



## An Efficient Hybrid Algorithm for Solving Multi Objective Linear Programming Model for Single Machine Problems

Y. Zare Mehrjerdi\*, S. Fereidouni & L. Emami Maibodi

*Yahia Zare Mehrjerdi*, Associate Professor, Department of Industrial Engineering, Yazd University, Yazd Iran,  
*Sepideh Fereidouni*, Master Students, Department of Industrial Engineering, Yazd University, Yazd Iran, [sepideh.fereidouni@gmail.com](mailto:sepideh.fereidouni@gmail.com)  
*Leila Emami Maibodi*, Master Students, Department of Industrial Engineering, Yazd University, Yazd Iran, [leila\\_em1986@yahoo.com](mailto:leila_em1986@yahoo.com)

### Keywords

Sequence of Single Machine  
Operation,  
Multi Objective Programming,  
Lateness,  
Weighted Tardiness,  
Meta- Heuristic

### ABSTRACT

*Due to the fact that the determination of an efficient scheduling solution in the sequence of single machine operation for multiple objective programming is important, especially in production planning, we are considering a single machine sequencing problem with minimizing the number of the tasks with lateness and weighted tardiness. In this article, the application of new optimization methods in sequencing problem and scheduling are in order. We propose the mathematical model for the problem under consideration first and then by introducing simulated annealing and genetic algorithm, as solution approaches, we test their efficiency for solving the proposed problem. At the end, to increase the efficiency of the proposed model a hybrid/meta-heuristic algorithm based upon the genetic algorithm is proposed. This method identifies a collection of efficient sequencing tools for objectives minimization.*

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

\*  
Corresponding author. Yahia Zare Mehrjerdi  
Email: [yazm2000@yahoo.com](mailto:yazm2000@yahoo.com)



# ارائه یک الگوریتم ترکیبی کارا جهت حل مدل برنامه ریزی خطی چندهدفه زمانبندی مسائل تک ماشین

یحیی زارع مهرجردی<sup>\*</sup>، سپیده فریدونی و لیلی امامی میبیدی

## چکیده:

از آنجا که تعیین برنامه‌های زمانبندی کارا در مسائل توالی عملیات برای معیارهای مختلف، از جمله مسائل مهم در برنامه‌ریزی تولید است، لذا در این مطالعه مساله توالی عملیات تک ماشین با معیارهای حداقل کردن تعداد کارهای دارای تاخیر و مجموع دیرکرد وزنی مورد بررسی قرار می‌گیرد. در این مقاله کاربرد روش‌های جدید بهینه‌سازی در مسائل توالی و زمانبندی مطرح می‌شود. ابتدا مدل ریاضی مساله برای اهداف موردنظر ارائه و سپس ضمن معرفی روش‌های شبیه‌سازی آنیلینگ و الگوریتم ژنتیک به عنوان روش‌های کاوشی، کارایی آنها در مساله موردنظر آزموده شده است. در پایان، جهت افزایش کارایی مدل الگوریتم ترکیبی برمبنای الگوریتم ژنتیک برای مساله ارائه شده است. این روش، مجموعه‌ای از توالی‌های کارا را به منظور حداقل کردن اهداف موردنظر مشخص می‌کند.

## کلمات کلیدی

توالی عملیات تک ماشین  
برنامه‌ریزی چندهدفه  
دیرکرد  
الگوریتم‌های فراابتکاری

## ۱. مقدمه

تعیین توالی و زمانبندی در سیستم‌های پیشرفته تولیدی از اهمیت ویژه‌ای برخوردار است. اقتصادی بودن این سیستم‌ها مشروط به داشتن یک برنامه تعیین توالی<sup>۱</sup> و زمانبندی<sup>۲</sup> مناسب می‌باشد. در واقع زمانبندی نوعی فعالیت تصمیم‌گیری است که با هدف بهینه‌سازی یک و یا چند هدف انجام می‌گیرد. درحالیکه، توالی در مورد ترتیب ورود کارها در سیستم مورد استفاده قرار

می‌گیرد. مشکلات تعیین توالی و زمانبندی قطعات در محیط‌های مختلف صنعتی یکسان نیست. چنانچه عمل کردن به قراردادهای در موعد تحویل موردنظر باشد، باید کمینه کردن تعداد قطعات دیرکردار یا کمینه کردن متوسط دیرکرد، موردتوجه قرار گیرد [۱،۲]. مطالعه جریمه‌های دیرکرد در مدل برنامه‌ریزی زمانبندی به عنوان یک زمینه تحقیقاتی جدید مطرح شده است. قبل از آن، سال‌ها تحقیقات موضوع زمانبندی تنها بر روی یک شاخص اندازه‌گیری باقاعده متمرکز بود [۳]. در این مطالعه، بادر نظر گرفتن مطلب فوق و اهمیت مسایل برنامه‌ریزی زمانبندی از مدل برنامه‌ریزی خطی چندهدفه، جهت حداقل کردن مجموع دیرکرد وزنی و تعداد کارهای دارای تاخیر در مسائل زمانبندی تک ماشین استفاده شده است. در این مقاله، ابتدا به بیان تاریخچه مسئله و جمع‌بندی از تحقیقات و مطالعات انجام شده پرداخته شده است. مشخص کردن اطلاعات ورودی، خروجی و تابع هدف

تاریخ وصول: ۹۰/۲/۱۸

تاریخ تصویب: ۹۰/۵/۳۰

<sup>\*</sup>نویسنده مسئول مقاله: دکتر یحیی زارع مهرجردی، استادیار دانشکده مهندسی صنایع، دانشگاه یزد، [yazm2000@yahoo.com](mailto:yazm2000@yahoo.com)

سپیده فریدونی، دانشجوی دوره کارشناسی ارشد، دانشکده مهندسی صنایع، دانشگاه یزد، [sepideh.fereidouni@gmail.com](mailto:sepideh.fereidouni@gmail.com)

لیلی امامی میبیدی، دانشجوی دوره کارشناسی ارشد، دانشکده مهندسی صنایع، دانشگاه یزد، [leila\\_em1986@yahoo.com](mailto:leila_em1986@yahoo.com)

<sup>۴</sup> شاخصی است که اولاً هدف کمینه کردن آن باشد و ثانیاً افزایش در مقدار آن، تنها در صورتیکه زمان تکمیل حداقل یکی از کارها در برنامه افزایش یابد، ممکن باشد.

