الگوریتم شبیه‌سازی تبرید استفاده شد.

تولید جمعیت اولیه

در مسئله مورد نظر تولید جمعیت اولیه دو بخش دارد. به عبارت دیگر، برای هر عضو از جمعیت باید نحوه رنگ آمیزی قطعات در زیرمسئله تولید کارگاهی و همچنین نحوه رنگ آمیزی محصولات در زیرمسئله مونتاژ مشخص شود. در این حالت برای هر دو زیرمسئله از رویه تصادفی استفاده شده است.

نحوه نمایش جواب‌ها

در این حالت ماتریسی به طول حداکثر رنگ در مجموع کل

جدول ۱. نحوه نمایش جواب

| | | | | | | | | | | |
|--|----|---|---|---|---|---|---|---|---|---|
| اعداد رنگی مجاز | ۱ | ۱ | ۱ | ۱ | ۱ | ۲ | ۲ | ۲ | ۲ | ۲ |
| اعداد تصادفی | ۱۰ | ۳ | ۵ | ۱ | ۹ | ۴ | ۸ | ۲ | ۶ | ۷ |
| اعداد رنگی تخصیص داده شده | ۲ | ۱ | ۱ | ۱ | ۲ | ۱ | ۲ | ۱ | ۲ | ۲ |
| اعداد رنگی تخصیص داده شده به هر عملیات | ۲ | | ۱ | | ۱ | | ۱ | | ۲ | |

جدول ۲. نحوه تبدیل جواب پیوسته به جواب گسسته

| اعداد تصادفی | ترتیب نزولی | جواب پیوسته تولیدشده | اعداد رنگی مجاز |
|--------------|-------------|----------------------|-----------------|
| ۵ | ۰/۹۵۸ | ۰/۲۱۵ | ۱ |
| ۲ | ۰/۸۱۵ | ۰/۸۱۵ | ۱ |
| ۱۰ | ۰/۷۲۶ | ۰/۴۵۱ | ۱ |
| ۷ | ۰/۶۸۴ | ۰/۶۴۷ | ۱ |
| ۴ | ۰/۶۴۷ | ۰/۹۵۸ | ۱ |
| ۶ | ۰/۵۱۸ | ۰/۵۱۸ | ۲ |
| ۳ | ۰/۴۵۱ | ۰/۶۸۴ | ۲ |
| ۸ | ۰/۳۲۴ | ۰/۳۲۴ | ۲ |
| ۱ | ۰/۲۱۵ | ۰/۱۱۶ | ۲ |
| ۹ | ۰/۱۱۶ | ۰/۷۲۶ | ۲ |

و به منظور اعتبارسنجی مدل، از نمونه‌های تصادفی برای آن استفاده شد که نحوه تولید آن‌ها در جدول ۳ آمده است.

جدول ۳. نحوه تولید داده برای نمونه‌های تصادفی

| مقادیر | نمادها | پارامترها |
|--------------|--------|-----------------------------|
| {۲،۳،...،۲۰} | P | تعداد محصولات |
| {۲،۳،...،۱۰} | n | تعداد قطعات |
| {۲،۳،...،۱۰} | m | تعداد ماشین‌آلات در مرحله ۱ |
| {۲،۳،۴،۵} | m' | تعداد ماشین‌آلات در مرحله ۲ |

علاوه بر این تنظیم مناسب پارامترها و انتخاب درست عملگرها در الگوریتم‌های فراابتکاری تأثیر بسزایی بر عملکرد و کارایی آن‌ها دارد که باید با توجه به شرایط مسئله مورد نظر تعیین شود. با وجود این، بیشتر پژوهشگران این تنظیمات را با استفاده از مقادیر مرجع که در مقالات مشابه پیشنهاد شده است انجام می‌دهند. در این پژوهش از روش تاگوچی به منظور تنظیم پارامترهای الگوریتم استفاده شد. در نهایت عملکرد روش‌های حل ارزیابی و با یکدیگر مقایسه شدند. برای ارزیابی روش‌های حل پیشنهادی، نمونه‌ها به سه گروه با اندازه کوچک، اندازه متوسط و اندازه بزرگ تقسیم شدند.

