

توسعه الگوریتم بهینه‌سازی انبوه ذرات جهت حل مدل یکپارچه برنامه‌ریزی تولید و سیستم تولید سلولی پویا

عاطفه کهنی اردکانی^۱، فرناز برزین‌پور^{۲*} و رضا توکلی‌مقدم^۳

^۱ کارشناسی ارشد مهندسی صنایع - دانشکده فنی و مهندسی - دانشگاه پیام‌نور تهران

^۲ استادیار دانشکده مهندسی صنایع - دانشگاه علم و صنعت ایران

^۳ استاد گروه مهندسی صنایع - دانشکده فنی و مهندسی - دانشگاه تهران

(تاریخ دریافت ۹۰/۴/۱۱، تاریخ دریافت روایت اصلاح‌شده ۹۰/۶/۲۳، تاریخ تصویب ۹۱/۱/۲۰)

چکیده

سیستم تولید سلولی، یکی از مهم‌ترین کاربردهای تکنولوژی گروهی است. مسئله تشکیل سلول و برنامه‌ریزی تولید، دو گام مهم در پیاده‌سازی این سیستم هستند. در این مقاله یک مدل جدید برای مسئله یکپارچه تشکیل سلول پویا و برنامه‌ریزی تولید با هدف حداقل‌سازی هزینه‌های کلی شامل هزینه‌های ماشین، حمل و نقل بین سلولی و درون سلولی، استقرار مجدد، مصرف ابزار، نگهداری موجودی و کمبود و برون‌سپاری قطعات با شرط در دسترس بودن ابزار، ارائه می‌شود. با توجه به NP-Hard بودن مدل پیشنهادی، یک الگوریتم بهینه‌سازی انبوه ذرات توسعه‌یافته برای حل مسئله طراحی می‌شود. در الگوریتم پیشنهادی، با استفاده از اطلاعات بهینه محلی و مقداردهی دوباره، بدترین ذرات پراکندگی جواب‌ها افزایش یافته و از همگرایی زودرس جلوگیری می‌شود. مقایسه نتایج الگوریتم پیشنهادی با نتایج حاصل از نرم‌افزار LINGO 8.0 در مسائل کوچک و با الگوریتم بهینه‌سازی انبوه ذرات استاندارد در مسائل با ابعاد مختلف، کارآمدی آن را نشان می‌دهد.

واژه‌های کلیدی: سیستم تولید سلولی پویا، تشکیل سلول، برنامه‌ریزی تولید، الگوریتم بهینه‌سازی انبوه ذرات

مقدمه

شبیه‌سازی محیط‌های تولید واقعی است. از جمله مزایای این یکپارچه‌سازی کاهش زمان راه‌اندازی، نگهداری مواد، موجودی در جریان ساخت و زمان پاسخ‌گویی به بازار، افزایش بهره‌وری و کیفیت تولید است [۲]. در تحقیقات اخیر در حوزه مسائل یکپارچه سیستم تولید سلولی پویا و برنامه‌ریزی تولید، بعضی از سیاست‌های برنامه‌ریزی تولید، مانند هزینه‌های ماشین، نگهداری موجودی، سفارش‌های عقب‌افتاده^۵ و برون‌سپاری قطعات^۶، برای تشکیل سلول‌های تولیدی در نظر گرفته شده است [۲،۳]؛ در حالی که بعضی از هزینه‌ها و محدودیت‌های تأثیرگذار مانند هزینه و محدودیت مصرف ابزار نادیده گرفته شده است. همچنین بررسی تأثیر سیاست‌های برنامه‌ریزی تولید بر نحوه تشکیل سلول، نشان‌دهنده انحراف زیادی از جواب‌های بهینه و ارائه برنامه تولید و ساختار سلولی با نوسانات زیادی بوده است [۲]. بنابراین ارائه راه‌حلی برای کاهش این نوسانات و طراحی مدلی جامع و کاربردی برای این مسئله کاری ضروری است.

سیستم تولید سلولی^۱ (CMS)، یکی از سیستم‌های کارآمد برای محیط‌های تولیدی با حجم و تنوع بالای محصولات است که وجود تقاضا و ترکیب متغیر محصولات، آن را به سمت پویا^۲ هدایت می‌کند. مراحل اجرای این سیستم شامل: تشکیل سلول^۳، برنامه‌ریزی و زمان‌بندی تولید و ارائه استقرار نهایی است [۱]. تشکیل سلول شامل طبقه‌بندی قطعات به خانواده‌های مشابه و تخصیص ماشین‌های مورد نیاز به سلول‌ها است. همچنین مدیریت منابع تولید و متعادل کردن آن‌ها بین دوره‌های متوالی با هدف حداقل‌سازی هزینه‌های تولید به عنوان برنامه‌ریزی تولید^۴ شناخته می‌شود [۲].

اغلب مرحله تشکیل سلول به دلیل اهمیت آن در صدر توجه محققان قرار گرفته و نسبت به سایر مراحل طراحی، تحقیقات بیشتری روی آن انجام شده است؛ در حالی که موضوع برنامه‌ریزی تولید نیز جزء مهم دیگری از این سیستم است که اهداف و نتایج آن بر نحوه تشکیل سلول تأثیرگذار است. یکپارچه‌سازی مفاهیم تشکیل سلول و برنامه‌ریزی تولید، نیاز اساسی برای مدل‌سازی و

همکاران [۸] یک مدل تشکیل سلول پویا با هدف حداقل سازی هزینه‌های کلی ارائه و برای حل آن از الگوریتم ممتیک^{۱۱} (MA) استفاده کردند و در [۹] نیز به مقایسه نتایج الگوریتم ژنتیک^{۱۲} (GA)، جستجوی ممنوع^{۱۳} (TS) و SA برای حل این مدل پرداختند.

جوادیان [۱۰] مدل ارائه شده در [۷] را با تقاضای پویای احتمالی توسعه داده و آن را به وسیله GA حل کرد. سعیدی و همکاران [۱۱] نیز از یک روش شبکه عصبی برای حل مدل تشکیل سلول پویا با هدف حداقل سازی هزینه‌های کلی استفاده کردند. صفایی و همکاران [۱۲] به بازبینی مقالاتی که بحث پویایی را در تولید سلولی در نظر گرفته بودند، پرداختند و سپس با افزودن هزینه حمل و نقل بین سلولی و درون سلولی به مدل ارائه شده در [۱۱]، آن را توسعه دادند.

تحقیقات متعددی، موضوع تشکیل سلول پویا و برنامه‌ریزی تولید را به طور مجزا مورد بررسی قرار داده‌اند؛ در حالی که یکپارچه‌سازی این دو موضوع سبب نزدیکی به شرایط واقعی خواهد شد. پتراو [۱۳] و هایر و همکاران [۱۴] نشان دادند که هنگام پیاده‌سازی سیستم تولید سلولی در یک محیط تولیدی، اهداف برنامه‌ریزی تولید در ساختار سلولی مؤثر است. اسکالر و همکاران [۱۵] یک مدل دو مرحله‌ای برای ترکیب این دو موضوع پیشنهاد دادند که در هر مرحله لیستی از سلول‌های بالقوه را با توجه به هزینه‌ها تهیه و با توجه به بهترین مجموعه از سلول‌ها، برنامه‌ریزی تولید را برای هر قطعه انجام دادند.

سانگ و هیتومی [۱۶] و چن [۱۷] به ارائه مدل‌هایی یکپارچه برای این موضوع پرداختند. اما آن‌ها فقط سیاست نگهداری موجودی را در نظر گرفته و از سیاست برون‌سپاری قطعات و سفارش‌های عقب‌افتاده صرف‌نظر کردند. چن و گو [۱۸] و دفرشا و چن [۱۹] در تحقیقاتی جداگانه فقط سیاست برون‌سپاری قطعات را در مدل در نظر گرفتند.