<sup>۲</sup>. Sequency

<sup>۳</sup>. Scheduling

Emmons (1975) یک الگوریتم شاخه و حد را برای حداقل کردن میانگین زمان در جریان ساخت وقتی حداقل کارهای دارای تاخیر با استفاده از الگوریتم Moore تعیین شده باشد، ارائه داد. در مطالعه‌ای که Nelson et al انجام دادند، مقدار حداقل تعداد کارهای دارای تاخیر برای محدوده مشخص شده با استفاده از قاعده کوتاه‌ترین زمان پردازش،<sup>1</sup> (SPT)، و الگوریتم Moore (1986) تعیین شدند. آنها نوع دیگری از الگوریتم Emmons را برای توسعه روشی جهت تولید همه جواب‌های بهینه پارا تو فراتر از محدوده مشخصی مورد استفاده قرار دادند. Azizoglu et al (2003) [7] مساله زمانبندی دودهدفه حداقل کردن حداکثر زودکرد و تعداد کارهای دارای تاخیر را بر روی یک ماشین مورد مطالعه قرار دادند. آنها فرض کردند زمان بیکاری مجاز نمی‌باشد. ابتدا، مساله حداقل کردن حداکثر زودکرد را درحالیکه تعداد کارهای دارای تاخیر در حداقل مقدار خود بودند، آزمودند. Eren & Guner (2006) [8] مساله تک ماشین دو معیاره با زمان‌های آماده‌سازی وابسته به ترتیب را مطالعه کردند. تابع هدف مورد بررسی کمینه کردن مجموع وزنی زمان‌های تکمیل و دیرکرد بود. مساله با استفاده از برنامه‌ریزی عدد صحیح مدل شد. مساله موردنظر NP-Hard بود. به همین دلیل برای حل مسایل با کارهای زیاد، از روش‌های هیوریستیک استفاده گردید. Tavakkoli-Moghaddam et al (2006) [9] رویکرد برنامه‌ریزی آرمانی فازی را برای حل مدل برنامه‌ریزی مختلط عدد صحیح مسائل تک ماشین با اهداف حداقل کردن زمان در جریان ساخت موزون و مجموع دیرکرد وزنی ارائه دادند. به‌علت تناقض میان اهداف، آنها رویکرد برنامه‌ریزی آرمانی فازی را برای مدل ریاضی توسعه داده شده پیشنهاد دادند. این رویکرد برپایه درجه تمایل تصمیم‌گیرنده<sup>2</sup> (DM) استوار بود. Chen & Sheen (2007) [10]، مساله تک ماشین را با هدف حداقل کردن مجموع زودکرد و دیرکرد وزنی، در رابطه با تعداد کارهای دارای تاخیر مورد بررسی قرار دادند. آنها n کار با موعدهای تحویل مشترک و وزن‌های متفاوت را برای زودکرد و دیرکرد در نظر گرفتند و یک الگوریتم بهینه پارا تو پیشنهاد دادند. Huo et al (2007) [11] مساله زمانبندی تک ماشین با دومعیار حداکثر دیرکرد وزنی و تعداد کارهای دارای تاخیر را مورد مطالعه قرار دادند. آنها اثبات‌های شدیداً NP-Hard برای مسایل زمانبندی هرگاه یکی از این دو معیار، ضابطه اصلی و دیگری معیار فرعی باشد، ارائه دادند. Keha, et al (2009) [12] در مقاله‌ای، کارایی محاسباتی چهار برنامه‌ریزی عدد صحیح مختلط مختلف،

در قسمت سوم مورد توجه قرار می‌گیرد. در ادامه، به معرفی الگوریتم‌های ژنتیک، شبیه‌سازی آنیلینگ و روش پیاده‌سازی آنها برای حل مساله پرداخته و نتایج حاصل از اجرای این الگوریتم‌ها روی مثال نمونه مورد بررسی و تحلیل واقع می‌شود. لازم بذکر است که الگوریتم‌ها در محیط MTLAB R2009a در کامپیوتر شخصی با، 2.66GHz کدنویسی و آزموده شد.

### ۱-۱. طرح مساله

مساله توالی عملیات و ترتیب چیدن کارها در حالت تک ماشین به‌منظور کمینه کردن تعداد کارهای دارای تاخیر و دیرکرد وزنی از مهمترین مسایل زمانبندی کارها است. این دومعیار دارای اهمیت زیادی از نظر تولیدکنندگان و مشتریان است. در دنیای واقعی تعداد کارهای دارای تاخیر، معیاری است که به عنوان نرخ عملکرد مدیران مورد توجه قرار می‌گیرد و مجموع دیرکرد وزنی به علت اینکه، هزینه‌هایی که در اثر دیرکرد به سیستم تحمیل می‌گردند را اندازه‌گیری می‌کند، نسبت به توابع دیگر همچون مجموع وزنی زمان‌های تکمیل حالت عمومی‌تری دارد و بیشتر مورد استفاده قرار می‌گیرند. از سوی دیگر، در مسائل واقعی معمولاً به جای درگیر شدن با یکی از معیارها با هر دو آنها سروکار داریم. از اینرو در این مطالعه سعی خواهد شد که از تصمیم‌گیری دو هدفه استفاده شود. در چنین حالتی امکان ارائه توالی که هر دو پارامتر را بهینه نماید، مگر در مواقع خاص، ناممکن است. لذا سعی می‌شود توالی‌های کارا برای مساله تعیین گردد.

### ۲. ادبیات و پیشینه موضوع

زمانبندی شامل برنامه‌ریزی و مرتب کردن کارها در یک توالی از کارها به منظور برآورده کردن نیازهای مشتریان است [4]. محیط تک ماشین خیلی ساده و نیز حالت خاصی از سایر مدل‌ها همچون ماشین‌های موازی و سری هستند. در این مدل، تنها از یک ماشین برای فرآیند بهینه تمام کارها و اندازه‌گیری عملکرد سیستم همچون زمان تکمیل، تاخیر، تعداد کارهای دارای تاخیر، زمان بیکاری، ماکزیمم زودکرد و دیرکرد و سایر معیارها استفاده می‌شود [4].

اغلب محققین در مسائل تک ماشین حداقل کردن یکی از معیارهای گفته شده را مورد بررسی قرار داده‌اند. با این وجود مسایل زمانبندی اغلب شامل بیش از یک معیار هستند، بنابراین به تحلیل‌های چندهدفه نیاز خواهد بود. Nelson et al (1986) [5] و Emmons (1975) [6] مسائل با میانگین زمان در جریان ساخت و تعداد کارهای دارای تاخیر را مورد بررسی قرار دادند.

<sup>1</sup> Shortest Processing Time

<sup>2</sup> Decision maker

(GA)<sup>۴</sup> و شبیه‌سازی آنیلینگ (SA)<sup>۵</sup> به عنوان روش کاوشی معرفی می‌شوند.

### ۱-۳. تعاریف و نمادها

به منظور شرح مدل،  $n$  را تعداد کار بدون انقطاع روی یک ماشین در نظر بگیرید. کار  $j$  بوسیله زمان عملیات، زمان تحویل و یک واحد جریمه دیرکرد تعریف می‌شود. فرضیات و نمادهای بکارگرفته شده در این مطالعه به صورت زیر است:

#### فرضیات مربوط به ماشین

۱. ماشین در هر زمان فقط قادر به انجام یک کار است یعنی هیچ فرآیندی ناتمام قطع نمی‌گردد. و هر کاری که فرآیند آن روی ماشین شروع گردد تا اتمام کامل روی ماشین قرار دارد.
۲. زمان آماده‌سازی ماشین برای کارهای متفاوت می‌تواند به صورت بخشی از زمان فرآیند تعریف شود و یا اینکه، این زمان را در نظر نگیرد.
۳. ماشین در طول زمانبندی به طور پیوسته در دسترس بوده و مادامی که کار هست بیکار نمی‌ماند.

#### فرضیات مربوط به کارها

۱. کارها بر یکدیگر برتری ندارند و تمام کارها باید فرآیند خود را به اتمام برسانند.
۲. وقتی کاری روی ماشین قرار گرفت عملیات تا انتهای کار ادامه پیدا می‌کند.
۳. زمان‌های فرآیند مستقل از زمانبندی و نحوه قرار گرفتن کارها در توالی هستند، یعنی زمان انجام یک کار از قبل مشخص است، در غیر اینصورت باید ذکر شود.

