

توسعه مسئله زمان‌بندی پروژه چندمهارته با ظرفیت متغیر از منابع محدود در طول زمان و ارائه الگوریتم جستجوی هارمونی برای حل آن

امیرحسین حسینیان^۱، وحید برادران^{۲*}، مهدی بشیری^۳

۱- دانشجوی دکتری، گروه مهندسی صنایع، دانشکده فنی و مهندسی، دانشگاه آزاد اسلامی
واحد تهران شمال، تهران، ایران

۲- استادیار، گروه مهندسی صنایع، دانشکده فنی و مهندسی، دانشگاه آزاد اسلامی واحد
تهران شمال، تهران، ایران

۳- استاد، گروه مهندسی صنایع، دانشکده فنی و مهندسی، دانشگاه شاهد، تهران، ایران

پذیرش: ۱۳۹۷/۳/۸

دریافت: ۱۳۹۶/۱۰/۹

چکیده

مسئله زمان‌بندی پروژه با منابع محدود و چندمهارته یکی از مسائل کاربردی تحقیق در عملیات است. در این نوع مسئله، کارکنان چندمهارته در طول انجام فعالیت‌های پروژه موردنیاز هستند. میزان دسترسی به کارکنان در طول افق برنامه‌ریزی پروژه، به دلایل وجود تعطیلات رسمی، تعطیلات آخر هفته، بیماری یا مرگ ثابت نیست. بنابراین در این مقاله، یک مدل ریاضی برای مسئله زمان‌بندی پروژه چندمهارته پیشنهاد می‌شود که در آن میزان دسترسی به منابع، متغیر و وابسته به زمان است. ارتباط بین فعالیت‌ها در مدل پیشنهادی از نوع تعمیم‌یافته در نظر گرفته شده است. هدف مدل پیشنهادی، کمینه‌سازی زمان تکمیل پروژه است. به منظور حل مدل پیشنهادی که از جمله مسائل NP-Hard است، یک الگوریتم فراابتکاری مبتنی بر روش جستجوی هارمونی (HS) توسعه داده شده است. در جهت افزایش تنوع جواب‌ها و کاهش احتمال گرفتار شدن الگوریتم پیشنهادی در بهینه محلی، دو عملگر تقاطع و

جهش جدید برای این الگوریتم طراحی شده است. کارایی الگوریتم پیشنهادی در حل چند مسئله نمونه، نسبت به دو الگوریتم بهینه‌سازی ازدحام ذرات (PSO) و ژنتیک (GA) مورد مقایسه قرار گرفته است. نتایج، نشان از برتری الگوریتم پیشنهادی از نظر شاخص‌های کیفیت جواب و زمان حل مسئله دارد.

واژگان کلیدی: بهینه‌سازی؛ جستجوی هارمونی؛ زمان‌بندی پروژه؛ منابع چندمهارته.

۱- مقدمه

زمان‌بندی پروژه با منابع محدود^۱ (RCPSP) از جمله مهم‌ترین مسائل در حوزه تحقیق در عملیات است که محدودیت دسترسی به منابع و وجود روابط پیش‌نیازی بین فعالیت‌ها، منجر به افزایش پیچیدگی آن می‌شود. مسئله RCPSP، یک مسئله بهینه‌سازی ترکیبی است که هدف آن زمان‌بندی مجموعه‌ای از فعالیت‌های مرتبط به هم، با توجه به محدودیت دسترسی به منابع و روابط پیش‌نیازی است؛ به طوری که زمان تکمیل پروژه^۲ کمینه گردد [۱]. بلازوویچ و همکاران [۲] نشان دادند که مسئله RCPSP از جمله مسائل NP-Hard است. بدین سبب، روش‌های حل دقیق ممکن است نتوانند در زمان محاسباتی معقول، موفق به یافتن جواب‌های بهینه مسئله شوند [۳]. با وجود پیچیدگی بالای RCPSP، برخی از فرضیات این مسئله در حالت کلاسیک، برای پروژه‌های دنیای واقعی بسیار محدودکننده هستند. به عنوان نمونه، در مسئله RCPSP کلاسیک، فرض بر این است که میزان دسترسی به منابع در هر دوره زمانی ثابت است [۴]؛ این در حالی است که در دنیای واقعی، ظرفیت منابع در طول زمان به دلیل وجود تعطیلات رسمی، تعطیلات آخر هفته، خرابی ماشین‌آلات و تجهیزات، بیماری کارکنان، عملیات نگهداری و تعمیرات، متغیر است [۵]. از این جهت، برخی از محققان مدل کلاسیک مسئله RCPSP را توسعه داده و میزان دسترسی به منابع را به صورت وابسته به زمان در نظر گرفته‌اند. یکی دیگر از مدل‌های توسعه‌یافته RCPSP کلاسیک، مسئله RCPSP چندمهارته^۳ (MSRCPSP) است که در آن هر یک از کارکنان قادر به انجام حداقل یک مهارت هستند و هر فعالیت، نیازمند یک یا چند مهارت است. هدف MSRCPSP، تعیین زمان شروع

1. Resource-constrained project scheduling problem (RCPSP)

2. Make-span

3. Multi-skilled resource-constrained project scheduling problem (MSRCPSP)

فعالیت‌ها و یافتن بهترین ترکیب نیروی کار چندمهارته برای انجام فعالیت‌ها است. پروژه‌های مرتبط با توسعه نرم‌افزار، صنایع شیمیایی، نگهداری و تعمیرات، تحقیق و توسعه از جمله نمونه‌هایی از کاربرد مسئله MSRCPSPP هستند [۶]. در مدل کلاسیک مسئله RCPSPP، فرض بر آن است که تمامی روابط پیش‌نیازی بین فعالیت‌های یک پروژه از نوع «پایان به آغاز»^۱ (FS) است. بر این اساس، یک فعالیت پیش‌نیاز باید تکمیل گردد تا امکان شروع فعالیت پس‌نیاز وجود داشته باشد. این در حالی است که در پروژه‌های دنیای واقعی ممکن است روابط پیش‌نیازی دیگری از جمله «پایان به پایان»^۲ (FF)، «آغاز به آغاز»^۳ (SS) و «آغاز به پایان»^۴ (SF) وجود داشته باشد. در صورتی که در یک مسئله RCPSPP، تمامی روابط پیش‌نیازی ذکر شده میان فعالیت‌ها وجود داشته باشد، روابط پیش‌نیازی را تعمیم یافته^۵ (GPR) می‌نامیم [۷]. در این مقاله، یک مدل ریاضی برای RCPSPP چندمهارته ارائه خواهد شد که در آن سطح دسترسی به منابع متغیر و وابسته به زمان بوده و روابط پیش‌نیازی تعمیم یافته بین فعالیت‌ها وجود دارد. مدل پیشنهادی با عنوان MSRCPSPPTRC-GPR شناخته می‌شود و هدف آن کمینه‌سازی زمان تکمیل پروژه است. مسئله MSRCPSPP کلاسیک، یک مسئله NP-Hard است [۸]. بنابراین، مسئله MSRCPSPPTRC-GPR به دلیل پیچیدگی بالاتر آن در دسته مسائل NP-Hard قرار می‌گیرد. در کنار ارائه مدل ریاضی، ارائه الگوریتمی فراابتکاری مبتنی بر روش جستجوی هارمونی^۶ (HS) برای حل مسئله، از اهداف و جنبه‌های نوآوری این پژوهش است. الگوریتم جستجوی هارمونی، یک روش فراابتکاری کارآمد است که از فرآیند نواختن موسیقی الهام گرفته شده است [۹]. با توجه به ساختار ساده این روش، احتمال همگرایی سریع این الگوریتم بسیار بالا است. با این حال، این امکان وجود دارد که الگوریتم جستجوی هارمونی در دام بهینه محلی گرفتار شود [۱۰]. از این جهت، در ارائه الگوریتم جستجوی هارمونی پیشنهادی (PHS) علاوه بر پیاده‌سازی فرآیند مرسوم جستجوی همسایگی که در حالت کلاسیک این الگوریتم مورد استفاده می‌گیرد،

-
1. Finish to Start (FS)
 2. Finish to Finish (FF)
 3. Start to Start (SS)
 4. Start to Finish (SF)
 5. Generalized precedence relations (GPR)
 6. Harmony search algorithm (HS)

دو عملگر تقاطع^۱ و جهش^۲ جدید برای آن طراحی شده‌است که امکان ایجاد جواب‌های متنوع و باکیفیت را فراهم می‌آورد. همچنین برای الگوریتم پیشنهادی، دو رویکرد جدید برای تولید فهرست‌های امکان‌پذیر فعالیت‌ها و فهرست منابع موردنیاز فعالیت‌ها توسعه داده شده است. به منظور آزمون کارایی الگوریتم پیشنهادی، عملکرد این روش با دو الگوریتم ژنتیک (GA) و بهینه‌سازی ازدحام ذرات (PSO) مقایسه خواهد شد.