در جدول‌های ۴، ۵ و ۶ نتایج حاصل از حل مسئله با استفاده از الگوریتم ژنتیک، همچنین نتایج حاصل از حدود بالا و پایین مسئله برای مسائل با اندازه‌های کوچک، بزرگ و متوسط آمده است. همچنین این جدول‌ها اعتبار و درستی حدود بالا و پایین مسئله را گزارش می‌کنند. با توجه به نتایج جدول ۴، بهترین حد پایین از میان حدود ارائه شده LB2 است و حدود دیگر تفاوت فراوانی با یکدیگر ندارند که این موضوع در شکل ۲ آمده است. همچنین از میان حدود بالای ارائه شده، ضعیف‌ترین حد ارائه شده UB1 است و دیگر حدود

براساس والد اول تغییر و قسمت اول فرزند دوم را تولید می‌کند. مشابه با حالت قبل برای قسمت دوم (زیرمسئله مونتاژ) نیز به همین ترتیب عمل می‌شود.

جهش

به منظور اجرای رویه جهش، پس از انتخاب دو ژن متفاوت به طور تصادفی ارزش‌های مرتبط با آن‌ها با یکدیگر تعویض می‌شوند.

انتخاب جمعیت

در این مقاله از چرخ رولت برای انتخاب کروموزوم‌ها به منظور عمل تقاطع استفاده شد. علاوه بر این به منظور گزینش و انتقال کروموزوم‌ها به نسل بعد الگوریتم شبیه‌سازی تبرید به کار گرفته شد. در این مرحله هر عضو از جمعیت جدید (فرزندان) با هر عضو از جمعیت قبل (والدین) مقایسه شدند. در صورتی که مقدار تابع هدف عضو جدید از مقدار تابع هدف عضو قبل بهتر یا با آن برابر باشد پذیرفته می‌شود. در غیر این صورت با احتمال شانس حضور در نسل بعد را خواهد داشت. در ابتدا فشار رویه انتخاب پایین است و گوناگونی افزایش می‌یابد، اما به مرور و با کاهش دما فشار رویه انتخاب افزایش و جست‌وجوی محلی قوت می‌یابد.

نتایج محاسباتی

در این قسمت به کمک نتایج محاسباتی، عملکرد و کارایی الگوریتم مورد نظر بررسی و مقایسه می‌شود. الگوریتم با استفاده از نرم‌افزار MATLAB (R2007a) کدنویسی و اجرا شده است. با توجه به اینکه اطلاعات مربوط به مسائل حل شده در مقالات مشابه برای مسئله مورد نظر وجود ندارد

را نشان می دهند. همچنین در شکل های ۶ و ۷ عملکرد تقریباً مشابه حدود بالا UB2 و UB3 نشان داده شده است و حد بالا UB1 عملکرد ضعیف تری از سایر حدود دارد؛ بنابراین با توجه به نتایج حاصل از حدود بالا و پایین ارائه شده عملکرد خوب حدود پایین LB2 و LB4، همچنین عملکرد ضعیف LB3 در همه مسائل با اندازه های متفاوت قابل مشاهده است. علاوه بر این با توجه به نتایج از میان حدود بالای ارائه شده ضعیف ترین حد UB1 است و سایر حدود مقادیر تقریباً مشابهی را نشان می دهند.

ارائه شده به طور تقریبی یکسان هستند (شکل ۵). علاوه بر این با توجه به جدول ۴، الگوریتم HGAPSO به جواب بهینه مشابه با جواب حاصل از GAMS دست می یابد. در مسائل با اندازه متوسط و بزرگ در جدول های ۵ و ۶ بهترین حد پایین LB2 و سپس LB4 است. در این مسائل با افزایش در ابعاد مسئله حد پایین LB3 نتایج ضعیف تری در مقایسه با سایر حدود به دست می آید که این مطلب در شکل های ۳ و ۴ قابل مشاهده است. همچنین با توجه به تأثیر گلوگاه، حدود پایین LB1 و LB5 در مسائل مختلف عملکردهای متفاوتی