#### نمادهای بکارگرفته شده

- $P_j$ : زمان پردازش کار  $j$  ( $j=1, 2, \dots, n$ )  
 $d_j$ : موعد تحویل کار  $j$  ( $j=1, 2, \dots, n$ )  
 $W_j$ : وزن کار  $j$  ( $j=1, 2, \dots, n$ )  
 $C_j$ : زمان تکمیل کار  $j$  ( $j=1, 2, \dots, n$ )  
 $R_j$ : زمان در دسترس قرار گرفتن کار،  
 $M$ : یک مقدار صحیح مثبت بزرگ،  
 $L_T$ : تعداد کارهای دارای تاخیر،  
 $w_j T_j = w_j \text{Max} \{0, C_j - d_j\}$  بطوریکه

<sup>۱</sup>(MIP)، را در مسائل گوناگون زمانبندی تک ماشین مورد مطالعه قرار دادند. نتایج حاصل از الگوریتم‌ها نشان داد، فرمولبندی MIP برای مسائل چندمعیاره کارایی بیشتری دارد. همچنین نتایج نشان داد برنامه‌ریزی عدد صحیح برای مسائل خاص بیشتر از مسائل عمومی مورد استفاده قرار گرفته و در عمل دارای کارایی بیشتری است. علاوه بر این، آنها دو مجموعه از نامعادلات را که می‌توانند برای بهبود فرمولبندی مورد استفاده قرار بگیرند، ارائه دادند.

لازم بذکر است که، قواعد گوناگونی برای توالی بهینه مدل‌های تک معیاره با اهداف متمایز وجود دارد (به عنوان نمونه، برای ساده‌ترین مدل مسائل تک ماشین بدون هیچ محدودیتی، قاعده کوتاهترین زمان فرآیند<sup>۲</sup> (SPT) برای حداقل کردن میانگین زمان در جریان ساخت، توالی بهینه را می‌دهد، درحالیکه قاعده زودترین موعد تحویل (EDD) ترتیب بهینه را برای حداقل کردن ماکزیمم تاخیر،  $T_{max}$ ، بدست می‌دهد). در حقیقت، هر تصمیم‌گیرنده براساس معیاری که در نظر دارد روشی متفاوت اتخاذ می‌کند. به-علاوه، هریک از این اهداف از دیدگاه‌های گوناگون با اهمیت هستند. باید توجه داشت که گاه این اهداف متضاد یکدیگر بوده و جواب مناسب برای یک هدف ممکن است برای هدف دیگر مناسب نباشد، ازاینرو، مسایل زمانبندی اغلب دارای طبیعت چندهدفه هستند [۴].

هدف از این مقاله، توسعه یک مدل برنامه‌ریزی خطی چندهدفه<sup>۳</sup> (MOLP)، به منظور حداقل کردن مجموع دیرکرد وزنی و تعداد کارهای دارای تاخیر، برای حل مسائل تک ماشین است. در ادامه از روش‌های ابتکاری جهت حل مدل پیشنهادی بهره گرفته شد.

### ۳. مدل ریاضی پیشنهادی

مسایل زمانبندی مدل‌های تک ماشین به لحاظ تنوع محدودیت-ها و توابع هدف دارای پیچیدگی‌هایی است که بعضاً حل آنها با روش‌های بهینه‌سازی معمول امکان‌پذیر نیست و یا مستلزم صرف وقت و هزینه بسیار است. در چند دهه اخیر گرایش محققان به روش‌های فراکاوشی بیشتر گشته است. این روش‌ها اغلب الهام گرفته از طبیعت هستند که برای حل مسائل در زمینه‌های مختلف از دهه ۱۹۶۰ مورد توجه قرار گرفتند، به‌گونه-ای که جواب‌های قابل قبول (نزدیک به بهینه) را با یک هزینه محاسباتی معقول جستجو می‌کنند، بی‌آنکه قادر به تضمین بهینگی مطلق جواب باشد [۱۳]. در این قسمت ابتدا به بیان تعاریف و نمادها پرداخته و سپس روش‌های الگوریتم ژنتیک

<sup>۴</sup>. Genetic Algorithm

<sup>۵</sup>. Simulated Annealing

<sup>۱</sup>. Mixed Integer Programming

<sup>۲</sup>. the shortest processing time

<sup>۳</sup>. Multi Objective Linear Programming

$\sum W_j T_j$ : مجموع دیرکرد،

$W_j T_j$ : جریمه دیرکرد وزنی کار  $j$ .

مینیمم‌سازی و علامت مثبت برای مسائل ماکزیم‌سازی استفاده می‌شود [۱۷].

### ۱-۲-۳. اجزاء و پارامترهای الگوریتم SA

- نمایش ساختار جواب،
- انتخاب جواب اولیه،

• مکانیزم ایجاد جواب همسایه: می‌توان از دو روش برای ایجاد همسایگی استفاده کرد: انتخاب جواب همسایگی از بین جواب‌های امکان‌پذیر به طور تصادفی، انتخاب جواب همسایگی برطبق ضابطه‌ای خاص براساس روش ابتکاری مخصوص به هر مساله. که روش اول بیشتر مورد استفاده قرار می‌گیرد (می‌توانید مرجع [۱۸] را ببینید).

• انتخاب دمای اولیه: چنانچه دمای اولیه پایین باشد، احتمال پذیرش جواب‌های بدتر کاهش یافته و ممکن است سیستم در جواب بهینه محلی باقی بماند. اگر بخواهیم جواب نهایی مستقل از جواب شروع باشد، دمای اولیه باید به اندازه کافی زیاد در نظر گرفته شود، تا امکان تعویض تقریباً آزاد جواب‌های همسایگی وجود داشته باشد، در غیر اینصورت جواب نهایی به سمت جواب شروع نزدیک خواهد شد. وایت نظریه معادل بودن  $T_0$  با انحراف استاندارد هزینه‌های سیستم از میانگین هزینه را مطرح نمود [۱۹]. در این روش دمای اولیه را معادل انحراف استاندارد مقادیر ارزش تابع هدف به تعداد دفعات معین اجرای برنامه در حالت ناپایدار<sup>۲</sup> در نظر گرفته و در هر مرحله براساس چند نمونه دمای اولیه محاسبه می‌شود (رابطه ۱).

$$T_0 = \sqrt{\frac{\sum_{j=1}^N (obj(j) - Mean)^2}{N - 1}} \quad (2)$$

• مکانیزم کاهش دما: ضابطه کاهش دما و حرکت به سمت سرد شدن سیستم، معمولاً به صورت تابعی به شکل زیر نشان داده می‌شود. (Cooling Function) [۲۰]:

$$T_{r+1} = \gamma T_r, 0.5 \leq \gamma \leq 0.99, \forall r = 1, 2, \dots, n \quad (3)$$

چنانچه  $\gamma$  مقدار بالایی داشته باشد، سیستم دیرتر سرد می‌شود و تعداد نقاط بیشتری از فضای جواب بررسی می‌شوند، که خود سبب افزایش زمان حل مساله خواهد شد.

$$X_{ij} = \begin{cases} 1 & \text{if job } j \text{ is scheduled after job } i \\ 0 & \text{otherwise} \end{cases}$$

**تعریف ۱.** تعداد کارهای دارای دیرکرد ( $L_T$ ): این تابع هدف به موعده تحویل وابسته بوده و به صورت  $1 \parallel \sum U_j$  نشان داده می‌شود که معادل با درصد کالاهایی می‌باشد که به موقع ارسال و تحویل مشتری می‌گردند:

$$U_j = \begin{cases} 1 & T_j \\ 0 & \text{otherwise} \end{cases} \quad L_T = \sum_{j=1}^n U_j \quad (1)$$

**تعریف ۲.** دیرکرد وزنی کل تک‌ماشینه (SMTWT)<sup>۱</sup>: هنگامیکه زمان تکمیل کار از موعده تحویل آن بزرگتر باشد، دیرکرد روی داده است. برای حل این مساله، معمولاً از روش شاخه و حد استفاده می‌شود (برای مطالعه بیشتر مرجع [۱۴] را ببینید).

