

## Solving a Robust Balancing & Sequencing of Robotic U-Shaped Assembly Line Problem by Harmony Search Algorithm

**Mahsa Sobhishoja**

MSc. In Industrial Engineering, Department of Industrial Engineering, Faculty of Engineering, Bu-Ali Sina University, Hamedan, Iran.

**Parvaneh Samouei\***

Assistant Professor, Department of Industrial Engineering, Faculty of Engineering, Bu-Ali Sina University, Hamedan, Iran

### Abstract

In recent years, robots have been widely used in assembly systems as called robotic assembly lines where a set of tasks have to be assigned to stations and each station needs to select one of the different robots to process the assigned tasks. Our focus is on u-type layouts because they are widely employed in many industries due to their efficiency and flexibility when compared to straight assembly lines. These lines offer more choices to group operations, a worker can be assigned to multiple stations located at entrance and exit sides. However, in many realistic situations, robots may be unavailable during the scheduling horizon for different reasons, such as breakdowns. This research deals with line balancing under uncertainty. The objective in this research is minimizing the cycle time for a given number of workstations and minimizing robot cost. This research deals with line balancing under uncertainty and presents one robust optimization model for balancing and sequencing of u-shaped robotic assembly line with considering set up times between task, failure robot times and preventive maintenance times for every robot. Since the NP-hard nature of the problem, multi-objective harmony search is developed to solve it. Numerical experiments demonstrated that by increasing uncertainty level, cost and cycle times increased.

**Keywords:** Balancing and Sequencing Assembly Line, Robust Optimization, Multi Objective Harmony Search, Robotic U-shaped, Set up Time Between Task

\*Corresponding Author: p.samouei@basu.ac.ir


**How to Cite:** Vol.19 No 63, Winter 2021



## حل مسئله استوار بالانس و توالی خط مونتاژ رباتیک U شکل با استفاده از الگوریتم جستجوی هارمونی

مهسا صبحی شجاع

کارشناس ارشد، دپارتمان مهندسی صنایع، دانشکده مهندسی، دانشگاه بوعلی سینا، همدان، ایران

پروانه سموئی 

استادیار، دپارتمان مهندسی صنایع، دانشکده مهندسی، دانشگاه بوعلی سینا، همدان، ایران

### چکیده

در سال‌های اخیر ربات‌ها به‌طور گسترده‌ای در سیستم مونتاژ با عنوان خطوط مونتاژ رباتیک مورد استفاده قرار گرفته است. در این خطوط مجموعه‌ای از فعالیت‌ها باید به ایستگاه‌ها تخصیص داده شوند و هر ایستگاه نیازمند انتخاب یکی از انواع ربات‌ها برای پردازش فعالیت‌های تخصیص‌یافته است. در شرایط واقعی ممکن است ربات‌ها در طی افق زمان‌بندی به دلایل مختلفی مانند خرابی از دسترس خارج شوند. این تحقیق در زمینه بالانس خط در شرایط عدم قطعیت صحبت می‌کند خطوط U-شکل به دلیل انعطاف‌پذیری و کارایی بیشتر نسبت به خطوط مستقیم، در بسیاری از صنایع مورد استفاده قرار گرفته‌اند. این خطوط گزینه‌های بیشتری برای تعیین فعالیت‌ها به ایستگاه‌های کاری ارائه می‌دهند و اپراتورها می‌توانند هم‌زمان به ایستگاه‌های کاری در هر دو سمت ورودی و خروجی سرویس دهند. هدف در این مسئله حداقل کردن زمان سیکل برای تعداد مشخصی ایستگاه‌های کاری و به حداقل رساندن هزینه احداث ربات است. این مقاله، یک مدل استوار برای مواجهه با عدم قطعیت در مسئله بالانس و توالی خط مونتاژ رباتیک U-شکل مدل ترکیبی ارائه می‌دهد همچنین زمان‌های آماده‌سازی بین فعالیت‌ها و زمان‌های خرابی و زمان نگهداری و تعمیرات هر ربات در آن در نظر گرفته می‌شود. از آنجایی که این مسئله یک مسئله NP-hard است لذا از الگوریتم فرا ابتکاری، جستجوی هارمونی چندهدفه جهت حل و بهینه‌سازی استفاده می‌شود. نتایج نشان می‌دهد با افزایش عدم قطعیت، مقادیر هزینه و زمان سیکل نیز افزایش می‌یابد.

**کلیدواژه‌ها:** بالانس و توالی خط مونتاژ، بهینه‌سازی استوار، جستجوی هارمونی چندهدفه، رباتیک U-شکل، زمان آماده‌سازی بین فعالیت‌ها

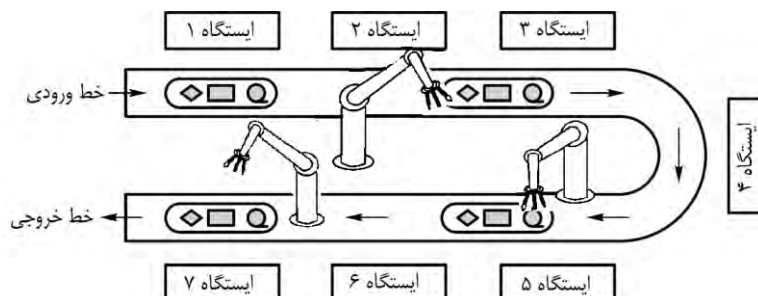
## مقدمه

از آنجا که رویکردهای سنتی تولید در پاسخگویی به تقاضای سریع مشتریان از اثربخشی کافی برخوردار نیست لذا، شرکت‌ها جهت برآورده کردن تقاضای مشتریان نیازمند به کارگیری سیستم‌های نوین تولید هستند یکی از این سیستم‌های نوین تولیدی، خط مونتاژ رباتیک است اهمیت خط مونتاژ رباتیک از آن رو است که توانسته است، جایگزین خوبی برای نیروی کار انسانی باشد که با مشکل متداول انجام کار تکراری روبرو است و باعث خستگی فیزیکی و ذهنی می‌شود (تومپولوس<sup>۱</sup>، ۲۰۱۴). یکی دیگر از این سیستم‌های نوین خطوط مونتاژ U شکل است که در سال‌های اخیر توجه ویژه‌ای به آن شده است، خط U شکل، این امکان را فراهم می‌سازد که اپراتورها در فاصله‌ای نزدیک به یکدیگر همکاری کنند که این امر باعث صرفه‌جویی در فضای مورد نیاز و ایجاد محیط کار امن‌تر برای اپراتورها می‌گردد علاوه بر این چنین خطی باعث به حداقل رساندن تعداد ایستگاه‌های کاری می‌گردد (میلتنبورگ<sup>۲</sup>، ۲۰۰۱)، توکساری<sup>۳</sup> و همکاران (۲۰۰۸)، وانگ<sup>۴</sup> و همکاران (۲۰۰۸)، سیرونکول و چیوتیما<sup>۵</sup> (۲۰۱۰)). علاوه بر این، به‌طور تجربی نشان داده شده است که بهره‌وری کار می‌تواند به‌طور قابل توجهی در خطوط U-شکل افزایش یابد (آس<sup>۶</sup> و همکاران، ۲۰۰۴).