جدول ۴. نتایج محاسباتی حل نمونه های اندازه کوچک

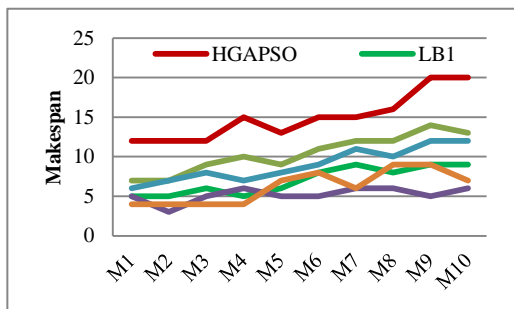
| Instance | Cplex | HGAPSO | LB1 | LB2 | LB3 | LB4 | LB5 | UB1 | UB2 | UB3 |
|----------|-------|--------|-----|-----|-----|-----|-----|-----|-----|-----|
| S1 | ۶ | ۶ | ۳ | ۶ | ۲ | ۴ | ۲ | ۷ | ۷ | ۷ |
| S2 | ۶ | ۶ | ۳ | ۵ | ۳ | ۳ | ۲ | ۶ | ۶ | ۶ |
| S3 | ۷ | ۷ | ۳ | ۶ | ۳ | ۴ | ۲ | ۱۰ | ۱۰ | ۱۱ |
| S4 | ۷ | ۷ | ۳ | ۵ | ۳ | ۴ | ۲ | ۱۱ | ۸ | ۸ |
| S5 | ۶ | ۶ | ۴ | ۶ | ۴ | ۵ | ۳ | ۹ | ۸ | ۸ |
| S6 | ۱۰ | ۱۰ | ۴ | ۶ | ۳ | ۴ | ۴ | ۱۴ | ۱۲ | ۱۰ |
| S7 | ۷ | ۷ | ۴ | ۶ | ۳ | ۵ | ۳ | ۱۴ | ۸ | ۱۲ |
| S8 | ۱۰ | ۱۰ | ۴ | ۶ | ۵ | ۵ | ۴ | ۱۳ | ۱۱ | ۱۱ |
| S9 | ۸ | ۸ | ۴ | ۵ | ۳ | ۴ | ۳ | ۱۵ | ۱۱ | ۱۱ |
| S10 | ۹ | ۹ | ۵ | ۵ | ۳ | ۵ | ۴ | ۱۸ | ۱۳ | ۱۳ |

جدول ۵. نتایج محاسباتی حل نمونه های اندازه متوسط

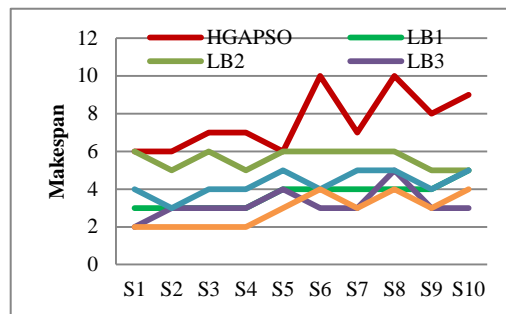
| Instance | Cplex | HGAPSO | LB1 | LB2 | LB3 | LB4 | LB5 | UB1 | UB2 | UB3 |
|----------|-------|--------|-----|-----|-----|-----|-----|-----|-----|-----|
| M1 | ۱۲ | ۱۲ | ۵ | ۷ | ۵ | ۶ | ۴ | ۲۳ | ۱۸ | ۱۷ |
| M2 | ۱۲ | ۱۲ | ۵ | ۷ | ۳ | ۷ | ۴ | ۲۵ | ۱۷ | ۱۷ |
| M3 | ۱۲ | ۱۲ | ۶ | ۹ | ۵ | ۸ | ۴ | ۲۶ | ۲۲ | ۲۰ |
| M4 | ۱۵ | ۱۵ | ۵ | ۱۰ | ۶ | ۷ | ۴ | ۲۹ | ۱۹ | ۱۸ |
| M5 | ۱۲ | ۱۳ | ۶ | ۹ | ۵ | ۸ | ۷ | ۲۴ | ۲۱ | ۱۸ |
| M6 | ۱۴ | ۱۵ | ۸ | ۱۱ | ۵ | ۹ | ۸ | ۳۰ | ۲۷ | ۲۶ |
| M7 | - | ۱۵ | ۹ | ۱۲ | ۶ | ۱۱ | ۶ | ۲۹ | ۲۵ | ۲۶ |
| M8 | - | ۱۶ | ۸ | ۱۲ | ۶ | ۱۰ | ۹ | ۳۳ | ۲۷ | ۲۷ |
| M9 | - | ۲۰ | ۹ | ۱۴ | ۵ | ۱۲ | ۹ | ۳۷ | ۲۹ | ۳۲ |
| M10 | - | ۲۰ | ۹ | ۱۳ | ۶ | ۱۲ | ۷ | ۴۳ | ۳۸ | ۳۱ |