۲- پیشینه پژوهش

در این بخش، مطالعات پیشین مربوط به سه زمینه (۱) RCPSP با میزان دسترسی متغیر به منابع، (۲) RCPSP چندمهارته و (۳) مسئله تخصیص کارکنان چندمهارته^۳ (SAP) بررسی می‌شود. مطالعات مرتبط با RCPSP در حالتی که ظرفیت منابع وابسته به زمان است، اندک هستند. بارتوش و همکاران [۱۱] ظرفیت منابع و احتیاجات فعالیت‌ها به منابع را وابسته به زمان در نظر گرفته‌اند. اسپرچر [۱۲] مدلی با در نظر گرفتن میزان دسترسی متغیر به منابع ارائه نمود. بوداکسومریری و کیم [۱۳] مدلی در زمینه RCPSP چندحالتی^۴ ارائه نموده‌اند که در آن، امکان انقطاع در روند انجام فعالیت‌ها، با توجه به تغییر در سطح دسترسی به منابع وجود دارد. هارتمن [۵] یک مدل ریاضی برای RCPSP ارائه داده است که در آن میزان دسترسی به منابع و احتیاج فعالیت‌ها از آن‌ها، وابسته به زمان است. در ارتباط با مسائل RCPSP چندمهارته و تخصیص کارکنان چندمهارته، تحقیقات متعددی انجام شده است که مشخصات پژوهش‌های پیشین در جدول ۱ قابل مشاهده است. در اکثر مطالعات، نوع رابطه پیش‌نیازی بین فعالیت‌ها از نوع «پایان به آغاز» در نظر گرفته شده است. این نوع رابطه، در جدول ۱ با FS نمایش داده شده است. روابط پیش‌نیازی تعمیم‌یافته در جدول ۱ با GPR نشان داده شده است. با بررسی جدول ۱، مشخص است که یکی از شکاف‌های تحقیقاتی، ارائه یک مدل RCPSP چندمهارته با سطح دسترسی متغیر به منابع و در نظر گرفتن روابط پیش‌نیازی تعمیم‌یافته بین فعالیت‌ها است.

1. Crossover
2. Mutation
3. Staff assignment problem (SAP)
4. Multi-mode RCPSP (MRCPSP)

جدول ۱ خلاصه‌ای از مشخصات پژوهش‌های پیشین در زمینه MSRCPS و SAP

روش حل	مدل				تابع هدف	سال	منبع
	دسترسی به منابع		نوع وابستگی				
	متغیر	ثابت	GPR	FS			
TS, SA		✓		✓	تعداد کارهای محول‌شده	۲۰۱۰	[۱۴]
ALNS		✓		✓	زمان تکمیل پروژه	۲۰۱۰	[۱۵]
CP		✓	✓		زمان تکمیل پروژه	۲۰۱۲	[۱۶]
DE		✓		✓	انحراف از زمان مورد انتظار	۲۰۱۳	[۱۷]
GA		✓		✓	زمان تکمیل پروژه؛ هزینه‌ها	۲۰۱۳	[۱۸]
CPLEX		✓		✓	زمان تکمیل پروژه	۲۰۱۳	[۱۹]
SS		✓		✓	زمان تکمیل پروژه	۲۰۱۳	[۲۰]
GA		✓		✓	ارزش فعلی خالص	۲۰۱۴	[۲۱]
TLBO		✓		✓	زمان تکمیل پروژه	۲۰۱۵	[۲۲]
GA, PSO		✓		✓	کل هزینه‌ها	۲۰۱۶	[۲۳]
MOIWO		✓		✓	زمان تکمیل پروژه؛ هزینه تخصیص منابع؛ کیفیت	۲۰۱۶	[۲۴]
NSCS, MOCS, FSCS		✓		✓	کل هزینه‌ها؛ ریسک دوباره‌کاری	۲۰۱۷	[۸]
NSGA-II		✓		✓	زمان تکمیل پروژه؛ هزینه‌ها؛ کارایی	۲۰۱۷	[۲۵]
PHS	✓		✓		زمان تکمیل پروژه	۲۰۱۸	این مقاله

۳- مدل ریاضی MSRCSPTRC-GPR

۳-۱- مفروضات مدل

برای نمایش ساختار پروژه از شبکه‌های گرهی^۱ (AON) شامل یک شبکه جهت‌دار $G(V, A)$ استفاده شده که V (مجموعه گره‌ها) نشان‌دهنده فعالیت‌های پروژه است و A نشانگر مجموعه‌ای از کمان‌ها است. کمان‌ها ایجادکننده روابط پیش‌نیازی بین فعالیت‌ها هستند [۱]. فعالیت‌ها به ترتیب از ۱ تا N شماره‌گذاری شده‌اند. دو فعالیت صفر و $(N+1)$ ، فعالیت‌های مجازی ابتدایی و انتهایی بوده که به ترتیب نشان‌دهنده شروع و پایان پروژه هستند. فعالیت‌های مجازی مدت‌زمان نداشته و نیاز آن‌ها به منابع صفر است. روابط پیش‌نیازی بین فعالیت‌ها از نوع تعمیم‌یافته است. میزان دسترسی به کارکنان در طول زمان متغیر است. برای اجرای هر یک از مهارت‌های موردنیاز، تعداد مشخصی از کارکنان در هر دوره موردنیاز است.

۳-۲- مجموعه‌ها

V : مجموعه فعالیت‌های پروژه $(j, j' = 0, 1, 2, \dots, N+1)$ ؛

A : مجموعه مهارت‌های موردنیاز پروژه $(k, k' = 1, 2, \dots, K)$ ؛

Ω : مجموعه کارکنان $(s, s' = 1, 2, \dots, S)$ ؛

Δ : مجموعه دوره‌های زمانی $(t, t' = 0, 1, 2, \dots, T)$ ؛

A : مجموعه کمان‌های نشان‌دهنده روابط پیش‌نیازی بین فعالیت‌ها؛

U_k : مجموعه فعالیت‌هایی که نیازمند مهارت k هستند؛

L_j : مجموعه مهارت‌های موردنیاز فعالیت j ؛

g_k : مجموعه کارکنانی که قادر به انجام مهارت k هستند.

1. Activity-on-node (AON)

۳-۳- مولفه‌ها

- P_j : مدت زمان فعالیت j ؛
- r_{jk} : تعداد کارکنان موردنیاز برای انجام مهارت k در فعالیت j ؛
- $\delta_{jj'}$: فاصله زمانی (تأخیر زمانی) بین فعالیت‌های j و j' ؛
- R_{kt} : تعداد کارکنان موجود در دوره t که قادر به انجام مهارت k هستند؛
- O_{sk} : برابر یک است، اگر کارگر s قادر به انجام مهارت k باشد.

۳-۴- متغیرهای تصمیم

- X_{jt} : $\left. \begin{array}{l} \text{اگر فعالیت } j \text{ در دوره } t \text{ شروع شود؛} \\ \text{در غیر این صورت.} \end{array} \right\}$
- H_{jst} : $\left. \begin{array}{l} \text{اگر کارگر } s \text{ شروع به انجام فعالیت } j \text{ در دوره } t \text{ نماید؛} \\ \text{در غیر این صورت.} \end{array} \right\}$
- λ_{jskt} : $\left. \begin{array}{l} \text{اگر کارگر } s \text{ در حال انجام مهارت } k \text{ از فعالیت } j \text{ در دوره } t \text{ باشد؛} \\ \text{در غیر این صورت.} \end{array} \right\}$
- η_{jkt} : $\left. \begin{array}{l} \text{اگر مهارت } k \text{ از فعالیت } j \text{ در دوره } t \text{ در حال انجام باشد؛} \\ \text{در غیر این صورت.} \end{array} \right\}$
- ST_j : زمان شروع فعالیت j ؛
- FT_j : زمان پایان فعالیت j ؛
- ω_{jk} : زمان شروع مهارت k از فعالیت j ؛
- ψ_{jk} : زمان تکمیل مهارت k از فعالیت j ؛
- Z : متغیر تابع هدف.