این مقاله، تخصیص و توالی فعالیت‌ها و تخصیص ربات‌ها به ایستگاه‌های کاری در یک خط مونتاژ رباتیک U-شکل را مدنظر قرار داده است. در این نوع از خط تولید ربات‌ها در یک خط U-شکل چیده شده و در آن عملیات مونتاژ محقق می‌گردد. یک خط مونتاژ رباتیک U-شکل در شکل ۱ نشان داده شده است. هر خط مونتاژ U-شکل را می‌توان به دو، خط ورودی و خط خروجی تقسیم کرد و ربات‌ها امکان سرویس‌دهی به هر دو خط را دارند (به‌عنوان مثال به‌طور هم‌زمان ربات ۱ به ایستگاه ۱ در خط ورودی و ایستگاه ۷ در خط خروجی سرویس می‌دهد)

- 
1. Thomopoulos
  2. Miltenburg
  3. Toksari
  4. Hwang
  5. Sirovetskul and Chutima
  6. Aase

شکل ۱. خط مونتاژ رباتیک U-شکل



در شرایط واقعی، فرایندهای مونتاژ به منابع مختلف عدم قطعیت، مانند تغییرات در زمان عملیات، تقاضا و غیره بستگی دارند. این تغییرات اهداف مونتاژ را تهدید می‌کند و محافظت در برابر آنها بسیار ضروری است. در میان این منابع عدم قطعیت، تغییرات در زمان‌های عملیات می‌تواند بسیار مهم باشد، به خصوص برای خطوطی که شامل ربات و خرابی‌های ناشی از استفاده ربات و در نهایت هزینه احتمالی است (هزیر و دولگوئی<sup>۱</sup>، ۲۰۱۵). به این منظور در این مقاله یک مدل استوار برای مسئله با در نظر گرفتن زمان خرابی و زمان نگهداری و تعمیر ربات توسعه یافته است. خطوط مونتاژ تک مدل برای تولید حجم بالا از محصولات یکسان طراحی شده است که برای تقاضای مشتری با تنوع بالا مناسب نیست. همچنین در موارد دنیای واقعی با توجه به هزینه‌های بالای مورد نیاز برای ساخت و نگهداری خط مونتاژ، حمل اجناس و غیره، سیستم‌های تولیدی، به سمت تولید چندین مدل مشابه از محصول به جای تک مدل سوق داده شدند. در چنین شرایطی، تولید مدل‌های ترکیبی برای تولید محصول، باعث ارضای تقاضاهای گوناگون و کاهش هزینه‌های تولید و انبار می‌شود (ربانی و همکاران، ۲۰۱۲). در اکثر مقالاتی که به مسئله توالی پرداخته‌اند می‌توان توالی را به دو حالت معنی کرد اول، توالی مدل در حالت‌هایی که نوع محصول از مدل‌های ترکیبی است و ترتیب انجام مدل‌ها را مشخص می‌کند؛ اما در این مقاله منظور، توالی فعالیت است که نشان‌دهنده تعیین نحوه ترتیب انجام فعالیت‌های مونتاژ است.

1. Hazir and Dolgui

## مروری بر مطالعات گذشته

ادبیات مرتبط با این مطالعه را می‌توان به ۳ دسته اصلی تقسیم‌بندی کرد که عبارت‌اند از: الف) بالانس خطوط مونتاژ رباتیک (U-شکل، ب) عدم قطعیت در مسائل خط مونتاژ و ج) استفاده از الگوریتم‌های فرا ابتکاری برای بالانس خطوط مونتاژ.

### پیشینه پژوهش

راینویچ و بوکچین<sup>۱</sup> (۱۹۹۱) اولین محققانی بودند که مسئله بالانس خط مونتاژ را مورد بررسی قرار دادند. آن‌ها در این مسئله تخصیص حجم مساوی از کار را میان مجموعه انواع ربات‌های موجود به ایستگاه‌های روی خط مورد بررسی قرار دادند. هدف این بود که تعداد ایستگاه‌های کاری را برای یک‌زمان سیکل داده‌شده به حداقل برسانند اما در مورد مسئله بالانس خط مونتاژ رباتیک (U-شکل برای اولین بار در ادبیات این نیلاکانتان و پونامبالم<sup>۲</sup> (۲۰۱۵) بودند که به بررسی این موضوع باهدف حداقل کردن زمان سیکل پرداختند. برای مسئله‌ای مشابه با دو تابع هدف حداقل‌سازی زمان سیکل و حداقل‌سازی کل مصرف انرژی نیلاکانتان و همکاران (۲۰۱۶) یک الگوریتم تکاملی ارائه دادند. نیلاکانتان و همکاران (۲۰۱۶) همچنین مسئله بالانس خط مونتاژ رباتیک (U-شکل را به‌منظور حداقل کردن زمان سیکل و هزینه مطالعه کردند و الگوریتم‌های بهینه‌سازی ازدحام ذرات و تفاضل تکاملی برای حل آن ارائه دادند. ربانی و همکاران (۲۰۱۶) مسئله بالانس خط مونتاژ رباتیک (U-شکل مدل ترکیبی را باهدف حداقل‌سازی هزینه‌های احداث ربات، هزینه‌های آماده‌سازی ربات، هزینه‌های آماده‌سازی وابسته به توالی و زمان سیکل بررسی کردند و از دو الگوریتم بهینه‌سازی چندهدفه، الگوریتم ژنتیک مرتب‌سازی نامغلوب و الگوریتم بهینه‌سازی ازدحام ذرات چندهدفه برای حل مسئله استفاده کردند. ژانگ<sup>۳</sup> و همکاران (۲۰۱۹) یک مدل غیرخطی چندهدفه جدید برای حداقل کردن انتشار کربن، انتشار سروصدا و زمان سیکل به‌طور هم‌زمان برای مسئله خط مونتاژ (U-شکل ارائه

1 Rubinovitz, and Bukchin

2. Nilakantan and Ponnambalam

3. Zhang

دادند علاوه بر این الگوریتم گرگ خاکستری ترکیبی پارتو برای حل مسئله بالانس چندهدفه طراحی شد و در نهایت این الگوریتم با پنج الگوریتم چندهدفه مشهور مقایسه شد و نتایج کارایی بیشتر الگوریتم پیشنهادی را نشان داد.

باید توجه داشت که اکثر تحقیقات خط مونتاژ، اطلاعات کاملی از تمام داده‌ها را فرض می‌گیرد. در حالی که در شرایط واقعی، فرآیندهای مونتاژ به منابع مختلف عدم قطعیت بستگی دارد برای این منظور می‌توانیم از بهینه‌سازی استوار استفاده کنیم که یکی از روش‌های بهینه‌سازی اساسی است که عدم قطعیت و اثرات آن را مدل می‌کند. هزیر و دولگوئی (۲۰۱۳) تحقیقی را در زمینه بالانس خط تحت عدم قطعیت ارائه دادند و دو مدل بهینه‌سازی استوار پیشنهاد دادند. عدم قطعیت فاصله‌ای برای زمان‌های عملیاتی در نظر گرفته شد؛ هدف تحقیق حداقل‌سازی تعداد ایستگاه‌های کاری بود و یک الگوریتم مبتنی بر تجزیه بندرز برای حل موارد بهینه در مقیاس بزرگ توسعه داده شد. در ادامه این دو نویسنده در سال ۲۰۱۵ مسئله بالانس خط مونتاژ نوع II-شکل را تحت عدم قطعیت با تشکیل یک مسئله همتای استوار و الگوریتم بهینه‌سازی آن ارائه دادند. در این مقاله فرض شد که تنها یک زیرمجموعه از زمان‌های عملیاتی، بدترین مقدار خود را می‌گیرند و برای حل مسئله، الگوریتم تجزیه بندرز پیشنهاد شد و با برخی از آزمایش‌های محاسباتی مورد ارزیابی قرار گرفت. پیرا و میراندا<sup>۱</sup> (۲۰۱۷) بر روی یک مسئله بالانس خط مونتاژ با عدم قطعیت در زمان‌های فعالیت مطالعه کردند. در این مقاله از روش برتسیماس و سیم برای مواجهه با عدم قطعیت استفاده شد. پیرا (۲۰۱۸) همچنین یک مسئله بالانس و تخصیص کارگر خط مونتاژ را در نظر گرفت در حالی که زمان‌های فعالیت در آن بین حدود بالا و پایین مشخص فرض شده بود و تابع هدف حداقل کردن حداکثر مقدار ممکن در میان سناریوهای موجود بود. سموئی و عشایری (۲۰۱۹) دو مدل ریاضی جدید برای مسئله بالانس خط مونتاژ نیمه خودکار مدل ترکیبی پیشنهاد دادند. در این مقاله فرمت‌های متفاوتی از اجرای عملیات، از اپراتور انسانی برای کمک به ربات تا راه‌حل رباتیک را شامل شده است. در مدل اول زمان سیکل مشخص و هدف حداقل کردن هزینه‌های ثابت و متغیر مربوط به آماده‌سازی خط مونتاژ بود اما مدل دوم شامل یک‌زمان سیکل متغیر بود که بین