کیون و همکاران [۳] در یک تحقیق کامل‌تر از سیاست برون‌سپاری قطعات و نگهداری موجودی به طور همزمان استفاده کردند. اما آن‌ها تأثیر برون‌سپاری قطعات بر استقرار مجدد را بررسی نکردند. صفایی و توکلی مقدم [۲] سیاست سفارش‌های عقب‌افتاده را نیز به مدل [۳] اضافه کرده و آن را برای مسائل با اندازه کوچک با

در این تحقیق یک مدل برنامه‌ریزی عدد صحیح غیرخطی مختلط برای مسئله یکپارچه تشکیل سلول و برنامه‌ریزی تولید پویا با هدف حداقل سازی هزینه‌های کلی شامل هزینه‌های ماشین، حمل و نقل بین سلولی^۷ و درون سلولی^۸، استقرار مجدد^۹، مصرف ابزار، نگهداری موجودی یا کمبود و برون‌سپاری قطعات با در نظر گرفتن محدودیت در برون‌سپاری قطعات به شرط در دسترس بودن ابزار ارائه می‌شود. در اکثر مقالات، هزینه افزودن یا کسر ماشین در هر دوره، یکسان فرض شده و برابر نصف مجموع هزینه‌های نصب، حرکت و جابجایی ماشین است که واقعی نبوده و در این تحقیق اصلاح شده است. شرط در دسترس بودن ابزار و هزینه آن نیز که از محدودیت‌های کاربردی در صنایع است به مدل پیشنهادی اضافه شده است.

در ادامه، در بخش مرور ادبیات تاریخچه، مهم‌ترین کارهای مرتبط و در بخش مدل‌سازی ریاضی، جزئیات مدل پیشنهادی ارائه می‌شود. یک مثال عددی برای اعتبارسنجی مدل در بخش مثال عددی و الگوریتم پیشنهادی بهینه‌سازی انبوه ذرات با استفاده از اطلاعات بهینه محلی^{۱۰} (LPSO) در بخش الگوریتم PSO، معرفی می‌شود. نتایج محاسباتی حاصل از الگوریتم پیشنهادی و مقایسه با جواب بهینه و الگوریتم PSO استاندارد برای مسائل با ابعاد مختلف در بخش تحلیل نتایج ارائه می‌شود. نتیجه‌گیری نهایی در بخش آخر مطرح می‌شود.

مرور ادبیات

CMS یکی از مهم‌ترین کاربردهای تکنولوژی گروهی است [۱]. CMS سنتی ایستا بوده و هر تغییری در تقاضای محصول در آن نادیده گرفته می‌شود. فانتھون [۴] به یک بازنگری جامع درباره تحقیقات که روی CMS انجام شده، پرداخته است.

تنوع بالا و تقاضای متغیر محصولات، CMS سنتی را به سمت پویایی هدایت می‌کند. در جدول (۱) مقایسه‌ای بین خصوصیات مهم‌ترین تحقیقات بررسی شده در مرور ادبیات و خصوصیات مدل پیشنهادی در این تحقیق ارائه شده است. آرهیول و همکاران [۵]، چن [۶] و وانگ و همکاران [۷] اولین کسانی بودند که مفهوم پویایی را در طراحی ساختار سلولی در نظر گرفتند. توکلی مقدم و

جدول ۱: مرور ادبیات سیستم‌های تولید سلولی پویا

روش حل	۱			۲			۳			۴			۵			۶			۷			۸			۹		
	الف	ب	-	الف	ب	-	الف	ب	ج	الف	ب	ج	الف	ب	ج	الف	ب	ج	الف	ب	ج	الف	ب	ج			
---	x						x																				
SA	x																										
MA																											
GA																											
SA																											
نرم افزار LINGO																											
تجزیه بندرز TS																											
TS																											
نرم افزار CPLEX																											
نرم افزار LINGO																											
LPSO	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x		

این تحقیق

مهم‌ترین خصوصیات مسائل سیستم تولید سلولی پویا

۱- تقاضای متغیر الف) تقاضای متغیر ب) تقاضای احتمالی ۲- ماشین‌های چندکاره ۳- مسیرهای چندگانه	۴- هزینه‌های ماشین الف) هزینه ثابت ماشین ب) هزینه عملیاتی ماشین ج) هزینه نگهداری ماشین ۵- هزینه مصرف ابزار	۶- هزینه حمل و نقل مواد الف) حمل و نقل بین سلولی ب) حمل و نقل درون سلولی ۷- تکرار ماشین	۸- هزینه‌های برنامه‌ریزی تولید الف) برون‌سپاری قطعات ب) نگهداری موجودی ج) سفارشات عقب‌افتاده ۹- کنترل میزان برون‌سپاری قطعات
---	--	--	--

مدل‌سازی ریاضی

در این بخش با توسعه مدل [۲]، یک مدل عدد صحیح غیرخطی برای مسئله یکپارچه برنامه‌ریزی تولید و تشکیل سلول پویا ارائه می‌شود. فرضیه‌ها، پارامترها و متغیرهای تصمیم مورد نیاز برای این مدل در ادامه آورده می‌شود:

فرضیه‌ها

- تقاضای هر قطعه در هر دوره متغیر و معلوم است.
- هزینه ثابت ماشین دربرگیرنده هزینه‌های بالاسری، نگهداری و اجاره است.
- هزینه متغیر هر ماشین وابسته به بار کاری اختصاص یافته به آن است.
- نگهداری موجودی، کمبود و برون‌سپاری قطعات استثنایی مجاز است.
- هر ماشین می‌تواند یک یا چند عملیات را بدون هزینه اضافی انجام دهد (انعطاف‌پذیری ماشین).

استفاده از LINGO 8.0 حل کردند. همچنین آن‌ها نشان دادند که تأثیر متقابل این دو موضوع ممکن است سبب انحراف تصمیم‌گیرنده از جواب بهینه شود. اما آن‌ها راهی برای کاهش این تأثیر ارائه ندادند.

با توجه به مطالعات بررسی شده، نیاز به تکامل مدل یکپارچه برنامه‌ریزی تولید و تشکیل سلول تولیدی در راستای کاهش تأثیر برون‌سپاری قطعات بر هزینه‌های استقرار مجدد وجود دارد. محدودیت در دسترس بودن ابزار نیز محدودیت کاربردی است که اغلب نادیده گرفته می‌شود. همچنین به دلیل NP-Hard بودن مسایل CMS استفاده از یک روش فراابتکاری ضروری است [۸]. در حالی که در همه تحقیقاتی که به یکپارچه‌سازی این موضوع پرداخته‌اند، از نرم‌افزارهای بهینه‌سازی که فقط قادر به حل مسائل با سایز کوچک است، استفاده شده است. بنابراین در این تحقیق از الگوریتم LPSO برای حل مدل و از الگوریتم PSO استاندارد برای بررسی عملکرد الگوریتم پیشنهادی استفاده می‌شود.

متغیرهای تصمیم

N_{mch} : تعداد ماشین نوع m تخصیص یافته به سلول c در دوره h

K_{mch}^+ : تعداد ماشین نوع m اضافه شده به سلول c در دوره h

K_{mch}^- : تعداد ماشین نوع m کسر شده از سلول c در دوره h

Q_{ph} : تعداد تقاضای قطعه p برای تولید داخلی در دوره h

S_{ph} : تعداد تقاضای قطعه p برای برون سپاری در دوره h

I_{ph} : سطح موجودی خالص (I_{ph}^+ : موجودی باقیمانده قطعه p در پایان دوره h , I_{ph}^- : کمبود قطعه p در پایان دوره h)

x_{jpmch} : اگر عملیات قطعه p روی ماشین m در سلول c در دوره h انجام شود یک و در غیر این صورت صفر است.

y_{ph} : اگر $Q_{ph} > 0$ یک و در غیر این صورت صفر است.

y'_{ph} : اگر $I_{ph}^+ \geq 0$ یک و اگر $I_{ph}^- > 0$ صفر است.