جدول ۶. نتایج محاسباتی حل نمونه های اندازه بزرگ

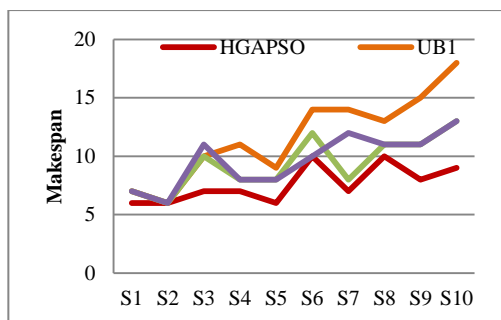
| Instance | HGAPSO | LB1 | LB2 | LB3 | LB4 | LB5 | UB1 | UB2 | UB3 |
|----------|--------|-----|-----|-----|-----|-----|-----|-----|-----|
| L1 | ۲۵ | ۸ | ۱۳ | ۴ | ۱۱ | ۶ | ۴۷ | ۳۸ | ۳۸ |
| L2 | ۲۳ | ۸ | ۱۵ | ۵ | ۱۳ | ۶ | ۶۱ | ۵۸ | ۴۰ |
| L3 | ۳۱ | ۷ | ۱۹ | ۵ | ۱۱ | ۸ | ۶۹ | ۶۴ | ۵۲ |
| L4 | ۳۷ | ۹ | ۱۷ | ۸ | ۱۲ | ۸ | ۷۴ | ۵۶ | ۵۷ |
| L5 | ۳۹ | ۹ | ۱۹ | ۸ | ۱۰ | ۹ | ۹۵ | ۸۸ | ۵۹ |
| L6 | ۴۵ | ۱۰ | ۲۴ | ۷ | ۱۳ | ۹ | ۹۹ | ۸۸ | ۸۱ |
| L7 | ۴۸ | ۱۱ | ۲۷ | ۶ | ۱۵ | ۱۰ | ۸۹ | ۶۹ | ۸۳ |
| L8 | ۵۷ | ۹ | ۳۰ | ۹ | ۱۱ | ۱۱ | ۱۰۳ | ۸۵ | ۹۵ |
| L9 | ۶۹ | ۱۲ | ۳۴ | ۶ | ۱۵ | ۱۱ | ۱۲۰ | ۹۹ | ۱۱۶ |
| L10 | ۷۶ | ۱۴ | ۴۱ | ۷ | ۱۶ | ۱۳ | ۱۴۲ | ۱۰۳ | ۱۲۳ |



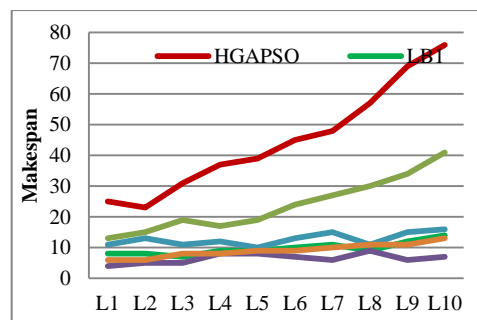
شکل ۳. حدود پایین ارائه شده برای مسائل با اندازه متوسط