### ۲-۳. شبیه‌سازی آنیلینگ

ایده‌ای که اساس روش SA را شکل داد برای اولین بار توسط متروپلیس و همکاران در ۱۹۵۳ ارائه شد که الگوریتمی برای شبیه‌سازی فرآیند آنیلینگ بود. ۳۰ سال بعد کرک پاتریک و همکاران [۱۵-۱۶] روشی پیشنهاد کردند که براساس آن می‌توان این شبیه‌سازی را برای جستجوی جواب در مسائل بهینه‌سازی به کاربرد. در روند این جستجو، الگوریتم SA با یک  $T_0$  (دمای اولیه) و  $S_0$  (جواب اولیه) شروع شده که ابتدا، این جواب، اولین ( $S_C$ ) و بهترین جواب ( $S_B$ ) است. یک جواب همسایگی ( $S_N$ ) با تعویض بعضی عناصر  $S_C$  بدست می‌آید. چنانچه تابع هدف به ازای  $S_N$  و  $S_C$  را با  $E_N$  و  $E_C$  نشان دهیم، اگر  $E_N$  از  $E_C$  بهتر باشد،  $S_N$  به عنوان  $S_C$  جدید پذیرفته می‌شود و در تکرار بعد  $S_N$  جدید تولید می‌شود. ولی چنانچه  $E_N$  از  $E_C$  بدتر باشد، از معیار متروپلیس استفاده شده و  $S_N$  با احتمال  $\exp(-\frac{\Delta E}{T})$  به عنوان  $S_C$  جدید پذیرفته می‌شود. در این رابطه  $\Delta E = E_N - E_C$  است و چنانچه  $E_C$  از  $E_B$  بهتر باشد،  $S_C$  به عنوان  $S_B$  جدید پذیرفته می‌شود. در غیر اینصورت  $S_B$  بدون تغییر باقی می‌ماند. SA این فرآیند را به تعداد  $L_n$  (طول زنجیره مارکوف) در هر سطح از دما تکرار و پارامتر  $T$  به آرامی با استفاده از تابع کاهش دما تا هنگامیکه شرط توقف برقرار گردد کاهش می‌یابد. لازم بذکر است که از علامت منفی برای مسائل

<sup>۲</sup>. None Stable

<sup>۱</sup>. Single machine total weighted tardiness

گام ۴: یک جواب همسایگی ( $S_N$ ) تولید کرده و مقدار تابع هدف ( $E_N$ ) را به ازای آن محاسبه کنید.  
 گام ۵: مقدار  $\Delta E = E_N - E_C$  را محاسبه کنید.  
 گام ۶: اگر  $\Delta E \leq 0$  جواب همسایگی را به عنوان جواب کنونی بپذیرید ( $S_C = S_N; E_C = E_N$ ) و به گام ۹ بروید.  
 گام ۷: اگر  $\Delta E > 0$  سپس یک عدد تصادفی در بازه  $[0, 1]$  تولید کنید.  
 گام ۸: اگر  $\text{Rand}[0, 1] > \exp(-\Delta E/T)$ ، جواب همسایگی را به عنوان جواب کنونی بپذیرید ( $S_C = S_N; E_C = E_N$ )، در غیر این صورت  $S_C$  و  $E_C$  بدون تغییر می‌مانند و به گام ۱۰ بروید.  
 گام ۹: اگر  $E_C < E_B$  قرار دهید  $E_C = E_B; S_C = S_B$  در غیر این صورت  $S_B$  و  $E_B$  بدون تغییر می‌مانند.  
 گام ۱۰: اگر تعداد تکرارها برابر طول زنجیره مارکوف گردید ( $iter = L_n$ )، به گام ۱۱ بروید، در غیر این صورت  $iter = iter + 1$  و به گام ۴ بروید.  
 گام ۱۲: بررسی شرایط توقف، اگر  $T \geq T_f$  به گام ۴ بروید، در غیر این صورت،  $S_B$  و  $E_B$  به عنوان جواب بهینه گزارش و متوقف شوید [۱۷].

### ۳-۳. معرفی الگوریتم ژنتیک

الگوریتم ژنتیک، شبیه‌سازی از سیر تکامل بیولوژیکی است که از اصل تکامل طبیعی داروین گرفته شده است. در این روش درمیان جواب‌های موجود، جواب‌های برتر که باعث بهبود تابع هدف می‌شوند برای تولید نسل بعدی جواب‌ها انتخاب می‌شوند و این سیر تکاملی تا یافتن جواب بهینه تکرار می‌شود. این الگوریتم اولین بار توسط هولند<sup>۳</sup> در ۱۹۷۵ ارائه شد [۱۳].

الگوریتم ژنتیک از توارث صفات والدین به فرزندان به وسیله ترکیب کروموزوم<sup>۴</sup>‌های والدین ایده می‌گیرد. در هر نسل، کروموزوم‌ها با استفاده از برخی معیارهای برازندگی<sup>۵</sup> ارزیابی می‌شوند. برای ایجاد نسل بعدی، کروموزوم‌های جدید که فرزند نامیده می‌شوند یا به وسیله عملگر تقاطع<sup>۶</sup> از کروموزوم نسل جاری و یا به وسیله اصلاح یک کروموزوم با استفاده از عملگر جهش شکل می‌گیرند. پس از چند نسل، الگوریتم به سمت بهترین کروموزوم، همگرا می‌شود که راه حل بهینه یا نزدیک به بهینه‌ای را برای مساله ارائه می‌کند.  
 گام‌های لازم الگوریتم ژنتیک جهت حل مسائل زمانبندی به صورت زیر می‌باشد:

• طول زنجیره مارکوف: طول زنجیره مارکوف،  $L_n$ ، تعداد تکرارهایی است که در یک دمای ثابت صورت می‌گیرد تا به حالت تعادل برسد.  
 • مکانیزم پذیرش جواب‌های نامزد شده: فرض کنید حل جاری منجر به تابع هدفی معادل  $E_C$  شده و نیز جواب همسایگی ایجاد شده دارای مقدار تابع هدف  $E_N$  باشد. در حالت ماکزیم‌سازی اگر  $\Delta E = E_N - E_C \geq 0$ ، جاری می‌شود و چنانچه  $E_N > E_B$  باشد،  $S_B = S_N$  قرار داده، در غیر این صورت  $S_B$  و  $E_B$  بدون تغییر باقی می‌مانند.  
 ۱.  $\Delta E = E_N - E_C < 0$ ، مقدار  $y = e^{+(\frac{\Delta E}{T})}$  با یک عدد تصادفی بین صفر و یک مورد مقایسه قرار می‌گیرد و اگر از آن عدد تصادفی بزرگتر بود، جواب همسایگی به عنوان  $S_C$  پذیرفته می‌شود، اما در غیر این صورت جواب همسایگی دیگری تولید می‌شود.  
 • معیارهای توقف الگوریتم SA: با توجه به اینکه الگوریتم‌های فوق ابتکاری هیچگونه شناختی نسبت به نقطه بهینه سراسری و بطور کلی درجه بهینه بودن جواب‌ها ندارد، معیاری برای توقف آنها مورد نیاز است. به طور کلی می‌توان در حالت استاتیک شرط توقف را چنین بیان کرد: رسیدن به یک دمای نهایی از پیش تعیین شده،  $T_f$ ، که کسر بسیار کوچکی از دمای اولیه است (رابطه ۳) و یا رسیدن به یک مقدار ماکزیمم از پیش تعیین شده برای تعداد کل تکرارها [۲۱]:

$$T_f = \beta T_0 \quad \beta \leq 0.01 \quad (4)$$

روش SA ابزاری برای ما فراهم می‌آورد تا بتوان از نقاط بهینه محلی<sup>۱</sup> فرار کرده و بوسیله پذیرش جواب‌های بدتر با یک احتمال مشخص، به نقطه بهینه سراسری<sup>۲</sup> دست یافت. این مورد یکی از نقاط قوت رویکرد SA است. برای مطالعه بیشتر در این زمینه می‌توانید مراجع [۲۱-۲۲] را ببینید. گام‌های لازم جهت شبیه‌سازی SA به صورت زیر می‌باشد:

گام ۱: پارامترهای ورودی ( $T_0; T_{final}; \gamma; L_n$ ، set  $T = T_0$ )  
 گام ۲: انتخاب جواب اولیه ( $S_0$ ) و محاسبه مقدار تابع هدف به ازای جواب اولیه ( $E_0$ ).  
 گام ۳: جواب اولیه را به عنوان بهترین جواب در نظر بگیرید  
 $(S_c = S_0; S_B = S_0; E_C = E_0; E_B = E_0; iter = 1)$

<sup>3</sup>. Holland

<sup>4</sup>. chromosome

<sup>5</sup>. Fitness

<sup>6</sup>. Crossover

<sup>1</sup>. Local Optimal

<sup>2</sup>. Global Optimal

محدودیت (۵) و (۵) رابطه میان زمان‌های تکمیل هر دو جفت از کارها را قید می‌کنند و براین موضوع تاکید دارند که در هر جفت از کارهای  $i$  و  $j$ ، کار  $j$  قبل از کار  $i$  و یا بالعکس انجام می‌شود (در این محدودیت مقدار  $M$  بزرگتر از مجموع زمان‌های فرآیند و ماکزیمم مقدار زمان‌های در دسترس کارها در نظر گرفته شده است)، محدودیت (۶) برای تعداد کارهای دارای دیرکرد، محدودیت (۷) تعیین‌کننده دیرکرد مربوط به هر کار و محدودیت‌های (۸) و (۹) نشان‌دهنده نوع متغیرهای تصمیم مساله است.

#### ۴. ارائه الگوریتم و روش حل

برای هدف موردنظر در این مقاله مثالی بیان شده که در قسمت اول با استفاده از روش‌های گفته شده حل و در پایان الگوریتم ترکیبی برای افزایش کارایی مدل ارائه شده است. داده‌های مثال در جدول (۱) آمده است. فرضیات زیر برای تولید داده‌ها در نظر گرفته شده است:

۱- زمان‌های انجام فرآیند  $(P_j)$ ، دارای توزیع یکنواخت گسسته بر بازه  $[۲۰ \ ۳۰]$  هستند.

۲- زمان‌های تحویل  $(d_j)$ ، دارای توزیع یکنواخت گسسته بر بازه  $[P^*(1-T-R/2), P^*(1-T+R/2)]$  هستند.

در حالت تک‌ماشین  $P = \sum_{j=1}^n P_j$  است [۴،۲۳].

مقادیر  $R$  و  $T$  به ترتیب از مجموعه  $\{۱ \text{ و } ۰/۶ \text{ و } ۰/۲\}$  و  $\{۰/۸ \text{ و } ۰/۴ \text{ و } ۰/۲\}$  انتخاب می‌شوند. هنگامیکه  $T$  بزرگتر انتخاب شود، زمان‌های تحویل در بازه‌های محدودتری تولید می‌شود ولی چنانچه خواسته باشیم زمان‌های تحویل دارای تنوع بیشتری باشند،  $R$  بزرگ انتخاب می‌شود (همان مراجع، به ترتیب صفحات ۹۲۲ و ۴۷). در این مطالعه  $R$  و  $T$  به ترتیب  $۰/۶$  و  $۰/۴$  انتخاب شدند. این در حقیقت یک راه کلاسیک برای تولید نمونه‌های تصادفی مساله زمانبندی است

۳- وزن کارها  $(W_j)$ ، توزیع یکنواخت گسسته بر بازه  $[۱۰ \ ۲۰]$  دارند.

۴- زمان‌های در دسترس  $(R_j)$ ، توزیع یکنواخت گسسته بر بازه  $[۱ \ ۱۰]$  دارند.

الف- گام‌های لازم برای حل مساله با استفاده از GA به صورت زیر است:

گام ۱. ابتدا مقادیر بهینه توابع هدف را با توجه به مدل برنامه‌ریزی خطی بیان شده در ۳-۴ بدست آورید. در اینصورت  $Z_1 = 791$  و  $Z_2 = 2$  خواهد شد.

#### Step 1:

Set  $k = 1$ .  
Select  $l$  initial sequences  $S_{1,1}, \dots, S_{l,1}$  using some heuristic.

#### Step 2:

Select the two best schedules among  $S_{k,1}, \dots, S_{k,l}$  and call these  $S_k^+$  and  $S_k^{++}$ .  
Select the two worst schedules among  $S_{k,1}, \dots, S_{k,l}$  and call these  $S_k^-$  and  $S_k^{--}$ .  
Generate two offspring  $S^*$  and  $S^{**}$  from parents  $S_k^+$  and  $S_k^{++}$ .  
Replace  $S_k^-$  and  $S_k^{--}$  with  $S^*$  and  $S^{**}$ .  
Keep all other schedules the same and go to Step 3.

#### Step 3:

Increment  $k$  by 1,  
If  $k = N$  then STOP,  
Otherwise go to Step 2[14].

#### ۳-۴. مدل برنامه‌ریزی خطی چندهدفه

در این مطالعه از برنامه‌ریزی خطی چندهدفه برای حل مدل تک ماشین بهره گرفته شده است:

$$\text{Min}Z_1 = \sum_{j=1}^n W_j T_j \quad \text{Min}Z_2 = L_T = \sum_{j=1}^n U_j \quad (5)$$

$$U_j = \begin{cases} 1 & T_j \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

$$S.t \quad C_j \geq P_j + R_j \quad \forall j \quad (7)$$

$$X_{ij} + X_{ji} = 1 \quad \forall i, j : i \neq j \quad (8)$$

$$C_i - C_j + MX_{ij} \geq P_i \quad \forall i, j : i \neq j \quad (9)$$

$$C_j - C_i + M(1 - X_{ij}) \geq P_j \quad \forall i, j : i \neq j \quad (10)$$

$$C_j \leq d_j + MU_j \quad \forall j \quad (11)$$

$$T_j \geq C_j - d_j \quad \forall j \quad (12)$$

$$X_{ij}, U_j \in \{0,1\} \quad \forall i, j : i \neq j \quad (13)$$

$$T_j, C_j \geq 0, \quad \forall j \quad (14)$$

محدودیت (۳) تضمین‌کننده بزرگتر بودن زمان تکمیل کار از زمان پردازش کار و زمان در دسترس قرار گرفتن کار  $j$ ، محدودیت (۴) نشان‌دهنده ترتیب ارتباط دو کار در برنامه زمانبندی شده،

گام ۵. از تقاطع دو نقطه‌ای استفاده و ۵۰ جمعیت جدید تولید کنید.

