

ارائه‌ی الگوریتمی فراابتکاری بر پایه‌ی شبیه‌سازی تبرید برای مسئله‌ی زمان‌بندی گروهی در محیط جریان کارگاهی انعطاف‌پذیر با زمان‌های آماده‌سازی وابسته به توالی

سعید راستی (کارشناسی ارشد)

دانشکده‌ی مهندسی صنایع، دانشگاه علوم و فنون مازندران

ناصر سلماسی (دانشیار)

دانشکده‌ی مهندسی صنایع، دانشگاه صنعتی شریف

در این تحقیق مسئله‌ی زمان‌بندی گروهی در محیط جریان کارگاهی (فلوشاپ) انعطاف‌پذیر، با در نظر گرفتن زمان‌های آماده‌سازی وابسته به توالی گروه‌ها و نیز تابع هدف کمیته‌سازی زمان تکمیل مورد نیاز برای پردازش کارهای داخل گروه‌ها ($FFm|fmls, Sptc|Cmax$) مورد بررسی قرار گرفته است. برای این مسئله یک مدل برنامه‌ریزی خطی عدد صحیح مختلط برای نخستین بار ارائه شده است. دو رویکرد فراابتکاری بر پایه‌ی شبیه‌سازی تبرید برای حل تقریبی مسئله توسعه داده شده است. مقایسه‌ی عملکرد الگوریتم‌های پیشنهادی در این تحقیق با دیگر الگوریتم موجود در ادبیات - که بر مبنای جست‌وجوی ممنوع است - نشان می‌دهد که الگوریتم شبیه‌سازی تبرید پیشنهادی به‌طور متوسط جواب‌ها را حدود ۳ درصد بهبود می‌دهد.

واژگان کلیدی: زمان‌بندی گروهی وابسته به توالی، جریان کارگاهی انعطاف‌پذیر، روش‌های فراابتکاری، شبیه‌سازی تبرید، مدل‌سازی ریاضی.

rasti.saeid@gmail.com
nsalmasi@sharif.edu

۱. مقدمه

کارها در همه‌ی مرحله‌ها پردازش شوند. به عبارت دیگر، مرحله‌های جهشی وجود دارد که بعضی از کارها در آن مرحله‌ها پردازش نمی‌شوند. اما هنوز تمامی کارها باید مسیر پردازشی را از مرحله‌ی اول (شروع) تا مرحله‌ی آخر (پایان) دنبال کنند. از این مسئله با عنوان خط جریان انعطاف‌پذیر^۱ (جریان کارگاهی انعطاف‌پذیر هیبریدی^۳) یاد می‌شود.^[۲]

در زمان‌بندی گروهی، پس از تخصیص گروه‌های کاری به سلول‌ها، گروه‌ها و کارهای متعلق به آن‌ها در هر سلول باید زمان‌بندی شوند. در این تحقیق فرض بر آن است که چندین گروه شامل تعداد مختلفی کار در هر گروه به یک سلول جریان کارگاهی انعطاف‌پذیر برای پردازش تخصیص داده شده‌اند. تمامی کارهای متعلق به یک گروه به منظور تکمیل پردازش‌شان باید مسیر یکسانی را طی کنند که شامل چند مرحله است. ممکن است هر مرحله شامل تعداد مختلفی از ماشین‌های (منابع) یکسان باشد که به صورت موازی قرار گرفته‌اند. در مرحله‌ی یک که بیش از یک ماشین به صورت موازی وجود داشته باشد، تمامی کارهای متعلق به یک گروه به صورت متوالی به روی یکی از ماشین‌ها پردازش می‌شوند بدون اینکه کاری متعلق به سایر گروه‌ها میان این کارها توسط ماشین اختصاص یافته پردازش شود. هدف تعیین

یکی از سیستم‌های تولید که در صنایع مختلف کاربرد رو به گسترشی دارد، «جریان کارگاهی انعطاف‌پذیر»^۱ است. در هر مرحله از محیط جریان کارگاهی فقط یک پردازش‌گر وجود دارد و کارها بر پایه‌ی آرایشی خطی از مرحله‌ی اول تا مرحله‌ی آخر پردازش می‌شوند. در برخی موارد به منظور افزایش ظرفیت پردازش کل سیستم، در مرحله‌های گلوگاه از چندین ماشین به صورت موازی استفاده می‌شود. این نوع سیستم تولیدی علاوه بر ویژگی‌های محیط جریان کارگاهی، از ویژگی‌های سیستم‌های تولیدی شامل ماشین‌های موازی نیز برخوردار است، از این رو، در ادبیات موضوع از آن با عنوان جریان کارگاهی انعطاف‌پذیر (جریان کارگاهی هیبریدی) یاد می‌شود. در محیط جریان کارگاهی انعطاف‌پذیر، با افزایش تعداد مرحله‌های دارای ماشین‌های موازی و نیز با افزایش تعداد ماشین‌های موازی موجود در این مرحله‌ها، انعطاف‌پذیری افزایش می‌یابد.^[۱] ماشین‌های موازی ممکن است یکسان، یکنواخت یا نامرتب باشند. در هر سیستم جریان کارگاهی انعطاف‌پذیر، کارها تمامی مرحله‌ها را طی می‌کنند و در هر مرحله پردازش می‌شوند. در بعضی موارد لازم نیست که تمامی

تاریخ: دریافت ۱۳۸۸/۷/۱۲، اصلاحیه ۱۳۸۹/۷/۲۷، پذیرش ۱۳۸۹/۹/۱۶.

توالی کارهای متعلق به هر گروه، و نیز توالی گروه‌ها به‌منظور بهینه‌سازی یک یا چند معیار کاربردی است. بنابراین، مسئله‌ی زمان‌بندی گروهی در دو سطح انجام می‌شود: ۱. تعیین توالی کارهای متعلق به هر گروه؛ ۲. تعیین توالی گروه‌ها. [۳] در این تحقیق، از تابع هدف کمیته‌سازی زمان تکمیل تمامی کارها به‌عنوان یکی از تابع هدف‌های رایج در مسائل زمان‌بندی استفاده می‌شود. به‌طور خلاصه، در این تحقیق مسئله‌ی زمان‌بندی گروهی در محیط جریان کارگاهی انعطاف‌پذیر، همراه با زمان‌های آماده‌سازی وابسته به توالی و تابع هدف کمیته‌سازی زمان تکمیل تمامی کارها مورد مطالعه قرار می‌گیرد.

زمان‌بندی گروهی جریان کارگاهی انعطاف‌پذیر عمدتاً مربوط به صنایع پردازشی چندمرحله‌یی با چندین پردازش‌گر در هر مرحله است. مسئله‌ی مورد بررسی در این تحقیق، در صنایع تولید و مونتاژ خودرو، [۴] صنایع نساجی [۵] و مونتاژ صفحه‌های مدار چاپی [۶] کاربرد دارد. در چنین صنایعی، به‌منظور افزایش ظرفیت کل یا برای متعادل‌سازی ظرفیت مراحل، در چندین مرحله ماشین‌ها به حالت موازی قرار می‌گیرند. از مزیت‌های اجرای سیستم تولید جریان کارگاهی انعطاف‌پذیر می‌توان به افزایش ظرفیت کل، متعادل‌سازی ظرفیت مراحل، حذف یا کاهش تأثیر مرحله‌های گلوگاهی بر ظرفیت‌های کارگاه، کاهش زمان آماده‌سازی، کاهش نیازمندی‌های انرژی، کاهش موجودی در جریان ساخت، تسهیل جریان قطعات و مواد، کاهش زمان تکمیل محصول و متمرکزسازی مسئولیت و بهبود روابط انسانی اشاره کرد. [۸]

محققین ضمن بررسی تحقیق‌های انجام‌شده در زمینه‌ی زمان‌بندی گروه‌های کاری با توجه به زمان‌های آماده‌سازی در محیط تک‌ماشین و ماشین‌های موازی، [۹] مسئله‌ی زمان‌بندی را با و بدون در نظر گرفتن فرض تکنولوژی گروهی (مجاز نبودن جداسازی کارهای متعلق به یک گروه هنگام پردازش آن‌ها) و با معیارهای عملکردی مختلف مورد بررسی قرار دادند. مسئله‌ی زمان‌بندی گروهی در حالت تک‌ماشین با تمرکز روی تعیین توالی گروه‌ها (سطح دوم مورد بحث در این تحقیق) با تابع هدف کمیته‌سازی مجموع دیرکردها بررسی شد. [۱۱] همچنین الگوریتمی بر پایه‌ی برنامه‌ریزی پویا برای تعیین توالی بهینه‌ی کارها به‌منظور کمیته‌کردن میانگین زمان تکمیل در مسئله‌ی تک‌ماشین ارائه شد. [۱۲] در سال ۱۹۹۵، یافته‌های تحقیقاتی در زمینه‌ی زمان‌بندی گروهی، با در نظر گرفتن تابع هدف کمیته‌سازی مجموع زمان‌های شناوری وزن‌دهی‌شده و بیشینه تأخیر مورد بررسی قرار گرفت [۱۳] و منجر به مطالعه‌ی قسمت‌هایی از زمان‌بندی گروهی شد که به در دسترس بودن دسته‌ها^۴ و پردازش آن‌ها مربوط می‌شد.

در سال ۱۹۸۵ برای نخستین بار یک الگوریتم چندجمله‌یی برای کمیته‌سازی زمان تکمیل مسئله‌ی زمان‌بندی گروهی دو ماشین با زمان‌های آماده‌سازی مستقل از توالی پیشنهاد شد. [۱۴] در همین راستا، محققین [۱۵] نسبت به ترکیب روش‌های ابتکاری تک‌مرحله‌یی پتروف (PT) [۱۶] با روش ابتکاری چندمرحله‌یی کمبل، دودک و اسمیت (CDS) [۱۷] برای حل مسئله‌ی زمان‌بندی گروهی چندمرحله‌یی m ماشین به هدف کمیته‌کردن زمان تکمیل کل کارها اقدام کردند. با توجه به عملکرد خوب روش ابتکاری پیشنهادی توسط لوگندران و نودتاسومیون (LN) [۱۸] در حل مسئله‌ی سطح ۱، آنان به‌منظور مقایسه‌ی عملکرد روش‌های ابتکاری LN و CDS به بررسی این روش‌ها پرداختند. [۱۹] بدین‌منظور آنها روش‌های LN و CDS را به‌طور جداگانه با روش ابتکاری PT ترکیب، و از هر یک برای حل کامل مسئله‌ی زمان‌بندی گروهی m ماشین با تابع هدف کمیته‌سازی زمان تکمیل کل استفاده کردند.

محققین ثابت کردند که مسئله‌ی زمان‌بندی کارها در محیط جریان کارگاهی دوماشینه با زمان‌های آماده‌سازی وابسته به توالی و تابع هدف کمیته‌سازی زمان

تکمیل، یک مسئله‌ی Np-hard است [۲۰] و الگوریتمی تقریبی برای حل آن ارائه کردند.

در نخستین تحقیق در زمینه‌ی زمان‌بندی جریان کارگاهی انعطاف‌پذیر، [۲۱] مسئله‌ی جریان کارگاهی انعطاف‌پذیر، در حالی که موجودی در جریان ساخت^۸ آن صفر بود، و تابع هدف کمیته‌سازی زمان تکمیل براساس رویکردی مبتنی بر روش شاخه و حد حل شد. برای حل دقیق مسئله زمان‌بندی خط جریان انعطاف‌پذیر با تابع هدف کمیته‌سازی زمان تکمیل و موجودی میانی محدود، یک مدل برنامه‌ریزی عدد صحیح مختلط ارائه شد. [۲۲] این مدل بعدها توسط طراح و به‌منظور زمان‌بندی دسته‌یی توسعه یافت. [۲۳] به‌منظور مقایسه‌ی قواعد تعیین توالی کار روی ماشین‌ها در مسئله‌ی زمان‌بندی جریان کارگاهی انعطاف‌پذیر با هدف کمیته‌سازی زمان تکمیل با زمان‌های آماده‌سازی وابسته به توالی یک بررسی آزمایشی انجام شد. [۲۴] در سال ۲۰۰۵ محققین از یک رویکرد ابتکاری بر پایه‌ی شبیه‌سازی تیرید برای حل مسئله زمان‌بندی جریان کارگاهی چندمرحله‌یی با ماشین‌های موازی نامرتب، زمان‌های آماده‌سازی مستقل، زمان‌های برداشت وابسته و تابع هدف کمیته‌سازی مجموع زمان‌های شناوری استفاده کردند. [۲۶] پس از آن، در سال ۲۰۰۷، مسئله‌ی $z = \sum_{i=1}^n u_i (1 + \lambda) + FF_m |S_{plc} / \lambda C_{max}$ را با ماشین‌های موازی نامرتب و $0 \leq \lambda \leq 1$ مورد مطالعه قرار دادند. [۲۷] آن‌ها از روش‌های ابتکاری سازنده^۹ برای تولید یک جواب اولیه‌ی خوب و از روش‌های فراابتکاری الگوریتم ژنتیک، شبیه‌سازی تیرید و جست‌وجو ممنوع برای بهبود جواب استفاده کردند. نتایج محاسباتی نشان می‌داد که عملکرد «رویکرد درج کار^{۱۰}» در میان الگوریتم‌های سازنده، و عملکرد «شبیه‌سازی تیرید» در میان الگوریتم‌های فراابتکاری نسبت به سایر روش‌ها بهتر است. پژوهش‌گران برای این مسئله یک مدل برنامه‌ریزی خطی عدد صحیح مختلط نیز ارائه کردند. همچنین برای مسئله‌ی جریان کارگاهی هیبریدی با زمان‌های آماده‌سازی وابسته به توالی یک الگوریتم مصون^{۱۱} ارائه کردند. [۲۸] آن‌ها نشان دادند که عملکرد الگوریتم ارائه‌شده نسبت به الگوریتم ژنتیک کوروز و آسکین [۲۵] بهتر است. همچنین در سال ۲۰۰۸، در مسئله‌ی زمان‌بندی خط جریان انعطاف‌پذیر با زمان‌های آماده‌سازی وابسته به توالی، سیاست‌های نگه‌داری پیشگیرانه‌ی مختلف و تابع هدف کمیته‌سازی زمان تکمیل لحاظ شد. [۲۹] بدین‌منظور از یک رویه‌ی جست‌وجوی همسایگی متغیر و چندین روش فراابتکاری و ابتکاری برای حل مسئله استفاده شد. در سال ۲۰۱۰ محققین داخلی به مطالعه‌ی مسئله‌ی زمان‌بندی جریان کارگاهی انعطاف‌پذیر پویا با زمان‌های آماده‌سازی وابسته پرداختند. [۳۰] آن‌ها از یک مدل شبیه‌سازی گسسته و همچنین هشت الگوریتم ابتکاری برای بررسی این مسئله بهره بردند.