مدل ریاضی پیشنهادی

مدل پیشنهادی عدد صحیح غیرخطی برای مسئله یکپارچه برنامه ریزی تولید و تشکیل سلول پویا با شرط در دسترس بودن ابزار به صورت زیر است:

$$\text{Min } Z = \sum_{h=1}^H \sum_{m=1}^M \sum_{c=1}^C \alpha_m N_{mch} + \sum_{h=1}^H \sum_{c=1}^C \sum_{p=1}^P \sum_{j=1}^{O_p} \sum_{m=1}^M \beta_m Q_{ph} t_{jpm} x_{jpmch} \quad (1-1)$$

$$+ \frac{1}{2} \sum_{h=1}^H \sum_{p=1}^P \sum_{j=1}^{O_p-1} \sum_{c=1}^C \left[\frac{Q_{ph}}{B_p^{\text{inter}}} \right] \gamma^{\text{inter}} \times \left| \sum_{m=1}^M x_{(j+1)pmch} - \sum_{m=1}^M x_{jpmch} \right| \quad (2-1)$$

$$+ \frac{1}{2} \sum_{h=1}^H \sum_{p=1}^P \sum_{j=1}^{O_p-1} \sum_{c=1}^C \left[\frac{Q_{ph}}{B_p^{\text{intra}}} \right] \gamma^{\text{intra}} \times \left(\sum_{m=1}^M \left| x_{(j+1)pmch} - x_{jpmch} \right| - \left| \sum_{m=1}^M x_{(j+1)pmch} - \sum_{m=1}^M x_{jpmch} \right| \right) \quad (3-1)$$

$$+ \sum_{h=1}^H \sum_{p=1}^P (\eta_p I_{ph}^+ + \rho_p I_{ph}^- + \lambda_p S_{ph}) \quad (4-1)$$

$$+ \sum_{h=1}^H \sum_{c=1}^C \sum_{p=1}^P \sum_{j=1}^{O_p} \sum_{m=1}^M Q_{ph} x_{jpmch} W_{jpm} \quad (5-1)$$

۶) هر قطعه در صورت در دسترس بودن ابزار، می تواند چند برنامه پردازشی داشته باشد که فقط یکی از آن ها استفاده می شود (انعطاف پذیری مسیر^{۱۴}).

پارامترها

P : تعداد انواع قطعات

C : حداکثر تعداد سلول ها

M : تعداد انواع ماشین

H : تعداد دوره ها

J : تعداد عملیات قطعه

G : تعداد ابزار

D_{ph} : تقاضای قطعه p در دوره h

B_p^{inter} : اندازه دسته تولیدی بین سلولی برای قطعه p

B_p^{intra} : اندازه دسته تولیدی درون سلولی برای قطعه p

γ^{inter} : هزینه حرکت بین سلولی برای هر دسته تولیدی

γ^{intra} : هزینه حرکت درون سلولی برای هر دسته تولیدی

α_m : هزینه ثابت ماشین m در هر دوره

β_m : هزینه متغیر ماشین m برای هر واحد زمانی

T_m : ظرفیت زمانی ماشین نوع m در هر دوره بر حسب ساعت

a_{jpm} : اگر عملیات j قطعه p بتواند روی ماشین نوع m انجام شود، برابر یک و در غیر این صورت برابر صفر است.

t_{jpm} : زمان اجرای عملیات j قطعه p روی ماشین m

λ_p : هزینه واحد برون سپاری قطعه p

η_p : هزینه نگهداری موجودی در واحد قطعه p در هر دوره

ρ_p : هزینه تأخیر سفارش ها در واحد قطعه p در هر دوره

H_m : هزینه نصب ماشین نوع m در یک سلول

R_m : هزینه برداشتن یا حذف ماشین نوع m از یک سلول

W_{jpm} : هزینه مصرف ابزار عملیات j قطعه p روی ماشین m

μ_{jpg} : اگر عملیات j قطعه p روی ماشین m به ابزار g نیاز داشته باشد، برابر یک و در غیر این صورت برابر صفر است.

σ_{gm} : اگر ابزار g روی ماشین m در دسترس باشد.

UC : حد بالای اندازه سلول

LO : حد پایین میزان برون سپاری قطعات

UO : حد بالای میزان برون سپاری قطعات

M : عدد مثبت بزرگ

نادیده گرفته می‌شود، در حالی که در صنایع نقش کاربردی دارد. عبارت (۱-۶) هزینه استقرار مجدد که جمع هزینه نصب ماشین‌های اضافه شده در یک دوره و هزینه برداشتن ماشین‌های کسر شده از یک دوره است را محاسبه می‌کند.

رابطه (۲) تضمین می‌کند که اگر قرار باشد یک بخش از تقاضای هر قطعه در یک دوره معین تولید شود، هر عملیات فقط به یک ماشین و یک سلول اختصاص می‌یابد. رابطه (۳) نشان می‌دهد که اختصاص عملیات به یک ماشین زمانی مجاز است که ابزار مورد نیاز آن در دسترس باشد. محدودیت (۴) تضمین‌کننده تأمین تقاضا و عدم تجاوز از ظرفیت ماشین است. محدودیت (۵) حد بالای اندازه سلول را نشان می‌دهد. رابطه (۶) نشان‌دهنده تعداد ماشین‌ها در دوره جاری در هر سلول است. رابطه (۷) برابر بالانس موجودی در طول دوره است و نشان می‌دهد که سطح موجودی (موجودی اضافی یا کمبود) هر قطعه در پایان هر دوره برابر سطح موجودی قطعه در پایان دوره قبلی به علاوه تعداد برون‌سپاری قطعات در دوره قبل منهای نرخ تقاضای قطعه در دوره جاری است. رابطه (۸) نشان می‌دهد که سطح خالص موجودی نیز باید در دوره پایانی صفر و تقاضای همه قطعات در طول افق برنامه‌ریزی تأمین شود. رابطه (۹) نشان‌دهنده حدود برون‌سپاری قطعات است. در صورت تعریف نکردن این رابطه، ممکن است میزان برون‌سپاری قطعات نامعقول باشد. به طور مثال ممکن است مقدار این متغیر برابر اعداد تک رقمی یا چند برابر تقاضای محصول در یک دوره باشد. روابط (۱۰) متمم رابطه (۲) است. در واقع اگر عملیات‌های مورد نیاز برای ساخت قطعات در رابطه (۲) قابل انجام باشد، بخشی از تقاضای آن می‌تواند در دوره معین در داخل تولید شود. روابط (۱۱) تضمین می‌کند که کمبود و نگهداری موجودی همزمان روی نمی‌دهد و دو متغیر وابسته هستند. رابطه (۱۲) نوع متغیرهای تصمیم را تعیین می‌کند.