۳-۵- مدل‌سازی مسئله

$$\text{Min}Z = \sum_{t=1}^T t.X_{(N+1)t} \quad (۱)$$

s.t.

$$\sum_{t=1}^T X_{jt} = 1 \quad ; \quad \forall j \quad (۲)$$

$$r_{jk}.\eta_{jkt} \leq R_{kt} \quad ; \quad \forall j, \forall k \in L_j, \forall t \quad (۳)$$

$$\lambda_{jskt} \leq O_{sk} \quad \forall j, \forall s, \forall k, \forall t \quad (۴)$$

$$\sum_{j=1}^{N+1} \sum_{k=1}^K \lambda_{jskt} \leq 1 \quad ; \quad \forall s, \forall t \quad (۵)$$

$$\eta_{jkt} \leq \sum_{s=1}^S \lambda_{jskt} \quad \forall j, \forall k, \forall t \quad (۶)$$

$$\omega_{jk} = \sum_{t=1}^T t.X_{jt} \quad ; \quad \forall j, \forall k \in L_j \quad (۷)$$

$$ST_j = \omega_{jk} \quad ; \quad \forall j, \forall k \in L_j \quad (۸)$$

$$\omega_{jk} = \omega_{jk'} \quad ; \quad \forall j, \forall (k \neq k') \in L_j \quad (9)$$

$$\psi_{jk} = \omega_{jk} + p_j \quad ; \quad \forall j, \forall k \in L_j \quad (10)$$

$$FT_j = ST_j + p_j \quad ; \quad \forall j \quad (11)$$

$$ST_{(N+1)} \leq T \quad ; \quad (12)$$

$$FT_j \geq \psi_{jk} \quad ; \quad \forall j, \forall k \in L_j \quad (13)$$

$$FT_j + \delta_{jj'} \leq ST_{j'} \quad ; \quad \forall (j, j') \in V; \forall (j, j') \in A_{FS} \quad (14)$$

$$ST_j + \delta_{jj'} \leq FT_{j'} \quad ; \quad \forall (j, j') \in V; \forall (j, j') \in A_{SF} \quad (15)$$

$$ST_j + \delta_{jj'} \leq ST_{j'} \quad ; \quad \forall (j, j') \in V; \forall (j, j') \in A_{SS} \quad (16)$$

$$FT_j + \delta_{jj'} \leq FT_{j'} \quad ; \quad \forall (j, j') \in V; \forall (j, j') \in A_{FF} \quad (17)$$

$$\sum_{s=1}^S \sum_{t=0}^T H_{jst} = \sum_{k=1}^K r_{jk} \quad ; \quad \forall j \quad (18)$$

$$\sum_{t=0}^T H_{jst} \leq 1 \quad ; \quad \forall j, \forall s \quad (19)$$

$$\sum_{j=0}^{N+1} \sum_{t'=t-pj+1}^t H_{jst'} \leq 1 \quad ; \quad \forall s, \forall t \quad (20)$$

$$H_{jst} \leq X_{jt} \quad ; \quad \forall j, \forall s, \forall t \quad (21)$$

$$X_{jt}, H_{jst}, \lambda_{jst}, \eta_{jkt} \in \{0, 1\} \quad ; \quad \forall j, \forall s, \forall k, \forall t \quad (22)$$

$$ST_j, FT_j, \omega_{jk}, \psi_{jk}, Z \geq 0 \quad ; \quad \forall j, \forall k, \forall t \quad (23)$$

رابطه (۱) نشان‌دهنده تابع هدف مدل است که در ارتباط با کمینه‌سازی زمان تکمیل پروژه است. محدودیت (۲) اطمینان می‌دهد که هر فعالیت دقیقاً یک‌بار آغاز شود. محدودیت (۳) تضمین می‌کند که میزان استفاده از یک منبع، از میزان دسترسی به آن بیشتر نشود. محدودیت (۴) نشان‌دهنده این است که نیروی کاری که به مهارت k از فعالیت j اختصاص یافته است، قادر به انجام این مهارت باشد. بر اساس محدودیت (۵)، هر نیروی انسانی تنها مجاز به انجام یک فعالیت در هر دوره است. ارتباط منطقی بین دو متغیر λ_{jst} و η_{jkt} در محدودیت (۶) نشان داده شده است. زمان شروع هر مهارت از طریق رابطه (۷) محاسبه می‌شود. رابطه (۸) این مفهوم را می‌رساند که زمان شروع هر فعالیت، برابر با زمان شروع مهارت‌های موردنیاز آن است. رابطه (۹) نشانگر این است که مهارت‌های یک فعالیت به صورت همزمان با هم شروع می‌شوند. زمان تکمیل مهارت‌های یک فعالیت، با استفاده از رابطه (۱۰) قابل محاسبه می‌باشند. زمان پایان فعالیت‌ها از طریق رابطه (۱۱) به دست می‌آیند. محدودیت (۱۲) تضمین می‌کند که زمان شروع فعالیت مجازی انتهایی باید کمتر از

زمان مرتبط با افق برنامه ریزی پروژه باشد. محدودیت (۱۳) نشان دهنده رابطه منطقی بین FT_j و ψ_{jk} است. محدودیت (۱۴) در ارتباط با روابط پیش نیازی «پایان به آغاز» است؛ درحالی که محدودیت (۱۵) نشان دهنده روابط پیش نیازی از نوع «آغاز به پایان» است. محدودیت (۱۶) بیانگر روابط «شروع به شروع» بین فعالیت است؛ درحالی که محدودیت (۱۷) مرتبط با روابط پیش نیازی «پایان به پایان» است. محدودیت (۱۸) تضمین می کند که تعداد کارکنانی که برای انجام یک فعالیت به آن تخصیص می یابند، برابر با تعداد نیروهای لازم برای انجام فعالیت است. محدودیت (۱۹) بیان می کند که حداکثر یک زمان شروع برای نیروی کار اختصاص یافته به یک فعالیت، تعیین می شود. بر اساس محدودیت (۲۰)، نیروی کار اختصاص یافته به مهارت های مختلف یک فعالیت باید مهارت ها را بدون وقفه انجام دهند. ارتباط منطقی بین دو متغیر H_{jst} و X_{jt} در محدودیت (۲۱) نشان داده شده است. محدودیت های (۲۲) و (۲۳) نوع متغیرهای تصمیم را در مدل ریاضی تعیین می نماید.

۴- الگوریتم جستجوی هارمونی پیشنهادی (PHS)

الگوریتم جستجوی هارمونی (HS) یکی از کاراترین الگوریتم ها در زمینه بهینه سازی ترکیبی است که از یک فرآیند موسیقایی برای یافتن بهترین هارمونی الهام گرفته شده است [۹]. الگوریتم جستجوی هارمونی از ۴ مرحله اصلی تشکیل شده است [۲۶، ۲۷]:

۱- ایجاد حافظه هارمونی (HM): حافظه هارمونی اولیه متشکل از جواب های تصادفی برای مسئله بهینه سازی است. برای مسئله ای که n بُعد دارد، یک حافظه هارمونی با اندازه HMS را می توان به صورت زیر نمایش داد. بردار $(HMS, \dots, q_n^i, \dots, q_2^i, q_1^i)$ یک جواب امکان پذیر برای مسئله است:

$$HM = \begin{bmatrix} q_1^1 & q_2^1 & \dots & q_n^1 \\ q_1^2 & q_2^2 & \dots & q_n^2 \\ \vdots & \vdots & \dots & \vdots \\ q_1^{HMS} & q_2^{HMS} & \dots & q_n^{HMS} \end{bmatrix} \quad (24)$$

۲- یک جواب جدید $[q'_1, q'_2, \dots, q'_n]$ بر اساس HM ایجاد می‌گردد. هر جزء از این جواب (q'_j) ، با توجه به نرخ انتخاب از حافظه هارمونی ($HMCR$) به دست می‌آید. نرخ انتخاب از حافظه هارمونی، به معنای انتخاب یک جزء از اعضای فعلی حافظه هارمونی است؛ بنابراین، مقدار $(1-HMCR)$ برابر احتمال تولید q'_j به صورت تصادفی است. در صورتی که q'_j از حافظه هارمونی تولید شود، یک عضو تصادفی از J امین بُعد از حافظه هارمونی انتخاب خواهد شد. مقدار q'_j می‌تواند با توجه به نرخ تنظیم کوک جهش یابد.