گام ۶. نرخ جهش را برابر ۰/۱ در نظر گرفته و برای هر کروموزم عدد تصادفی بین صفر و یک تولید و چنانچه عدد حاصل از نرخ جهش بزرگتر بود، عملیات جهش را برای آن کروموزم انجام دهید.

گام ۷. به شمارنده ۱ واحد اضافه نموده و برازندگی توالی‌های بدست آمده را محاسبه کنید (توجه داشته باشید در این مرحله باید تعداد کروموزم‌ها دقیقاً برابر مقدار اولیه باشد).

گام ۸. شرایط توقف را بررسی کنید (رسیدن به تعداد ثابت تکرار)، در غیراینصورت به گام ۷ بروید. نتایج حاصل از GA در جدول ۲ آمده است. لازم بذکر است که تعداد زایش‌ها برابر ۵۰۰ در نظر گرفته شده است.

جدول ۱. مساله با ۷ کار

| Jobs  | ۱  | ۲   | ۳  | ۴  | ۵  | ۶   | ۷  |
|-------|----|-----|----|----|----|-----|----|
| $w_j$ | ۱۶ | ۱۳  | ۱۲ | ۱۶ | ۱۹ | ۱۲  | ۱۷ |
| $P_j$ | ۲۴ | ۲۱  | ۲۶ | ۲۹ | ۲۹ | ۳۰  | ۲۰ |
| $d_j$ | ۹۴ | ۱۸۰ | ۸۰ | ۸۶ | ۹۳ | ۱۷۶ | ۹۶ |
| $r_j$ | ۸  | ۳   | ۵  | ۳  | ۹  | ۶   | ۱۰ |

ب- گام‌های لازم جهت حل مدل با استفاده از الگوریتم SA به‌قرار زیر می‌باشد:

گام ۱. انتخاب جواب اولیه و سایر پارامترهای مدل (در اینجا با توجه به وجود الگوریتم بهینه برای تعداد کارهای دارای تاخیر، جواب اولیه را از الگوریتم هاجسون بدست می‌آوریم. دمای اولیه، از انحراف استاندارد ۱۵ توالی موجه، طول زنجیره مارکوف، برابر ۶۰، دمای پایانی، که از رابطه (۳) بخش ۱-۲-۳ بدست می‌آید (بادر نظر گرفتن  $\beta=0.01$ ). و تعیین مقدار عددی  $\gamma$ ، که در اینجا برابر 0.95 در نظر گرفته شده است).

گام ۲. مکانیزم ایجاد جواب همسایگی: کارهایی را به طور تصادفی جهت تعویض برای بدست آوردن جواب همسایگی انتخاب کنید. باتوجه به اینکه احتمال انتخاب هر کار ۰/۱۴۲ است، یک عدد تصادفی بین  $U(0,1)$  تولید کنید، چنانچه این عدد از ۰/۱۴۲ کمتر بود کار ۱ با ۲ جابه‌جا شود، اگر بین ۰/۲۸۵ - ۰/۱۴۲ بود، کار ۲ با ۳، و به همین ترتیب، اگر بین ۰/۸۵۷ - ۰/۲۸۵ بود، کار ۳ با ۴ جابه‌جا شود.

گام ۳. امکان‌پذیری جواب را با برقرار بودن محدودیت‌ها مورد بررسی قرار دهید. چنانچه توالی حاصل امکان‌پذیر نبود با استفاده از قاعده تولید جواب همسایگی جواب دیگری تولید کنید تا محدودیت‌ها را ارضا کند.

گام ۲. شمارنده را برابر ۱ قرار دهید و تعدادی از توالی‌های امکان‌پذیر را به عنوان جمعیت اولیه در نظر بگیرید (این جمعیت شامل تعدادی کروموزم است که هر کروموزم، از ۷ ژن و هر ژن نشان-دهنده شماره کار در توالی موردنظر است، تشکیل شده است). در اینجا جمعیت اولیه برابر ۵۰ فرض شده است (با استفاده از قاعده تولید اعداد تصادفی، ۷ عدد تصادفی تولید و بزرگترین عدد را به کار شماره ۱ و کوچکترین را به کار شماره ۷ اختصاص دهید. این کار را تاجایی که ۵۰ کروموزم موجه بدست آید تکرار کنید، به عنوان مثال، اولین کروموزم به صورت زیر تولید می‌شود:

$$P_1 = (0.45, 0.71, 0.11, 0.98, 0.23, 0.67, 0.75) = (5, 3, 7, 1, 6, 4, 2)$$

گام ۳. برازندگی توالی‌های بدست آمده را محاسبه کنید. یک روش ساده جهت تشکیل یک تابع برازندگی ترکیبی، روش L-P متریک می‌باشد که در این مطالعه از آن استفاده شده است. منظور از روش‌های L-P متریک حداقل کردن انحراف تابع هدف موجود از یک مدل چندهدفه نسبت به راه حل ایده‌آل می‌باشد. بنابراین تابع برازندگی بصورت زیر خواهد شد:

$$L-P = \left\{ \sum_{j=1}^k \gamma_j \left[ \frac{f_j(x_j^*) - f_j(x)}{f_j(x_j^*)} \right]^p \right\}^{\frac{1}{P}} \quad (15)$$

که،

$x_j^*$ : راه حل ایده‌آل در بهینه‌سازی هدف  $f_j$ ،

$x$ : بیانگر یک راه حل مفروض،

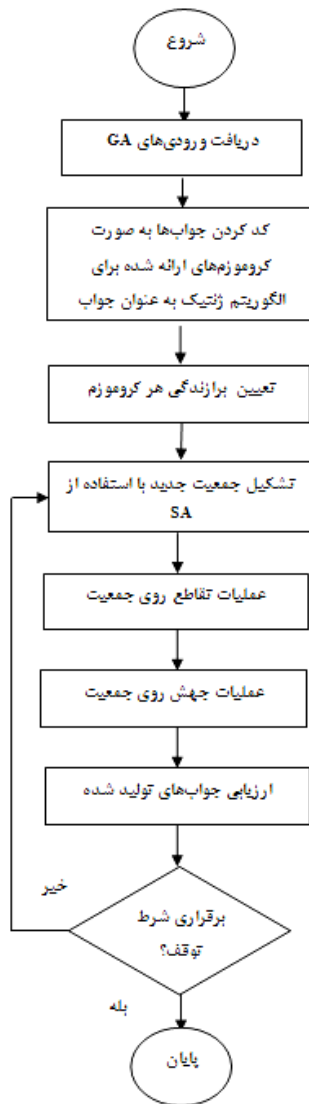
$\gamma_j (j > 0)$ : نشان دهنده درجه اهمیت برای هدف  $f_j$ ،

و تابع سازگار L-P به منظور حداقل کردن انحرافات از ایده‌آل باید کمینه گردد.