در ارزیابی مسئله‌ی زمان‌بندی گروهی جریان کارگاهی انعطاف‌پذیر با زمان‌های آماده‌سازی مستقل از توالی و تابع هدف کمیته‌سازی زمان تکمیل تمامی کارها [۱] که در سال ۲۰۰۵ صورت گرفت، عملکرد سه الگوریتم حل ابتکاری مختلف، بر پایه‌ی ترکیب روش ابتکاری تک‌مرحله‌یی PT و روش ابتکاری چندمرحله‌یی LN، مورد مقایسه قرار گرفت. پس از آن در سال ۲۰۰۶، محققین سه الگوریتم بر پایه‌ی جست‌وجوی ممنوع برای مسئله‌ی زمان‌بندی گروهی $FF_m |fmls, S_{plc} | C_{max}$ ارائه کردند. [۳] آنان سه سازکار مختلف برای تعیین جواب اولیه‌ی بهتر در نظر گرفتند. در این تحقیق از طراحی آزمایشات برای تحلیل زمان تکمیل و زمان محاسباتی به‌عنوان دو متغیر پاسخ جدا استفاده شده است. در همین سال محققین داخلی نیز مسئله‌ی زمان‌بندی گروهی جریان کارگاهی هیبریدی با زمان‌های آماده‌سازی وابسته به توالی و تابع هدف کمیته‌سازی زمان تکمیل را مورد بررسی قرار دادند. [۳۱] آن‌ها برای حل این مسئله دو روش فراابتکاری بر پایه‌ی شبیه‌سازی تیرید و الگوریتم ژنتیک را توسعه دادند.

۳. مدل ریاضی

در سال ۲۰۱۰، محققین یک مدل برنامه‌ریزی خطی عدد صحیح مختلط برای حل مسئله‌ی جریان کارگاهی با زمان‌های آماده‌سازی وابسته به توالی گروه‌ها و تابع هدف کمینه‌سازی مجموع زمان‌های شناوری ارائه کردند.^[۲۲] در این تحقیق، مدل ریاضی پیشنهاد شده برای مسئله‌ی زمان‌بندی گروهی در محیط جریان کارگاهی انعطاف‌پذیر با زمان‌های آماده‌سازی وابسته به توالی گروه‌ها بسط داده شده است. در این مدل ضمن معرفی مفهوم شیار (slot)، در هر مرحله و روی هر ماشین به تعداد گروه‌های موجود شیار در نظر می‌گیریم. هر گروه در یک شیار قرار می‌گیرد و هر شیار حداکثر یک گروه را در خود جای می‌دهد. در مسائل دنیای واقعی، گروه‌ها تعداد کارهای مختلفی دارند. از آنجا که هر گروه را می‌توان به هر شیاری تخصیص داد، به منظور تسهیل توسعه‌ی مدل ریاضی، فرض می‌کنیم تعداد کارهای تمامی گروه‌ها (b_{max}) مساوی، و برابر با تعداد کارهای گروهی است که بیشترین تعداد کار را دارد. اگر یک گروه تعداد کارهای کم‌تری نسبت به b_{max} داشته باشد، آنگاه فرض می‌کنیم که آن گروه به‌میزان این اختلاف (تعداد کارهای واقعی - b_{max}) کارهای مجازی دارد. پارامترها، متغیرهای تصمیم و مدل ریاضی این مسئله عبارت است از:

۱.۳. پارامترها

m	تعداد مراحل
c	اندیس مرحله، $c = 1, 2, \dots, m$
m_c	تعداد ماشین‌های موازی در مرحله‌ی c
k, k'	اندیس‌های مورد استفاده برای ماشین‌ها، $k = 1, 2, \dots, m_c$
g	تعداد گروه‌ها
p, l	اندیس گروه، $p, l = 1, 2, \dots, g$
b_{max}	بیشترین تعداد کارهای موجود در یک گروه
j, j'	اندیس کار، $j, j' = 1, 2, \dots, b_{max}$
i, i'	اندیس شیار، $i, i' = 1, 2, \dots, g$
M	یک عدد بزرگ
t_{cpj}	- برای کارهای حقیقی: زمان پردازش کار j از گروه p در مرحله‌ی c
	- برای کارهای مجازی: $\forall c, p, j, 0$
t'_{cpj}	- برای کارهای حقیقی: زمان پردازش کار j از گروه p در مرحله‌ی c
	- برای کارهای مجازی: $-M$
S_{pic}	زمان آماده‌سازی برای گروه l در مرحله‌ی c اگر گروه p گروه قبلی باشد، $\forall c, p, l; p \neq l$
T_{cp}	جمع زمان‌های پردازش کارهای گروه p در مرحله‌ی c
	$T_{cp} = \sum_{j=1}^{b_p} t_{cpj}$

۲.۳. متغیرهای تصمیم

X_{ekij} زمان تکمیل کار j در i امین شیار روی ماشین k در مرحله‌ی c ، $\forall c, k, i, j$

با توجه به کاربرد گسترده‌ی مسئله‌ی زمان‌بندی گروهی جریان کارگاهی انعطاف‌پذیر در صنعت، بررسی بیشتر آن ضروری به نظر می‌رسد. با توجه به تحقیقات کم انجام شده در این زمینه، ناحیه‌ی وسیعی برای تحقیقات بیشتر در این خصوص وجود دارد. این تحقیق پیرامون توسعه و بهبود عملکرد یک رویکرد فراابتکاری بر پایه‌ی شبیه‌سازی تبرید برای کمینه‌سازی زمان تکمیل تمامی کارها در زمان‌بندی گروهی در محیط جریان کارگاهی انعطاف‌پذیر با زمان‌های آماده‌سازی وابسته به توالی متمرکز شده است. در این تحقیق سیاست مورد استفاده برای تعیین اولویت گروه‌ها و کارها در مرحله‌های دوم الی آخر بر پایه‌ی ترتیب حضور گروه‌ها و کارها در هر مرحله، یا به عبارتی (FIFO)^[۲۳] است.

برای توسعه‌ی یک الگوریتم کارا که به جواب قابل قبولی برای این مسئله‌ی پیچیده بینجامد، باید به این سؤال پاسخ داد که چه ترکیبی از عوامل مهم الگوریتم منجر به عملکرد بهتر آن می‌شود؟

۲. ویژگی‌های مسئله

بر اساس نمادهای ارائه شده توسط محققین،^[۲۲] مسئله‌ی مورد نظر را می‌توان با نماد $FF_m | fmls, S_{pic} | C_{max}$ نمایش داد، که در آن:

- FF_m نشان‌گر یک محیط جریان کارگاهی انعطاف‌پذیر، شامل m مرحله‌ی پردازش است که در آن دست‌کم یک مرحله دارای بیش از یک ماشین به صورت موازی است. ماشین‌های موازی در هر مرحله یکسان‌اند.

- $fmls$ نشان‌گر آن است که کارها به گروه‌های مختلف تخصیص داده می‌شوند، هر گروه p ($p = 1, 2, \dots, g$) شامل n_p کار است و تعداد کارها در گروه‌ها ممکن است متفاوت باشد.

- S_{pic} نشان‌گر آن است که زمان آماده‌سازی ماشین‌ها برای پردازش گروه‌ها وابسته به توالی گروه‌هاست. زمان آماده‌سازی یک گروه (l گروه) روی هر ماشین در هر مرحله (مرحله‌ی c) وابسته به گروه بلافاصله پردازش شده‌ی قبلی (گروه p) روی همان ماشین در آن مرحله (مرحله‌ی c) است. زمان آماده‌سازی هر گروه در هر مرحله ممکن است متفاوت باشد، اما این زمان برای تمامی ماشین‌های متعلق به یک مرحله یکسان است.

- C_{max} نشان می‌دهد که تابع هدف مسئله‌ی زمان‌بندی مورد نظر کمینه‌سازی زمان تکمیل تمامی کارهاست.

محدودیت‌ها و فرض‌های مختلف ممکن است به مسئله‌های زمان‌بندی گروهی مختلف در محیط جریان کارگاهی انعطاف‌پذیر منجر شود. برای تعریف بهتر و روشن‌تر مسئله‌ی مورد بررسی در این تحقیق، فرض‌های در نظر گرفته شده عبارت‌اند از:

- زمان‌های آماده‌سازی گروه‌ها مستقل از حضور کار در هر مرحله‌اند (فرایند آماده‌سازی یک گروه روی یک ماشین ممکن است قبل از رسیدن کارهای آن گروه به روی ماشین انجام شود).

- کارها روی ماشین‌ها بدون انقطاع پردازش می‌شوند (وقفه در طول پردازش یک کار مجاز نیست).

- ظرفیت ذخیره‌ی کارها در میان مرحله‌ها نامحدود است.

- زمان جابه‌جایی کارها در بین مرحله‌ها صفر است.

- تمامی کارها در آغاز افق زمان‌بندی در دسترس‌اند.

- ماشین‌ها همواره در دسترس‌اند و خرابی برای ماشین‌ها وجود ندارد.

$$X_{ckij'} - X_{ckij} + M(1 - Y_{ckijj'}) \geq \sum_{p=1}^g W_{ckip} t_{epj'} \quad (14)$$

$$X_{ckij} - X_{(c-1)k'i'j} + M[2 - (W_{ckip} + W_{(c-1)k'i'p})] \geq t_{epj} \quad (15)$$

$$\forall c, k, k', i, i', j, p$$

$$CT_{cki} = \max\{X_{ckij}\} \quad \forall c, k, i \quad (16)$$

$$CT_{\max} \geq CT_{mki} \quad \forall i, k = 1, 2, \dots, m_m \quad (17)$$

$$X_{ckij}, C_{cki}, O_{cki} \geq 0 \quad (18)$$

$$W_{ckip}, A_{ckipl}, Y_{ckijj'} \in \{0, 1\} \quad j < j' \quad (19)$$

تابع هدف مدل بر پایه کمیته‌سازی زمان تکمیل تمامی کارها در آخرین مرحله در معادله ۱ ارائه شده است. محدودیت ۲ تضمین می‌کند که هر گروه (به جز گروه صفر) در هر مرحله به یک شیار موجود از میان شیارهای ماشین‌های آن مرحله (به جز شیار صفر) تخصیص داده شود. محدودیت ۳ تخصیص حداکثر یک گروه به هر شیار را در هر مرحله و روی هر ماشین تضمین می‌کند. محدودیت ۴ ایجاب می‌کند که در هر مرحله و روی هر ماشین گروه صفر در شیار صفر قرار گیرد. زمان آماده‌سازی یک گروه روی یک ماشین، وابسته به گروه پردازش‌شده قبلی روی همان ماشین است؛ این مطلب توسط محدودیت ۵ به مدل اعمال شده است.

محدودیت‌های ۶ و ۷ نشان‌گر رابطه‌ی بین متغیرهای W و A هستند؛ طبق محدودیت ۶ اگر گروه p به شیار i تخصیص داده شود آنگاه ممکن است گروه l به شیار $i + 1$ تخصیص داده شود، و طبق محدودیت ۷ اگر گروه l به شیار $i + 1$ تخصیص داده شود، آنگاه ممکن است گروه p به شیار i تخصیص داده شود. از آنجا که تابع هدف کمیته‌سازی زمان تکمیل کل است، بین شیارهای پرتخصیص داده‌شده شیاری خالی برای پردازش گروه‌ها روی ماشین‌ها وجود نخواهد داشت. به عبارت دیگر شیارهای اشغال‌شده روی هر ماشین یکی بعد از دیگری قرار خواهند گرفت.

محدودیت ۸ تضمین می‌کند که شیار خالی بین شیارهای پرتقرار نگیرد. محدودیت ۹ زمان آماده‌سازی مورد نیاز برای یک شیار روی یک ماشین را تعیین می‌کند. زمان آماده‌سازی یک شیار روی یک ماشین براساس گروه تخصیص داده‌شده به آن شیار و گروه تخصیص یافته به شیار قبلی تعیین می‌شود. زمان تکمیل شیار صفر روی هر ماشین برابر با صفر است؛ این مطلب در محدودیت ۱۰ نشان داده شده است. زمان تکمیل گروه‌های تخصیص‌یافته به شیارهای روی ماشین‌های مرحله اول توسط محدودیت ۱۱ تعیین می‌شود. این محدودیت نشان می‌دهد که زمان تکمیل یک گروه تخصیص‌یافته به یک شیار برابر با جمع زمان تکمیل گروه تخصیص داده‌شده به شیار قبلی، زمان آماده‌سازی مورد نیاز برای گروه موجود در این شیار، و جمع زمان‌های پردازش کارهای متعلق به این گروه است. محدودیت ۱۲ برای یافتن زمان تکمیل کارهای متعلق به گروه‌ها به مدل اضافه شده است. زمان تکمیل یک کار بزرگ‌تر مساوی مجموع زمان تکمیل گروه پردازش‌شده در شیار قبلی، زمان راه‌اندازی برای این گروه، و زمان پردازش کار مورد نظر است.

محدودیت‌های ۱۳ و ۱۴ برای تعیین توالی پردازش کارهای متعلق به یک گروه در نظر گرفته شده‌اند. اگر در یک مرحله، کار z از یک گروه بعد از کار z' از همان گروه پردازش شود، اختلاف بین زمان‌های تکمیل آن‌ها در آن مرحله باید بزرگ‌تر از زمان پردازش کار z یا مساوی آن باشد. با اعمال محدودیت ۱۵، پردازش یک کار در یک مرحله وقتی شروع می‌شود که پردازش آن در مرحله قبلی تمام شده باشد. به عبارت دیگر زمان تکمیل یک کار در یک مرحله باید بزرگ‌تر یا مساوی مجموع

$$W_{ckip} \quad \forall c, k, i = 0, \dots, g, p = 0, \dots, g$$

۱؛ اگر گروه p به شیار i از ماشین k در مرحله c تخصیص داده شود،
در غیر این صورت،
۰؛ اگر در i امین شیار از ماشین k در مرحله c کار j' بعد از کار j پردازش شود،
در غیر این صورت،
۰؛ زمان تکمیل گروه اختصاص یافته به i امین شیار روی ماشین k در مرحله c ،
زمان آماده‌سازی برای گروه تخصیص داده شده به شیار i روی ماشین k در مرحله c ،
۱؛ اگر روی ماشین k در مرحله c گروه p به شیار i و گروه l به شیار $i + 1$ تخصیص داده شود،
در غیر این صورت،
۰؛

۳.۳. مدل

$$\text{Minimize } Z = C_{\max} \quad (1)$$

Subject to :

$$\sum_{k=1}^{m_c} \sum_{i=1}^g W_{ckip} = 1 \quad \forall c, p \quad (2)$$

$$\sum_{p=1}^g W_{ckip} \leq 1 \quad \forall c, k, i \quad (3)$$

$$W_{ck0} = 1 \quad \forall c, k \quad (4)$$

$$\sum_{p=0}^g \sum_{l=1}^g A_{ckipl} \leq 1 \quad (p \neq l) \quad i = 0, \dots, g-1 \quad (5)$$

$$A_{ckipl} \leq W_{ckip} \quad \forall c, k, p, l, p \neq l \quad (6)$$

$$A_{ckipl} \leq W_{ck(i+1)l} \quad i = 0, \dots, g-1 \quad (7)$$

$$\sum_{k=1}^{m_c} \sum_{i=0}^g \sum_{p=0}^g \sum_{l=1}^g A_{ckipl} = g \quad (p \neq l) \quad \forall c \quad (8)$$

$$O_{cki} = \sum_{p=0}^g \sum_{l=1}^g A_{ck(i-1)pl} S_{plc} \quad (p \neq l) \quad \forall c, k, i \quad (9)$$

$$CT_{rk0} = 0 \quad \forall c, k \quad (10)$$

$$CT_{\backslash ki} = CT_{\backslash k(i-1)} + O_{\backslash ki} + \sum_{p=1}^g W_{\backslash kip} T_{\backslash p} \quad \forall i, k = 1, \dots, m_1 \quad (11)$$

$$X_{ckij} \geq CT_{ck(i-1)} + O_{cki} + \sum_{p=1}^g W_{ckip} t_{epj} \quad \forall c, k, i, j \quad (12)$$

$$X_{ckij} - X_{ckij'} + M Y_{ckijj'} \geq \sum_{p=1}^g W_{ckip} t_{epj} \quad \forall c, k, i, j, j'; j < j' \quad (13)$$

مراحل اولیه‌ی شبیه‌سازی تیرید حرکت‌های بدترکننده‌ی جواب پذیرفته می‌شوند، اما در مراحل پایانی احتمال پذیرش جواب‌های بهبوددهنده بیشتر است. در هر مرحله، SA جواب‌های جدیدی در همسایگی جواب جاری تولید می‌کند و به سمت یکی از آن‌ها حرکت می‌کند تا شرایط توقف حاصل شود. عملکرد SA توسط فاکتورهای نظیر دمای اولیه (T_0)، دمای پایانی (T_f) و تعداد تکرارها در هر دما (L_{max}) تحت تأثیر قرار می‌گیرد.