۳- حافظه هارمونی به‌روز می‌شود. جواب جدید تولیدشده در مرحله دوم، در صورتی که مقدار برازندگی بهتری نسبت به بدترین عضو در HM داشته باشد، جایگزین آن خواهد شد؛ در غیر این صورت، جواب جدید ایجادشده حذف می‌شود.

۴- مراحل (۲) و (۳) تا جایی که شرط توقف برآورده نشود، ادامه می‌یابند. شرط توقف غالباً حداکثر تعداد تکرار است. مولفه‌های نرخ انتخاب از حافظه هارمونی ($HMCR$) و نرخ تنظیم کوک (PAR)، دو مولفه پایه‌ای الگوریتم جستجوی هارمونی هستند که بر تعیین اجزای یک جواب و سرعت همگرایی الگوریتم اثرگذار هستند. مولفه $HMCR$ میزان استفاده از داده‌های موجود در حافظه هارمونی را مشخص می‌نماید. هر عضو از جواب جدید برای تنظیم یا عدم تنظیم کوک مورد بررسی قرار می‌گیرد. هر جزء از یک جواب با احتمال PAR تحت تنظیم کوک قرار می‌گیرد. این عملیات به صورت زیر انجام می‌شود [۹]:

$$q'_j = \begin{cases} q'_j + rand(0,1) \times B_w & \text{با احتمال } PAR, \text{ داریم;} \\ q'_j & \text{با احتمال } PAR, \text{ داریم;} \end{cases} \quad (25)$$

در رابطه بالا، B_w نشان‌دهنده پهنای باند است. مقدار $rand()$ یک عدد تصادفی است که با توزیع یکنواخت بین صفر و یک تعیین می‌شود.

۴-۱- روند کدگذاری و کدگشایی

در این مقاله، هر جواب به صورت یک بردار $(2 \times N)$ و N نشان‌دهنده تعداد فعالیت‌های پروژه است. به عبارت دیگر، هر سطر از یک جواب شامل N موقعیت (ژن)

بوده و هر موقعیت مرتبط با یک فعالیت است. سطر اول هر جواب، متشکل از مجموعه‌ای از اعداد تصادفی در بازه $[0,1]$ است که باید تبدیل به یک فهرست فعالیت امکان‌پذیر از نظر روابط پیش‌نیازی شود. در این فهرست، هر فعالیت در موقعیتی بعد از فعالیت‌های پیش‌نیاز خود قرار می‌گیرد. سطر دوم نیز همانند سطر نخست از مجموعه‌ای از اعداد تصادفی در بازه $[0,1]$ تشکیل شده است که این سطر باید تبدیل به فهرست منابع موردنیاز فعالیت‌ها شود. به‌منظور ایجاد یک فهرست فعالیت امکان‌پذیر، در گام نخست یک برادر تصادفی در بازه $[0,1]$ بر اساس قانون بزرگ‌ترین ارزش موقعیت^۱ ایجاد می‌شود. بر طبق این قانون، مقادیر تصادفی کمتر مربوط به فعالیت‌هایی است که زودتر اجرا می‌شوند. این در حالی است که مقادیر بزرگ‌تر، مربوط به فعالیت‌هایی است که دیرتر انجام می‌شوند [۲۸].

در جهت تولید فهرست امکان‌پذیر منابع، چند مجموعه تحت عنوان CW ^۲ برای هر فعالیت در نظر گرفته می‌شود. مجموعه CW_{jk} نشان‌دهنده مجموعه کارکنانی است که قادر به انجام مهارت k از فعالیت j هستند. تعداد مجموعه‌هایی که برای هر فعالیت لحاظ می‌شود، برابر تعداد مهارت‌های موردنیاز آن فعالیت است. با فرض آن‌که فعالیت «۵» نیازمند دو مهارت «۱» و «۳» است، مجموعه‌های CW_{51} و CW_{53} به ترتیب نشان‌دهنده مجموعه کارکنانی هستند که توانایی انجام مهارت‌های «۱» و «۳» از فعالیت «۵» را دارند. فرض کنید این دو مجموعه به‌صورت $CW_{51} = \{1, 2, 4, 6, 8, 10, 12\}$ و $CW_{53} = \{1, 5, 7, 9, 10, 11, 12\}$ باشند. تخصیص کارکنان موردنیاز یک فعالیت در NRS ^۳ گام صورت می‌پذیرد. تعداد گام‌ها برای یک فعالیت، برابر تعداد کارکنان موردنیاز برای انجام تمامی مهارت‌های موردنیاز آن فعالیت است. در هر گام، یک کارگر به یک فعالیت تخصیص می‌یابد. روند تخصیص یک کارگر به مهارت k از فعالیت j به ترتیب زیر است:

۱. عدد تصادفی مرتبط با مهارت k در اندازه (تعداد اعداد) مجموعه CW_{jk} ضرب می‌شود.

1. Largest position value rule
 2. Capable Workforces (CW)
 3. Number of required steps (NRS)

۲. عدد به دست آمده در گام اول، به سمت بالا گرد می‌شود. عدد حاصل تحت عنوان U شناخته می‌شود.

۳. با توجه به بند دوم، U اُمین عضو از مجموعه CW_{jk} به فعالیت j تخصیص می‌یابد.

۴. U اُمین عضو از تمامی مجموعه‌های CW حذف خواهد شد. حذف نمودن U اُمین عضو از تمامی مجموعه‌های CW موجب می‌شود که هر کارگر تنها به یک مهارت از یک فعالیت تخصیص یابد. جدول ۲، جزئیات محاسبات لازم برای تولید فهرست منابع امکان‌پذیر را برای فعالیت ۵ نشان می‌دهد. در دو گام اول، کارگر شماره ۱۲ و ۱ برای انجام مهارت نوع ۱ در این فعالیت انتخاب شده و در یک گام دیگر، کارگر شماره ۱۰ برای انجام مهارت ۳ از این فعالیت تخصیص یافته است.

جدول ۲ محاسبات لازم برای تولید یک فهرست منابع امکان‌پذیر برای فعالیت ۵

مهارت	گام	CW_{jk}	محاسبات	کارگر تخصیص‌یافته	توضیحات
۱	۱	$CW_{\Delta_1} = \{1, 2, 4, 6, 8, 10, 12\}$ $CW_{\Delta_3} = \{1, 5, 7, 9, 10, 11, 12\}$	$\lceil 7 \times 0.9058 \rceil = 7$	"۱۲"	فرد «۱۲» از CW_{Δ_1} و CW_{Δ_3} حذف می‌شود
	۲	$CW_{\Delta_1} = \{1, 2, 4, 6, 8, 10\}$ $CW_{\Delta_3} = \{1, 5, 7, 9, 10, 11\}$	$\lceil 6 \times 0.1270 \rceil = 1$	"۱"	فرد «۱» از CW_{Δ_1} و CW_{Δ_3} حذف می‌شود
۳	۳	$CW_{\Delta_1} = \{2, 4, 6, 8, 10\}$ $CW_{\Delta_3} = \{5, 7, 9, 10, 11\}$	$\lceil 5 \times 0.6324 \rceil = 4$	"۱۰"	-

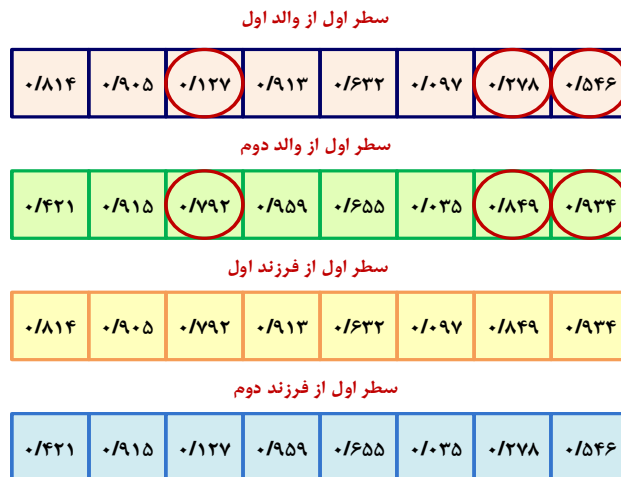
پس از تولید جواب‌ها، نیاز است که هر جواب کُدگشایی شود تا زمان تکمیل پروژه به دست آید. در این مقاله، از روش برنامه تولید زمان بندی سری^۱ (S-SGS) [۵] که یک روش ابتکاری پرکاربرد در مبانی نظری موضوع است، برای کُدگشایی جواب‌ها استفاده شده است.