1. Pereira and miranda

حدود بالا و پایین تغییر می‌کند. این حدود تغییرات تقاضای مورد انتظار آینده را بازتاب می‌کند، هدف این مدل حداقل کردن هزینه و زمان سیکل بود و برای دستیابی به نتایج بهتر و قابل‌اعتماد یک راه‌حل استوار برای سطوح مختلف اطمینان ( $\alpha$ ) ارائه دادند.

ازکان<sup>۱</sup> (۲۰۱۹) مسئله بالانس و زمان‌بندی فعالیت‌ها را در خطوط مونتاژ موازی با زمان‌های راه‌اندازی وابسته به توالی معرفی کردند. آنها یک مدل برنامه‌ریزی ریاضی باینری برای آن مسئله پیشنهاد دادند و مسئله را به کمک الگوریتم انجماد تدریجی حل نمودند. ژو و وو<sup>۲</sup> (۲۰۲۰) از بهینه‌سازی دو هدفه بر مبنای روش‌های تجزیه برای مسئله بالانس خطوط مونتاژ روباتیک استفاده کردند و در تحقیق خود توازن میان مصرف انرژی و بهره‌وری را مدنظر قرار دادند. ییلماز<sup>۳</sup> (۲۰۲۰) به این موضوع اشاره کرد که در خیلی از مقالات مرتبط با مسائل خطوط مونتاژ روباتیک، شرایط یک محیط قطعی در نظر می‌گیرند و به ماهیت عدم قطعیت زمان‌های عملیات توجه نمی‌کنند. این در حالی است که در خطوط مونتاژ روباتیک  $U$  شکل، بالانس خطوط تحت تأثیر عدم قطعیت زمان فعالیت‌های مونتاژ است. لئو و همکاران<sup>۴</sup> (۲۰۲۱) از احتمال سناریو و اعداد فازی مثلثی برای تقاضای غیرقطعی مسائل بالانس خطوط مونتاژ مدل‌های ترکیبی استفاده کردند و مسئله را به کمک الگوریتم ژنتیک حل کردند. جین و همکاران<sup>۵</sup> (۲۰۲۱) نیز از مدل‌های حداقل سازی حداکثر تأسّف برای مسائل بالانس خطوط مونتاژ با محدودیت منابع استفاده کردند و از الگوریتم بهینه‌سازی نهنگ برای حل کمک گرفتند. همچین ژنک و همکاران<sup>۶</sup> (۲۰۲۱) یک مدل بهینه‌سازی مختلط عدد صحیح استوار با آزادسازی زمان سیکل برای بالانس خطوط مونتاژ مدل‌های ترکیبی با عملیات دستی ارائه کردند. اخیراً نیز سوباسزک و همکاران<sup>۷</sup> (۲۰۲۲) یک مدل زمان‌بندی استوار با توجه به دو فاکتور عدم قطعیت خرابی‌های ماشین‌ها و زمان‌های عملیات متغیر ارائه کردند. با توجه به ساختار ترکیبی مسئله بالانس و توالی خط مونتاژ هنگامی که اندازه مسئله بزرگ می‌شود، رسیدن به یک راه‌حل بهینه دشوار می‌شود. از این رو،

---

1 Özcan

2 Zhou and Wu

3 Yilmaz

4 Liu et al.

5 Jin et al.

6 Zhang et al.

7. Sobaszek et al.



الگوریتم‌های فرا ابتکاری برای حل مسئله پیشنهاد می‌شود. الگوریتم ژنتیک رایج‌ترین الگوریتم فرا ابتکاری مورداستفاده در مسائل خط مونتاژ و بعدازآن، شبیه‌سازی تبرید و بهینه‌سازی کلونی مورچگان است. فرا ابتکاری‌ها از پرکاربردترین روش‌های مورداستفاده برای حل مسائل بالانس خط چندهدفه است. نورمحمدی و همکاران (۲۰۱۳) به مسئله بالانس خط مونتاژ U-شکل چندهدفه پرداختند. به حداکثر رساندن کارایی خط و به حداقل رساندن تغییرات حجم کار، اهداف در نظر گرفته شده است. این مقاله یک الگوریتم رقابت استعماری که با تکیه بر تکامل اجتماعی و سیاسی الهام گرفته شده است ارائه می‌دهد. برای نشان دادن کارایی الگوریتم پیشنهادی، نتایج مرتبط را با یک الگوریتم ژنتیک مقایسه کردند، نتایج محاسباتی نشان داد که الگوریتم پیشنهادی از الگوریتم ژنتیک بهتر است. پریرا و همکاران (۲۰۱۸) یک الگوریتم ترکیبی مبتنی بر ممتیک را برای حل مسئله بالانس خط مونتاژ رباتیک به منظور حداقل کردن هزینه تجهیزات ارائه دادند. تا جایی که مشخص است، اما در رابطه با بالانس و خط مونتاژ که با الگوریتم جستجوی هارمونی حل شده است مقاله پرنومو و وی<sup>۱</sup> (۲۰۱۴) است. آنها یک مدل ریاضی دو هدفه برای به حداکثر رساندن میزان تولید و توزیع حجم کاری در یک خط مونتاژ دوطرفه با محدودیت‌های موقعیتی پیشنهاد دادند. ارزیابی مجموعه جواب‌های بهینه پارتو بین الگوریتم پیشنهادی و الگوریتم ژنتیک دو هدفه نشان داد که روش پیشنهادی دارای نسبت و همگرایی بهتر برای مسائل سایز کوچک و متوسط و مجموعه راه‌حل‌های بهتری برای مسائل سایز بزرگ است.