۱.۴. الگوریتم شبیه‌سازی تیرید پیشنهادی

به منظور بهبود کیفیت جواب و کارایی روش جست‌وجو، سه سازوکار مختلف به الگوریتم شبیه‌سازی تیرید استاندارد اضافه می‌شود. برای افزایش کیفیت جواب، از چندین سازوکار ویژه برای تولید جواب همسایه و از یک فهرست کاندید برای انتخاب بهترین همسایه بهره برده می‌شود. رویه‌ی جست‌وجو در هر دما در طول فرایند تیرید به تعداد از پیش تعیین شده‌ی تکرار می‌شود. برای فرار از تله‌ی بهینه‌ی محلی از «جهش شدید» استفاده می‌شود. «جهش شدید» هنگامی مورد استفاده قرار می‌گیرد که در طول فرایند جست‌وجو مقدار تابع هدف بهترین جواب بدون تغییر باقی بماند. در این حالت در هر یک از مراحل دو گروه تصادفی انتخاب شده و جای آن‌ها عوض می‌شود.

۲.۴. جواب اولیه (IS) ۱۳

چنان که پیش‌تر گفته شد، الگوریتم شبیه‌سازی تیرید برای شروع حرکت احتیاج به یک جواب اولیه دارد. در این تحقیق از دو رویکرد برای تعیین این جواب استفاده شده است.

۱.۲.۴. سازوکارهای تولید جواب اولیه در رویکرد اول

در این رویکرد یک توالی از گروه‌ها و کارها در مرحله‌ی اول نشان‌دهنده‌ی یک جواب اولیه است. برای تعیین یک توالی از گروه‌ها و کارها در مرحله‌ی اول از روش‌های زیر استفاده می‌کنیم:

جواب اولیه‌ی ۱ (IS1). این روش ساده‌ترین روش تولید جواب اولیه است. در این روش گروه‌ها و کارها به صورت تصادفی مرتب می‌شوند. به عنوان مثال، مسئله‌ی را در نظر بگیرید که در آن سه گروه و در هر گروه سه کار وجود دارد. جواب اولیه با استفاده از این سازوکار ممکن است چنین باشد:

$$G_1 : J_{11} - J_{12} - J_{13}, \quad G_2 : J_{21} - J_{22} - J_{23},$$

$$G_3 : J_{31} - J_{32} - J_{33}$$

جواب اولیه‌ی ۲ (IS2). در این روش به منظور حفظ سادگی، توالی گروه‌ها همچنان به صورت تصادفی تعیین می‌شود.^[۱۳] اما برای تعیین توالی کارهای داخل هر گروه، از نتایج به دست آمده از تحقیق‌های انجام گرفته در زمینه‌ی مسئله‌ی زمان‌بندی ماشین‌های موازی یکسان با تابع هدف کمینه‌سازی زمان تکمیل استفاده می‌کنیم. از آنجا که جریان کارگاهی انعطاف پذیر شامل چندین مرحله است، ابتدا زمان اجرای جمعی^{۱۴} را برای هر کار با جمع کردن زمان‌های اجرای آن در تمام مرحله‌ها محاسبه می‌کنیم. در ادامه، توالی کارهای هر گروه براساس بزرگ‌ترین زمان اجرای جمعی تعیین می‌شود؛ یعنی کاری که دارای بزرگ‌ترین زمان اجرای جمعی است در ابتدای توالی قرار می‌گیرد. اگر زمان اجرای جمعی دو کار مساوی باشد کاری که شماره‌اش کوچک‌تر است ابتدا قرار می‌گیرد.

جواب اولیه‌ی ۳ (IS3). در این سازوکار از قاعده‌ی بزرگ‌ترین زمان پردازش

زمان تکمیل آن کار در مرحله‌ی قبل و زمان پردازش آن در مرحله‌ی مورد نظر باشد. محدودیت ۱۶ تضمین می‌کند که زمان تکمیل یک گروه در یک مرحله برابر با زمان تکمیل آخرین کار آن گروه است که در آن مرحله پردازش می‌شود. محدودیت ۱۷ نشان می‌دهد که زمان تکمیل تمامی کارها بزرگ‌تر یا مساوی زمان تکمیل گروه‌هایی است که در مرحله‌ی آخر پردازش می‌شوند. محدودیت‌های ۱۸ و ۱۹ نشان‌دهنده‌ی نوع متغیرهای تصمیم‌گیری است.

اگر فرض کنیم در هر مرحله یک ماشین وجود داشته باشد، آنگاه مسئله‌ی مورد بررسی در این تحقیق به مسئله‌ی زمان‌بندی گروهی در محیط جریان کارگاهی با زمان‌های آماده‌سازی وابسته به توالی و تابع هدف کمینه‌سازی زمان تکمیل کل کاهش می‌یابد. محققین نشان داده‌اند^[۲۴] که مسئله‌ی جریان کارگاهی با زمان‌های آماده‌سازی وابسته به توالی گروه‌ها و تابع هدف کمینه‌سازی آخرین کار قویاً Np-hard است. از آنجا که مسئله‌ی مورد بررسی در این تحقیق با ساده‌سازی قابل تبدیل به مسئله‌ی تحقیقی شیار و همکاران^[۲۴] است، می‌توان آن را نیز به عنوان یک مسئله‌ی Np-hard در نظر گرفت.

۴. شبیه‌سازی تیرید

با توجه به Np-hard بودن مسئله‌ی مورد بررسی، برای حل کارآمد مسئله‌های بزرگ در مقیاس صنعتی، یک الگوریتم فراابتکاری بر پایه‌ی شبیه‌سازی تیرید پیشنهاد شده است. الگوریتم مورد نظر شامل دو مرحله جست‌وجو است: جست‌وجوی خارجی و جست‌وجوی داخلی. جست‌وجوی خارجی شامل جست‌وجوی بهترین توالی گروه‌ها و تخصیص گروه‌ها به ماشین‌های موجود در هر مرحله است. با در دست بودن یک توالی از گروه‌ها، در فرایند جست‌وجوی داخلی بهترین توالی کارها در هر گروه جست‌وجو می‌شود. جواب نهایی از ترکیب بهترین توالی گروه‌ها با بهترین توالی کارهای هر گروه که منجر به کم‌ترین زمان تکمیل برای مسئله‌ی مورد نظر می‌شود، به دست می‌آید. ساختار کامل این الگوریتم در قسمت‌های بعد توضیح داده شده است.

الگوریتم شبیه‌سازی تیرید (SA)، ایده و سازوکار موجود در تیرید فیزیکی یک جامد را شبیه‌سازی می‌کند. این ایده اولین بار در سال ۱۹۵۳ مطرح شد^[۲۵] و براساس آن، ماده به صورت سیستمی از ذرات در نظر گرفته می‌شود و فرایند خنک شدن با کاهش تدریجی دمای سیستم تا رسیدن به یک حالت پایدار، حالت انجام، شبیه‌سازی می‌شود. این الگوریتم بعدها توسعه داده شد^[۲۶] و برای حل مسئله‌های ترکیبی و دیگر مسئله‌های بهینه‌سازی به کار برده شد. مزیت عمده‌ی الگوریتم SA نسبت به دیگر روش‌های فراابتکاری اجتناب از گرفتار شدن الگوریتم جست‌وجو در تله‌ی بهینه‌ی محلی است.

الگوریتم شبیه‌سازی تیرید استاندارد با تولید تصادفی جواب اولیه شروع می‌شود، و پس از ارزیابی آن به وسیله‌ی تابع هزینه یک همسایه‌ی جدید تولید خواهد شد. چنانچه جواب همسایه‌ی جدید مقدار تابع هزینه را بهبود دهد، پذیرفته خواهد شد. در حالتی که مقدار تابع هزینه‌ی جواب همسایه بدتر باشد، به شرط ارضاء رابطه‌ی $r > e^{-\Delta/T}$ باز هم می‌توان آن را پذیرفت. در این رابطه، r یک عدد تصادفی است که با استفاده از توزیع تصادفی یکنواخت بین ۰ و ۱ تولید می‌شود؛ Δ اختلاف بین مقادیر تابع هزینه‌ی جواب جاری و همسایه‌ی جدید، و T دمای جاری است. این دما که مقداری مثبت است، با پیشرفت الگوریتم مطابق رویه‌ی خنک‌سازی از یک مقدار نسبتاً بالا به سمت یک مقدار نزدیک به صفر حرکت می‌کند. بنابراین، در

۲.۲.۳.۱.۲: اگر $\Delta \leq 0$ قرار بده $S_G^* = S_T^*$ و $fc = fc + 1$.

۲.۲.۳.۱.۲: اگر $\Delta > 0$ ، یک عدد تصادفی $X \sim U(0, 1)$ تولید کن؛

۲.۲.۳.۱.۲: اگر $X > e^{-\Delta/T}$ ، قرار بده $S_T^i = S_T^n$ و $iter = iter + 1$.

۲.۲.۳.۱.۲: در غیر این صورت یک جهش شدید در S_T^i ایجاد کن.

۲.۲: قرار بده $T = T \times \alpha$ ، $iter = 0$ و $acts = 0$.

مرحله ۳: بهترین جواب پیدا شده تاکنون، S_G^* ، را به عنوان جواب نهایی برگردان.

جواب اولیه ۴ (IS۴). در این سازوکار از بزرگ‌ترین زمان پردازش در هر دو سطح -- تعیین توالی گروه‌ها و توالی کارهای داخل گروه‌ها -- استفاده می‌شود. توالی گروه‌ها براساس بزرگ‌ترین زمان پردازش تجمعی مشخص می‌شود. زمان پردازش تجمعی هر گروه به صورت جمع زمان اجرای همه کارهای گروه مورد نظر در تمام مرحله‌ها و کمیته زمان‌های آماده‌سازی برای گروه مورد نظر در تمام مرحله‌ها محاسبه می‌شود. پس از مشخص شدن توالی گروه‌ها توالی کارهای داخل گروه‌ها براساس بزرگ‌ترین زمان اجرای تجمعی مشخص می‌شود.

برای تعیین توالی کارها در مرحله‌های دوم تا مرحله‌ی آخر از قاعده‌ی FIFO استفاده می‌کنیم، به این صورت که توالی گروه‌ها در یک مرحله براساس زمان تکمیل اولین کار آن‌ها در مرحله‌ی قبل تعیین می‌شود. به عبارت دیگر، گروهی که اولین کار آن در مرحله‌ی $c - 1$ زودتر تکمیل شده باشد، در مرحله‌ی c زودتر پردازش می‌شود. برای تخصیص کار اول گروه‌ها به ماشین‌ها در هر مرحله‌ی c ، $c = 1, 2, \dots, m$ از یک الگوریتم حریصانه^{۱۷} استفاده می‌کنیم، یعنی کار به ماشینی تخصیص داده می‌شود که با توجه به زمان‌های آماده‌سازی هرچه سریع‌تر تکمیل شود.

۲.۲.۴. سازوکارهای تولید جواب اولیه در رویکرد دوم

در این رویکرد برای نشان دادن جواب اولیه، باید گروه‌های تخصیص داده شده به هر ماشین، توالی گروه‌های روی هر ماشین و همچنین توالی کارهای داخل گروه‌ها، در مرحله‌ی اول مشخص باشد. در این رویکرد برای تعیین جواب اولیه از روش‌های زیر استفاده می‌کنیم:

جواب اولیه‌ی ۱ (IS۱). این روش ساده‌ترین روش تولید جواب اولیه است. در این روش، گروه‌ها به صورت تصادفی به ماشین‌های مرحله‌ی اول تخصیص داده می‌شوند، به طوری که به هر ماشین دست‌کم یک گروه تخصیص داده شود. توالی گروه‌های تخصیص داده شده به هر ماشین و همچنین توالی کارهای متعلق به هر گروه به صورت تصادفی تعیین می‌شود. به عنوان مثال، جواب اولیه برای مسئله‌ی نمونه با استفاده از این سازوکار عبارت است از: S مخفف مرحله و M مخفف ماشین (است)

$$S1 - M1 : G1 : J11 - J12 - J13$$

$$S1 - M2 : G2 : J21 - J22 - J23, \quad G3 : J31 - J32 - J33$$

جواب اولیه‌ی ۲ (IS۲). در این روش به منظور حفظ سادگی، تخصیص گروه‌ها به ماشین‌ها همچنان به صورت تصادفی انجام می‌شود، با این تفاوت که برای تعیین توالی گروه‌های روی هر ماشین و همچنین کارهای داخل هر گروه، از روش ابتکاری بزرگ‌ترین زمان پردازش استفاده می‌کنیم. توالی گروه‌ها براساس بزرگ‌ترین زمان پردازش تجمعی مشخص می‌شود. پس از مشخص شدن توالی گروه‌ها توالی کارهای داخل گروه‌ها براساس بزرگ‌ترین زمان اجرای تجمعی مشخص می‌شود.

جواب اولیه‌ی ۳ (IS۳). در این سازوکار از قاعده بزرگ‌ترین زمان پردازش در

در هر دو سطح -- تعیین توالی گروه‌ها و توالی کارهای داخل گروه‌ها -- استفاده می‌شود.^[۲] در این روش مرحله‌ی (ماشینی) که دارای بزرگ‌ترین زمان پردازش تجمعی برای همه گروه‌هاست، تحت عنوان مرحله‌ی (ماشین) کلیدی در نظر گرفته می‌شود. از آنجا که در این مسئله زمان‌های آماده‌سازی وابسته به توالی است، مفهوم کمیته زمان راه‌اندازی را معرفی و از آن در محاسبه‌ی بزرگ‌ترین زمان پردازش تجمعی استفاده می‌کنیم. کمیته زمان راه‌اندازی برای هر گروه در هر مرحله کم‌ترین زمان از بین زمان‌های راه‌اندازی گروه مورد نظر در مرحله‌ی مربوطه است. بنابراین زمان پردازش تجمعی^{۱۵} برای هر مرحله (ماشین) به صورت جمع کمیته زمان‌های آماده‌سازی همه گروه‌ها در آن مرحله و زمان اجرای کارهای متعلق به همه گروه‌ها در مرحله‌ی مورد نظر محاسبه می‌شود. اگر زمان پردازش تجمعی دو مرحله (ماشین) مساوی باشد مرحله‌ی (ماشینی) که شماره‌ی کوچک‌تری دارد به عنوان مرحله‌ی (ماشین) کلیدی در نظر گرفته می‌شود. پس از پیدایش مرحله‌ی (ماشین) کلیدی، توالی کارهای داخل گروه‌ها براساس بزرگ‌ترین زمان اجرا^{۱۶} روی ماشین کلیدی و توالی گروه‌ها براساس بزرگ‌ترین زمان پردازش تجمعی روی ماشین کلیدی مشخص می‌شود.