مثال عددی

با توجه به اینکه مدل پیشنهادی در این تحقیق توسعه‌یافته، مدل ارائه شده در [۲] است، برای ارزیابی عملکرد آن، یک مثال عددی مطابق با داده‌های مثال اول در [۲] با استفاده از نرم‌افزار LINGO 8.0 حل شده است.

$$+ \sum_{h=1}^H \sum_{m=1}^M \sum_{c=1}^C (H_m K_{mch}^+ + R_m K_{mch}^-) \quad (۶-۱)$$

S.t

$$\sum_{m=1}^M \sum_{c=1}^C a_{jpm} x_{jpmch} = y_{ph} \quad \forall j, p, h \quad (۲)$$

$$\sum_{p=1}^P \sum_{j=1}^{Q_p} Q_{ph} t_{jpm} x_{jpmch} \leq T_m N_{mch} \quad \forall m, c, h \quad (۳)$$

$$x_{jpmch} \leq \mu_{jpg} \times \sigma_{gm} \quad \forall j, p, m, c, h, g \quad (۴)$$

$$\sum_{m=1}^M N_{mch} \leq UC \quad \forall c, h \quad (۵)$$

$$N_{mch} = N_{mc(h-1)} + K_{mch}^+ - K_{mch}^- \quad \forall m, c, h \quad (۶)$$

$$I_{ph} = I_{p(h-1)} + Q_{ph} + S_{p(h-1)} - D_{ph} \quad \forall p, h \quad (۷)$$

$$I_{ph}^+ - I_{ph}^- = I_{ph}, \quad I_{pH}^+ - I_{pH}^- = 0 \quad \forall p \quad (۸)$$

$$LO \leq S_{ph} \leq UO \quad \forall p, h \quad (۹)$$

$$Q_{ph} \leq M y_{ph}, \quad Q_{ph} \geq y_{ph} \quad \forall p, h \quad (۱۰)$$

$$I_{ph}^+ \leq M y'_{ph}, \quad I_{ph}^- \leq M (1 - y'_{ph}) \quad \forall p, h \quad (۱۱)$$

$$x_{jpmch}, y_{ph}, y'_{ph} \in \{0, 1\}$$

$$N_{mch}, K_{mch}^+, K_{mch}^-, Q_{ph}, I_{ph}^+, I_{ph}^-, S_{ph} \geq 0, \text{int} \quad (۱۲)$$

در مدل پیشنهادی، تابع هدف شامل عبارت‌های (۱-۱)، (۲-۱)، (۳-۱)، (۴-۱)، (۵-۱) و (۶-۱) است. عبارت (۱-۱) جمع هزینه‌های ثابت و متغیر ماشین است. هزینه ثابت، مدل را مجبور به استفاده حداکثر از ماشین‌ها می‌کند. هزینه متغیر نیز، جمع بار کاری تخصیص‌یافته به هر ماشین در هر سلول است و آن را بالانس می‌کند. عبارت (۲-۱) نشان می‌دهد که اگر دو عملیات متوالی در دو سلول جداگانه انجام شود، به حرکت بین‌سلولی نیاز است. عبارت (۳-۱) نشان می‌دهد که دو عملیات متوالی به حرکت درون‌سلولی نیاز دارد، اگر هر دو عملیات در یک سلول ولی روی دو ماشین مختلف انجام شود. عبارت (۴-۱) جمع هزینه‌های برنامه‌ریزی تولید شامل هزینه نگهداری موجودی، کمبود و هزینه‌های برون‌سپاری قطعات است. عبارت (۵-۱) هزینه مصرف ابزار را در صورت استفاده از آن‌ها محاسبه می‌کند. اغلب این هزینه

(رابطه ۹) در این تحقیق است. در واقع این محدودیت هزینه برون‌سپاری قطعات را کاهش و با استفاده حداکثری از ماشین‌ها، هزینه ثابت و متغیر ماشین را افزایش می‌یابد. اعمال این تغییرات و اصلاح هزینه استقرار مجدد (رابطه ۶-۱) و اصلاح رابطه بالانس موجودی (رابطه ۷) به ترتیب هزینه استقرار مجدد و نگهداری موجودی را نیز کاهش می‌دهد.

ساختار سلولی پیشنهادی مثال اول در [۲] شکل ۱ و ساختار پیشنهادی در این تحقیق در شکل ۲ نشان داده شده است. C_j و P_j ، به ترتیب سلول و قطعه شماره j و $M_j(n)$ ، n ماشین از نوع j را نشان می‌دهد. مقایسه ساختارهای سلولی، متعادل بودن برنامه تولید و کاهش استقرار مجدد در ساختار سلولی پیشنهادی این تحقیق را به خوبی نشان می‌دهد. در واقع به دلیل محدود نبودن میزان برون‌سپاری قطعات در مدل [۲]، در دوره دوم میزان برون‌سپاری زیاد و ساختار سلولی موجود در دوره اول، تغییر زیادی داشته است و این موضوع سبب استفاده نکردن از حداکثر ظرفیت‌های داخلی می‌شود.

برای قابل مقایسه بودن نتایج دو مثال، محدودیت در دسترس بودن ابزار از مدل حذف شده است. مسئله نمونه شامل ۵ قطعه (شامل ۳ عملیات)، ۵ ماشین، ۳ سلول و ۳ دوره است. مقادیر سایر پارامترهای مورد استفاده که در [۲] وجود ندارد، در جدول (۲) آورده شده است.

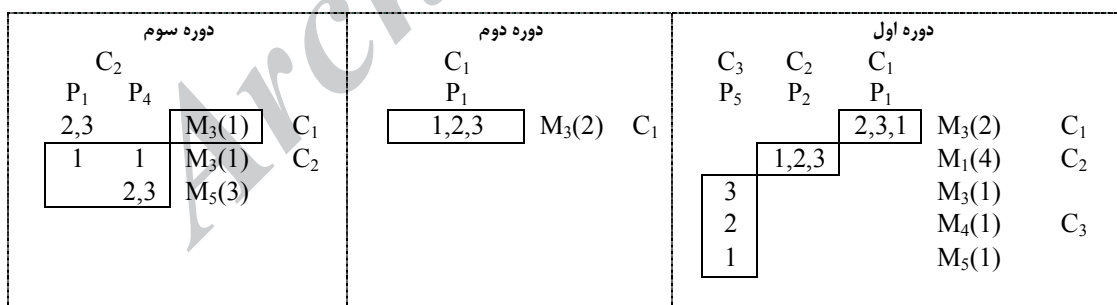
جدول ۲: توزیع داده‌های مورد استفاده

پارامتر	مقدار	پارامتر	مقدار
S_m	$\alpha_m/3$	LO	$0.5 \times D_{ph}$
R_m	$\alpha_m/4$	UO	$2 \times D_{ph}$

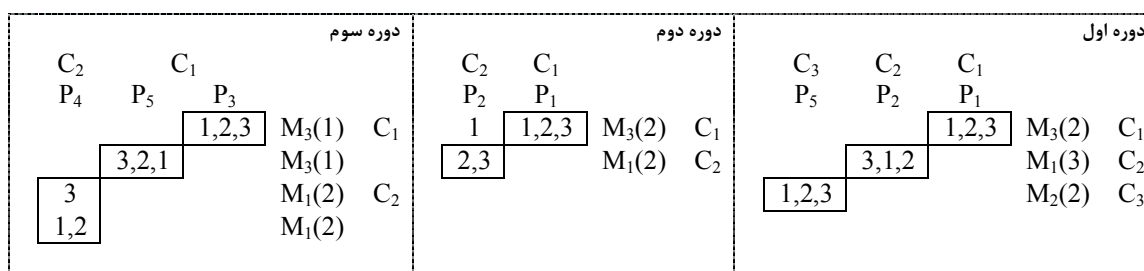
مقایسه نتایج جدول (۳) نشان می‌دهد که مقدار تابع هدف در این تحقیق حتی بدون در نظر گرفتن هزینه مصرف ابزار از تابع هدف [۲] کمتر است. در حالی که هزینه برون‌سپاری قطعات و حرکت بین‌سلولی کاهش و هزینه‌های ثابت و متغیر ماشین و حرکت درون‌سلولی افزایش یافته است. در حالی که هزینه برون‌سپاری قطعات و حرکت بین‌سلولی کاهش و هزینه‌های ثابت، متغیر ماشین و حرکت درون‌سلولی افزایش یافته است. این تغییرات به دلیل افزودن محدودیت برون‌سپاری قطعات

جدول ۳: مقایسه نتایج مثال اول در مدل ریاضی پیشنهادی و مدل [۲]

مرجع	مقدار هدف	هزینه ثابت ماشین	هزینه متغیر ماشین	هزینه حرکت بین سلولی	هزینه حرکت درون سلولی	هزینه استقرار مجدد	هزینه نگهداری موجودی	هزینه کمبود	هزینه برون‌سپاری
[۲]	144609	26100	46114	645	521	7375	14588	650	48616
این تحقیق	134493	33100	54320	384	886	4850	13263	490	26520