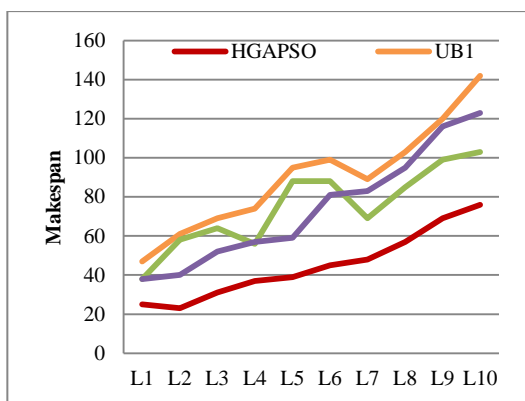
شکل ۲. حدود پایین ارائه شده برای مسائل با اندازه کوچک



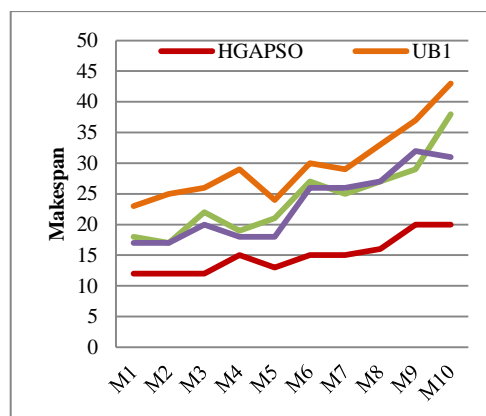
شکل ۵. حدود بالای ارائه شده برای مسائل با اندازه کوچک



شکل ۴. حدود پایین ارائه شده برای مسائل با اندازه بزرگ



شکل ۷. حدود بالای ارائه شده برای مسائل با اندازه بزرگ



شکل ۶. حدود بالای ارائه شده برای مسائل با اندازه متوسط

عملکرد خوب الگوریتم HGAPSO را نشان می‌دهند. از میان حدود ارائه شده LB2 بهترین حد پایین و LB3 و UB1 بدترین حد پایین و بالا و برای سایر حدود نتایج مشابه و نزدیک به هم مشاهده شد. علاوه بر این به منظور پژوهش‌های آتی محورهای متعددی پیشنهاد می‌شود که از مهم‌ترین آن‌ها در نظر گرفتن سایر الگوریتم‌های ابتکاری و فراابتکاری و تقلیل مسئله مورد نظر به سایر مسائل مشابه موجود مانند مسیریابی وسایل نقلیه و... است.

نتیجه‌گیری

در این پژوهش حل مسئله تولید کارگاهی همراه با یک مرحله مونتاژ موازی بررسی شد. مسئله مدنظر به مسئله رنگ‌آمیزی گراف تقلیل داده و سپس حدود بالا و پایین برای آن ارائه شد. به دلیل پیچیدگی بالای مسئله مورد بررسی برای حل آن از الگوریتم ترکیبی ژنتیک و بهینه‌سازی انبوه ذرات استفاده شد. نتایج محاسباتی عملکرد مناسب حدود بالا و پایین ارائه شده و همچنین