در بازارهای رقابتی امروز تولید به موقع و مقرون به صرفه در حال تبدیل شدن به مسئله مهمی است. برای تولیدکنندگان، استفاده بهینه از منابع ماشین با اطمینان از یک برنامه کارآمد و استوار در سطح عملیاتی امری حیاتی محسوب می‌شود. در یک زمان بندی بلندمدت، سیستم‌های تولیدی به دلیل استفاده زیاد همواره در معرض خرابی و زوال قرار دارند که بر روی دسترس پذیری ماشین و پایداری سیستم تأثیر می‌گذارد. اکثر تحقیقات انجام شده در حوزه خطوط مونتاژ رباتیک، دسترسی همیشگی به ربات را در نظر می‌گیرند. این در حالی است که در شرایط واقعی، ممکن است ربات‌ها در طی افق زمان بندی به دلایل مختلفی،

1. Purnomo and Wee



مانند خرابی، نگهداری و تعمیر برنامه‌ریزی‌شده، از دسترس خارج شوند. این ویژگی در دسترس بودن، پیچیدگی را برای هر مسئله خط مونتاژ، بیشتر می‌کند. به‌غیر از نگهداری و تعمیر برنامه‌ریزی‌شده که بر روی دسترسی دستگاه تأثیر می‌گذارد، وقفه‌های غیرمنتظره دیگری نیز نظیر مثال خرابی، لغو یا ورود سفارش جدید یا تغییرات تاریخ باعث می‌شود برنامه به‌سرعت قابل اجرا نباشد. برای مقابله با این اختلالات که موجب تأخیر در زمان‌های پردازش مونتاژ می‌شود می‌توان از رویکرد برنامه‌ریزی استوار برای نمایش سطوح مختلف محافظه‌کاری تصمیم‌گیرندگان در دنیای واقعی استفاده می‌شود. علاوه بر این، با مطالعات انجام‌شده ما، پژوهشی را نیافتیم که در این حوزه و به‌خصوص برای خطوط رباتیک U شکل اقدامی را انجام داده باشند. این موضوعات در کنار در نظر گرفتن زمان آماده‌سازی وابسته به توالی بین فعالیت‌ها برای خطوط مونتاژ مدل‌های ترکیبی و ارائه الگوریتمی جهت حل این دسته از مسائل؛ موضوعاتی است که در این مقاله به آن پرداخته می‌شود.

### مدل مفهومی

یک مسئله بالانس خط مونتاژ رباتیک U-شکل مدل ترکیبی را می‌توان به شرح زیر توصیف کرد:  $m$  مدل محصول به‌طور هم‌زمان در هر سفارش، مونتاژ می‌شوند و فعالیت مونتاژ توسط ربات‌هایی بر روی یک مجموعه از ایستگاه‌های کاری انجام می‌پذیرد، هر ربات می‌تواند در هر دو طرف خطوط U-شکل کار کند (خطوط ورودی و خروجی). فعالیت‌های موردنیاز هر مدل دارای یک مجموعه از پیش تعیین‌شده از روابط پیش‌نیازی هستند، اما شباهت بین مدل‌ها امکان ترکیب تمام فعالیت‌ها را تنها با یک نمودار پیش‌نیازی با  $N_T$  فعالیت را فراهم می‌کند. مسئله بالانس خط مونتاژ رباتیک U-شکل مدل ترکیبی نوع ۲ تخصیص  $N_T$  فعالیت به  $N_S$  ایستگاه کاری و تخصیص یکی از  $N_r$  ربات موجود به هر ایستگاه کاری برای پردازش فعالیت‌های تخصیص‌یافته است به‌منظور به حداقل رساندن زمان سیکل برای یک تعداد مشخصی از ایستگاه‌های کاری. با تخصیص ربات از میان ربات‌های موجود با کارکرد و هزینه‌های متفاوت به هر ایستگاه کاری به دنبال حداقل کردن هزینه‌های ربات نیز می‌باشیم.

- مفروضات و ویژگی‌های اساسی مرتبط با مدل‌سازی مسئله به شرح زیر است:
۱. ربات‌ها می‌توانند به هر ایستگاه تخصیص داده شوند و هر ایستگاه می‌تواند شامل تنها یک ربات باشد، ربات‌ها می‌توانند هر فعالیتی را انجام دهند و زمان فعالیت‌ها بستگی به نوع ربات اختصاص داده شده دارد.
  ۲. هر فعالیت تنها به یک ربات قابل تخصیص است و در هر زمان هر ربات می‌تواند تنها یک فعالیت را پردازش کند، در نتیجه هر فعالیت می‌تواند تنها به یک ایستگاه تخصیص یابد و قابل تقسیم بین ایستگاه‌های کاری مختلف نیست.
  ۳. یک مجموعه‌ای از مدل‌های مشابه یک محصول روی خط U-شکل مونتاژ می‌شوند. فعالیت‌های مونتاژ برای مدل‌های مختلف تقریباً مشابه است و فعالیت‌های مشترک بین مدل‌های مختلف در ایستگاهی مشترک انجام می‌شوند. برای مدل‌های مختلف روابط پیش‌نیازی میان آن‌ها ثابت است و یک گراف پیش‌نیازی ترکیبی برای همه مدل‌ها در نظر گرفته می‌شود. زمان پردازش مربوط به یک فعالیت ممکن است برای مدل‌های مختلف متفاوت باشد.
  ۴. زمان‌های آماده‌سازی شامل تنها زمان آماده‌سازی بین فعالیت‌ها است و این زمان‌ها قطعی و ثابت است. در صورتی که زمان فعالیتی برای یک مدل صفر باشد زمان آماده‌سازی برای آن فعالیت صفر در نظر گرفته می‌شود.
  ۵. هر ایستگاه کاری شامل دو زیر ایستگاه است. (زیر ایستگاه یک در خط ورودی و زیر ایستگاه دو در خط خروجی)
  ۶. تعداد ایستگاه‌های کاری ثابت می‌باشند و تعداد ربات‌های موجود برای تخصیص بیش از تعداد ایستگاه‌های کاریست که تخصیص از میان آن‌ها صورت می‌گیرد و هر ربات می‌تواند به حداکثر یک ایستگاه کاری تخصیص یابد.
  ۷. در کل ۵ نوع ربات موجود است زمان پردازش فعالیت‌ها، زمان نگهداری پیشگیرانه، هزینه ربات‌ها، زمان تعمیرات در هر نوع ربات یکسان است. هزینه احداث هر ربات با ربات دیگر متفاوت و غیرقطعی است.
  ۸. خرابی ربات‌ها از یک نوع است، زمان تعمیرات پس از خرابی احتمالی و از تابع یکنواخت تبعیت می‌کند.

۹. زمان عملیات شامل چهار بخش زمان پردازش فعالیت، زمان آماده‌سازی، زمان نگهداری پیشگیرانه و زمان تعمیر در صورت بروز خرابی است. زمان پردازش فعالیت‌ها قطعی و با توجه به نوع ربات زمان‌ها نیز متفاوت‌اند، زمان آماده‌سازی قطعی زمان عملیات نگهداری پیشگیرانه و زمان تعمیر پس از خرابی ربات غیرقطعی و برای هر نوع ربات متفاوت است.

۱۰. خط مونتاژ یکنواخت است و هیچ انبار میانگیری بین ایستگاه‌های کاری وجود ندارد. ایستگاه‌های کاری بسته هستند.

۱۱. از زمان شروع یک فعالیت تا زمان اتمام آن یک‌بار خرابی ممکن است رخ دهد و در صورت خرابی ادامه کار پس از تعمیرات روی همان قطعه انجام می‌شود. (ضایعاتی برای قطعه وجود ندارد).