شکل ۱: ساختار سلولی پیشنهادی مثال اول در [۲]

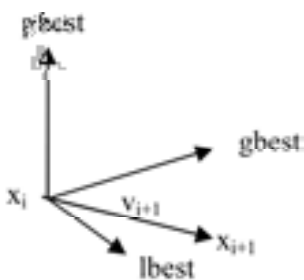


شکل ۲: ساختار سلولی پیشنهادی مثال اول در این تحقیق

اطلاعاتی بیشتر، فضای جستجو و پراکندگی الگوریتم را افزایش می‌دهد. بنابراین در الگوریتم پیشنهادی که به اختصار LPSO نامیده می‌شود، از بهترین جوابی که تا کنون توسط l همسایه بی‌واسطه هر ذره ($lbest^i$) به دست آمده است، نیز استفاده می‌شود. در الگوریتم LPSO هر مرحله مکان و سرعت هر ذره به ترتیب با استفاده از روابط (۱۴) و (۱۵) به روز می‌شود:

$$v_i(t+1) = w \cdot v_i(t) + c_1 r_1 [pbest_i(t) - x_i(t)] + c_2 r_2 [lbest_i(t) - x_i(t)] + c_3 r_3 [gbest(t) - x_i(t)] \quad (15)$$

در رویکرد پیشنهادی یک ذره برای حرکت از مکان قبلی (x_i) به مکان جدید (x_{i+1}) با استفاده از اطلاعات $pbest$ ، $lbest$ و $gbest$ را تعیین و طبق شکل ۴ حرکت می‌کند.



شکل ۴: نحوه حرکت یک ذره در LPSO پیشنهادی

مراحل الگوریتم PSO پیشنهادی

نحوه نمایش جواب: در اینجا بردار مکان (x_i) و سرعت (v_i) ذره به صورت یک ماتریس 1×7 بعدی است که هر یک از درایه‌های آن یک ماتریس چندبعدی است. شکل ۵ نحوه نمایش این ماتریس را نشان می‌دهد. به طور مثال N_{mch} یک ماتریس $m \times c \times h$ بعدی از تعداد ماشین‌ها است.

$$x_i, v_i = [N_{mch} \quad K_{mch} \quad Q_{ph} \quad SU_{ph} \quad I_{ph} \quad X_{jpmch} \quad Y_{ph}]$$

شکل ۵: نحوه نمایش جواب

نحوه تولید جواب اولیه: جواب اولیه الگوریتم به طور تصادفی تولید شده است، به نحوی که در محدودیت‌های مدل پیشنهادی صدق کند؛ در صورت عدم بررسی امکان‌پذیری جواب در این مرحله، تعدادی جواب غیرقابل قبول تولید خواهد شد که زمان الگوریتم را در

الگوریتم PSO

معرفی الگوریتم PSO استاندارد

روش PSO یک تکنیک بهینه‌سازی تصادفی بر مبنای جمعیت است که ایده اولیه آن توسط ابره‌ه‌ارت و کندی [۲۰] معرفی شد. در این روش هر جواب که موقعیت یک پرنده در فضای جستجو است، ذره 15 نامیده می‌شود. هر ذره دارای یک مقدار شایستگی 16 و یک مؤلفه سرعت 17 ، که مسیر حرکت و مکان هر ذره را تعیین می‌کند، است.

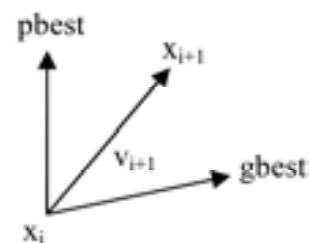
الگوریتم PSO استاندارد با یک گروه از جواب‌های تصادفی شروع و با به روز کردن سرعت و مکان هر ذره به جستجو در فضای مسئله می‌پردازد. برای به روز کردن سرعت در هر مرحله، از بهترین جواب برای هر ذره به دست آمده ($pbest^{18}$) و بهترین جواب سراسری که تا کنون توسط همه ذرات به دست آمده است ($gbest^{19}$) استفاده می‌شود. سرعت و مکان هر ذره در هر مرحله به ترتیب و با استفاده از روابط (۱۳) و (۱۴) به روز می‌شود:

$$v_i(t+1) = w \cdot v_i(t) + c_1 r_1 [pbest_i(t) - x_i(t)] + c_2 r_2 [gbest(t) - x_i(t)] \quad (13)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (14)$$

به طوری که t شمارنده هر مرحله و i شمارنده هر ذره و x_i موقعیت و v_i سرعت ذره i ام است. همچنین w وزن اینرسی و c_1 ، c_2 ، c_3 ضرایب شتاب و r_1 ، r_2 ، r_3 اعداد تصادفی هستند.

در این الگوریتم یک ذره برای حرکت از مکان قبلی (x_i) به مکان جدید (x_{i+1}) با استفاده از اطلاعات $pbest$ و $gbest$ را تعیین و طبق شکل ۳ حرکت می‌کند.



شکل ۳: نحوه حرکت یک ذره در PSO استاندارد

معرفی الگوریتم PSO پیشنهادی

موضوع مهم در الگوریتم PSO استاندارد نرخ همگرایی سریع آن بوده که ممکن است منجر به قرار گرفتن در بهینه محلی شود. واضح است که استفاده از منابع

نتایج محاسباتی

برای ارزیابی عملکرد الگوریتم پیشنهادی ۱۰ مسئله با ابعاد مختلف (جدول ۴) بر اساس ادبیات موضوع در نظر گرفته شده است، به طوری که مسئله ۱ تا ۵ از [۲] و مسئله ۶ تا ۱۰ از [۱۲] تعریف شده است. برای حل این مسائل، الگوریتم‌ها در نرم‌افزار MATLAB 7.8 کدنویسی شده است. رایانه مورد استفاده در این کار پنتیوم IV با پردازنده 2.6GHz و 2GB حافظه است.

جدول ۴: مشخصات مسائل نمونه

شماره مسئله	ابعاد مسئله	تعداد متغیرهای تصمیم
	$P(O_p) \times M \times C \times H \times G$	
1	$2(2) * 2 * 2 * 2 * 2$	80
2	$3(2) * 2 * 2 * 3 * 2$	162
3	$3(2) * 3 * 2 * 3 * 3$	216
4	$4(3) * 3 * 2 * 3 * 2$	477
5	$6(3) * 5 * 3 * 3 * 4$	1017
6	$8(3) * 6 * 3 * 3 * 5$	1554
7	$10(5) * 7 * 3 * 3 * 6$	3459
8	$12(3) * 10 * 3 * 3 * 7$	3654
9	$14(3) * 12 * 3 * 3 * 8$	5028
10	$16(4) * 14 * 3 * 3 * 10$	8634

تنظیم پارامتر

پارامترهای ورودی دو الگوریتم PSO و LPSO شامل $c_1, c_2, c_3, \omega, \alpha, \beta, \tau_1, \tau_2, \tau_3, l$ و k است. c_1, c_2, c_3 و m مؤثرین پارامترها در نتیجه نهایی الگوریتم‌ها هستند [۲۱]. نتایج [۲۱] نشان می‌دهد که مقدار مناسب برای c_1, c_2 و c_3 به ترتیب $[1, 1/5]$ ، $[0, 1/75]$ و $[0, 25/75]$ است. همچنین به دلیل وابستگی مقادیر m و n به یکدیگر، ۳ زوج عدد به صورت $(250, 70)$ ، $(400, 50)$ و $(150, 100)$ برای آن‌ها پیشنهاد شده است. مقادیر l و k نیز از بین مقادیر ۱۰٪، ۱۵٪، ۲۰٪ و ۲۵٪، مقدار n انتخاب می‌شود.