نشانه‌گذاری

s : یک جواب ممکن

T_f : دمای پایانی

S_G^* : بهترین جواب به دست آمده تاکنون

L_{max} : تعداد تکرار در هر دما

S_T^* : بهترین جواب به دست آمده در دمای T

$iter$: شمارنده‌ی تعداد تکرارها در دمای T

S_T^i : جواب اولیه در دمای T

$acts$: شمارنده‌ی تعداد S_T^* های تغییر یافته در دمای T

S_T^n : جواب همسایه در دمای T

fc : شمارنده تعداد S_G^* های تغییر یافته در فرایند تبرید

SIG : حد بالای ضریب انجماد

α : ضریب کاهش دما

SIT : حد بالای ضریب پایان جست‌وجو در دمای T

الگوریتم پیشنهادی بر پایه‌ی شبیه‌سازی تبرید

مرحله ۱: (ارزش‌دهی اولیه)

۱.۱: دمای اولیه (T_0) ، دمای نهایی، T_f و نرخ خنک‌سازی α را مقداردهی کن و قرار بده $T = T_0$.

۲.۱: یک برنامه زمانی اولیه، S_T^i تولید کن (بخش ۲.۴) و قرار بده $S_G^* = S_T^* = S_T^i$ ، $iter = 0$ و $acts = 0$.

۳.۱: قرار بده $iter = 0$ و $acts = 0$.

مرحله ۲: تا وقتی که انجماد صورت نگرفته $(T > T_f)$ و $(fc < SIG)$ ، مراحل زیر را تکرار کن:

۱.۲: تا وقتی که $(iter < L_{max})$ و $(acts < SIT)$ ، مراحل زیر را تکرار کن:

۱.۱.۲: یک برنامه‌ی زمانی همسایه برای S_T^i ، S_T^n پیدا کن (بخش ۳.۴).

۲.۱.۲: مقدار $\Delta = f(S_T^n) - f(S_T^i)$ را محاسبه کن.

۳.۱.۲: اگر $\Delta \leq 0$ قرار بده $S_T^i = S_T^n$ و $iter = iter + 1$.

۱.۳.۱.۲: مقدار $\Delta = f(S_T^n) - f(S_T^*)$ را محاسبه کن.

۲.۳.۱.۲: اگر $\Delta \leq 0$ قرار بده $S_T^* = S_T^n$ و $acts = acts + 1$.

۱.۲.۳.۱.۲: مقدار $\Delta = f(S_T^*) - f(S_G^*)$ را محاسبه کن.

در تمامی مرحله‌ها به دست می‌آید. توالی کارهای داخل گروه‌ها برای هر یک از همسایه‌های به‌دست آمده بدون تغییر باقی می‌ماند. برای جواب ورودی چنین عمل می‌کنیم:

با استفاده از سازوکار تعویض جفتی جزئی، جست‌وجوی داخلی برای توالی گروهی جواب ورودی با هدف تعیین بهترین همسایه‌ی کاری متناظر با آن انجام می‌شود. به‌منظور مقداردهی اولیه‌ی فهرست کاندیدای داخلی، فهرست کاندیدای خارجی و بهترین زمان تکمیل، توالی کاری، توالی گروهی و زمان تکمیل جواب ورودی به ترتیب در ICL ، OCL و BM قرار می‌گیرد. همه‌ی همسایه‌های کاری جواب ورودی در تمامی مرحله‌ها با استفاده از آشفتگی‌های جزءبندی شده روی کارهای متعلق به هر گروه به دست آمده (به‌عنوان مثال اولین همسایه‌ی کاری با تعویض دو کار اول گروه اول در مرحله‌ی اول به دست می‌آید) و زمان تکمیل هر یک محاسبه می‌شود. توالی دارای کم‌ترین زمان تکمیل به‌عنوان ورودی بعدی در ICL ، توالی گروهی متناظر با آن در OCL و زمان تکمیل مربوط به این جواب در BM قرار می‌گیرد.

برای هر یک از همسایه‌های گروهی جواب ورودی چنین عمل می‌کنیم:

ابتدا زمان تکمیل همسایه‌ی مورد نظر با توالی اولیه‌ی کارها را با مقدار قرار گرفته در BM مقایسه می‌کنیم، اگر مدت آن کم‌تر باشد این جواب جایگزین بهترین جواب قبلی می‌شود (توالی کارهای آن در ICL ، توالی گروهی متناظر با آن در OCL و زمان تکمیل آن در BM قرار می‌گیرد). سپس آشفتگی‌های جزءبندی شده به ترتیب روی کارهای متعلق به هر یک از گروه‌ها در مرحله‌ی که دچار آشفتگی گروهی شده انجام و زمان تکمیل هر جواب محاسبه می‌شود. زمان تکمیل بهترین جواب با مقدار قرار گرفته در BM مقایسه و در صورتی که کم‌تر از آن باشد، عمل جایگزینی این جواب با بهترین جواب قبلی انجام می‌شود. نمودار جریان تابع تولید جواب همسایه در ضمیمه‌ی ۱ نشان داده شده است. در این تابع وقتی مرحله‌ی c ($c = 1, 2, \dots, m$) دچار آشفتگی گروهی می‌شود، زمان‌بندی کارها و گروه‌ها از مرحله‌ی ۱ تا $c - 1$ بدون تغییر باقی می‌ماند و زمان‌بندی کارها و گروه‌ها برای مرحله‌های $m, \dots, c + 1$ طبق قاعده‌ی FIFO و الگوریتم حریصانه انجام می‌شود (برای تخصیص گروه‌ها به ماشین‌ها در تمام مرحله‌ها از الگوریتم حریصانه استفاده می‌کنیم).

۳.۳.۴. سازوکار تولید جواب همسایه در رویکرد دوم

در این رویکرد از سازوکار تعویض جفتی، حرکت انتقالی و تعویض جفتی جزئی برای تابع تولید همسایگی استفاده می‌کنیم. ورودی این تابع یک جواب کامل (توالی موجه) و خروجی آن بهترین همسایه‌ی متناظر با آن جواب است. در این تابع برای تعیین همسایه‌های گروهی جواب ورودی در هر مرحله چنین عمل می‌کنیم (گام‌های دوم و سوم برای مرحله‌هایی که دارای بیش از یک ماشین هستند انجام می‌شود). گام اول: با استفاده از سازوکار حرکت انتقالی، تمامی همسایه‌های گروهی هر ماشین که بیش از یک گروه به آن تخصیص داده شده تعیین می‌شود. به‌عنوان مثال، فرض می‌کنیم که در مسئله‌ی با چهار گروه و دو ماشین در مرحله‌ی اول، جواب اولیه مطابق زیر است:

$$S_1 - M_1 : G_1, G_2$$

$$S_1 - M_2 : G_3, G_4$$

چون هر دو ماشین دارای بیش از یک گروه‌اند، می‌توانیم با استفاده از سازوکار فوق همسایه‌های گروهی آن‌ها را تعیین کنیم. همسایه‌های گروهی ماشین‌های اول و دوم

هر سه سطح -- تخصیص گروه‌ها به ماشین‌ها، تعیین توالی گروه‌ها و توالی کارهای داخل گروه‌ها -- استفاده می‌شود. در این روش پس از پیدایش مرحله‌ی (ماشین) کلیدی، تخصیص گروه‌ها به ماشین‌ها و همچنین توالی گروه‌های روی هر ماشین براساس بزرگ‌ترین زمان پردازش تجمعی روی ماشین کلیدی و الگوریتم حریصانه و توالی کارهای داخل گروه‌ها براساس بزرگ‌ترین زمان اجرا روی ماشین کلیدی انجام می‌شود.

جواب اولیه‌ی ۴ (IS۴). در این سازوکار از بزرگ‌ترین زمان پردازش در هر سه سطح -- تخصیص گروه‌ها به ماشین‌ها، تعیین توالی گروه‌ها، و توالی کارهای داخل گروه‌ها -- استفاده می‌شود. تخصیص گروه‌ها به ماشین‌ها و همچنین توالی گروه‌های روی هر ماشین براساس بزرگ‌ترین زمان پردازش تجمعی، الگوریتم حریصانه، و توالی کارهای داخل گروه‌ها براساس بزرگ‌ترین زمان اجرای تجمعی مشخص می‌شود.

در این رویکرد نیز برای تعیین توالی گروه‌ها و کارها در مرحله‌های ۲ تا m از قاعده‌ی FIFO، و برای تخصیص گروه‌ها به ماشین‌ها در مرحله‌های $m, \dots, c = 1$ از الگوریتم حریصانه استفاده می‌کنیم. همانند رویکرد اول، جواب اولیه‌ی کامل که شامل توالی گروه‌ها و کارها در تمام مرحله‌هاست، با تکمیل کردن جواب اولیه طبق قاعده‌ی FIFO و الگوریتم حریصانه به دست می‌آید.

۳.۴. همسایگی

در الگوریتم شبیه‌سازی تیرید پیشنهادی، برای یافتن بهترین همسایه‌ی متناظر با هر جواب از دو مرحله جست‌وجو استفاده می‌کنیم. در جست‌وجوی خارجی، هر توالی جدید از گروه‌ها با انجام آنچه که آشفتگی خارجی روی توالی گروه‌ها نامیده می‌شود، به دست می‌آید. برای ایجاد آشفتگی روی گروه‌ها از دو سازوکار حرکت انتقالی و تعویض جفتی^[۲۷] استفاده می‌شود. در جست‌وجوی داخلی نیز، برای به دست آوردن هر توالی جدید از کارها از آشفتگی روی کارها استفاده می‌کنیم. برای ایجاد آشفتگی روی کارها از سازوکار تعویض جفتی جزئی استفاده می‌شود. در این تحقیق از دو رویکرد برای تعیین جواب همسایه استفاده شده است.

۱.۳.۴. سازوکارهای همسایگی

از این سازوکارها برای حرکت از یک جواب ممکن به یک جواب ممکن دیگر در فضای جواب مسئله استفاده می‌شود.

سازوکار حرکت انتقالی. این سازوکار مکان یک گروه انتخاب شده را تغییر می‌دهد؛ بدین معنا که یک گروه دلخواه G_r از مکان r به مکان i انتقال داده می‌شود، در حالی که ترتیب سایر گروه‌ها بدون تغییر باقی می‌ماند. اگر $1 \leq r < i \leq g$ این انتقال را «انتقال به راست» و اگر $1 \leq i < r \leq g$ آن را «انتقال به چپ» می‌نامند. در این همسایگی جواب جاری با g گروه دارای $(g - 1)^2$ همسایه است.

سازوکار تعویض جفتی. این سازوکار جای دو گروه اختیاری G_i و G_r را ($1 \leq i, r \leq g$ و $i \neq r$) عوض می‌کند. این گروه‌ها در مکان‌های r و i قرار دارند. در این همسایگی جواب جاری با g گروه دارای $g \times (g - 1) / 2$ همسایه است. سازوکار تعویض جفتی جزئی همانند تعویض جفتی است با این تفاوت که این تعویض شامل تعویض گردشی نیز هست. در این همسایگی جواب جاری با n کار دارای n همسایه است.

۲.۳.۴. سازوکار تولید جواب همسایه در رویکرد اول

در این تابع از سازوکارهای تعویض جفتی و تعویض جفتی جزئی استفاده می‌کنیم. ورودی این تابع یک جواب کامل و خروجی آن بهترین همسایه‌ی متناظر با آن جواب است. با استفاده از سازوکار تعویض جفتی، همه‌ی همسایه‌های گروهی جواب ورودی

عبارت است از:

۵. نتایج محاسباتی

در این بخش کیفیت جواب‌های ایجاد شده توسط الگوریتم‌های فرایندکاری توسعه یافته بر پایه شبیه‌سازی تبرید با هدف شناسایی بهترین الگوریتم مورد مقایسه قرار می‌گیرند. سپس عملکرد الگوریتم برتر ارائه شده در این تحقیق را با تنها الگوریتم موجود در ادبیات، یعنی الگوریتم ارائه شده بر پایه جست‌وجوی ممنوع^[۳] مقایسه می‌کنیم. به منظور مقایسه الگوریتم‌ها، زمان تکمیل کل کارها (مقدار تابع هدف مسئله) و زمان سپری شده برای حل مسئله‌های نمونه -- به‌عنوان دو متغیر پاسخ که به ترتیب نشان‌دهنده‌ی کارایی و بهره‌وری الگوریتم است -- در نظر گرفته می‌شوند. زبان برنامه‌نویسی ++C برای کدنویسی الگوریتم‌های ارائه شده در این تحقیق مورد استفاده قرار گرفته است. بدین منظور از رایانه‌یی با پردازش‌گر پنتیوم چهار ۲ گیگاهرتز، با ۱ گیگابایت حافظه با دسترسی تصادفی (RAM) و سیستم عامل ویندوز XP برای انجام آزمایش‌ها استفاده شده است.

۱.۵. مجموعه داده‌ها

داده‌های مورد نیاز برای تعریف یک مسئله شامل زمان پردازش کارها، زمان‌های آماده‌سازی هر گروه به‌روی هر ماشین، تعداد مرحله‌ها و تعداد ماشین‌های موازی موجود در هر مرحله‌اند.

اندازه‌ی مسئله یک عامل مهم برای ایجاد مسئله‌های تصادفی است، به طوری که مسئله‌های بزرگ‌تر نیازمند صرف زمان بیشتر برای حل توسط الگوریتم‌های فرایندکاری‌اند. اندازه‌ی مسئله در مسائل زمان‌بندی گروهی جریان کارگاهی انعطاف‌پذیر توسط سه پارامتر: تعداد مرحله‌ها (ns)، تعداد گروه‌ها (ng) و تعداد کارهای متعلق به هر گروه (nz) تعیین می‌شود.^[۱۹] محققین برای ارزیابی الگوریتم‌های ابداعی خودشان مسئله‌هایی تصادفی در سه اندازه‌ی کوچک، متوسط و بزرگ ایجاد کردند. تعداد مرحله‌ها از محدوده‌ی ۲ تا ۳، ۴ تا ۶ و ۷ تا ۹ به ترتیب برای مسئله‌های کوچک، متوسط و بزرگ انتخاب شد. تعداد گروه‌ها از محدوده‌ی ۳ تا ۵، ۶ تا ۹ و ۱۰ تا ۱۲ به ترتیب برای مسئله‌های کوچک، متوسط و بزرگ انتخاب شد در حالی که تعداد کارهای هر گروه از محدوده‌ی ۲ تا ۵، ۶ تا ۹ و ۱۰ تا ۱۲ به ترتیب برای سه اندازه مختلف انتخاب شد.

همه‌ی پارامترهای یادشده و نیز زمان اجرای کارها و زمان‌های آماده‌سازی با استفاده از توزیع تصادفی یکنواخت گسسته بین حدود تعریف شده تعیین شده‌اند. زمان اجرای کارهای موجود در هر گروه با استفاده از توزیع یکنواخت گسسته، از محدوده‌ی ۱ تا ۲۰ و زمان‌های آماده‌سازی گروه‌ها از محدوده‌ی ۰ تا ۱۰۰ تولید شده است.