اندیس‌ها و پارامترها:

$j, k, l$ . اندیس فعالیت

$S$  اندیس ایستگاه کاری

$r$  اندیس ربات‌ها

$m$  اندیس نوع مدل محصول

$N_t$  تعداد کل فعالیت‌ها

$N_s$  تعداد کل ایستگاه‌های کاری

$N_r$  تعداد کل ربات‌ها

$N_m$  تعداد کل مدل‌های محصول

$P_{imr}$  زمان پردازش فعالیت  $i$  برای مدل  $m$  توسط ربات  $r$

$Q_{im}$  اگر زمان پردازش فعالیت  $i$  برای مدل  $m$  مثبت باشد ۱ و در غیر این صورت ۰

$FS_{ijm}$  زمان آماده‌سازی بین فعالیت  $i$  و  $j$  برای مدل  $m$

$t_r^p$  زمان انجام نگهداری پیشگیرانه روی ربات  $r$

$t_r$  زمان انجام تعمیر روی ربات  $r$  پس از خرابی

$C_r$  هزینه احداث ربات  $r$

$A_{ij}$  اگر فعالیت  $i$  پیش‌نیاز (مستقیم و غیرمستقیم) فعالیت  $j$  باشد ۱ و در غیر این صورت ۰

$R_{ir}$  اگر از زمان شروع فعالیت  $i$  تا زمان پایان فعالیت  $i$  توسط ربات  $r$ ، خرابی رخ دهد ۱ و

در غیر این صورت ۰

$h_{ir}$  اگر نگهداری پیشگیرانه برای ربات  $r$  قبل از فعالیت  $i$  ام انجام شود،  $h_{ir} = 1$ ؛ در

غیر این صورت،  $h_{ir} = 0$

$bigM$  یک عدد مثبت خیلی بزرگ

متغیرها:

$x_{is}$  اگر فعالیت  $i$  به ایستگاه کاری  $s$  در خط ورودی تخصیص یابد؛ ۱ در غیر این صورت

۰

$y_{is}$  اگر فعالیت  $i$  به ایستگاه کاری  $s$  در خط خروجی تخصیص یابد ۱ و در غیر این

صورت ۰

$u_{rs}$  اگر ربات  $r$  به ایستگاه کاری  $s$  تخصیص یابد ۱ و در غیر این صورت ۰

$w_{ijs}$  اگر فعالیت  $i$  پیش از فعالیت  $j$  در ایستگاه کاری  $s$  انجام شود ۱ و در غیر این صورت

۰

$Z_{ijms}$  اگر فعالیت  $j$  بلافاصله بعد از فعالیت  $i$  برای مدل  $m$  در ایستگاه کاری  $s$  انجام

شود ۱ و در غیر این صورت ۰

$CT$  زمان سیکل

$Cost$  هزینه احداث ربات‌ها

در این مقاله، یک مدل برنامه‌ریزی مختلط عدد صحیح جدید ارائه شده است. مدل ریاضی

پیشنهاد شده بر اساس فرمول ریاضی است که توسط آکینار و بایکاسوگلو<sup>۱</sup> (۲۰۱۴)،

اوکساز<sup>۲</sup> و همکاران (۲۰۱۷)، لو<sup>۳</sup> و همکاران (۲۰۱۵) ارائه شده، فرمول مدل ریاضی به

شرح زیر است:

1. Akpinar and Baykasoglu

2. Oksuz

3. Lu

$$\min CT = \max_{\substack{1 < s < N_s \\ 1 < m < N_m}} \left\{ \sum_{r=1}^{N_r} \sum_{i=1}^{N_t} (h_{ir} t_r^p + p_{imr} + t_r R_{ir}) (x_{is} + y_{is}) u_{rs} + \sum_{\substack{i=1 \\ i \neq j}}^{N_t} \sum_{j=1}^{N_t} Z_{ijms} FS_{ijm} \right\} \quad (1)$$

$$\min Cost = \sum_{s=1}^{N_s} \sum_{r=1}^{N_r} c_r u_{rs} \quad (2)$$

$$\sum_{s=1}^{N_s} (x_{is} + y_{is}) = 1 \quad \forall i = \{1.2. \dots. N_t\} \quad (3)$$

$$\sum_{i=1}^{N_t} (x_{is} + y_{is}) \geq 1 \quad \forall s = \{1.2. \dots. N_s\} \quad (4)$$

$$\sum_{r=1}^{N_r} u_{rs} = 1 \quad \forall s = \{1.2. \dots. N_s\} \quad (5)$$

$$\sum_{s=1}^{N_s} u_{rs} \leq 1 \quad \forall r = \{1.2. \dots. N_r\} \quad (6)$$

$$\sum_{s=1}^{N_s} (N_s - s + 1) (x_{is} - x_{js}) \geq 0 \quad \forall i, j \in \{1.2. \dots. N_t\} | A_{ij} = 1 \quad (7)$$

$$\sum_{s=1}^{N_s} (N_s - s + 1) (y_{js} - y_{is}) \geq 0 \quad \forall i, j \in \{1.2. \dots. N_t\} | A_{ij} = 1 \quad (8)$$

$$w_{ijs} + w_{jis} + Z_{ijms} + Z_{jims} \leq 2 \left( 1 - (x_{is} + y_{is}) + (x_{js} + y_{js}) \right) \quad (9)$$

$$\forall i, j \in \{1.2. \dots. N_t\}; m \in \{1.2. \dots. N_m\}; s = \{1.2. \dots. N_s\} \quad (10)$$

$$w_{ijs} + w_{jis} + Z_{ijms} + Z_{jims} \leq 2 \left( 1 + (x_{is} + y_{is}) - (x_{js} + y_{js}) \right) \quad (11)$$

$$\forall i, j \in \{1.2. \dots. N_t\}; m \in \{1.2. \dots. N_m\}; s = \{1.2. \dots. N_s\} \quad (12)$$

$$w_{iis} = 0 \quad \forall i \in \{1.2. \dots. N_t\}; s = \{1.2. \dots. N_s\} \quad (13)$$

$$w_{ijs} + w_{jis} + bigM \times \left( 2 - (x_{is} + y_{is}) - (x_{js} + y_{js}) \right) \geq 1 \quad (14)$$

$$\forall i, j \in \{1.2. \dots. N_t\} | j \neq i; s = \{1.2. \dots. N_s\}$$

$$w_{ijs} + w_{jis} - bigM \times \left( 2 - (x_{is} + y_{is}) - (x_{js} + y_{js}) \right) \leq 1 \quad (15)$$

$$\forall i, j \in \{1.2 \dots N_t\} | j \neq i; s = \{1.2. \dots N_s\} \quad ($$

$$A_{ij} \left( (x_{is} + y_{is}) + (x_{js} + y_{js}) - 1 \right) \leq w_{ijs} \quad (۱۵)$$

$$\forall i, j \in \{1.2 \dots N_t\} | j \neq i; s = \{1.2. \dots N_s\} \quad ($$

$$w_{iks} + w_{kjs} - 1 \leq w_{ijs} \quad \forall i, j, k \in \{1.2 \dots N_t\} | j \neq i \neq k; s = \{1.2. \dots N_s\} \quad (۱۶)$$

$$w_{ijs} + bigM \times (2 - x_{is} - y_{js}) \geq 1 \quad \forall i, j \in \{1.2 \dots N_t\}; s = \{1.2. \dots N_s\} \quad (۱۷)$$

$$\sum_{j=1}^{N_t} \sum_{s=1}^{N_s} Z_{ijms} \leq 1 \quad \forall i = \{1.2. \dots N_t\}; m \in \{1.2 \dots N_m\} \quad (۱۸)$$

$$Z_{ijms} + Z_{jims} \leq 1 \quad \forall i, j \in \{1.2 \dots N_t\} | j \neq i; m \in \{1.2 \dots N_m\}; s = \{1.2. \dots N_s\} \quad (۱۹)$$

$$\sum_{s=1}^{N_s} Z_{iims} = 0 \quad \forall i = \{1.2. \dots N_t\}; m \in \{1.2 \dots N_m\} \quad (۲۰)$$