به طور قطع بررسی نتایج مدل پیشنهادی برای همه مقادیر ارایه شده در [۲۱] کاری بسیار زمان‌بر است. بنابراین در اینجا فقط حدود بالا و پایین c_1, c_2 و ۳ زوج پیشنهادی m و n در نظر گرفته می‌شود.

رسیدن به جواب مطلوب طولانی می‌کند. بردار سرعت اولیه نیز که یک ماتریس 1×7 بعدی است با مقادیر صفر پر می‌شود.

نحوه تولید جواب جدید: با توجه به پیوسته بودن الگوریتم LPSO و گسسته‌بودن مدل پیشنهادی، در هر مرحله پس از تولید، بردار مکان ذره از روی بردار سرعت که شامل اعداد پیوسته است، به نزدیک‌ترین اعداد صحیح گرد می‌شود و سپس امکان‌پذیری جواب‌های جدید با توجه به محدودیت‌ها بررسی می‌شود.

نحوه تعیین lbest: ابتدا ذرات را بر اساس کمترین مقدار تابع هدف مرتب و سپس بهترین جوابی که تا کنون توسط l (۵ درصد از ذرات) همسایه بی‌واسطه هر ذره به دست آمده است، انتخاب می‌شود.

مقداردهی دوباره به تعدادی از ذرات: تکنیک دیگری که برای جلوگیری از همگرایی سریع در الگوریتم استفاده شده است، مقداردهی دوباره k ذره از بدترین ذرات (۱۰٪ از ذرات با بیشترین تابع هدف) در هر مرحله است.

الگوریتم LPSO پیشنهادی از الگوی زیر پیروی می‌کند:

قدم ۱) برای $i=1$ تا ماکزیمم تعداد ذرات (n)

الف) مقداردهی اولیه ذرات با مکان‌های تصادفی

امکان‌پذیر (برقراری تمام محدودیت‌ها) و سرعت صفر

ب) قرار دادن مقدار اولیه ذرات در مقدار اولیه $pbest$

ج) محاسبه $lbest$ و $gbest$

د) محاسبه مقدار شایستگی ذرات

قدم ۲) انجام مراحل زیر تا زمان رسیدن به ماکزیمم تعداد تکرار (m):

الف) برای $i=1$ تا ماکزیمم تعداد ذرات (n)

الف-۱) محاسبه سرعت و مکان جدید ذرات به

ترتیب بر اساس فرمول (۱۴) و (۱۵)

الف-۲) بررسی امکان‌پذیری مکان جدید ذرات (در

همه محدودیت‌ها صدق کند)

الف-۳) محاسبه مقدار شایستگی ذرات

الف-۴) به روزرسانی مقادیر $pbest$ ، $lbest$ و $gbest$

الف-۵) انتخاب k ذره از بدترین ذرات و مقداردهی

اولیه دوباره به آن‌ها

ورودی الگوریتم‌ها برابر $c_1=1/5$ ، $c_2=1$ ، $c_3=0/75$ ، $n=400$ ، $m=50$ ، $l=60$ و $k=80$ است. مقادیر r_1 ، r_2 ، r_3 و w به شکل تصادفی از بازه $[0,1]$ انتخاب می‌شود.

تحلیل نتایج محاسباتی

تحقیقات بررسی‌شده در بخش مرور ادبیات نشان می‌دهد، اکثر مقالات مشابه فقط به توسعه مدل یکپارچه پرداخته و در معدودی از آن‌ها نیز که مدل حل شده است از نرم‌افزارهای بهینه‌سازی استفاده شده است. بنابراین امکان مقایسه نتایج الگوریتم LPSO پیشنهادی با نتایج سایر الگوریتم‌ها وجود ندارد.

در نتیجه در این تحقیق، از الگوریتم PSO استاندارد برای بررسی عملکرد الگوریتم پیشنهادی استفاده می‌شود. با توجه به اینکه نرم‌افزار فقط در ۵ مسئله نمونه اول، در زمان مشخص (حداکثر ۳ ساعت) به جواب رسیده است، فقط در این دسته از مسائل، امکان مقایسه جواب الگوریتم LPSO پیشنهادی و PSO استاندارد با جواب بهینه نرم‌افزار وجود دارد. مقایسه نتایج به دست آمده از نرم‌افزار LINGO 8.0 و الگوریتم PSO یا LPSO بر اساس معیار درصد خطا (%Error) در جدول (۵) آورده شده است. معیار %Error به کمک رابطه (۱۸) محاسبه می‌شود:

$$\% \text{ Error} = \frac{(\text{Best}_{\text{PSO or LPSO}} - \text{Opt}_{\text{LINGO}})}{\text{Opt}_{\text{LINGO}}} \times 100 \quad (18)$$

به طوری که Best_{PSO} یا $\text{Best}_{\text{LPSO}}$ معادل بهترین نتیجه به دست آمده در ۱۰ بار اجرای الگوریتم PSO یا LPSO، $\text{Opt}_{\text{LINGO}}$ معادل جواب بهینه به دست آمده از نرم‌افزار LINGO و Time معادل زمان محاسباتی بر حسب ثانیه است.

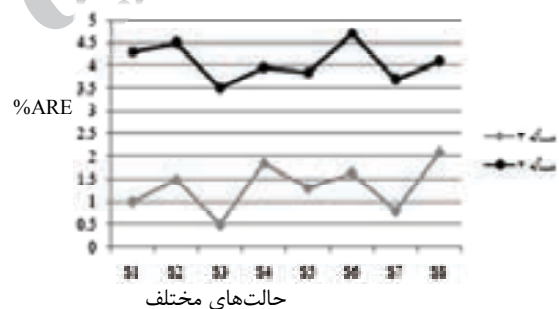
بر اساس نتایج جدول (۵)، متوسط درصد خطای الگوریتم PSO نسبت به جواب بهینه نرم‌افزار، برابر ۵/۹٪ و متوسط درصد خطای الگوریتم LPSO نسبت به جواب بهینه نرم‌افزار، برابر ۲/۲٪ است. بنابراین با توجه به کمتر بودن خطای محاسباتی الگوریتم LPSO نسبت به PSO و برابر بودن تقریبی متوسط زمان محاسباتی دو الگوریتم، استفاده از این روش برای حل مسائل کوچک برای مدل پیشنهادی مناسب‌تر است.

به دلیل کم بودن تأثیر پارامتر c_2 در کیفیت نتایج [۲۱] مقدار آن برابر ۱ فرض شده است. ۱۹۲ حالت مختلف برای آزمایش پارامترها تولید می‌شود. مسئله نمونه ۳ و ۴ برای همه حالات به تعداد ۱۰ بار اجرا شده است [۲۲]. از دو معیار متوسط نسبی خطا (ARE) (فرمول ۱۶) و انحراف معیار (S.D) (فرمول ۱۷) برای مقایسه نتایج تنظیم پارامتر استفاده شده است [۲۲].

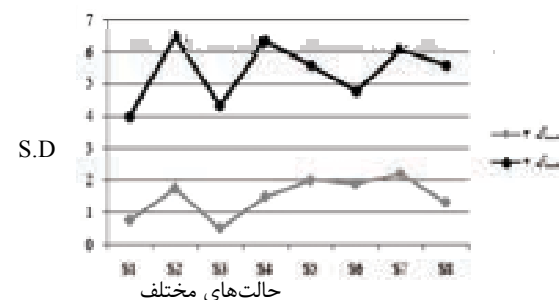
$$\% \text{ ARE} = \frac{1}{n} \sum_{i=1}^n \frac{\text{Mean}_i - \text{Best}_i}{\text{Best}_i} \times 100 \quad (16)$$

$$\text{S.D} = \frac{1}{n} \sum_{i=1}^n (\text{Mean}_i - \text{Best}_i)^2 \quad (17)$$

به طوری که Best_i برابر بهترین مقدار و Mean_i برابر متوسط مقدار تابع هدف برای ۱۰ بار اجرای مسئله i است. برای نمونه، نتایج تنظیم پارامتر برای ۸ حالت، نمودار (۱) و (۲) آورده شده است که به طور مثال در حالت S_1 ، $c_1=0/75$ ، $c_2=1$ ، $c_3=0/75$ ، $n=250$ ، $m=70$ ، $l=40$ و $k=60$ است.