برخلاف مسئله‌ی زمان‌بندی گروهی جریان کارگاهی که انعطاف‌پذیری در آن مطرح نیست، انعطاف‌پذیری در مسئله‌ی جریان کارگاهی انعطاف‌پذیر به‌وسیله‌ی تعداد ماشین‌های موازی یکسان در هر مرحله تعریف می‌شود. در این راستا، محققین انعطاف‌پذیری را در در سه سطح پایین، متوسط و بالا در نظر گرفتند.^[۱۹] در انعطاف‌پذیری پایین یک‌سوم مرحله‌ها دارای ماشین‌های موازی‌اند، در انعطاف‌پذیری متوسط دوسوم مرحله‌ها، و در انعطاف‌پذیری بالا تمامی مرحله‌ها دارای ماشین‌های موازی‌اند. برای تعیین این که کدام مرحله‌ها دارای ماشین‌های موازی‌اند، یک جایگشت تصادفی از مرحله‌های تولید و سپس انعطاف‌پذیری در تعداد مرحله‌ها ضرب می‌شود ($d = flex \times ns$). اگر حاصل ضرب یک عدد صحیح نباشد، عدد تولیدشده به سمت عدد صحیح بالاتر گرد می‌شود. فرض می‌شود که اولین d مرحله دارای

$$S1 - M1 : G1, G2 \quad S - M1 : G2, G1$$

$$S1 - M2 : G2, G2 \quad S - M2 : G2, G2$$

گام دوم: همه‌ی ترکیب‌های دوتایی ماشین‌های موجود در مرحله‌ی مورد بررسی در نظر گرفته می‌شود. برای هر ترکیب تمام حالت‌های تعویض‌های جفتی بین ماشین‌ها تعیین می‌شود. هر یک از این تعویض‌ها نشان‌دهنده‌ی یک همسایه‌ی گروهی است. به‌عنوان مثال، جواب اولیه‌ی جزئی فوق دارای یک ترکیب دوتایی (۲،۱) است. این ترکیب دارای چهار تعویض جفتی است:

$$S1 - M1 : G2, G2 \quad S1 - M1 : G2, G2$$

$$S1 - M2 : G2, G1 \quad S1 - M2 : G1, G2$$

$$S1 - M1 : G1, G2 \quad S1 - M1 : G1, G2$$

$$S1 - M2 : G2, G2 \quad S1 - M2 : G2, G2$$

گام سوم: همه‌ی ترکیب‌های دوتایی ماشین‌های موجود در مرحله‌ی مورد بررسی، که در آن‌ها ماشین اول دارای بیش از یک گروه است، در نظر گرفته می‌شود. برای هر ترکیب، گروه‌های روی ماشین اول به ترتیب به ماشین دوم منتقل می‌شوند. هر انتقال نشان‌دهنده‌ی یک همسایه‌ی گروهی است. برای هر یک از این انتقال‌ها گام اول تکرار می‌شود. به‌عنوان مثال، جواب اولیه‌ی جزئی فوق دارای دو ترکیب دوتایی (۲،۱) و (۲،۱) است. هر یک از این ترکیب‌ها دارای دو همسایه است:

$$S1 - M1 : G2 \quad S1 - M1 : G1$$

$$S1 - M2 : G2, G2, G1 \quad S1 - M2 : G2, G2, G2$$

$$S1 - M1 : G2 \quad S1 - M1 : G2$$

$$S1 - M2 : G1, G2, G2 \quad S1 - M2 : G1, G2, G2$$

برای یافتن بهترین برنامه‌ی زمانی متناظر با جواب ورودی و هر یک از همسایه‌های گروهی آن، مطابق آنچه که در رویکرد اول شرح داده شد عمل می‌کنیم. در این تابع نیز وقتی مرحله‌ی c ($c = 1, 2, \dots, m$) دچار آشفتگی گروهی می‌شود، زمان‌بندی کارها و گروه‌ها از مرحله‌ی 1 تا $c-1$ بدون تغییر باقی می‌ماند، اما برای مرحله‌های $1, \dots, m$ طبق قاعده‌ی FIFO و الگوریتم حریم‌انجام می‌شود (تخصیص گروه‌ها به ماشین‌ها در مرحله‌های $1, \dots, c$ مشخص است. برای تخصیص گروه‌ها به ماشین‌ها در مرحله‌های $1, \dots, m$ از الگوریتم حریم‌انجام استفاده می‌کنیم).

۴.۴. شرایط توقف

معیارهای توقف به دو دسته تقسیم می‌شوند:

۱. معیارهای خروج از دمای T ؛ اگر یکی از معیارهای زیر ارضا شود الگوریتم از دمای T خارج می‌شود:
 - رسیدن به بیشترین تعداد تکرار در دمای $(L_{max})T$ ؛
 - رسیدن به حد بالای تعداد دفعات تغییر بهترین جواب در دمای $(SI_T)T$.
۲. معیارهای توقف الگوریتم؛ اگر یکی از معیارهای زیر ارضا شود الگوریتم خاتمه می‌یابد:
 - رسیدن به دمای پایانی (T_f) ؛
 - رسیدن به حد بالای تعداد دفعات تغییر بهترین جواب فرایند تبرید (SI_G) .

واریانس یک‌طرفه استفاده شده و نتایج آن در جدول ۳ نمایش داده شده است. براساس نتایج حاصله، شواهد معنی‌داری مبنی بر تأثیر سازوکار تولید جواب اولیه وجود ندارد. نمودار میانگین و فاصله برای مسئله‌های کوچک را می‌توان در شکل ۱ مشاهده کرد. در صد انحراف نسبی با استفاده از رابطه زیر محاسبه می‌شود:

$$RPD = \frac{Alg_{sol} - Min_{sol}}{Min_{sol}}$$

که در آن Alg_{sol} جواب به دست آمده توسط الگوریتمی معین برای یک مسئله‌ی مشخص است و Min_{sol} کمینه جواب به دست آمده برای آن مسئله است.

مسئله‌های متوسط

برای مسئله‌های متوسط، میانگین RPD سازوکارهای تولید جواب اولیه در جدول ۴ و نتایج تحلیل واریانس در جدول ۵ نمایش داده شده است. نتایج نشان می‌دهد که شواهد معنی‌داری مبنی بر تأثیر سازوکار تولید جواب اولیه وجود ندارد. نمودار میانگین و فاصله برای مسئله‌های متوسط را می‌توان در شکل ۲ مشاهده کرد.

مسئله‌های بزرگ

برای مسئله‌های بزرگ، میانگین RPD و نتایج تحلیل واریانس به ترتیب در جداول ۶ و ۷ نمایش داده شده است. براساس نتایج حاصله، شواهد معنی‌داری مبنی بر تأثیر سازوکار تولید جواب اولیه وجود ندارد. نمودار میانگین و فاصله برای مسئله‌های بزرگ را می‌توان در شکل ۳ مشاهده کرد.

۲.۳.۵. مقایسه‌ی کارایی سازوکارهای تولید جواب اولیه در رویکرد دوم

در این رویکرد نیز از زمان تکمیل کل به‌عنوان یک اندازه‌ی نشان‌دهنده‌ی کارایی سازوکارهای تولید جواب اولیه استفاده می‌شود.

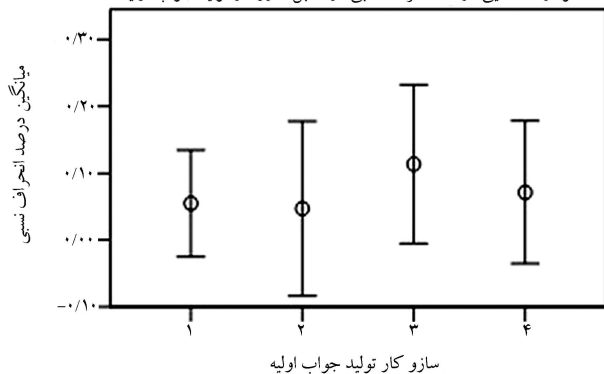
جدول ۲. میانگین درصد انحراف نسبی برای مسئله‌های کوچک.

جواب اولیه	IS۱	IS۲	IS۳	IS۴
میانگین درصد انحراف نسبی	۰/۰۵۴۷	۰/۰۴۷۰	۰/۱۱۳۶	۰/۰۷۱۲

جدول ۳. نتایج تحلیل واریانس برای مسئله‌های کوچک.

منبع	df	SS	MS	F	P value
جواب اولیه	۳	۰/۰۱۳	۰/۰۰۴	۰/۵۵۶	۰/۶۵۱
خطا	۵۶	۰/۴۴۸	۰/۰۰۸		
مجموع	۵۹	۰/۴۶۱			

نمودار فاصله‌ی درصد انحراف نسبی در مقابل سازوکار تولید جواب اولیه



شکل ۱. نمودار میانگین و فاصله برای مسئله‌های کوچک.

ماشین‌های موازی‌اند. [۳۰] برای مرحله‌های دارای ماشین‌های موازی، تعداد ماشین‌ها با احتمال برابر ۲ یا ۳ در نظر گرفته می‌شود. با مشخصات فوق، ۵ مسئله‌ی کوچک، ۵ مسئله‌ی متوسط و ۵ مسئله‌ی بزرگ تولید می‌شود که از آن‌ها برای تنظیم پارامترهای الگوریتم شبیه‌سازی تیرید پیشنهادی استفاده خواهد شد. برای مقایسه‌ی سازوکارهای تولید جواب اولیه و الگوریتم‌های پیشنهادی از ۴۵ مسئله‌ی ارائه شده توسط پژوهش‌گران [۳] استفاده می‌شود.

۲.۵. تنظیم پارامترها

به‌منظور افزایش کارایی الگوریتم‌های پیشنهادی، تنظیم پارامترهای الگوریتم ضرورت می‌یابد. پارامترهای مورد نظر بر کارایی الگوریتم‌های مبتنی بر شبیه‌سازی تیرید و کیفیت جواب به‌دست آمده تأثیر مستقیم دارند. این پارامترها عبارت‌اند از: دمای اولیه (T_0)، بیشینه تعداد تکرار در هر دما (L_{max})، حد بالای ضریب پایان جست‌وجو در دمای T (SI_T)، و حد بالای ضریب انجماد (SI_G). برای تنظیم پارامترها، ابتدا مقادیر پارامترهای فوق را با توجه به اندازه مسئله آن‌قدر بزرگ انتخاب می‌کنیم که تقریباً مطمئن شویم بهترین جواب با استفاده از الگوریتم مورد نظر به دست آمده است. سپس با کاهش تدریجی تک‌تک پارامترها زمان حل مسئله را طوری کاهش می‌دهیم که همان جواب قبلی دوباره به دست آید. این کار را تا جایی ادامه می‌دهیم که دیگر کاهش هیچ‌یک از پارامترها امکان‌پذیر نباشد. نتایج تنظیم پارامترها در جدول ۱ ارائه شده است.

دمای اولیه با توجه به اندازه مسئله از محدوده‌ی ۱۰۰ تا ۴۰۰ انتخاب می‌شود. دمای پایانی ۰/۵ و ضریب کاهش دما ۰/۹۵ است. بیشینه تکرار در هر دما با توجه به اندازه مسئله از بازه ۱۰ تا ۳۰ انتخاب می‌شود. حد بالای ضریب انجماد با توجه به اندازه مسئله از بازه ۱۰ تا ۵۰ و حد بالای ضریب پایان جست‌وجو در دمای T نیز با توجه به اندازه مسئله از محدوده‌ی ۱۰ تا ۲۰ انتخاب می‌شود. برای تعیین پارامترها از ۱۵ مسئله‌ی تصادفی که روش تولید آن‌ها در قسمت قبل شرح داده شد، استفاده شده است.

۳.۵. تعیین جواب اولیه

برای به دست آوردن یک جواب اولیه‌ی خوب، کارایی چهار سازوکار تولید جواب اولیه در هر رویکرد و برای هر اندازه مسئله بررسی می‌شود. در این مطالعه، برای ارزیابی نتایج از تحلیل واریانس یک‌طرفه استفاده می‌شود.

۱.۳.۵. مقایسه‌ی کارایی سازوکارهای تولید جواب اولیه در رویکرد اول

ما از زمان تکمیل کل به‌عنوان اندازه‌ی که نشان‌دهنده‌ی کارایی سازوکارهای تولید جواب اولیه است استفاده می‌کنیم.

مسئله‌های کوچک

میانگین درصد انحراف نسبی (RPD) سازوکارهای تولید جواب اولیه در جدول ۲ نشان داده شده است. برای مقایسه‌ی کارایی سازوکارهای تولید جواب اولیه، از تحلیل

جدول ۱. تنظیم پارامترها.

اندازه مسئله	ضریب			
	SI_G	SI_T	L_{max}	T_0
کوچک	۲۰	۱۰	۱۰	۱۰۰
متوسط	۳۰	۱۰	۲۰	۱۵۰
بزرگ	۵۰	۲۰	۳۰	۴۰۰

جدول ۴. میانگین درصد انحراف نسبی برای مسئله‌های متوسط.

IS۴	IS۳	IS۲	IS۱	جواب اولیه
۰/۰۵۲۴	۰/۰۳۸۳	۰/۰۶۷۵	۰/۰۳۱۸	میانگین درصد انحراف نسبی

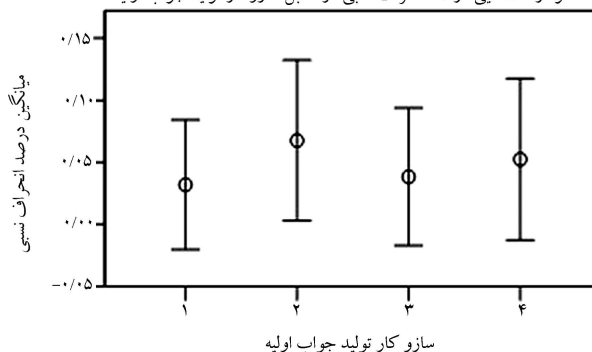
مسئله‌های کوچک
 برای مسئله‌های کوچک، میانگین RPD و نتایج تحلیل واریانس به ترتیب در جداول ۸ و ۹ نمایش داده شده است. نتایج نشان می‌دهد که شواهد معنی‌داری مبنی بر تأثیر سازوکار تولید جواب اولیه وجود ندارد. نمودار میانگین و فاصله برای مسئله‌های کوچک را می‌توان در شکل ۴ مشاهده کرد.

جدول ۵. نتایج تحلیل واریانس برای مسئله‌های متوسط.

منبع	df	SS	MS	F	P value
جواب اولیه	۳	۰/۰۰۰۴	۰/۰۰۰۱	۰/۵۴۴	۰/۶۵۹
خطا	۵۶	۰/۱۱۵	۰/۰۰۰۲		
مجموع	۵۹	۰/۱۱۹			

مسئله‌های متوسط
 برای مسئله‌های متوسط، میانگین RPD و نتایج تحلیل واریانس به ترتیب در جداول ۱۰ و ۱۱ نمایش داده شده است. نتایج نشان می‌دهد که شواهد معنی‌داری مبنی بر تأثیر سازوکار تولید جواب اولیه وجود ندارد. نمودار میانگین و فاصله برای مسئله‌های متوسط را می‌توان در شکل ۵ مشاهده کرد.