$$\left( e_{is} Q_{im} + e_{js} Q_{jm} - 1 \right) - \left| \sum_{k=1}^{N_t} (w_{iks} Q_{km}) - \sum_{l=1}^{N_t} (w_{jls} Q_{lm}) - 1 \right| \leq Z_{ijms} \quad (۲۱)$$

$$\forall i, j \in \{1.2 \dots N_t\} | j \neq i; m \in \{1.2 \dots N_m\}; s = \{1.2. \dots N_s\} \quad (۲۲)$$

$$x_{is} \in \{0.1\} \quad \forall i \in \{1.2 \dots N_t\}; s = \{1.2. \dots N_s\} \quad (۲۳)$$

$$y_{is} \in \{0.1\} \quad \forall i \in \{1.2 \dots N_t\}; s = \{1.2. \dots N_s\} \quad (۲۴)$$

$$u_{rs} \in \{0.1\} \quad \forall r \in \{1.2 \dots N_r\}; s = \{1.2. \dots N_s\} \quad (۲۵)$$

$$w_{ijs} \in \{0.1\} \quad \forall i, j \in \{1.2 \dots N_t\}; s = \{1.2. \dots N_s\} \quad ($$

$$Z_{ijms} \in \{0.1\} \quad \forall i, j \in \{1.2 \dots N_t\}; m \in \{1.2 \dots N_m\}; s = \{1.2. \dots N_s\} \quad (۲۶)$$

رابطه (۱)، به حداقل رساندن زمان سیکل برای تعداد معینی از ایستگاه‌های کاری است و رابطه (۲)، حداقل کردن هزینه‌های احداث ربات در ایستگاه‌های کاری است. رابطه (۳) تضمین می‌کند که هر فعالیت  $i$  تنها به یک ایستگاه کاری  $s$  و تنها به یکی از دو خط ورودی یا خط خروجی اختصاص داده شود. رابطه (۴) تضمین می‌کند که به هر ایستگاه‌های کاری  $s$  حداقل یک فعالیت یا در خط ورودی یا در خط خروجی تخصیص

یابد. رابطه (۵) تضمین می‌کند که هر ایستگاه کاری  $S$  می‌تواند تنها شامل یک ربات باشد. رابطه (۶) تضمین می‌کند که هر ربات  $I$  می‌تواند حداکثر به یک ایستگاه کاری تخصیص یابد. رابطه (۷) تضمین می‌کند که اگر  $I$ ، فعالیت پیش‌نیاز مستقیم فعالیت  $J$  باشد و فعالیت  $J$  به خط ورودی ایستگاه  $S$  تخصیص یابد، آنگاه فعالیت  $I$  نیز باید به خط ورودی همان ایستگاه یا ایستگاه‌های قبلی تخصیص یابد، به‌طور مشابه رابطه (۸) تضمین می‌کند که اگر  $I$  فعالیت پیش‌نیاز مستقیم فعالیت  $J$  باشد و فعالیت  $I$  به خط خروجی ایستگاه  $S$  تخصیص یابد، آنگاه فعالیت  $J$  نیز باید به خط خروجی همان ایستگاه یا ایستگاه‌های بعدی تخصیص یابد. رابطه (۹) تضمین می‌کند که اگر تنها فعالیت  $I$  به ایستگاه کاری  $S$  یا به خط ورودی یا خط خروجی تخصیص یابد و فعالیت  $J$  تخصیص نیابد هیچ‌گونه توالی و آماده‌سازی بین آنها در ایستگاه  $S$  در نظر گرفته نشود. به‌طور مشابه رابطه (۱۰) نیز تضمین می‌کند که اگر تنها فعالیت  $J$  به ایستگاه کاری  $S$  یا به خط ورودی یا خط خروجی تخصیص یابد و فعالیت  $I$  تخصیص نیابد هیچ‌گونه توالی و آماده‌سازی بین آنها در ایستگاه کاری  $S$  در نظر گرفته نشود و رابطه (۱۱) تضمین می‌کند که اگر هیچ‌کدام از دو فعالیت  $I$  و  $J$  به ایستگاه کاری  $S$  تخصیص نیابد توالی و آماده‌سازی برای آنها در ایستگاه  $S$  در نظر گرفته نشود. رابطه (۱۲) تضمین می‌کند که هیچ توالی بین یک فعالیت  $I$  و خودش در ایستگاه کاری  $S$  در نظر گرفته نشود. رابطه (۱۳) تضمین می‌کند که اگر هر دو فعالیت  $I$  و  $J$  به ایستگاه کاری  $S$  تخصیص یابد حداقل یک توالی بین آنها در ایستگاه کاری  $S$  در نظر گرفته شود و رابطه (۱۴) تضمین می‌کند که اگر هر دو فعالیت  $I$  و  $J$  به ایستگاه کاری  $S$  تخصیص یابد حداکثر یک توالی بین آنها در ایستگاه کاری  $S$  در نظر گرفته شود. رابطه (۱۵) تضمین می‌کند که اگر هر دو فعالیت  $I$  و  $J$  به ایستگاه کاری  $S$  تخصیص یابد و فعالیت  $I$  پیش‌نیاز مستقیم فعالیت  $J$  باشد در این صورت فعالیت  $I$  باید پیش از فعالیت  $J$  در ایستگاه کاری  $S$  پردازش شود. رابطه (۱۶) تضمین می‌کند که اگر در ایستگاه کاری  $S$  هر سه فعالیت  $I$  و  $J$  و  $k$  تخصیص یابد و فعالیت  $I$  پیش از فعالیت  $k$  انجام شود و فعالیت  $k$  پیش از فعالیت  $J$  انجام شود در این صورت فعالیت  $I$  نیز باید پیش از فعالیت  $J$  انجام شود. رابطه (۱۷) تضمین می‌کند که اگر در ایستگاه کاری  $S$  هر دو فعالیت  $I$  و  $J$  تخصیص یابند و فعالیت  $I$  به خط ورودی و فعالیت  $J$  به خط خروجی تخصیص یابد در این صورت فعالیت  $I$  باید پیش از



فعالیت  $J$  انجام شود. رابطه (۱۸) تضمین می‌کند که حداکثر یک آماده‌سازی بعد از فعالیت  $i$  از مدل  $m$  در نظر گرفته شود. رابطه (۱۹) تضمین می‌کند که اگر دو فعالیت  $J_i$  از مدل  $m$  در ایستگاه کاری  $S$  تخصیص یافته باشند حداکثر یک آماده‌سازی بین این دو فعالیت انجام شود. رابطه (۲۰) تضمین می‌کند که هیچ آماده‌سازی بین یک فعالیت  $i$  و خودش در ایستگاه کاری  $S$  در نظر گرفته نشود. رابطه (۲۱) تضمین می‌کند که اگر دو فعالیت  $J_i$  از مدل  $m$  در ایستگاه کاری  $S$  تخصیص یافته باشند و توالی انجام فعالیت‌ها به این صورت باشد که  $i$  دقیقاً پیش از فعالیت  $J$  انجام شود یک عملیات آماده‌سازی بعد از اتمام فعالیت  $i$  و پیش از شروع فعالیت  $J$  لازم است. رابطه‌های (۲۲)، (۲۳)، (۲۴)، (۲۵)، (۲۶) متغیرهای باینری مسئله را نشان می‌دهند.