۲-۴- عملگر تقاطع پیشنهادی

به منظور استفاده از عملگر تقاطع پیشنهادی، دو جواب به صورت تصادفی از میان جواب‌های موجود در حافظه هارمونی انتخاب می‌شوند. جواب‌هایی که به عنوان والد انتخاب شده‌اند، به ترتیب با P_1 و P_2 نمایش داده می‌شوند. عملگر تقاطع، تعداد RAN عدد صحیح تصادفی در بازه $[1, N]$ تولید می‌کند. RAN برابر تعداد موقعیت‌هایی است که عدد تصادفی مربوط به آن‌ها در هر والد باید تغییر نماید. در ادامه، تعداد RAN عدد تصادفی صحیح و غیرتکراری در بازه $[1, N]$ تولید می‌شود تا موقعیت‌هایی که اعداد تصادفی مربوط به آن‌ها باید تغییر کنند، مشخص شود. با فرض آنکه تعداد ۸ فعالیت در پروژه وجود دارد، برای $RAN = 3$ ، اعداد تصادفی سه موقعیت باید تغییر کنند. این سه موقعیت در بازه $[1, 8]$ تعیین می‌شوند. با توجه به آنکه ساختار نمایش جواب در این مقاله دو سطری است، عملگر تقاطع طی دو مرحله جواب‌های جدید را تولید می‌کند. جواب‌های جدید ایجاد شده توسط عملگر تقاطع با CH_1 و CH_2 نمایش داده می‌شوند.

مرحله اول: برای ایجاد سطر اول در CH_1 ، اعداد تصادفی موجود در موقعیت‌های انتخاب شده از P_1 به موقعیت‌های متناظر با خود در P_2 منتقل خواهند شد. برای تولید سطر اول CH_2 ، اعداد تصادفی موجود در موقعیت‌های انتخاب شده از P_2 ، به موقعیت‌های متناظر با خود در P_1 منتقل می‌شوند. با فرض آنکه $RAN = 3$ است، سه موقعیت سوم، هفتم و هشتم به صورت تصادفی انتخاب شده‌اند تا اعداد تصادفی مرتبط با آن‌ها تغییر نماید. شکل ۱ نمونه‌ای از نحوه ایجاد سطرهای اول CH_1 و CH_2 را نشان می‌دهد.

1. Serial schedule generation scheme (S-SGS)



شکل ۱ نحوه ایجاد سطرهای اول CH_1 و CH_2

مرحله دوم: تعداد RAN فعالیت انتخاب خواهند شد تا اعداد تصادفی مربوط به آن‌ها در سطر دوم تغییر نمایند. برای ایجاد سطر دوم در CH_1 ، اعداد تصادفی مربوط به فعالیت‌های انتخاب‌شده در P_1 ، جای خود را به اعداد تصادفی موجود در موقعیت‌های مربوط به این فعالیت‌ها در P_2 می‌دهند. برای تولید سطر دوم CH_2 ، اعداد تصادفی مربوط به فعالیت‌های انتخاب‌شده در P_2 ، جای خود را به اعداد تصادفی موجود در موقعیت‌های مربوط به این فعالیت‌ها در P_1 می‌دهند. با فرض آنکه فعالیت‌های ۳، ۴ و ۶ برای تغییر انتخاب شده‌اند، شکل ۲ روند ایجاد سطرهای دوم برای فرزندان را نشان می‌دهد.

۳-۴- عملگر جهش پیشنهادی

عملگر جهش پیشنهادی با انتخاب یک جواب به صورت تصادفی از حافظه هارمونی شروع به کار می‌کند. پس از آن، یک عدد تصادفی صحیح در بازه $[1, N]$ تولید می‌شود تا تعداد موقعیت‌هایی که باید اعداد تصادفی مربوط به آن‌ها تغییر کنند، مشخص شود. پس از تعیین موقعیت‌ها، اعداد تصادفی در بازه $[0, 1]$ به تعداد موردنیاز تولید می‌شود و جایگزین اعداد فعلی در موقعیت‌های انتخاب‌شده خواهند شد. شکل ۳ جوابی را نشان می‌دهد که به تصادف برای انجام جهش انتخاب شده است؛ با فرض

آنکه موقعیت های ۲، ۵ و ۸ برای تغییر انتخاب شده اند. جواب جهش یافته در شکل ۳ قابل مشاهده است.

۲	۳	۵	۶	۱	۸	۴	۷	سطر اول از فرزند اول
۰/۶۷	۰/۷۵, (۰/۳۹, ۰/۶۵)	(۰/۳۱, ۰/۸۲), ۰/۰۳	(۰/۶۴, ۰/۱۸)	۰/۱۶, ۰/۱۱	(۰/۱۳, ۰/۱۴), ۰/۸۹	۰/۵۲	۰/۰۱	سطر دوم از والد اول
۱	۴	۲	۵	۳	۶	۷	۸	سطر اول از فرزند دوم
۰/۵۰, ۰/۴۹	۰/۳۱	۰/۰۹	(۰/۴۳, ۰/۷۹), ۰/۰۴	۰/۱۷, (۰/۷۰, ۰/۰۳)	(۰/۲۳, ۰/۲۵)	۰/۰۵	(۰/۹۵, ۰/۱۹), ۰/۶۱	سطر دوم از والد دوم
۲	۳	۵	۶	۱	۸	۴	۷	سطر دوم از فرزند اول
۰/۶۷	۰/۱۷, (۰/۷۰, ۰/۰۳)	(۰/۳۱, ۰/۸۲), ۰/۰۳	(۰/۲۳, ۰/۲۵)	۰/۱۶, ۰/۱۱	(۰/۱۳, ۰/۱۴), ۰/۸۹	۰/۳۱	۰/۰۱	سطر دوم از فرزند اول
۱	۴	۲	۵	۳	۶	۷	۸	سطر دوم از فرزند دوم
۰/۵۰, ۰/۴۹	۰/۵۲	۰/۰۹	(۰/۴۳, ۰/۷۹), ۰/۰۴	۰/۱۷, (۰/۳۹, ۰/۶۵)	(۰/۶۴, ۰/۱۸)	۰/۰۵	(۰/۹۵, ۰/۱۹), ۰/۶۱	سطر دوم از فرزند دوم