نمودار فاصله بی درصد انحراف نسبی در مقابل سازوکار تولید جواب اولیه



شکل ۲. نمودار میانگین و فاصله برای مسئله‌های متوسط.

جدول ۶. میانگین درصد انحراف نسبی برای مسئله‌های بزرگ.

IS۴	IS۳	IS۲	IS۱	جواب اولیه
۰/۰۱۳۶	۰/۰۶۳۸	۰/۰۱۶۵	۰/۰۴۶۹	میانگین درصد انحراف نسبی

مسئله‌های بزرگ
 برای مسئله‌های بزرگ، میانگین RPD و نتایج تحلیل واریانس به ترتیب در جداول ۱۲ و ۱۳ نمایش داده شده است. نتایج نشان می‌دهد که شواهد معنی‌داری مبنی بر تأثیر سازوکار تولید جواب اولیه وجود ندارد. نمودار میانگین و فاصله برای مسئله‌های بزرگ را می‌توان در شکل ۶ مشاهده کرد.

از نقطه نظر بهره‌وری (سرعت تعیین جواب)، سازوکارهای تولید جواب اولیه در هر دو رویکرد عملکرد یکسانی دارند. از آنجا که بین سازوکارهای تولید جواب اولیه در هر دو رویکرد -- چه به لحاظ کارایی و چه به لحاظ بهره‌وری -- تفاوت معنی‌داری دیده نمی‌شود، برای تولید جواب اولیه در هر رویکرد از سازوکار اول که دارای محاسبات کم‌تری است استفاده می‌کنیم.

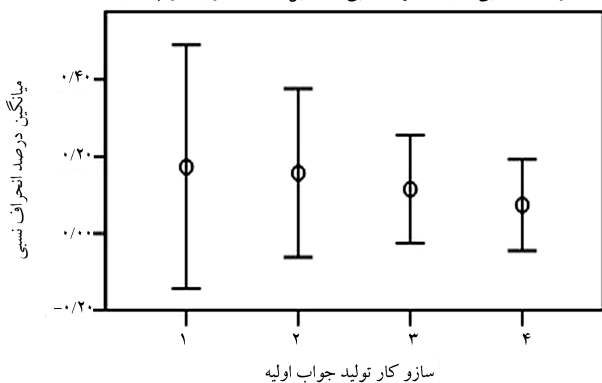
جدول ۸. میانگین درصد انحراف نسبی برای مسئله‌های کوچک

IS۴	IS۳	IS۲	IS۱	جواب اولیه
۰/۰۷۳۹	۰/۱۱۴۸	۰/۱۵۶۹	۰/۱۷۲۸	میانگین درصد انحراف نسبی

جدول ۹. نتایج تحلیل واریانس برای مسئله‌های کوچک.

منبع	df	SS	MS	F	P value
جواب اولیه	۳	۰/۰۳۰	۰/۰۱۰	۰/۳۳۵	۰/۸۰۰
خطا	۵۶	۱/۶۹۰	۰/۰۳۰		
مجموع	۵۹	۱/۷۲۰			

نمودار فاصله بی درصد انحراف نسبی در مقابل سازوکار تولید جواب اولیه

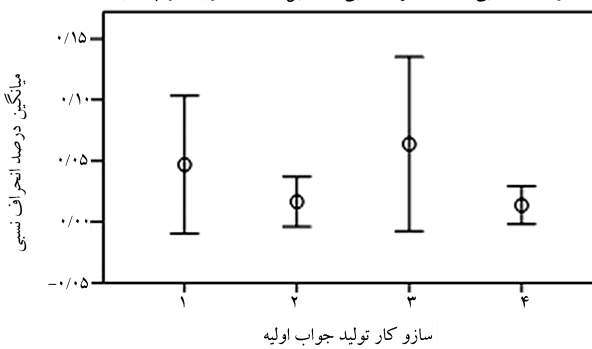


شکل ۴. نمودار میانگین و فاصله برای مسئله‌های کوچک.

جدول ۷. نتایج تحلیل واریانس برای مسئله‌های بزرگ.

منبع	df	SS	MS	F	P value
جواب اولیه	۳	۰/۰۰۰۹	۰/۰۰۰۳	۲/۰۲۸	۰/۱۵۰
خطا	۵۶	۰/۰۵۶	۰/۰۰۰۱		
مجموع	۵۹	۰/۰۶۵			

نمودار فاصله بی درصد انحراف نسبی در مقابل سازوکار تولید جواب اولیه



شکل ۳. نمودار میانگین و فاصله برای مسئله‌های بزرگ.

جدول ۱۰. میانگین درصد انحراف نسبی برای مسئله‌های متوسط.

IS۴	IS۳	IS۲	IS۱	جواب اولیه
۰/۰۳۸۵	۰/۰۱۷۸	۰/۰۲۷۵	۰/۰۳۶۹	میانگین درصد انحراف نسبی

۴.۵. مقایسه‌ی الگوریتم‌های پیشنهادی

در این قسمت، چهار الگوریتم فراابتکاری پیشنهاد شده بر پایه‌ی SA بررسی و مقایسه می‌شود. از این‌رو، زمان تکمیل کل و زمان سپری شده برای حل مسئله به‌عنوان دو معیار که به ترتیب نشان‌دهنده‌ی کارایی و بهره‌وری‌اند در نظر گرفته می‌شود و الگوریتم‌ها به‌طور جداگانه روی مسئله‌های کوچک، متوسط و بزرگ آزمایش می‌شوند. الگوریتم‌های مورد مقایسه عبارت‌اند از:

A۱: شبیه‌سازی تبرید استاندارد بر پایه‌ی رویکرد اول با سازوکار تولید جواب اولیه IS۱.

A۲: شبیه‌سازی تبرید پیشنهادی بر پایه‌ی رویکرد اول با سازوکار تولید جواب اولیه IS۱.

B۱: شبیه‌سازی تبرید استاندارد بر پایه‌ی رویکرد دوم با سازوکار تولید جواب اولیه IS۱.

B۲: شبیه‌سازی تبرید پیشنهادی بر پایه‌ی رویکرد دوم با سازوکار تولید جواب اولیه IS۱.

۱.۴.۵. مقایسه‌ی کارایی الگوریتم‌ها

ما از زمان تکمیل کل به‌عنوان اندازه‌ی نشان‌دهنده‌ی کارایی الگوریتم‌ها استفاده می‌کنیم، و در هر بار اجرای هر الگوریتم برای هر مسئله زمان تکمیل متناظر را ذخیره می‌کنیم.

مسئله‌های کوچک

برای مسئله‌های کوچک، میانگین RPD در جدول ۱۴ و نتایج تحلیل واریانس در جدول ۱۵ نمایش داده شده است. براساس نتایج حاصله، دست‌کم یک الگوریتم به‌لحاظ کارایی متفاوت از سایرین است. از این‌رو، ما از روش کم‌ترین اختلاف قابل توجه (LSD) فیشر استفاده می‌کنیم. با توجه به نتایج ارائه‌شده در جدول ۱۶ الگوریتم‌های A۱ و A۲ با اطمینان ۹۵٪ ترجیح داده می‌شوند و اختلاف قابل توجهی بین آن‌ها وجود ندارد. همچنین B۲ برتر از B۱ است. نمودار میانگین و فاصله برای مسئله‌های کوچک را می‌توان در شکل ۷ مشاهده کرد.

مسئله‌های متوسط

برای مسئله‌های متوسط، میانگین RPD و نتایج تحلیل واریانس به ترتیب در جداول ۱۷ و ۱۸ نمایش داده شده است. براساس نتایج حاصله، دست‌کم یک الگوریتم به‌لحاظ کارایی متفاوت از سایرین است. با توجه به جدول ۱۹ الگوریتم A۲ با اطمینان ۹۵٪ برتر از سایر الگوریتم‌هاست. نمودار میانگین و فاصله برای مسئله‌های متوسط را می‌توان در شکل ۸ مشاهده کرد.

جدول ۱۴. میانگین درصد انحراف نسبی برای مسئله‌های کوچک.

الگوریتم	A۱	A۲	B۱	B۲
میانگین درصد انحراف نسبی	۰/۰۰۰۰	۰/۰۰۰۰	۰/۰۵۱۸	۰/۰۳۱۰

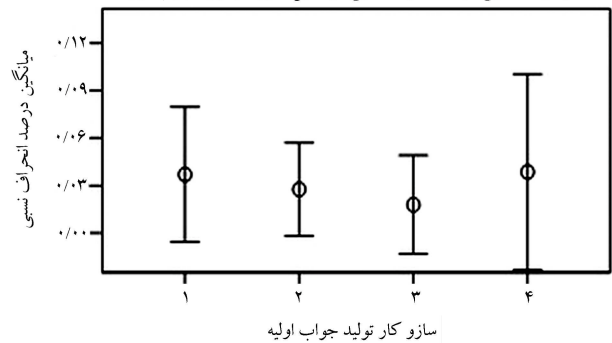
جدول ۱۵. نتایج تحلیل واریانس برای مسئله‌های کوچک.

منبع	df	SS	MS	F	P value
الگوریتم	۳	۰/۰۰۰۹	۰/۰۰۰۳	۳۵,۳۹۹	۰/۰۰۰۰
خطا	۵۶	۰/۰۰۰۵	۰/۰۰۰۰۰۸۵		
مجموع	۵۹	۰/۰۰۱۴۵			

جدول ۱۱. نتایج تحلیل واریانس برای مسئله‌های متوسط.

منبع	df	SS	MS	F	P value
جواب اولیه	۳	۰/۰۰۰۱	۰/۰۰۰۰۳	۰/۳۷۸	۰/۷۷۰
خطا	۵۶	۰/۰۰۵۶	۰/۰۰۰۱		
مجموع	۵۹	۰/۰۰۵۷			

نمودار فاصله‌ی درصد انحراف نسبی در مقابل سازوکار تولید جواب اولیه



شکل ۵. نمودار میانگین و فاصله برای مسئله‌های متوسط.

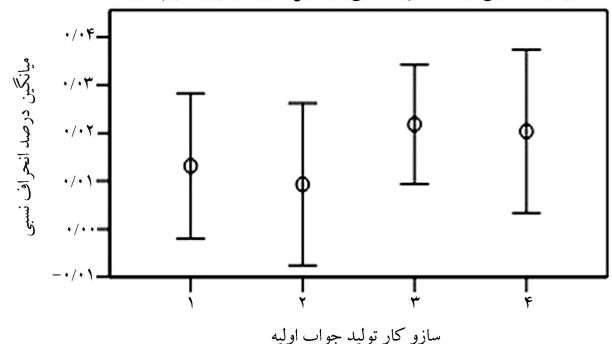
جدول ۱۲. میانگین درصد انحراف نسبی برای مسئله‌های بزرگ.

IS۴	IS۳	IS۲	IS۱	جواب اولیه
۰/۰۲۰۳	۰/۰۲۱۸	۰/۰۰۹۳	۰/۰۱۳۱	میانگین درصد انحراف نسبی

جدول ۱۳. نتایج تحلیل واریانس برای مسئله‌های بزرگ.

منبع	df	SS	MS	F	P value
جواب اولیه	۳	۰/۰۰۰۱	۰/۰۰۰۰۳	۱,۱۳۷	۰/۳۶۴
خطا	۵۶	۰/۰۰۰۷۳	۰/۰۰۰۰۱		
مجموع	۵۹	۰/۰۰۰۸۳			

نمودار فاصله‌ی درصد انحراف نسبی در مقابل سازوکار تولید جواب اولیه

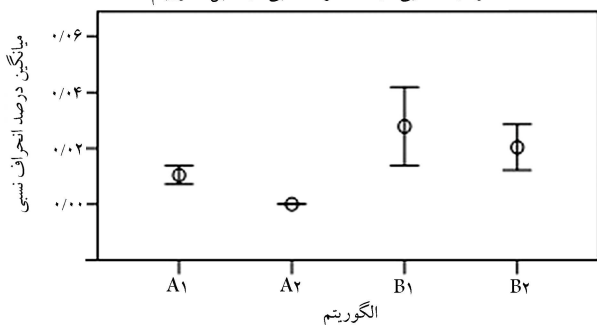


شکل ۶. نمودار میانگین و فاصله برای مسئله‌های بزرگ.

جدول ۱۶. نتایج روش کم‌ترین اختلاف قابل توجه فیشر برای مسئله‌های کوچک.

الگوریتم‌ها	اختلاف میانگین‌ها	کمترین اختلاف قابل توجه	اختلاف قابل توجه در سطح ۹۵٪
A۱ و A۲	۰/۰۰۰۰۰	۰/۰۰۰۹۹	خیر
A۱ و B۱	۰/۰۰۵۱۸	۰/۰۰۰۹۹	بله
A۱ و B۲	۰/۰۰۳۱۰	۰/۰۰۰۹۹	بله
A۲ و B۱	۰/۰۰۵۱۸	۰/۰۰۰۹۹	بله
A۲ و B۲	۰/۰۰۳۱۰	۰/۰۰۰۹۹	بله
B۱ و B۲	۰/۰۰۲۰۸	۰/۰۰۰۹۹	بله

نمودار فاصله‌ی درصد انحراف نسبی در مقابل الگوریتم



شکل ۸. نمودار میانگین و فاصله برای مسئله‌های متوسط.

مسئله‌های بزرگ

برای مسئله‌های بزرگ، میانگین *RPD* و نتایج تحلیل واریانس به ترتیب در جداول ۲۰ و ۲۱ نمایش داده شده است. با توجه به جدول ۲۲ روشن است که الگوریتم A۲ با اطمینان ۹۵٪ برتر از سایر الگوریتم‌هاست. نمودار میانگین و فاصله برای مسئله‌های بزرگ را می‌توان در شکل ۹ مشاهده کرد.

۲.۴.۵. مقایسه‌ی بهره‌وری الگوریتم‌ها

در این قسمت، از زمان سپری‌شده‌ی CPU به‌عنوان یک شاخص بهره‌وری استفاده کرده و الگوریتم‌ها را براساس آن مقایسه می‌کنیم. در هر بار اجرای هر الگوریتم برای هر مسئله، زمان سپری‌شده‌ی CPU را ذخیره می‌کنیم.

مسئله‌های کوچک

میانگین زمان سپری‌شده برای حل مسئله‌های نمونه و نتایج تحلیل واریانس برای مسئله‌های کوچک به ترتیب در جداول ۲۳ و ۲۴ نمایش داده شده است. نتایج نشان

جدول ۲۰. میانگین درصد انحراف نسبی برای مسئله‌های بزرگ.

الگوریتم	A۱	A۲	B۱	B۲
میانگین درصد انحراف نسبی	۰/۰۱۲۷	۰/۰۰۰۰	۰/۰۳۴۴	۰/۰۲۳۳

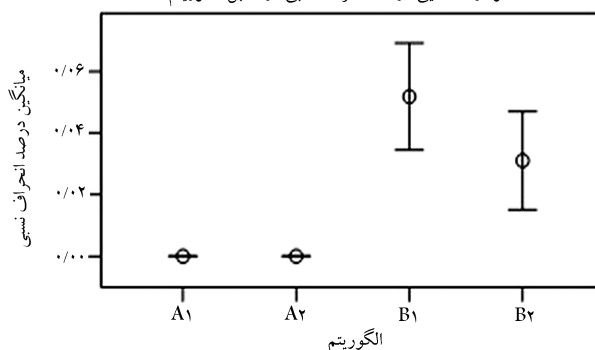
جدول ۲۱. نتایج تحلیل واریانس برای مسئله‌های بزرگ.