نمودار ۱: نتایج تنظیم پارامتر مسئله ۳ و ۴ بر اساس ARE



نمودار ۲: نتایج تنظیم پارامتر مسئله ۳ و ۴ بر اساس S.D

با مقایسه نتایج تنظیم پارامتر، حالات مختلف بر اساس کمترین مقدار ARE و S.D، بهترین پارامترهای

جدول ۵: مقایسه نتایج حل مسائل با اندازه کوچک به وسیله LINGO، PSO و LPSO

LPSO, LINGO	PSO, LINGO	LPSO		PSO		LINGO		شماره مسئله
		Time	Best	Time	Best	Time	Opt.	
0.3	1	275	42690	230	42985	60	42547	1
0.4	3.1	310	7684	282	79884	554	77425	2
1.5	4.6	389	101165	355	104425	4143	99823	3
4.1	10	456	114398	494	121524	7260	109825	4
4.6	9.8	597	167137	642	175450	10800	159698	5
2.2	5.9	405.4	---	400.6	---	4563.4	---	متوسط

جدول ۶: مقایسه نتایج حل مسائل با اندازه‌های مختلف به وسیله PSO و LPSO

LPSO و PSO		LPSO			PSO			شماره مسئله
%PRAS	%PRBS	Time	Ave.	Best	Time	Ave.	Best	
1.7	1.8	275	42400	42190	230	43150	42985	1
1	2.7	310	79156	77684	282	79945	79884	2
7.4	6.8	389	99254	99165	355	107285	196425	3
4.5	3.3	456	119745	117398	494	125346	121524	4
5.7	6	597	169845	165437	642	180125	175450	5
10	16	712	209176	193194	753	233175	232042	6
12.9	14	890	261640	259013	879	300248	299514	7
10.1	11.4	986	315640	309785	1013	351145	349895	8
20.9	22	1375	367950	361895	1236	465315	461974	9
15	14.9	1640	419346	418226	1487	493184	491626	10
8.9	9.9	763	---	---	737.1	---	---	متوسط

$$\% \text{ PRAS} = \frac{(\text{Ave}_{\text{PSO}} - \text{Ave}_{\text{LPSO}})}{\text{Ave}_{\text{PSO}}} \times 100 \quad (20)$$

به طوری که Ave_{PSO} یا Ave_{LPSO} ، متوسط بهترین نتیجه به دست آمده در ۱۰ بار اجرای الگوریتم PSO یا LPSO است.

نتایج جدول (۶)، به طور متوسط ۹/۹٪ بهبود را در بهترین جواب‌های LPSO نسبت به PSO و ۸/۹٪ بهبود را در متوسط جواب‌های LPSO نسبت به PSO نشان می‌دهد. این بهبود برای هر دو معیار PRAS و PRBS در مسائل کوچک ناچیز بوده و با افزایش ابعاد مسئله افزایش یافته است. این مسئله نشان‌دهنده توانایی بالاتر الگوریتم LPSO نسبت به PSO در یافتن جواب‌های نزدیک به بهینه در مسائل با اندازه واقعی است. همچنین بررسی

با توجه به اینکه نرم‌افزار LINGO 8.0 قادر به حل مسائل ۶ تا ۱۰ در زمان قابل قبول نیست، امکان مقایسه جواب الگوریتم‌ها با جواب بهینه وجود ندارد. بنابراین برای بررسی عملکرد الگوریتم پیشنهادی در مسائل با ابعاد مختلف، ۱۰ مسئله نمونه فقط به کمک دو روش PSO و LPSO حل و از معیارهای درصد بهبود بهترین جواب^{۲۱} (PRBS)، درصد بهبود متوسط جواب^{۲۲} (PRAS) برای مقایسه کارایی دو الگوریتم استفاده می‌شود. مقایسه نتایج الگوریتم‌ها بر اساس معیارهای PRBS، PRAS و زمان در جدول (۶) آورده شده است. مقدار عددی PRBS و PRAS به ترتیب از رابطه (۱۹) و (۲۰) قابل محاسبه است [۲۳].

$$\% \text{ PRBS} = \frac{(\text{Best}_{\text{PSO}} - \text{Best}_{\text{LPSO}})}{\text{Best}_{\text{PSO}}} \times 100 \quad (19)$$

نمودار (۴) نشان می‌دهد که استفاده از اطلاعات $lbest$ و مقداردهی اولیه بدترین ذرات در هر مرحله از الگوریتم LPSO، فضای جستجوی مسئله و پراکندگی جواب‌ها را افزایش و باعث خروج از بهینه محلی شده است.

نتیجه‌گیری

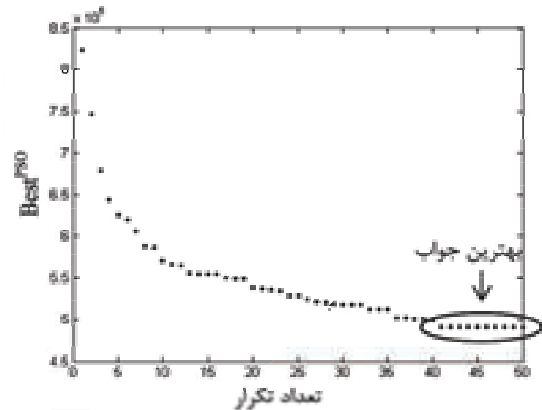
در این مقاله یک مدل عدد صحیح غیرخطی مختلط برای مسئله یکپارچه تشکیل سلولی پویا و برنامه‌ریزی تولید با هدف حداقل‌سازی هزینه‌های کلی (هزینه‌های ماشین، حمل و نقل بین‌سلولی و درون‌سلولی، استقرار مجدد، مصرف ابزار، نگهداری موجودی یا کمبود و برون‌سپاری قطعات) با شرط در دسترس بودن ابزار ارائه شده است. ارزیابی مدل پیشنهادی نشان داد که در نظر گرفتن هدف حداقل‌سازی هزینه‌های کلی و محدودیت برون‌سپاری قطعات، نوسانات شدید در برنامه تولیدی و تغییرات متعدد در ساختار سلول را کنترل می‌کند. همچنین در نظر گرفتن هزینه نصب یا برداشتن ماشین به هزینه مصرف ابزار به سایر هزینه‌ها، سبب نزدیکی به شرایط صنایع واقعی شده است. بنابراین این مدل قادر است ساختار نزدیک بهینه برای هر سلول و برنامه تولیدی کارآمدی را ارائه دهد.

مقایسه نتایج الگوریتم LPSO پیشنهادی در مسائل کوچک با نرم‌افزار LINGO 8.0 و در مسائل با ابعاد مختلف با الگوریتم PSO استاندارد، کارایی آن را نشان می‌دهد. افزایش فضای جستجو با استفاده از اطلاعات $lbest$ و افزایش پراکندگی جواب‌ها با استفاده از مقداردهی اولیه بدترین ذرات در هر مرحله، منجر به کاهش سرعت همگرایی الگوریتم و خروج از بهینه محلی شده است.