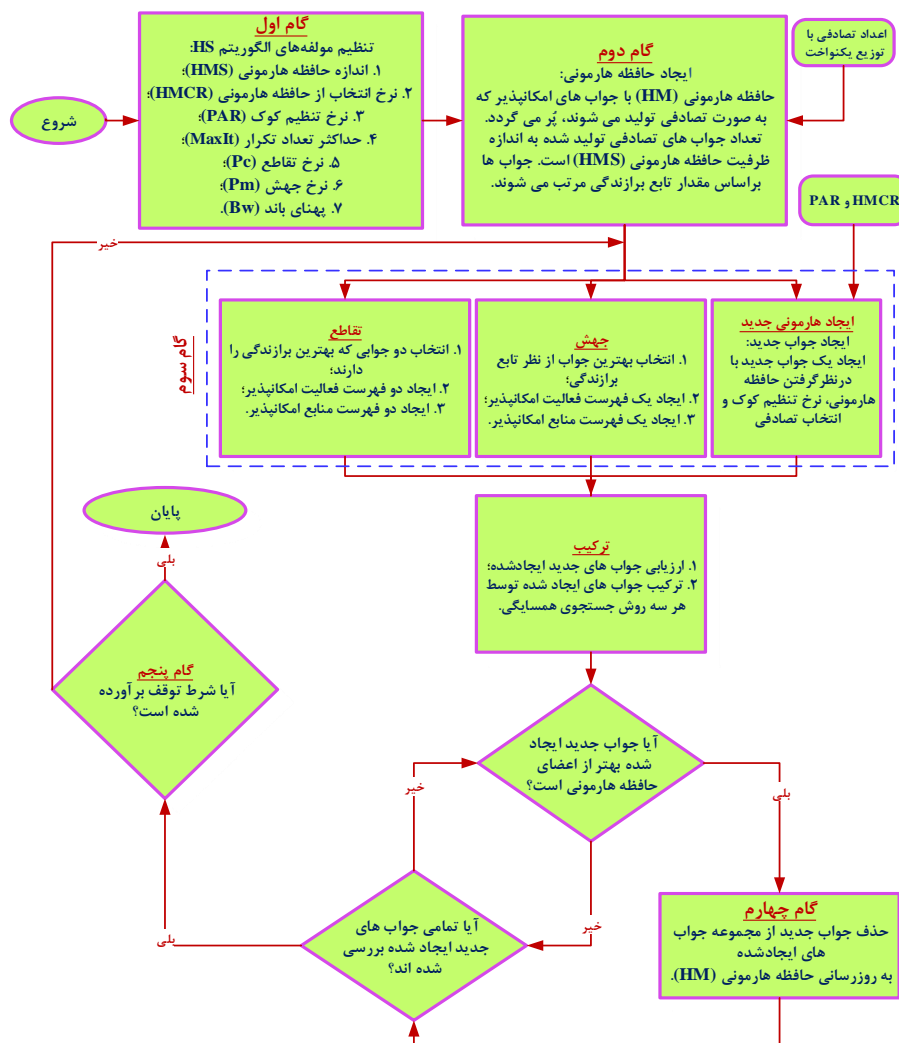
شکل ۲ نحوه ایجاد سطرهای دوم CH_1 و CH_2

سطر اول جواب انتخاب شده	۰/۸۱	۰/۹۰	۰/۱۲	۰/۶۳	۰/۰۹	۰/۲۷	۰/۵۴	۰/۹۵
سطر دوم جواب انتخاب شده	(۰/۱۵, ۰/۴۸)	۰/۸۰	۰/۱۴	۰/۴۲	(۰/۹۱, ۰/۷۹)	۰/۹۵	(۰/۸۴, ۰/۰۳), ۰/۷۶,	۰/۶۹
سطر اول جواب جهش یافته	۰/۸۱	۰/۴۳	۰/۱۲	۰/۶۳	۰/۷۶	۰/۲۷	۰/۵۴	۰/۳۴
سطر دوم جواب جهش یافته	(۰/۱۵, ۰/۴۸)	۰/۳۸	۰/۱۴	۰/۴۲	۰/۴۴, ۰/۶۴	۰/۹۵	(۰/۸۴, ۰/۰۳), ۰/۷۶,	۰/۲۷

شکل ۳ نحوه عملکرد عملگر جهش پیشنهادی

۴-۴- شرط توقف

در این مطالعه، حداکثر تعداد تکرار به‌عنوان شرط توقف الگوریتم پیشنهادی در نظر گرفته شده است. شکل ۴ ساختار الگوریتم PHS را نشان می‌دهد.



شکل ۴ فلوجارت الگوریتم پیشنهادی

۵- یافته‌های پژوهش

۵-۱- تنظیم مولفه‌های مدل پیشنهادی

برای ارزیابی مدل پیشنهادی، تعداد ۳۰ مسئله استاندارد به تصادف از پایگاه مسائل نمونه زمان‌بندی پروژه^۱ (PSPLIB) شامل دو دسته مسائل کوچک (مسئله ۱ تا ۱۵) و بزرگ (مسئله ۱۶ تا ۳۰) انتخاب شده‌اند. این مسائل شامل نمونه‌های ۳۲ فعالیتی و ۱۲۲ فعالیتی هستند. از آنجایی که برای مدل پیشنهادی در این مقاله، تاکنون مسئله استاندارد به خصوصی ارائه نشده است، برخی مولفه‌های موردنیاز مدل در بند ۳-۳ به صورت تصادفی تولید شده‌اند.

۵-۲- تنظیم مولفه‌های الگوریتم‌ها

در این مطالعه، از روش تاگوچی برای تنظیم مولفه‌های ورودی الگوریتم‌ها استفاده شده است. ابتدا سه سطح مطابق جدول ۳ برای مولفه‌های هر سه الگوریتم فراابتکاری (پیشنهادی، GA و PSO) در نظر گرفته شده و سه مسئله نمونه با ۱۲۲ فعالیت به‌ازای هر سطح حل شده است. هر مسئله ۱۰ مرتبه اجرا شده تا نتایج قابل‌اعتمادی به دست آید. در نتیجه برای هر آزمایش، تعداد ۳۰ خروجی حاصل خواهد شد. بهترین جواب در میان ۱۰ مرتبه اجرا، به‌عنوان نتیجه هر مسئله در نظر گرفته شده است. زمان تکمیل پروژه برای هر آزمایش باید تبدیل به شاخص «درصد انحراف نسبی»^۲ (RPD) شود. مقدار RPD با استفاده از رابطه زیر، قابل‌محاسبه است:

$$RPD = \frac{Method_{C_{max}} - Best_{C_{max}}}{Best_{C_{max}}} \times 100 \quad (26)$$

در رابطه بالا، $Method_{C_{max}}$ نشان‌دهنده زمان تکمیل حاصل‌شده توسط روش حل مسئله است. $Best_{C_{max}}$ برابر بهترین زمان تکمیل به‌دست‌آمده در میان تمامی مقادیر است. در مرحله بعدی، میانگین درصد انحراف نسبی برای هر سه مسئله نمونه محاسبه می‌شود. در ادامه با استفاده از رویکرد تاگوچی، سطوح مطلوب

1. Project scheduling problem library (PSPLIB)
2. Relative percentage deviation (RPD)

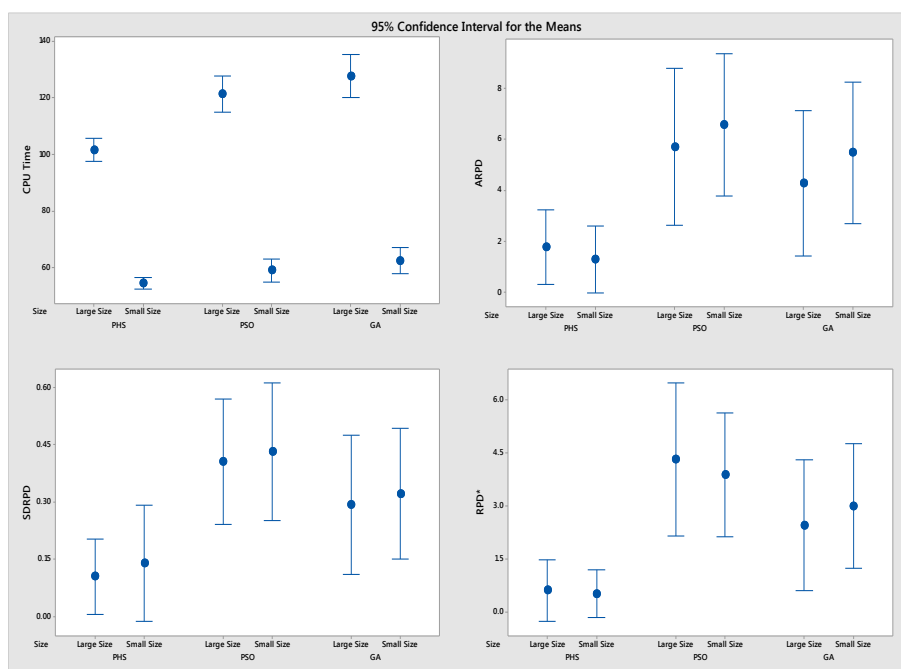
مولفه‌های هر سه الگوریتم مطابق ستون آخر جدول ۳ تعیین شده است. هر سه الگوریتم در محیط نرم‌افزار Matlab R2015b برنامه‌نویسی شده‌اند و اجرای برنامه آن‌ها توسط یک رایانه شخصی با مشخصات (4- 2.33 GHz Intel Quad Core CPU, GB RAM) صورت پذیرفته است.