منبع	df	SS	MS	F	P value
الگوریتم	۳	۰/۰۰۰۳	۰/۰۰۰۱	۷/۵۹۰	۰/۰۰۰۲
خطا	۵۶	۰/۰۰۰۷۳	۰/۰۰۰۱۳		
مجموع	۵۹	۰/۰۰۱۰۳			

جدول ۲۲. نتایج روش کم‌ترین اختلاف قابل توجه فیشر برای مسئله‌های بزرگ.

الگوریتم‌ها	اختلاف میانگین‌ها	کمترین اختلاف قابل توجه	اختلاف قابل توجه در سطح ۹۵٪
A۱ و A۲	۰/۰۱۰۴	۰/۰۰۰۷۰	بله
A۱ و B۱	۰/۰۱۷۴	۰/۰۰۰۷۰	بله
A۱ و B۲	۰/۰۰۰۹۹	۰/۰۰۰۷۰	بله
A۲ و B۱	۰/۰۰۲۷۸	۰/۰۰۰۷۰	بله
A۲ و B۲	۰/۰۰۲۰۳	۰/۰۰۰۷۰	بله
B۱ و B۲	۰/۰۰۰۷۵	۰/۰۰۰۷۰	بله

نمودار فاصله‌ی درصد انحراف نسبی در مقابل الگوریتم



شکل ۷. نمودار میانگین و فاصله برای مسائل کوچک.

جدول ۱۷. میانگین درصد انحراف نسبی برای مسئله‌های متوسط.

الگوریتم	A۱	A۲	B۱	B۲
میانگین درصد انحراف نسبی	۰/۰۱۰۴	۰/۰۰۰۰	۰/۰۲۷۸	۰/۰۲۰۳

جدول ۱۸. نتایج تحلیل واریانس برای مسئله‌های متوسط.

منبع	df	SS	MS	F	P value
الگوریتم	۳	۰/۰۰۰۲	۰/۰۰۰۰۶۷	۱۶/۴۲۰	۰/۰۰۰۰
خطا	۵۶	۰/۰۰۰۲۲	۰/۰۰۰۰۰۴		
مجموع	۵۹	۰/۰۰۰۴۲			

جدول ۱۹. نتایج روش کم‌ترین اختلاف قابل توجه فیشر برای مسئله‌های متوسط.

الگوریتم‌ها	اختلاف میانگین‌ها	کمترین اختلاف قابل توجه	اختلاف قابل توجه در سطح ۹۵٪
A۱ و A۲	۰/۰۱۰۴	۰/۰۰۰۷۰	بله
A۱ و B۱	۰/۰۱۷۴	۰/۰۰۰۷۰	بله
A۱ و B۲	۰/۰۰۰۹۹	۰/۰۰۰۷۰	بله
A۲ و B۱	۰/۰۰۲۷۸	۰/۰۰۰۷۰	بله
A۲ و B۲	۰/۰۰۲۰۳	۰/۰۰۰۷۰	بله
B۱ و B۲	۰/۰۰۰۷۵	۰/۰۰۰۷۰	بله

جدول ۲۵. میانگین زمان سپری شده برای حل مسئله‌های متوسط.

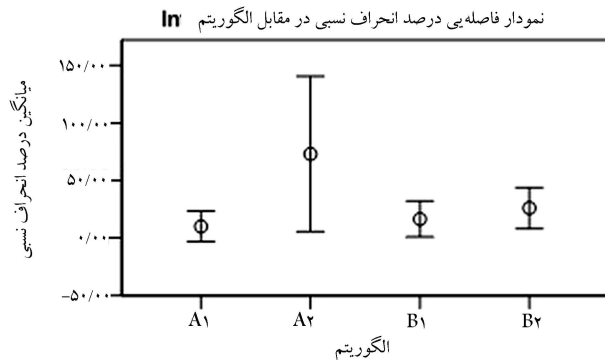
الگوریتم	A ₁	A ₂	B ₁	B ₂
میانگین زمان CPU	۱۰	۷۳	۱۶٫۴	۲۶

جدول ۲۶. نتایج تحلیل واریانس برای مسئله‌های متوسط.

منبع	df	SS	MS	F	P value
الگوریتم	۳	۱۲۲۱۳٫۳۵	۴۰۷۱٫۱۲	۴٫۷۲۲	۰٫۰۱۵
خطا	۵۶	۴۸۲۷۶٫۲	۸۶۲٫۰۷۵		
مجموع	۵۹	۶۰۴۸۹٫۵۵			

جدول ۲۷. نتایج روش کم‌ترین اختلاف قابل توجه فیشر برای مسئله‌های متوسط.

الگوریتم‌ها	اختلاف میانگین‌ها	کمترین اختلاف قابل توجه	اختلاف قابل توجه در سطح ۹۵٪
A ₁ و A ₂	۶۳	۳۲٫۴۲۲۵	بله
A ₁ و B ₁	۶٫۴	۳۲٫۴۲۲۵	خیر
A ₁ و B ₂	۱۶	۳۲٫۴۲۲۵	خیر
A ₂ و B ₁	۵۶٫۶	۳۲٫۴۲۲۵	بله
A ₂ و B ₂	۴۷	۳۲٫۴۲۲۵	بله
B ₁ و B ₂	۹٫۶	۳۲٫۴۲۲۵	خیر



شکل ۱۱. نمودار میانگین و فاصله برای مسئله‌های متوسط.

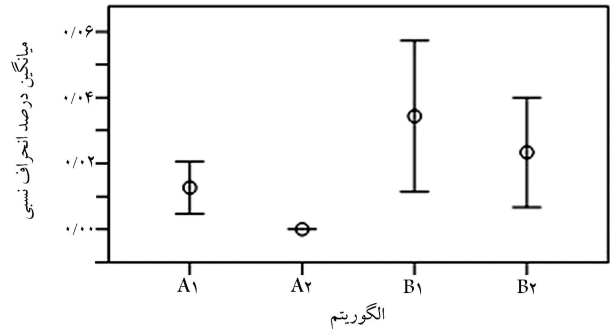
مسئله‌های بزرگ

برای مسئله‌های بزرگ، میانگین زمان سپری شده برای حل مسئله‌های نمونه، نتایج تحلیل واریانس و روش LSD به ترتیب در جداول ۲۸ تا ۳۰ نمایش داده شده است. با توجه به این نتایج، واضح است که الگوریتم‌های A₁ و B₁ با اطمینان ۹۵٪ به الگوریتم‌های A₂ و B₂ ترجیح داده می‌شوند. الگوریتم A₁ دارای نتایج بهتری از B₁ است اما این اختلاف از لحاظ آماری معنی‌دار نیست. نمودار میانگین و فاصله برای مسئله‌های بزرگ را می‌توان در شکل ۱۲ مشاهده کرد.

۵.۵. مقایسه‌ی الگوریتم با الگوریتم‌های موجود در ادبیات

مسئله‌ی مورد بررسی در این تحقیق پیش‌تر مورد مطالعه قرار گرفته^[۳] و توسط یک روش فرآیندکاری برپایه‌ی جست‌وجوی ممنوع حل شده است. برای نشان‌دادن کارایی

نمودار فاصله‌ی درصد انحراف نسبی در مقابل الگوریتم



شکل ۹. نمودار میانگین و فاصله برای مسئله‌های بزرگ.

می‌دهد که اختلاف معنی‌داری بین الگوریتم‌ها وجود ندارد. نمودار میانگین و فاصله برای مسئله‌های کوچک را می‌توان در شکل ۱۰ مشاهده کرد.

مسئله‌های متوسط

برای مسئله‌های متوسط، میانگین زمان سپری شده برای حل مسئله‌های نمونه و نتایج تحلیل واریانس به ترتیب در جداول ۲۵ و ۲۶ نمایش داده شده است. نتایج حاصله نشان‌گر آن است که دست‌کم یک الگوریتم در میانگین پاسخ متفاوت از سایرین است. برای تحلیل بیشتر، از روش LSD استفاده می‌کنیم که نتایج آن در جدول ۲۷ ارائه شده است. نتایج نشان می‌دهد که، الگوریتم‌های A₁ و B₁ از الگوریتم A₂ با اطمینان ۹۵٪ برترند. الگوریتم A₁ دارای نتایج بهتری از B₁ و B₂ است اما این اختلاف‌ها از لحاظ آماری معنی‌دار نیست. نمودار میانگین و فاصله برای مسئله‌های متوسط را می‌توان در شکل ۱۱ مشاهده کرد.

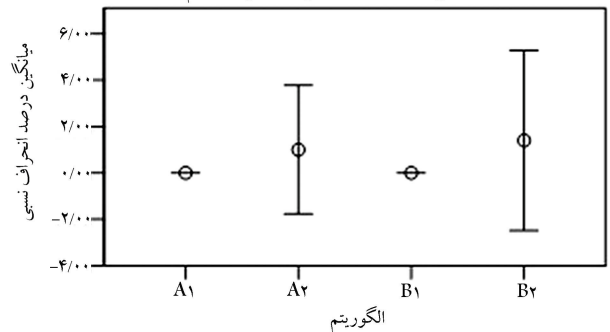
جدول ۲۳. میانگین زمان سپری شده برای حل مسئله‌های کوچک.

الگوریتم	A ₁	A ₂	B ₁	B ₂
میانگین زمان CPU	۰٫۰۰۰۰	۱٫۰۰۰۰	۰٫۰۰۰۰	۱٫۴۰۰

جدول ۲۴. نتایج تحلیل واریانس برای مسئله‌های کوچک.

منبع	df	SS	MS	F	P value
الگوریتم	۳	۷٫۶۰۰	۲٫۵۳۳	۰٫۶۸۵	۰٫۵۷۴
خطا	۵۶	۲۰۷٫۲۰۰	۳٫۷۰۰		
مجموع	۵۹	۲۱۴٫۸۰۰			

نمودار فاصله‌ی درصد انحراف نسبی در مقابل الگوریتم



شکل ۱۰. نمودار میانگین و فاصله برای مسئله‌های کوچک.

جدول ۲۸. میانگین زمان سپری شده برای حل مسئله‌های بزرگ.

الگوریتم	A۱	A۲	B۱	B۲
میانگین زمان CPU	۱۴۵/۶	۴۱۵/۴	۱۸۳/۸	۳۴۱

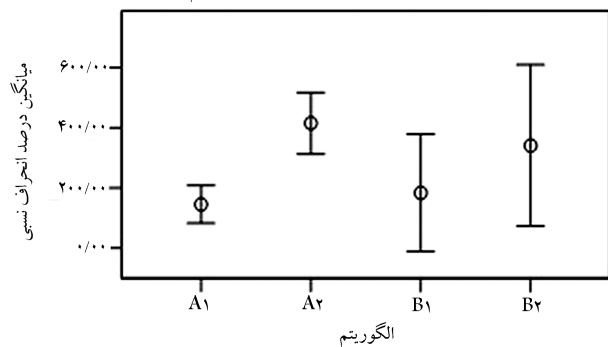
جدول ۲۹. نتایج تحلیل واریانس برای مسئله‌های متوسط.

منبع	df	SS	MS	F	P value
الگوریتم	۳	۲۴۵۳۹۷/۷۵	۸۱۷۹۹/۲۵	۴/۰۵۷	۰/۰۲۵
خطا	۵۶	۱۱۲۸۹۸۵/۲	۲۰۱۶۰/۴۵		
مجموع	۵۹	۱۳۷۴۳۸۲/۹۵			

جدول ۳۰. نتایج روش کم‌ترین اختلاف قابل توجه فیشر برای مسئله‌های بزرگ.

الگوریتم‌ها	اختلاف میانگین‌ها	کمترین اختلاف قابل توجه	اختلاف قابل توجه در سطح ۹۵٪
A۱ و A۲	۲۶۹/۸۰۰۰	۱۵۶/۷۹۲	بله
A۱ و B۱	۳۸/۲۰۰۰	۱۵۶/۷۹۲	خیر
A۱ و B۲	۱۹۵/۴۰۰۰	۱۵۶/۷۹۲	بله
A۲ و B۱	۲۳۱/۶۰۰۰	۱۵۶/۷۹۲	بله
A۲ و B۲	۷۴/۴۰۰۰	۱۵۶/۷۹۲	خیر
B۱ و B۲	۱۵۷/۲۰۰۰	۱۵۶/۷۹۲	بله

نمودار فاصله‌ی درصد انحراف نسبی در مقابل الگوریتم



شکل ۱۲. نمودار میانگین و فاصله برای مسئله‌های بزرگ.

الگوریتم برتر ارائه شده در این تحقیق، ۴۵ مسئله‌ی آزمایشی را که پیش‌تر حل شده [۳] با الگوریتم A۲ نیز حل کرده و نتایج آن‌ها را مقایسه می‌کنیم.

به منظور بررسی عملکرد الگوریتم A۲ نسبت به الگوریتم TS از آزمون فرض نمونه‌های جفت شده (t زوجی) در سه سطح کوچک، متوسط و بزرگ استفاده می‌کنیم. با استفاده از این آزمون می‌خواهیم نشان دهیم که آیا اختلاف مشاهده شده بین جواب‌هایی که با استفاده از دو الگوریتم به دست آمده، معنی‌دار است یا خیر؛ به عبارت دیگر آیا الگوریتم A۲ منجر به تولید جواب‌های بهتری نسبت به الگوریتم TS شده است یا خیر. در آزمون برای میانگین دو جامعه فرض بر استقلال نمونه‌های تصادفی است؛ اگر نمونه‌ها به هم وابسته باشند از آزمون نمونه‌های جفت شده (t زوجی) استفاده می‌کنیم. در این مسئله نمونه‌ها مستقل نیستند زیرا داده‌ها در واقع زوج شده‌اند. بنابراین نمونه‌ی ۱ را جواب‌های به دست آمده از الگوریتم TS و نمونه‌ی ۲ را

جواب‌های به دست آمده از الگوریتم A۲ برای ۴۵ مسئله‌ی فوق در نظر می‌گیریم. می‌خواهیم با استفاده از سطح معنی‌دار بودن ۰/۰۵، فرض صفر برابری میانگین جواب به دست آمده با استفاده از دو الگوریتم را در مقابل فرض کم‌تر بودن میانگین جواب‌های به دست آمده از الگوریتم A۲ نسبت به الگوریتم TS آزمون کنیم.

$$\begin{cases} H_0: \mu_1 = \mu_2 \\ H_1: \mu_1 > \mu_2 \end{cases}$$

فرض می‌کنیم نمونه‌ها به صورت $x_i, y_i, i = 1, 2, \dots, n$ باشد x_i نمونه‌ی به دست آمده از الگوریتم TS و y_i نمونه‌ی حاصل از الگوریتم A۲ است، در این صورت اختلاف بین جفت داده‌ها را به صورت $d_i = x_i - y_i, i = 1, 2, \dots, n$ تعریف می‌کنیم.