گام ۴. مقادیر حاصل از گام ۳ را به ترتیب نزولی مرتب، و عملیات تقاطع را برای ۲۵ توالی انجام دهید (نحوه انتخاب: از میان ۲۵ توالی اول که دارای مقدار برازندگی بیشتری می‌باشند ۴۰٪ را به طور تصادفی انتخاب کنید، ۱۰ توالی، ۶۰٪ باقی را از میان ۲۵ توالی دوم و ۱۵ توالی باقی‌مانده گروه اول انتخاب کنید).

۱. فاصله متریک بمنظور سنجش نزدیکی یک راه حل موجود نسبت به راه حل ایده‌آل مورد استفاده واقع می‌شود. در این روش  $1 \leq P \leq \infty$  بیانگر پارامتر مشخص‌کننده روش‌های L-P و درجه تاکید به انحرافات موجود است، بگونه‌ای که هرچه این ارزش بزرگتر باشد تاکید بیشتری بر بزرگترین انحرافات خواهد بود. معمولاً ارزش‌های  $p=1$  و  $p=2$  و  $p=\infty$  در محاسبات بکار گرفته می‌شوند. در این مطالعه نیز به ازای  $P=1$  و  $P=2$  محاسبات انجام گرفته است.





شکل ۱. روند کلی حل روش ترکیبی GA و SA

### ۵. نتیجه‌گیری

اگرچه نمی‌توان ادعا کرد که روش‌های کاوشی قادر به یافتن بهینه مطلق هستند، و همگرایی آنها در صورتی اثبات می‌شود که تعداد تکرارها به بی‌نهایت میل کند، لکن در بسیاری از مسائل از جمله مسئله زمانبندی همیشه یافتن بهینه مطلق مدنظر نیست، بلکه قصد اصلی یافتن پاسخی رضایتبخش با صرف زمان و هزینه-های محاسباتی معقول است. در این مطالعه توانایی‌های الگوریتم-های هیوریستیک GA و SA در حل مساله بهینه‌سازی توام تعداد کارهای دارای تاخیر و مجموع دیرکرد وزنی در مدل تک ماشین از نظر دقت و سرعت مورد بررسی قرار گرفت. از مزیت‌های روش SA می‌توان به سادگی الگوریتم و سرعت همگرایی و بهبود جواب نهایی اشاره کرد. انتخاب جواب اولیه مناسب و سازگار با شرایط مساله نیز در سرعت همگرایی و بهبود جواب موثر است. با استفاده از الگوریتم ترکیبی و استفاده از جواب‌های همسایگی SA

گام ۴. مقدار تابع هدف را به ازای جواب همسایگی تولید شده برای تابع هدف مجموع دیرکرد وزنی محاسبه و با مقدار تابع هدف حاصل از گام اول براساس مطالب گفته شده در بخش ۱-۲-۳ مورد ارزیابی قرار دهید.

گام ۵. تا وقتی که تعداد تکرارها برابر زنجیره مارکوف نشده این فرآیند را تکرار کنید. سپس با استفاده از رابطه (۲) بخش ۱-۲-۳، ضریب کاهش دما را محاسبه و با دمای نهایی مقایسه کنید. چنانچه شرط توقف برقرار گردید،  $S_B$  و  $E_B$  را گزارش، درغیراینصورت به گام ۲ بروید. نتایج حاصل از الگوریتم SA به ازای پارامترهای ورودی مختلف در جدول ۳ آمده است.

جدول ۲. نتایج حاصل از حل مساله با استفاده از GA

| تابع هدف                  | جمعیت | نرخ جهش | توالی کارها     |
|---------------------------|-------|---------|-----------------|
| $Z_1 = 791$<br>$Z_2 = 2$  | ۵۰    | ۰/۱     | {۴,۱,۵,۷,۳,۶,۲} |
| $Z_1 = 2462$<br>$Z_2 = 4$ | ۵۰    | ۰/۱     | {۴,۳,۵,۶,۲,۷,۱} |

ج- گام‌های لازم جهت حل مدل با استفاده از الگوریتم ترکیبی GA و SA:

در مرحله تشکیل جمعیت جدید، هر کدام از کروموزم‌های جمعیت اولیه به عنوان جواب آغازین الگوریتم SA در نظر گرفته می‌شوند. جواب همسایگی با استفاده از مکانیزم اعداد تصادفی تولید و گام‌های لازم شبیه‌سازی آنیلینگ انجام می‌شود ( $T_0 = 60, (\gamma = 0.95, L = 60, \beta = 0.01)$  توالی که شرط توقف را برآورده می‌کند، به عنوان کروموزم جمعیت جدید انتخاب می‌شود (پارامترهای الگوریتم ژنتیک، اندازه جمعیت ۵۰ و نرخ جهش ۰/۱) در نظر گرفته شد. این فرآیند که برای تمامی کروموزم‌های جمعیت اولیه انجام می‌شود. باعث می‌گردد، حتی توالی‌هایی که توابع هدف مناسب ندارند، با احتمال مشخصی جهت بررسی مورد استفاده قرار گیرند. این عمل باعث افزایش سرعت همگرایی مساله می‌گردد. روند کلی روش ترکیبی SA و GA در شکل (۱) نشان داده شده است.

این روش ابزاری برای ما فراهم می‌آورد تا بتوان از نقاط بهینه محلی فرار کرده و بوسیله پذیرش جواب‌های بدتر با احتمال مشخص، به نقطه بهینه سراسری دست یافت. نتایج حاصل در جدول ۴ نشان داده شده‌اند. همان‌طور که از بررسی آماری جمعیت جواب‌ها، دیده می‌شود در روش ترکیبی الگوریتم ژنتیک و شبیه‌سازی آنیلینگ جمعیت علاوه بر اینکه پراکندگی مناسب را دارد، سرعت خوبی برای میل متوسط جواب‌ها به جواب‌های مناسب را دارا می‌باشد. در واقع همین باعث سرعت و دقت بالای این الگوریتم نسبت به روش‌های قبلی شده است.

مجاز"، پایان‌نامه کارشناسی ارشد، دانشگاه صنعتی اصفهان، دانشکده مهندسی صنایع و مرکز برنامه‌ریزی سیستم‌ها، سال ۱۳۸۲.

[۳] زهتابان، م.، "الگوریتم توالی عملیات با هدف کمینه کردن دیرکرد و زودکرد"، پایان‌نامه کارشناسی ارشد، دانشکده فنی دانشگاه تهران، شهریور ۱۳۷۸.

[۴] برهانی داریان، ع.ر.، شهیدی، ل.، "بررسی کاربرد الگوریتم شبیه‌سازی آنیلینگ در بهره‌برداری بهینه از منابع آب و مقایسه آن با دیگر روش‌های کاوشی"، نشریه بین‌المللی علوم مهندسی دانشگاه علم و صنعت ایران، ویژه‌نامه مندرسی عمران، جلد ۱۹، شماره ۸، سال ۱۳۸۷، صفحه ۳۱-۴۰.

[۵] محسنی موحد، ا.، "تهیه مدل ریاضی بهینه‌سازی عملکرد هیدرولیکی کانال‌های آبیاری با استفاده از SA"، رساله دکتری کشاورزی، دانشگاه تربیت مدرس، ۱۳۸۲.