جدول ۳ سطوح مختلف برای مولفه‌های ورودی الگوریتم‌ها

الگوریتم‌ها	مولفه‌ها	نمادها	سطوح مولفه‌ها			مقدار بهینه
			سطح ۱	سطح ۲	سطح ۳	
GA	نرخ تقاطع	pc	۰/۸۰	۰/۸۵	۰/۹۰	۰/۸۰
	نرخ جهش	pm	۰/۰۵	۰/۱۰	۰/۱۵	۰/۱۰
	اندازه جمعیت	$Npop$	۵۰	۱۰۰	۱۵۰	۱۵۰
	تعداد تکرار	$MaxIt$	۱۰۰	۱۵۰	۲۰۰	۲۰۰
PSO	اندازه جمعیت	$Npop$	۵۰	۱۰۰	۱۵۰	۱۵۰
	تعداد تکرار	$MaxIt$	۱۰۰	۱۵۰	۲۰۰	۲۰۰
	ضریب اجتماعی	$C1$	۱	۱/۵	۲	۱
	ضریب ادراک	$C2$	۱	۱/۵	۲	۱
	وزن اینرسی	INW	۰/۷۵	۰/۸۰	۰/۸۵	۰/۸
PHS	اندازه حافظه هارمونی	HMS	۵۰	۱۰۰	۱۵۰	۱۵۰
	تعداد تکرار	$MaxIt$	۱۰۰	۱۵۰	۲۰۰	۲۰۰
	نرخ انتخاب از حافظه هارمونی	$HMCR$	۰/۷۵	۰/۸۵	۰/۹۵	۰/۸۵
	نرخ تنظیم کوک	PAR	۰/۱۰	۰/۳۰	۰/۵۰	۰/۱۰
	نرخ تقاطع	pc	۰/۸۰	۰/۸۵	۰/۹۰	۰/۸۰
	نرخ جهش	pm	۰/۰۵	۰/۱۰	۰/۱۵	۰/۱۰
	پهنای باند	B_w	۰/۱۰	۰/۱۵	۰/۲۰	۰/۱۰

۳-۵- مقایسه عملکرد الگوریتم‌ها

در این بخش، عملکرد الگوریتم PHS با عملکرد الگوریتم‌های بهینه‌سازی ازدحام ذرات (PSO) و ژنتیک (GA) [۳۱،۳۰،۲۹] مورد مقایسه قرار می‌گیرد. عملکرد الگوریتم‌ها بر اساس چند معیار سنجش عملکرد مورد مقایسه قرار می‌گیرند. این معیارها عبارت‌اند از: ۱- میانگین RPD (ARPD)، ۲- انحراف متوسط RPD (SDRPD)، ۳- بهترین مقدار RPD به دست آمده توسط روش حل (RPD*)، و ۴- زمان محاسبات (CPU time). هرچه مقادیر این شاخص‌ها کمتر باشد، نشان‌دهنده عملکرد بهتر الگوریتم است. هر مسئله نمونه، ۵ بار توسط هر الگوریتم حل شده است و شاخص‌های بالا برای آن‌ها محاسبه شده است. شکل ۵ فواصل اطمینان و میانگین شاخص‌های SDRPD، RPD*، ARPd و زمان محاسبات هر الگوریتم را به تفکیک مسائل کوچک و بزرگ نشان می‌دهد.

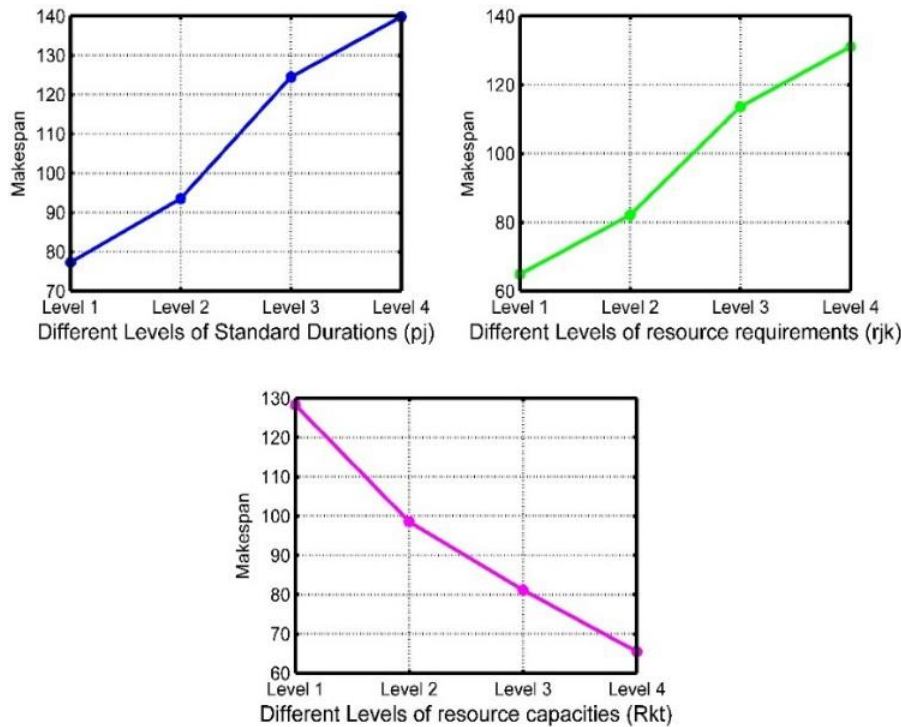


شکل ۵ فواصل اطمینان شاخص‌های عملکردی الگوریتم‌ها به تفکیک مسائل کوچک و بزرگ

همان‌طور که در تمامی نمودارهای شکل ۵ مشخص است، میانگین شاخص‌های عملکردی (ARPD، SDRPD و RPD*) الگوریتم پیشنهادی (PHS) از دو الگوریتم دیگر به‌طور معنی‌داری برای هر دو دسته مسائل کوچک و بزرگ، کمتر (بهتر) است. همچنین کیفیت جواب مسائل کوچک و بزرگ برای هر الگوریتم تقریباً مشابه است. از طرفی، زمان حل مسئله با الگوریتم پیشنهادی به مراتب از دو الگوریتم دیگر برای هر دو گروه مسائل کوچک و بزرگ پایین‌تر است. مسائل با ابعاد کوچک توسط نرم‌افزار GAMS نسخه ۲۴/۱/۲ حل شده‌اند و جواب بهینه مربوط به مسائل با ابعاد کوچک به دست آمده‌اند. مقایسه جواب‌های بهینه با جواب‌های الگوریتم پیشنهادی نشان می‌دهد که الگوریتم پیشنهادی در ۶۷ درصد از نمونه‌های آزمایشی توانسته است در زمان محاسباتی معقولی، به جواب بهینه دست یابد.

۴-۵- اعتبارسنجی مدل

در این بخش، صحت مدل پیشنهادی با انجام تحلیل حساسیت موردسنجش قرار می‌گیرد. به این ترتیب، مقادیر مولفه‌های ورودی تغییر می‌یابند تا تأثیر این تغییرات بر زمان تکمیل پروژه مشخص شود. مهم‌ترین این مولفه‌ها شامل مدت زمان استاندارد فعالیت‌ها (p_j)، میزان احتیاجات فعالیت‌ها به منابع (r_{jk}) و میزان دسترسی به منابع (R_{kt}) است. به منظور انجام تحلیل حساسیت، برای هر مولفه ۴ سطح در نظر گرفته می‌شود. مقادیر مولفه‌ها از سطح اول تا سطح چهارم افزایش می‌یابند. شکل ۶ اثر مقادیر مختلف مولفه‌ها را بر زمان تکمیل پروژه نشان می‌دهد. با توجه به شکل ۶، افزایش مولفه‌های مدت زمان فعالیت‌ها (p_j) و تعداد منابع موردنیاز (r_{jk}) منجر به افزایش قابل توجه زمان تکمیل پروژه می‌گردد. از طرف دیگر، افزایش مقدار دسترسی به منابع (R_{kt}) منجر به کاهش معنی‌دار زمان موردنیاز برای تکمیل پروژه خواهد شد.