### روش پژوهش

پس از مدل‌سازی صورت گرفته، مدل توسط مثال‌های عددی مختلف حل شد و تحلیل حساسیت‌های مختلف برای آن انجام شد که تمام تحلیل‌های صورت گرفته صحت و اعتبار مدل را تأیید نمودند. یکی از رویکردهای بهینه‌سازی تحت شرایط عدم قطعیت، بهینه‌سازی استوار است. روش‌های دیگری نیز برای مقابله با عدم قطعیت وجود دارد سه رویکرد اصلی عبارت‌اند از:

- بهینه‌سازی تصادفی
- بهینه‌سازی فازی
- بهینه‌سازی استوار

مدل‌های بهینه‌سازی تصادفی برای جانشینی قطعیت در زمانی که پارامترهای مسئله ناشناخته و احتمالی هستند، مورد استفاده قرار می‌گیرد. این مدل‌ها زمانی قابل استفاده هستند که توزیع عدم قطعیت پارامترها مشخص و معلوم باشد. منطق فازی به جای پاسخ، آن سؤال را به دو بخش درست و نادرست تقسیم‌بندی نمی‌کند، بلکه در اصل به یک محدوده جواب در بین آنها توسعه داده می‌شود. منطق فازی و بهره‌گیری از آن برای مدل‌سازی عدم قطعیت تا حد زیادی به در دسترس بودن خیره و ماهیت پارامتر عدم قطعیت بستگی دارد. در بهینه‌سازی استوار، دو رویکرد قابل‌بحث است، استوار شدنی یا وجه و استوار بهینگی. استوارشدنی

بدین معناست که جواب باید برای تمامی و یا اکثریت حالات ممکن پارامترهای دارای عدم قطعیت شدنی باقی بماند. استوار بهینگی بدین معناست که مقدار تابع هدف به ازای جواب برای همه و یا بیشتر حالات ممکن پارامترهای دارای عدم قطعیت، نزدیک به مقدار بهینه باقی بماند. بنابر دلایل ذکر شده نتیجه می‌شود که برای مدل‌سازی عدم قطعیت، در مدل‌های استوار، به آگاهی از توزیع پارامترهای دارای عدم قطعیت و یا خبره احتیاج نیست و تنها لازم است تا حدود پارامتر دارای عدم قطعیت را دانست. لذا استفاده از رویکرد بهینه‌سازی استوار در مقایسه با بهینه‌سازی فازی و بهینه‌سازی تصادفی برای مسئله موجود در این مقاله، از ارجحیت و انعطاف بیشتری برخوردار است.

وابسته به تعریف استواری و چگونگی مجموعه عدم قطعیت داده‌ها، روش‌های بهینه‌سازی استوار متمایز می‌شوند. در یک تعریف کلی، در یک مسئله بهینه‌سازی تحت شرایط عدم قطعیت، جواب بهینه استوار جوابی است که در اکثریت حالات ممکن برای داده‌های غیرقطعی، موجه بماند و نسبت به مقدار بهینه تابع هدف انحراف کمتری داشته باشد. فرم کلی یک مسئله بهینه‌سازی خطی به صورت رابطه (۲۷) است:

$$\begin{aligned} \min c^T x \\ \text{s. t.} \\ Ax \leq b \end{aligned} \quad (27)$$

اگر در مدل فوق، حداقل یکی از مؤلفه‌های بردارهای  $c$  و  $b$  و یا ماتریس  $A$  غیرقطعی باشد، آنگاه یک مسئله بهینه‌سازی تحت شرایط عدم قطعیت حاصل می‌شود.

$\xi = [c, A, b]$  را به عنوان داده‌های مدل (۲۷) تعریف می‌کنیم. مجموعه  $U$  را به عنوان تمام حالات ممکن برای  $\xi$  (داده‌های مسئله) در نظر بگیرید. واضح است اگر  $|U| = 1$  باشد، آنگاه مدل فوق یک مسئله بهینه‌سازی قطعی است. ؛ اما در اکثر مسائل دنیای واقعی  $|U| \geq 1$  است و مسئله بهینه‌سازی تحت شرایط عدم قطعیت حاصل می‌شود.

یکی از روش‌های بهینه‌سازی استوار، روش فاصله‌ای بن تال<sup>۱</sup> و همکاران (۲۰۰۹) است که بر اساس آن، برای هر یک از داده‌های غیرقطعی مسئله، یک فاصله بسته کران‌دار عدم قطعیت در نظر گرفته می‌شود و جواب استوار به گونه‌ای است که در اکثریت حالات

ممکن برای داده‌ها در فاصله‌های متناظر آنها شدنی است. به منظور روشن شدن روش مذکور، مسئله بهینه‌سازی غیرقطعی رابطه (۲۸) را در نظر بگیرید:

$$\min z = \sum_j a_{0j} x_j + b_0 \quad (28)$$

s. t.

$$\sum_j a_{ij} x_j \leq b_0 \quad \forall i = 1, 2, \dots, m$$

که در آن برای هر  $i = \{1, 2, \dots, m\}$  داریم:  $b_i \in [a_{ij}^L, a_{ij}^U]$  و  $a_{ij} \in [a_{ij}^L, a_{ij}^U]$  حال تعریف می‌کنیم:

$$\tilde{a}_{ij} = a_{ij} + \xi_{ij} \hat{a}_{ij} \quad (29)$$

$$\tilde{b}_i = b_i + \xi_{i0} \hat{b}_i \quad (30)$$

جایی که  $a_{ij}$  و  $b_i$  مقادیر اسمی داده‌ها، و  $\hat{a}_{ij}$  و  $\hat{b}_i$  حداکثر انحراف هر داده از مقدار اسمی هستند توجه کنید که اگر داده‌ای از مسئله قطعی باشد، آنگاه کران بالا و پایین فاصله غیرقطعی متناظر با آن باهم برابر، و برابر با مقدار اسمی می‌شوند. پس از نرمال‌سازی فاصله عدم قطعیت هر داده، می‌توان مدل (۲۸) را به صورت رابطه (۳۱) بازنویسی کرد:

$$\min z = \sum_j a_{0j} x_j + b_0 + \sum_j \xi_j \hat{a}_{0j} x_j + \xi_0 \hat{b}_0 \quad (31)$$

s. t.

$$\sum_j a_{ij} x_j + \sum_j \xi_{ij} \hat{a}_{ij} x_j \leq b_i + \xi_{i0} \hat{b}_i \quad \forall i = 1, 2, \dots, m$$

که تمامی مقادیر  $\xi_j$  و  $\xi_0$  در فاصله  $[-1, 1]$  قرار دارند. پس از نرمال‌سازی، در روش فاصله‌ای بن تال و همکاران (۲۰۰۹)، مدل (۳۲) به عنوان همتای استوار مدل (۳۱) معرفی می‌شود:

$$\min z = \max_{\xi_j, \xi_0} \left( \sum_j a_{0j} x_j + b_0 + \sum_j \xi_j \hat{a}_{0j} x_j + \xi_0 \hat{b}_0 \right) \quad (32)$$

s. t.

$$\sum_j a_{ij} x_j + \Theta_i \max_{\xi_{ij}, \xi_i} \left( \sum_j \xi_{ij} \hat{a}_{ij} x_j - \xi_i \hat{b}_i \right) \leq b_i \quad \forall i = 1, 2, \dots, m$$

که در آن،  $\Theta_i$  پارامتر کنترل سطح عدم قطعیت در هر محدودیت است. اگر در تابع هدف و هر یک از محدودیت‌های مدل (۳۲)،  $\xi$ ها به طور مستقل از هم در فاصله  $[-1, 1]$  تغییر

کنند، آنگاه فضای عدم قطعیت Box بر مسئله حاکم می‌شود و همتای استوار با مدل رابطه (۳۳) معادل است (لی<sup>۱</sup> و همکاران (۲۰۱۱)):

$$\begin{aligned} \min z \\ \text{s. t.} \\ z \geq \sum_j a_{0j} x_j + b_0 + \psi_0 \left( \sum_j \hat{a}_{0j} |x_j| + \hat{b}_0 \right) \quad (33) \\ \sum_j a_{ij} x_j + \psi_i \sum_j \hat{a}_{ij} |x_j| \leq b_i - \psi_i \hat{b}_i \quad \forall i = 1, 2, \dots, m \end{aligned}$$

جایی که پارامتر  $\psi$  سطح ریسک گریزی یا پوشش عدم قطعیت را در پارامترهای هر محدودیت نشان می‌دهد. توجه کنید که اگر  $\psi = 0$  باشد به این معناست که مقادیر اسمی جایگزین داده‌های غیرقطعی می‌شوند.؛ و اگر  $\psi = 1$  باشد، آنگاه پوشش عدم قطعیت به صورت کامل لحاظ شده و جواب بهینه استوار حاصل شده از مدل (۳۳) به گونه‌ای است که در تمام حالات ممکن برای داده‌های غیرقطعی، شدنی است.