[6] Tavakkoli-Moghaddam, R., Javadi, B., Jolani, F., Ghodrathnama, A., "The Use of a Fuzzy Multi-Objective Linear Programming for Solving a Multi-Objective Single Machine Scheduling Problem", Applied Soft Computing, 10, 2010, pp. 919-925.

[7] Nelson, R.T., Sarin, R.K., Daniels, R.L., "Scheduling with Multiple Performance", 1986.

[8] Emmons, H. "A Note on a Scheduling Problem with Dual Criteria". Naval Research Logistics 22, 1975, pp. 615-616.

[9] Azizoglu, M., Kondakci, S., Köksalan, M., "Single Machine Scheduling with Maximum Earliness and Number Tardy", Computers and Industrial Engineering 45 (2), 2003, pp. 257-268.

[10] Eren, T., Güner, E., "Abicriteria Scheduling with Sequence-Dependent Setup Times", Applied Mathematics and Computation 179 (1), 2006, pp. 378-385.

[11] Tavakkoli-Moghaddam, R., Javadi, B., Safaei, N., "Solving a Mixed-Integer Model of a Single Machine Scheduling Problem by a Fuzzy Goal Programming Approach", Wseas Transactions on Business and Economics 3 (2), 2006, pp. 45-52.

[12] Chen, W.Y., Sheen, G.J., "Single-Machine Scheduling with Multiple Performance Measures: Minimizing Job-Dependent Earliness and Tardiness Subject to the Number of Tardy Jobs", Int. J. Production Economics 109, 2007, pp. 214-229.

[13] Huo, Y., Leung, J.Y.-T., Zhao, H., "Bi-Criteria Scheduling Problems: Number of Tardy Jobs and

و نیز در نظر گرفتن جواب‌های بد با احتمال مشخص کارایی الگوریتم را بهبود بخشیدیم. بررسی‌ها نشان داد که الگوریتم ترکیبی شبیه‌سازی آنیلینگ و ژنتیک از توانایی خوبی از نظر سرعت و میزان بهینه بودن جواب‌ها نسبت به سایر الگوریتم‌ها برخوردار بوده است.

از سوی دیگر، هرچند سرعت همگرایی SA خوب است، ولی امکان همگراشدن جواب‌ها قبل از رسیدن به جواب بهینه مطلق وجود دارد و لذا در مورد این الگوریتم، پارامترهای موجود را باید به طور مناسب انتخاب کرد، زیرا انتخاب نامناسب آنها می‌تواند تأثیر بسزایی در جواب الگوریتم داشته باشد. در مقابل، الگوریتم ژنتیک نسبت به شبیه‌سازی آنیلینگ از سرعت کمتری برخوردار بوده و احتمال کمتری وجود دارد که به بهینه‌های محلی همگرا شود از سوی دیگر به دلیل ساختار کروموزوم‌ها، امکان عدم همگرایی و پیدانشدن جواب یا پیداشدن جواب نامناسب مسأله وجود دارد. به همین دلیل از الگوریتم تلفیقی برای کاهش سرعت همگرایی و پراکندگی جواب‌ها استفاده شد.

### جدول ۳. نتایج حاصل از حل مسأله با استفاده از الگوریتم SA

| تابع هدف              | $T_0$ | $T_{final}$ | $\gamma$ | توالی کارها     |
|-----------------------|-------|-------------|----------|-----------------|
| $Z_1=805$<br>$Z_2=3$  | ۵۳    | ۰/۵۳        | ۰/۹۵     | {۴,۳,۵,۷,۱,۶,۲} |
| $Z_1=1953$<br>$Z_2=4$ | ۵۳    | ۰/۵۳        | ۰/۹۵     | {۲,۷,۱,۳,۴,۵,۶} |

### جدول ۴. مقایسه روش پیشنهادی SA & GA در پیدا کردن جواب مسأله با ۷ کار

| تابع هدف              | روش  | زمان بدست آوردن جواب (ثانیه) | توالی کارها     |
|-----------------------|------|------------------------------|-----------------|
| $Z_1=791$<br>$Z_2=2$  | GA   | ۱/۱۸۳۹                       | {۴,۱,۵,۷,۳,۶,۲} |
| $Z_1=2462$<br>$Z_2=4$ | GA   | ۱/۲۷۸۹                       | {۴,۳,۵,۶,۲,۷,۱} |
| $Z_1=805$<br>$Z_2=3$  | SA   | ۰/۱۷۹۹                       | {۴,۳,۵,۷,۱,۶,۲} |
| $Z_1=1953$<br>$Z_2=4$ | SA   | ۰/۲۱۸۵                       | {۲,۷,۱,۳,۴,۵,۶} |
| $Z_1=765$<br>$Z_2=2$  | GA + | ۰/۵۵۶۷                       | {۴,۱,۵,۷,۳,۶,۲} |
| $Z_1=1856$            | SA   | ۰/۸۷۶۵                       | {۶,۳,۴,۲,۵,۱,۷} |

### مراجع

[۱] مصلحی، ق.، "کمینه‌سازی مجموع بیشینه‌های زودکرد و دیرکرد در مسائل Flow shop"، رساله دوره دکترای مهندسی صنایع، دانشگاه تربیت مدرس، دانشکده فنی مهندسی، زمستان ۱۳۷۸.

[۲] واسعی، م.، "الگوریتم بهینه تعیین توالی عملیات در مسأله یک ماشین با مجموع بیشینه زودکرد و بیشینه دیرکرد و بیکاری

*Maximum Weighted Tardiness*”, European Journal of Operational Research 177(1), 2007, pp. 116–134.

- [14] Keha, A.B., Khowala, K., Fowler, J., “*Mixed Integer Programming Formulations for Single Machine Scheduling Problems*”. Computers & Industrial Engineering 56, 2009, pp. 357–367.
- [15] Pinedo, M., L., *Scheduling Theory, Algorithms, and Systems*, Third Edition, Springer, 2008.
- [16] Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., “*Optimization by Simulated Annealing*”, Science, 220, 1983, pp. 671–680.
- [17] Jeong, S.J., Kim, K.S., Lee, Y.H., “*The Efficient Search Method of Simulated Annealing using Fuzzy Logic Controller*”, Expert Systems with Applications 36, 2009, pp. 7099–7103.
- [18] Ozcan, U., “*Balancing stochastic Two-Sided Assembly Lines: A Chance Constrained, Piecewise-Linear, Mixed Integer Program and a Simulated Annealing Algorithm*”, European Journal of Operational Research 205, 2010, pp. 81–97.
- [19] White, S.R., “*Concept of Scale in Simulated Annealing*”, Proceeding IEEE International Conference on Computer Design, Portchester, 1983.
- [20] Vidal, R.V.V., *Applied Simulated Annealing*, Springer-Verlag, 1993.
- [21] Vasan, A., Komaragiri Srinivasa, R., “*Comparative analysis of Simulated Annealing, Simulated Quenching and Genetic Algorithms for Optimal Reservoir Operation*”, Applied Soft Computing 9, 2009, pp. 274–281.
- [22] Reeves, C.R., “*Modern Heuristic Techniques for Combinatorial Problems*”, Oxford, Blackwell Scientist Publications, 1993.
- [23] Loukil, T., Teghem, J., Tuytens, D., “*Solving Multi-Objective Production Scheduling Problems using Metaheuristics*,” European Journal of Operational Research, 161, 2005, pp.42–61.