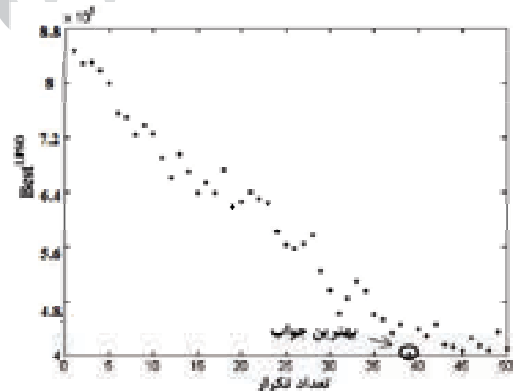
در تحقیقات بعدی، ارائه مدل ریاضی با متغیرهای احتمالی یا فازی مطلوب است. همچنین با توجه به ماهیت چندهدفه مسئله یکپارچه‌سازی برنامه‌ریزی تولید و تشکیل سلولی، شناسایی سایر اهداف، بررسی تأثیر آن‌ها بر یکدیگر و حل آن با روش‌های چندهدفه پیشنهاد می‌شود.

نتایج نشان می‌دهد که متوسط زمان اجرای هر دو الگوریتم به طور تقریبی برابر است.

برای بررسی روند همگرایی جواب‌ها در دو الگوریتم مورد نظر، از مسئله نمونه ۱۰ استفاده و نتایج در نمودار (۳) و (۴) نشان داده شده است. با توجه به اینکه روند تولید جواب‌های اولیه در هر دو الگوریتم به صورت تصادفی و امکان‌پذیر است، نقطه شروع هر دو به طور تقریبی نزدیک به هم است.



نمودار ۳: روند تولید جواب‌های الگوریتم PSO در مسئله ۱۰



نمودار ۴: روند تولید جواب‌های الگوریتم LPSO در مسئله ۱۰

نمودار (۳) به خوبی نشان می‌دهد که مقادیر جواب‌ها در الگوریتم PSO در ۱۰ تکرار اول به سرعت کاهش یافته و از تکرار ۱۰ تا ۴۰ با سرعت کمتری کاهش یافته و در ۱۰ تکرار آخر به طور کامل ثابت شده است. سرعت همگرایی بالای الگوریتم PSO در تکرارهای اولیه، ممکن است منجر به همگرایی زودرس و توقف در یک نقطه بهینه محلی شود.

مراجع

- 1- Soleymanpour, M., Vrat, P. and Shankar, R. (2002). "A transiently chaotic neural network approach to the design of cellular manufacturing." *International Journal of Production Research*, Vol.10, No. 40, PP. 2225–2244.
- 2- Safaei, N. and Tavakkoli-Moghaddam, R. (2009). "Integrated multi-period cell formation and subcontracting production planning in dynamic cellular manufacturing systems." *International Journal of Production Economics*, Vol. 120, No. 2, PP. 301-314.
- 3- Kioon, S.A., Bulgak, A.A. and Bektas, T. (2009). "Integrated cellular manufacturing systems design with production planning and dynamic system reconfiguration." *European Journal of Operations Research*, Vol. 192, PP. 414–428.
- 4- Fantahun, M., Defersha, F. and Chen, M. (2006). "A comprehensive mathematical model for the design of cellular manufacturing systems." *International Journal Production Economics*, Vol. 103, No. 1, PP. 767–783.
- 5- Rheault, M., Drolet, J.R. and Abdounour, G. (1996). "Dynamic Cellular Manufacturing Systems (DCMS)." *Computers Industrial Engineering*, Vol. 31, No. 1-2, PP. 143-146.
- 6- Chen, M. (1998). "A mathematical programming model for system reconfiguration in a dynamic cellular manufacturing environment." *Annals of Operations Research*, Vol. 77, PP. 109–128.
- 7- Wang, T.Y., Wu, K.B. and Liu, Y.W. (2001). "A simulated annealing algorithm for facility layout problems under variable demand in Cellular Manufacturing Systems." *Computers in industry*, Vol. 46, PP. 181-188.
- 8- Tavakkoli-Moghaddam, R., Safaei, N. and Babakhani, M. (2005). "Solving a dynamic cell formation problem with machine cost and alternative process plan by memetic algorithms." *SAGA*, Springer-Verlag Berlin Heidelberg, LNCS 3777, PP. 213–227.
- 9- Tavakkoli-Moghaddam, R., Aryanezhad, B., Safaei, N. and Azaron, A. (2005). "Solving a dynamic cell formation problem using metaheuristics." *Mathematics and Computation*, Vol. 170, No. 2, PP. 761–780.
- 10- Javadian, N. (2003). Design of cellular manufacturing systems in a dynamic and probabilistic condition using genetic algorithms and comparing them with the optimal solution. M.Sc. thesis in Industrial Engineering, Mazandaran University of Science and Technology, Tehran, Iran.
- 11- Saidi-Mehrabad, M. and Safaei, N. (2007). "A new model of dynamic cell formation by a neural approach." *International Journal Advance Manufacturing Technology*, Vol. 33, PP. 1001-1009.
- 12- Safaei, N., Saidi-Mehrabad, M. and Jabal-Ameli, M.S. (2008). "A hybrid simulated annealing for solving an extended model of dynamic cellular manufacturing system." *European Journal of Operational Research*, Vol. 185, PP. 563–592.
- 13- Petrov, V. (1968). Flow line group production planning. Business Publications, London, PP. 142–145.
- 14- Hyer, N.L. and Wemmerlov, U. (1982). "MRP/GT: a framework for production planning and control of cellular manufacturing." *Decision Sciences*, Vol. 13, PP. 681–701.
- 15- Schaller, J.E., Selcuk Erengu, S. and Vakharia, A. (1998). "A methodology for integrating cell formation and production planning in cellular manufacturing." *Annals of Operations Research*, Vol. 77, PP. 1–21.
- 16- Song, S. and Hitomi, K. (1996). "Integrating the production planning and cellular layout for flexible cellular manufacturing." *Production Planning and Control*, Vol. 7, No. 6, PP. 585–593.
- 17- Chen, M. (2001). "A model for integrated production planning in cellular manufacturing systems." *Integrated Manufacturing System*, Vol. 12, No. 4, PP. 275-284.

- 18- Chen, M. and Cao, D. (2004). "Coordinating production planning in cellular manufacturing environment using Tabu search." *Computers & Industrial Engineering*, Vol. 46, PP. 571–588.
- 19- Defersha, F. and Chen, M. (2006). "A comprehensive mathematical model for the design of cellular manufacturing systems." *International Journal of Production Economics*, Vol. 103, PP. 767–783.
- 20- Kennedy, J. and Eberhart, R.C. (1995). "Particle Swarm Optimization." In of IEEE International Conference on Neural Networks, Vol. 5, PP. 1942-1948.
- 21- Kallami-heris, S.M. (2008) "The effect of population and coefficients learning in PSO algorithm." Proc., 3rd International Conference on Information and Knowledge Technology, Ferdowsi University of Mashhad.
- 22- Kahfi, A. (2010). Solving an integrated model for production planning and Dynamic Cellular Manufacturing System by a multi-objective meta-heuristic method. M.Sc. thesis in Industrial Engineering, Payame Noor University, Tehran, Iran.
- 23- Barzinpour, F. (2004). Scheduling in cellular manufacturing systems with setup times. Ph.D. thesis in Industrial Engineering, Tarbiat Modarres University, Tehran, Iran.

واژه های انگلیسی به ترتیب استفاده در متن

- 1- Cellular Manufacturing System
- 2- Dynamic
- 3- Cell Formation
- 4- Production Planning
- 5- Backorder
- 6- Subcontracting
- 7- Inter Cell Movement Cost
- 8- Intra Cell Movement Cost
- 9- Reconfiguration Cost
- 10- Particle Swarm Optimization with Local Best
- 11- Memetic Algorithm
- 12- Genetic Algorithm
- 13- Tabu Search
- 14- Routing Flexibility
- 15- Particle
- 16- Fitness Function
- 17- Velocity
- 18- Personal Best
- 19- Global Best
- 20- Local Best
- 21- Percentage Reduction in the Best Solution
- 22- Percentage Reduction in the Average Solution