منابع

1. Lee, C. Y., Cheng, T. C. E. and Lin, B. M. T. (1993). "Minimizing the Makespan in the-Machine Assembly-Type Flowshop Scheduling Problem", *Management Science*, Vol. 39, No. 5, PP. 616-625.
2. Al Anzi, F. S., and Allahverdi, A. (2013). "An Artificial Immune System Heuristic for Two-Stage Multi-Machine Assembly Scheduling Problem to Minimize Total Completion Time", *Journal of Manufacturing Systems*, Vol. 32, No. 4, PP. 825– 830.
3. Allahverdi, A., and Al Anzi, F. S. (2006). "A Branch and Bound Algorithm for Three-Machine Flowshop Scheduling Problem to Minimize Total Completion Time with Separate Setup Times", *European Journal of Operational Research*, Vol. 169, No. 3, PP. 767-780.
4. Seyedi, I., Maleki Daronkolaei, A., and Kalashi, F. (2012). "Tabu Search and Simulated Annealing for New Three-Stage Assembly Flow Shop Scheduling with Blocking", *Interdisciplinary Journal of Contemporary Research in usiness*, Vol. 4, No. 8, PP. 394-402.
5. Navaei, J. et al. (2014). "Heuristics for an Assembly Flow Shop with Non-Identical Assembly Machines and Sequence Dependent Setup Times to Minimize Sum of Holding and Delay Costs", *Computers and Operations Research*, Vol. 44, No. 4, PP. 52–65.
6. Hariri, A. M. A., and Potts, C. N. (1997). "A Branch and Bound Algorithm for the Two-Stage Assembly Scheduling Problem", *European Journal of Operational Research*, Vol. 103. No. 3, PP. 547-556.
7. Yokoyama, M. (2001). "Hybrid Flow Shop Scheduling with Assembly Operations", *International Journal of Production Economics*, Vol. 73, PP. 103-116.
8. Sung, Ch.S., and Juhn, J. (2009). "Makespan Minimization for a 2-Stage Assembly Scheduling Problem Subject to Component Available Time Constraint", *International Journal of Production Economics*, Vol. 119, No. 2, PP. 392-401.
9. Sung, C. S., and Kim, H. Ah. (2008). "A Two-Stage Multiple-Machine Assembly Scheduling Problem for Minimizing Sum of Completion Times", *International Journal of Production Economics*, Vol. 113, No. 2, PP. 1038-1048.
10. Fattahi, P., Hosseini, S. M. H., and Jolai, F. (2012). "A Mathematical Model and Extension Algorithm for Assembly Flexible Flow Shop Scheduling Problem", *International Journal of Advance Manufacture Technology*, Vol. 65, No. 5-8, PP. 787-802.
11. Garey, M. R., and Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, LA: Freeman.
12. Sotskov, Y. N., and Tanaev, V. S. (1976). "Chromatic Polynomial of a Mixed Graph", *Vestsi Akademii Navuk BSSR, Seryya Fizika-Matematychnykh Navuk, Minsk*. Vol. 140, No. 6, PP. 20–23. Russian.
13. Sotskov, Y. N., Dolgui, A., and Werner, F. (2001). "Mixed graph coloring for unit-time job-shop scheduling", *International Journal of Mathematical Algorithms*, Vol. 2, No. 4, PP. 289–323.
14. Al Anzi, F. S. et al. (2006). "Using Mixed Graph Coloring to Minimize Total Completion Time in Job Shop Scheduling", *applied Mathematics and omputation*, Vol. 182, No. 2, PP. 1137–1148.
15. Kouider, A. et al. (2015). "Mixed Integer Linear Programs and Tabu Search Approach to Solvemixed Graph Coloring for Unit-Time Job Shop Scheduling", In *IEEE International Conference on Automation Science and Engineering (CASE)*, August 24–28, Gothenburg, Sweden, 1177–1181.
16. Kouider, A. et al. (2017). "Mixed Graph Colouring For Unit-Time Scheduling", *International Journal of Production Research*, Vol. 55, No. 6, PP. 1720-1729.
17. Shen, J. W. (2003). "Solving the Graph Coloring Problem Using Genetic Programming", In *Genetic Algorithms and Genetic Programming at Stanford 2003: 187-196*, Stanford Bookstore.
18. Daneshamooz, F., Jabbari, M., and Fattahi, P. (2013). "A Model for Jobshop Scheduling with a Parallel Assembly Stage to Minimize Makespan", *Journal of Industrial Engineering Research in Production Systems*, Vol. 2, No. 4, PP. 39-53.

19. Setak, M. et al. (2014). "Capacitated Multi-Depot Vehicle Routing Problem with Inter-Depot Routes", *Journal of Industrial Engineering, University of Tehran, Special Issue*, Vol. 48, PP. 11-18.
20. Kao, Y. T., and Zahara, E. (2008). "A Hybrid Genetic Algorithm and Particle Swarm Optimization for Multimodal Functions", *Applied Soft Computing*, Vol. 8, No. 2, PP. 849-857.
21. Eberhart, R., and Kennedy, J. (1995). "A New Optimizer Using Particle Swarm Theory", 6th Int. Symposium on Micro Machine and Human Science, Nagoya, Japan, PP. 4-39.
22. Kiani, M. et al. (2015). "An Efficient Genetic Algorithm for a Vehicle Routing Problem Considering the Competency of Working Teams", *Journal of Industrial Engineering, University of Tehran*, Vol. 49, No. 2, PP. 257-271.

واژه‌های انگلیسی به ترتیب استفاده در متن

1. Scheduling
2. Job Shop
3. Parallel Assembly
4. Graph Coloring