۱.۵.۵. مسئله‌های کوچک

فرض صفر را، در صورتی که $t \geq t_{0.05, 14} = 1.761$ رد می‌کنیم. آماره‌ی آزمون به صورت زیر است:

$$\begin{aligned} t_{(n-1)} &= \frac{\bar{d}}{S_d / \sqrt{n}} \\ \bar{d} &= \frac{1}{n} \sum d_i \\ S_d^2 &= \frac{1}{n-1} \sum (d_i - \bar{d})^2 \\ t &= \frac{1.133}{\frac{1.646}{\sqrt{15}}} = 0.508 \end{aligned}$$

چون $t = 1.752 < t_{0.05, 14} = 1.761$ کوچک‌تر از $t_{0.05, 14}$ است، فرض صفر را نمی‌توانیم رد کنیم. در نتیجه تفاوت معنی‌داری بین الگوریتم‌های A۲ و TS وجود ندارد.

۲.۵.۵. مسئله‌های متوسط

هرگاه $t \geq t_{0.05, 14} = 1.761$ باشد فرض صفر را رد می‌کنیم. آماره‌ی آزمون به صورت زیر است:

$$t = \frac{26.933}{\frac{21.595}{\sqrt{15}}} = 4.830$$

چون $4.830 > t_{0.05, 14} = 1.761$ است، فرض صفر را رد می‌کنیم، و نتیجه می‌گیریم که به طور متوسط الگوریتم A۲ کاراتر از الگوریتم TS است.

۳.۵.۵. مسئله‌های بزرگ

هرگاه $t \geq t_{0.05, 14} = 1.761$ باشد، فرض صفر را رد می‌کنیم. آماره‌ی آزمون به صورت زیر است:

$$t = \frac{97.867}{\frac{39.634}{\sqrt{15}}} = 9.563$$

چون $9.563 > t_{0.05, 14} = 1.761$ است، فرض صفر را رد می‌کنیم و نتیجه می‌گیریم که به طور متوسط الگوریتم A۲ دارای کارایی بالاتری نسبت به الگوریتم TS است.

۶.۵. بررسی کارایی الگوریتم پیشنهادی با استفاده از مدل ریاضی

به منظور بررسی کارایی الگوریتم A۲، از بین ۴۵ مسئله‌ی آزمایشی حل شده توسط الگوریتم پیشنهادی قبلی [۳] مسئله‌هایی را که با استفاده از مدل ریاضی قابل حل‌اند

جدول ۳۱. نتایج حاصل از حل مسئله‌های کوچک با مدل ریاضی.

مسئله	**اندازه مسئله	زمان تکمیل کل		
		A2	TS	OS*
۱	$2 \times 3 \times 9$	۴۱۴	۴۱۴	۴۱۴
۲	$2 \times 3 \times 9$	۲۷۳	۲۷۳	۲۷۳
۳	$2 \times 3 \times 9$	۲۷۶	۲۷۶	۲۷۶
۴	$2 \times 4 \times 12$	۴۰۵	۴۰۵	۴۰۵
۵	$2 \times 4 \times 12$	۱۸۵	۱۸۵	۱۸۵
۶	$2 \times 4 \times 12$	۲۲۷	۲۲۷	۲۲۷
۷	$3 \times 4 \times 11$	۴۳۶	۴۳۶	۴۳۶
۸	$3 \times 4 \times 11$	۴۱۵	۴۱۵	۴۱۵
۹	$3 \times 4 \times 11$	۲۷۰	۲۷۶	۲۷۰

* حل بهینه (Optimal solution)

** تعداد مراحل پردازشی \times تعداد گروه‌ها \times مجموع تعداد کارها

تولید جواب اولیه در هر دو رویکرد یکسان است. از نقطه نظر بهره‌وری (سرعت تعیین جواب) نیز همه‌ی سازوکارها در هر دو رویکرد یکسان عمل می‌کنند. با توجه به نتایج محاسباتی به دست آمده، جست‌وجوی خارجی دارای تأثیر بیشتری روی کیفیت جواب به دست آمده است و با محدودکردن جست‌وجوی داخلی و تمرکز بر جست‌وجوی خارجی می‌توان در زمان کم‌تر نتایج قابل قبولی به دست آورد. در جست‌وجوی خارجی در رویکرد اول استفاده از سازوکار تعویض جفتی و در رویکرد دوم استفاده از سازوکارهای حرکت انتقالی و تعویض جفتی دارای نتایج بهتری نسبت به سایر حالت‌هاست. بررسی نتایج محاسباتی به دست آمده از حل مسئله‌های با اندازه‌های مختلف، برتری مسلم کارایی الگوریتم‌های مبتنی بر رویکرد اول را در مقایسه با الگوریتم‌های مبتنی بر رویکرد دوم آشکار می‌سازد.

در این تحقیق همچنین شبیه‌سازی تبرید پیشنهاد شده با شبیه‌سازی تبرید استاندارد مورد مقایسه قرار گرفت. نتایج محاسباتی نشان می‌دهد که کارایی شبیه‌سازی تبرید پیشنهادی بهتر از شبیه‌سازی تبرید استاندارد است. اما بهره‌وری شبیه‌سازی تبرید استاندارد بیشتر از شبیه‌سازی تبرید پیشنهادی است. مقایسه‌ی الگوریتم‌های فرابابتکاری براساس مسئله‌های کوچک، متوسط و بزرگ با توجه به دو معیار بهترین زمان تکمیل و زمان سپری شده‌ی CPU برای به دست آوردن بهترین زمان تکمیل، نشان می‌دهد که A2 و A1 به ترتیب از لحاظ کارایی و بهره‌وری بهترین الگوریتم‌ها هستند. مقایسه‌ی الگوریتم A2 با دیگر الگوریتم ارائه شده برای مسئله‌ی مورد بررسی [۲] نشان می‌دهد که الگوریتم A2 در حل مسئله‌های متوسط و بزرگ کاراتر از الگوریتم لوگندران و همکاران است. از نقطه نظر بهره‌وری، دو الگوریتم قابل مقایسه نیستند زیرا مشخصات رایانه‌هایی که دو الگوریتم روی آن‌ها اجرا شده‌اند متفاوت است.

با آزادکردن فرض‌های محدودکننده می‌توان مسائل جدیدی برای تحقیقات آینده ایجاد کرد. به عنوان مثال، در این تحقیق فرض شده است که ماشین‌های موازی یکسان‌اند و بنابراین، می‌توان ماشین‌های موازی را یکپارچه یا نامرتب در نظر گرفت. یک فرض ساده‌کننده‌ی دیگر در نظر گرفتن زمان‌های آماده‌سازی فقط بین گروه‌هاست، در صورتی که می‌توان زمان‌های آماده‌سازی بین کارهای متعلق به گروه‌ها را نیز در نظر گرفت (زمان آماده‌سازی چندگانه). ارائه‌ی حد پایین برای مسئله‌ی مورد نظر به منظور ارزیابی کیفیت الگوریتم‌های فرابابتکاری ارائه شده برای مسئله را نیز می‌توان به عنوان یک تحقیق برای آینده در نظر گرفت. به علاوه، مسئله‌ی مورد نظر را با تابع هدف کمیته‌سازی مجموع زمان شناوری نیز می‌توان بررسی کرد.

۶. نتیجه‌گیری

در این تحقیق مسئله‌ی زمان‌بندی گروهی وابسته به توالی در محیط جریان کارگاهی انعطاف‌پذیر با تابع هدف کمیته‌سازی زمان تکمیل مورد نیاز برای پردازش کارهای داخل گروه‌ها مورد بررسی قرار گرفت. برای این مسئله یک مدل برنامه‌ریزی خطی عدد صحیح مختلط ارائه شد. پس از اثبات Np-hard بودن مسئله، دو رویکرد مختلف برای تعیین جواب اولیه و همسایگی در نظر گرفته شد. در هر رویکرد از چهار سازوکار مختلف برای پیدا کردن جواب اولیه در سطوح محاسباتی مختلف استفاده شده است. برای بررسی روش‌های مختلف، ۶۰ مسئله‌ی آزمایشی در اندازه‌های کوچک، متوسط و بزرگ تولید شده است. نتایج نشان می‌دهد که کارایی روش‌های

پانویس

- flexible flow shops (FFS)
- flexible flow Line (FFL)
- hybrid flexible flow shop
- batch
- Petrov
- Campbell, Dudek and Smith
- Logendran and Nudtasomboon
- in- process buffers
- constructive heuristics
- insertion-based approach
- immune algorithm
- first in first out
- initial solution (IS)

- cumulative run time (CRT)
- cumulative processing time (CPT)
- longest run time (LRT)
- Greedy algorithm
- inside candidate list
- outside candidate list
- best makespsn

منابع

- Logendran, R.; Carson, S. and Hanson, E. "Group scheduling in flexible flow shops", *International Journal of Production Economics*, **96**, pp. 143-155 (2005).

2. Zandieh, M. and Fatemi Ghomi, S.M.T. "A framework and a classification scheme for modeling production systems", *Proceedings of the Second National Industrial Engineering Conference*, Yazd University, Yazd, Iran, pp 308-315 (2003).
3. Logendran, R.; deSzoeko, P. and Barnard, F. "Sequence-dependent group scheduling problems in flexible flow shops", *International Journal of Production Economics*, **102**, pp. 66-86 (2006).
4. Agnetis, A.; Pacifici, A.; Rossi, F.; Lucertini, M. and Nicoletti, S. and Nicolò, F. "Scheduling of flexible flow lines in an automobile assembly plant", *European Journal of Operational Research*, **97**, pp. 348-62 (1997).
5. Karacapilidis, N.I. and Pappis, C.P. "Production planning and control in textile industry: A case study", *Computers in Industry*, **30**, pp. 127-44 (1996).
6. Alisantoso, D.; Khoo, L.P. and Jiang, P.Y. "An immune algorithm approach to the scheduling of a flexible PCB flow shop", *International Journal of Advanced Manufacturing Technology*, **22**, pp. 819-27 (2003).
7. Hsieh, J.C.; Chang, P.C. and Hsu, L.C. "Scheduling of drilling operations in printed circuit board factory", *Computers & Industrial Engineering*, **44**, pp. 461-73 (2003).
8. Heragu, S., *Facilities Design*, PWS, Boston (1997).
9. Liaee, M.M. and Emmons, H. "Scheduling families of jobs with setup times", *International Journal of Production Economics*, **51**, pp. 165-176 (1997).
10. Nakamura, N.; Yoshida, T. and Hitomi, K. "Group production scheduling for minimum total tardiness: part I", *AIIE Transactions*, **10**, pp. 157-162 (1978).
11. Ozden, M.; Egbelu, P.J. and Iyer, A.V. "Job scheduling in a group technology environment for a single facility", *Journal of Computer Industrial Engineering*, **9**, pp. 67-72 (1985).
12. Baker, K.R. "Scheduling the production of components at a common facility", *IIE Trans*, **20**, pp. 32-35 (1988).
13. Webster, S. and Baker, K.R. "Scheduling groups of jobs on a single machine", *Oper Res*, **43**, pp. 692-703 (1995).
14. Ham, I.; Hitomi, K. and Yoshida, T., *Group Technology: Applications to Production Management*, Kluwer Academic, Hingham (1985).
15. Allison, J.D. "Combining Petrov's heuristic and the CDS heuristic in group scheduling problems", *In Proceedings of the 12th Annual Conference on Computers and Industrial Engineering*, Orlando, FL, pp. 457-461 (1990).
16. Petrov, V.A., *Flow Line Group Production Planning*, Business Publications, London (1966).
17. Campbell, H.G.; Dudek, R.A. and Smith, M.L. "A heuristic algorithm for the n job, m machine sequencing problem", *Manage Sci*, **16**, pp. B630-B637 (1970).
18. Logendran, R. and Nudtasomboon, N. "Minimizing the makespan of a group scheduling problem: A new heuristic", *Int J Prod Econ*, **22**, pp. 217-230 (1991).
19. Logendran, R.; Mai, L. and Talkington, D. "Combined heuristics for bi-level group scheduling problems", *International Journal of Production Economics*, **38**, pp. 133-145 (1995).
20. Gupta, J.N.D. and Darrow, W.P. "The two-machine sequence dependent flowshop scheduling problem", *European Journal of Operational Research*, **24**, pp. 439-446 (1986).
21. Salvador, M.S., *A Solution to a Special Case of Flow Shop Scheduling Problems*, Elmaghraby, S.E. (Ed.), Symposium on the Theory of Scheduling and its Applications, Springer, Berlin, pp. 83-91 (1973).
22. Sawik, T.J. "Mixed integer programming for scheduling flexible flow lines with limited intermediate buffers", *Mathematical and Computer Modeling*, **31**, pp. 39-52 (2000).
23. Sawik, T.J. "An exact approach for batch scheduling in flexible flow lines with limited intermediate buffers", *Mathematical and Computer Modeling*, **36**, pp. 461-471 (2002).
24. Kurz, M.E. and Askin, R.G. "Comparing scheduling rules for flexible flow lines", *Int J Prod Econ*, **85**, pp. 371-388 (2003).
25. Kurz, M.E. and Askin, R.G. "Scheduling flexible flow lines with sequence-dependent setup times", *European Journal Operations Research*, **159**(1), pp. 66-82 (2004).
26. Low, C.h. "Simulated annealing heuristic for flow shop scheduling problems with unrelated parallel machines", *Computers & Operations Research*, **32**, pp. 2013-2025 (2005).
27. Jungwattanakit, J.; Reodechaa, M.; Chaovalitwongsea, P. and Werner, F. "A comparison of scheduling algorithms for flexible flow shop problems with unrelated parallel machines, setup times, and dual criteria", *Computers & Operations Research*, **28**(05), pp. 1-23 (2007).
28. Zandieh, M.; Fatemi Ghomi, S.M.T. and Moattar Hussein, S.M. "An immune algorithm approach to hybrid flow shops scheduling with sequence-dependent setup times", *Appl Math Comput*, **180**, pp. 111-127 (2006).
29. Naderi, B.; Zandieh, M. and Fatemi Ghomi, S.M.T. "A study on integrating sequence dependent setup time flexible flow lines and preventive maintenance scheduling", *J Intell Manuf*, **20**, pp. 683-694 (2008).
30. Kia, H.R.; Davoudpour, H. and Zandieh, M. "Scheduling a dynamic flexible flow line with sequence-dependent setup times: A simulation analysis", *International Journal of Production Research*, (48), pp.4019-4042 (2010).
31. Zandieh, M.; Dorri, B. and Khamseh, A.R. "Robust metaheuristics for group scheduling with sequence-dependent setup times in hybrid flexible flow shops", *Int J Adv Manuf Technol*, **43**, pp. 767-778 (2008).
32. Pinedo, M., *Scheduling Theory, Algorithms, and Systems*, Prentice-Hall, Englewood Cliffs, NJ (2008).
33. Salmasi, N.; Logendran, R. and Skandari, M.R. "Total flow time minimization in a flowshop sequence-dependent group scheduling problem", *Computers & Operations Research*, (37), pp. 199-212 (2010).
34. Schaller, J.E.; Gupta, J.N.D. and Vakharia, A.J. "Scheduling a flow line manufacturing cell with sequence dependent family setup times", *European Journal of Operational Research*, **125**, pp. 324-339 (2000).
35. Metropolis, N.; Rosenbluth, A.W.; Rosenbluth, M.N.; Teller, A.H. and Teller, E. "Equation of state calculation by east computing machines", *J Chem Phys*, **21**, pp. 1087-1091 (1953).
36. Kirkpatrick, S.; Gelatt Jr, C.D. and Vecchi, M.P. "Optimization by simulated annealing", *Science*, **220**, pp. 671-80 (1983).



نمودار جریان تابع تولید جواب همسایه.