شکل ۶ اثر مولفه‌های مدل بر زمان تکمیل پروژه

۶- نتیجه‌گیری و پیشنهادها

در این مقاله، یک مدل ریاضی برای مسئله زمان‌بندی پروژه چندمهارته با ظرفیت متغیر از منابع محدود و روابط پیش‌نیازی تعمیم‌یافته بین فعالیت‌ها (MSRCSPTRC-GPR) ارائه شد. با توجه به پیچیدگی بالای مسئله، یک الگوریتم جستجوی هارمونی تحت عنوان PHS برای حل مسئله پژوهش توسعه داده شد. در الگوریتم پیشنهادی، دو رویکرد جدید برای تولید فهرست امکان‌پذیر فعالیت‌ها و تعیین ترکیب کارکنانی که به فعالیت‌ها تخصیص می‌یابند، ارائه شده است. جهت یافتن جواب‌های متنوع و باکیفیت، دو عملگر تقاطع و جهش جدید نیز برای PHS پیشنهاد شده است. این دو عملگر توانسته‌اند روند جستجوی همسایگی را در الگوریتم PHS تقویت نمایند. عملکرد روش PHS نسبت به دو الگوریتم فراابتکاری PSO و GA در ۳۰ مسئله استاندارد موردسنجش قرار گرفته است. نتایج، نشان از برتری الگوریتم PHS از نظر

اکثر معیارهای سنجش کیفیت جواب و زمان حل مسئله دارد. همچنین مقایسه جواب‌های به‌دست‌آمده توسط PHS، PSO و GA در ۱۵ مسئله نمونه‌ای با جواب‌های بهینه‌ای که توسط نرم‌افزار GAMS به‌دست‌آمده‌اند، نشان‌دهنده آن است که الگوریتم پیشنهادی در اکثر مسائل موفق به یافتن جواب‌های بهینه شده است. مطالعه مشابه دیگر در ارتباط با در نظر گرفتن امکان انقطاع برای فعالیت‌ها در مدل MSRCPSPTRC-GPR می‌تواند زمینه‌ای برای تحقیقات آتی به‌شمار آید.

۷- منابع

- [1] Hartmann, S., & Briskorn, D., (2010). A survey of variants and extensions of the resource-constrained project scheduling problem, *European Journal of Operational Research*, 207(1), 1-14.
- [2] Blazewicz, J., Lenstra, J.K. & Kan, A., (1983). Scheduling subject to resource constraints: Classification and complexity, *Discrete Applied Mathematics*, 5(1), 11-24.
- [3] Shahbazi, S., Sajjadi, S.M., & Jolai, F., (2018). A Simulation-Based Optimization Model for Determining the Sequence of Implementing Projects Related To New Product Development, *Modern Researches in Decision Making*, 2(4), 129-152.
- [4] Chen, J., Fowler, J., Kempf, K., & Mason, S., (2015). Multi-mode resource-constrained project scheduling Problems with non-preemptive activity splitting, *Computers & Operations Research*, 53(2), 275-287.
- [5] Hartmann, S., (2013). Project scheduling with resource capacities and requests varying with time: a case study, *Flexible Services and Manufacturing Journal*, 25(1), 74-93.
- [6] Néron, E., & Baptista, D., (2002). Heuristics for multi-skill project scheduling problem, International Symposium on combinatorial optimization (CO'2002), Paris, France.
- [7] Ghafoori, S., & Taghizadeh Yazdi, M.R., (2017). Proposing a Multi-Objective Mathematical Model for RCPSP and Solving It with Firefly and Simulated Annealing algorithms, *Modern Researches in Decision Making*, 1(4), 117-142.

- [8] Maghsoudlou, H.M., Nadjafi, B., & Niaki, S.T.A., (2017). Multi-skilled project scheduling with level-dependent rework risk; three multi-objective mechanisms based on cuckoo search, *Applied Soft Computing*, 54, 46-61.
- [9] Lee, K. S. & Geem, Z. W., (2005). A New Meta-Heuristic Algorithm for Continuous Engineering Optimization: Harmony Search Theory and Practice, *Computer Methods in Applied Mechanics and Engineering*, 194, 3902-3933.
- [10] Rastgar, I., & Sahraeian, R., (2013). Developing harmony search algorithm for solving optimization problems: a case study in parallel machine production scheduling problem, *Journal of Industrial Engineering Research in Production Systems*, 1(1), 57-71.
- [11] Bartusch, M., Mohring, R.H., & Radermacher, F.J., (1988). Scheduling project networks with resource constraints and time windows, *Annals of Operations Research*, 16(1), 201-240.
- [12] Sprecher, A., (1994). Resource-constrained project scheduling: exact methods for the multi-mode case, Number 409 in lecture notes in economics and mathematical systems. Springer, Berlin, DOI: 10.1007/978-3-642-48397-4.
- [13] Buddhakulsomsiri, J., & Kim, D.S., (2006). Properties of multi-mode resource-constrained project scheduling problems with resource vacations and activity splitting, *European Journal of Operational Research*, 175(1), 279-295.
- [14] Ho, S., & Leung, J., (2010). Solving a manpower scheduling problem for airline catering using metaheuristics, *European Journal of Operational Research*, 202(3), 903-921.
- [15] Cordeau, J., Laporte, G., Pasin, F., & Ropke, S., (2010). Scheduling Technicians and Tasks in a Telecommunications Company, *Journal of Scheduling*, 13(4), 393-409.
- [16] Liu, S., & Wang, C., (2012). Optimizing linear project scheduling with multi-skilled crews, *Automation in Construction*, 24, 16-23.
- [17] Kazemipoor, H., Tavakkoli-Moghaddam, R., Shahnazari-Shahrezaei, P., & Azaron, A., (2013). A differential evolution algorithm to solve multi-skilled

- project portfolio scheduling problems, *The International Journal of Advanced Manufacturing Technology*, 64(5-8), 1099-1111.
- [18] Myszkowski, P.B., & Skowronski, M., (2013). Specialized genetic operators for multi skill resource-constrained project scheduling problem, 19th International Conference on Soft Computing MENDEL, 57-62.
- [19] Mehmanchi, E., & Shadrokh, S., (2013). Solving a New Mixed Integer Non-Linear Programming Model of the Multi-Skilled Project Scheduling Problem Considering Learning and Forgetting Effect, *Proceedings of the 2013 IEEE IEEM*, Bangkok, Thailand.
- [20] Kazemipoor, H., Tavvakoli-Moghaddam, E., & Sharezaei, P., (2013). Solving a novel multi-skilled project scheduling model by scatter search, *South African Journal of Industrial Engineering*, 24(1), 121-135.
- [21] Tabrizi, B.H., Tavvakoli-Moghaddam, R., & Ghaderi, S.F., (2014). A two-phase method for a multi-skilled project scheduling problem with discounted cash flows, *Scientia Iranica*, 21(3), 1083-1095.
- [22] Zheng, H., Wang, L., & Zheng, X., (2015). Teaching-learning-based optimization algorithm for multiskill resource constrained project scheduling problem, *Soft Computing*, 21(6), 1537-1548.
- [23] Javanmard, S., Nadjafi, B., & Niaki, S.T.A., (2016). Preemptive multi-skilled resource investment project scheduling problem; mathematical modelling and solution approaches, *Computers and Chemical Engineering*, 96(1), 55-68.
- [24] Maghsoudlou, H.M., Nadjafi, B., & Niaki, S.T.A., (2016). A multi-objective invasive weeds optimization algorithm for solving multi-skill multi-mode resource constrained project scheduling problem, *Computers and Chemical Engineering*, 8(1), 157-169.
- [25] Chen, R., Liang, C., Gu, D., and Leung, J., (2017). A multi-objective model for multi-project scheduling and multi-skilled staff assignment for IT product development considering competency evolution, *International Journal of Production Research*, 55(21), 1-29.

- [26] Geem, Z.W., Kim J.H. & Loganathan G.V. (2001). A new heuristic optimization algorithm: harmony search. *Simulations*, 76, 60–68.
- [27] Kalivarapu, J., Jain, S., & Bag, S., (2015). An improved harmony search algorithm with dynamically varying bandwidth, *Engineering Optimization*, 48(7), 1091-1108.
- [28] Wang, L., Pan, Q.K., & Tasgetiren, M.F., (2011). A hybrid harmony search algorithm for the blocking permutation flow shop scheduling problem, *Computers & Industrial Engineering*, 61(1), 76–83.
- [29] Bagher, M., Kassae, M., Allam Tabriz, A., & Zandieh, M., (2018). Truck Scheduling in Distribution Systems with Multiple Cross Docks and No Intermediate Storage, *Modern Researches in Decision Making*, 2(4), 1-27.
- [30] Mahdavi, A.M., (2007). Designing Quality Measurement Model of Information System Services Based on Genetic Algorithm, *Management Research in Iran*, 11(20), 235-263.
- [31] Asadi, R., Kazemzadeh, R., Nakhaei Kamal Abadi, I., & Bagherinejad, Z., (2013). A Joint Ordering Model for Automotive parts in Two-level Supply Chain, *Management Research in Iran*, 17(2), 1-18.