### الگوریتم جستجوی هارمونی چندهدفه

جستجوی هارمونی یک روش فرا ابتکاری است که به طور گسترده برای حل مسائل مختلف بهینه‌سازی مورد استفاده قرار می‌گیرد. این الگوریتم، مانند سایر الگوریتم‌های فرا ابتکاری، تلاش می‌کند تا مسائل بهینه‌سازی ترکیبی سخت را از طریق یک تصادفی کنترل‌شده، حل کند. سهولت استفاده و توانایی آن برای ارائه یک راه‌حل مناسب برای مسائل دنیای واقعی از جستجوی هارمونی، یک روش حل مسائل بهینه‌سازی استوار و محبوب ساخته است.

در مسائل بهینه‌سازی چندهدفه، دو یا چند هدف باید به صورت هم‌زمان بهینه شوند که عموماً در تضاد با یکدیگر قرار دارند. برای بهینه‌سازی چندهدفه راه‌های متفاوتی ارائه شده است که ساده‌ترین آنها، استفاده از ضرایب وزنی و تبدیل آن به یک مسئله تک هدفه است؛ اما راه‌حل بهتر، استفاده از مفهوم بهینه پارتو است. در این روش مجموعه‌ای از جواب‌های بهینه ارائه می‌شود که هیچ‌یک بر دیگری برتری مطلق ندارد. یکی از روش‌های بهینه‌سازی مبتنی بر مفهوم بهینه پارتو، الگوریتم ژنتیک مرتب‌سازی نامغلوب-دو است که

1. Li

کارایی مناسبی دارد؛ اما کارهای انجام گرفته در این زمینه با استفاده از جستجوی هارمونی شامل موارد زیر است:

الف) همانطور که میدانیم هارمونی جدید در مرحله به روزرسانی با ضعیف ترین هارمونی مقایسه می شود تا در صورت برتری جایگزین آن شود. در این روش، در مرحله به روزرسانی اگر هارمونی جدید بتواند بدترین هارمونی حافظه را مغلوب نماید جایگزین آن می شود. بقیه مراحل جستجو مشابه الگوریتم پایه است.

ب) در روش دیگری که ارائه شده، در مرحله تولید هارمونی جدید، یک حافظه هارمونی جدید تولید می شود. در مرحله به روزرسانی حافظه جدید و قبلی با یکدیگر ترکیب شده و بر اساس مفهوم بهینه پارتو رتبه بندی شده و هارمونی های دارای رتبه بهتر، حافظه هارمونی را در تکرار بعد تشکیل می دهند همچنین معیار فاصله ازدحامی نیز برای تعیین اولویت هارمونی های هم رتبه مورد استفاده قرار گیرد سایر مراحل مشابه الگوریتم پایه است (سیواسابرامانی و سواراپ<sup>۱</sup>، ۲۰۱۱)

کارایی الگوریتم MOHS<sup>۲</sup> بستگی به تعریف پارامترهای کنترل زیر دارد:

۱. احتمال انتخاب از حافظه هارمونی<sup>۳</sup> (HMCR): با استفاده از حافظه این اطمینان حاصل می شود که متغیرهای خوب و جدیدی به عنوان عناصر راه حل های جدید انتخاب شوند. معمولاً مقداری بین ۰/۷ تا ۰/۹۵ برای این پارامتر انتخاب می شود.

۲. احتمال تنظیم گام<sup>۴</sup> (PAR): برای کنترل تغییرات می توان از پارامتر احتمال تنظیم گام استفاده کرد که به عبارت دیگر احتمال تغییر جزئی مقدار انتخاب شده از حافظه هارمونی است که معمولاً مقداری بین ۰/۱ تا ۰/۵ به این پارامتر تخصیص داده می شود.

۳. حداکثر تکرارها<sup>۵</sup> (MaxIt): تعداد راه حل های ایجاد شده با MaxIt تعریف می شود که وابسته به اندازه مسئله است. در این مقاله  $MaxIt = 100$  است.

1. Sivasubramani and Swarup

۲. Multi Objective Harmony Search

۳. Harmony Memory Considering Rate

۴ Pitch Adjustment Rate

۵. Maximum Iteration

۴. اندازه حافظه هارمونی<sup>۱</sup> (HMS): تعداد ردیف‌ها در این ماتریس حافظه به تعداد HMS است که هر ردیف به معنی یک راه‌حل یا جواب مسئله است که به صورت تصادفی تولید شده‌اند.

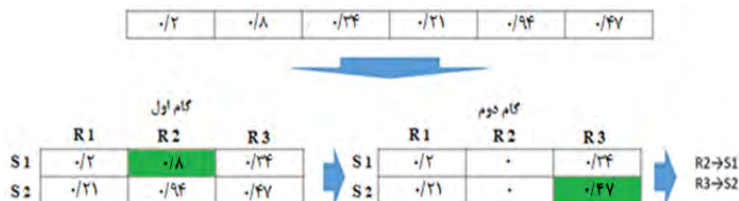
### روش نمایش جواب

برای نمایش جواب در این مسئله از یک رشته اعداد اعشاری بین صفر و یک که به صورت تصادفی ایجاد شده‌اند استفاده می‌کنیم که با نگاهی که در ادامه ذکر می‌شود، به یک جواب از مسئله ختم می‌گردد. بخش اول نمایش جواب از (تعداد ربات \* تعداد ایستگاه) عضو تشکیل شده است. در این ماتریس  $S * R$ ، هر سطر مربوط به یک ایستگاه و هر ستون مربوط به یک ربات است. به ترتیب از ایستگاه اول، در سطر مربوط به هر ایستگاه، مقدار بیشینه ستون‌های ماتریس، نشان‌دهنده ستون ربای است که به آن ایستگاه تخصیص داده می‌شود. پس از تخصیص آن ربات به ایستگاه مورد نظر، ستون مربوط به آن ربات صفر می‌شود و برای ایستگاه‌های بعدی (سطرهای بعدی) تا تخصیص ربات به همه ایستگاه‌ها این مراحل تکرار می‌شود. شکل ۲ را ببینید طول رشته اعداد اعشاری ۶ است از آنجایی که در مثال مورد نظر ۲ ایستگاه و ۳ ربات برای تخصیص در نظر گرفته شده است، این رشته جواب به یک ماتریس  $۲ * ۳$  که سطر آن تعداد ایستگاه و ستونش تعداد ربات را نشان می‌دهد، تبدیل می‌شود. در گام اول، سطر اول بیشترین مقدار مربوط به ستون دوم است پس ربات دوم را به ایستگاه اول تخصیص می‌دهیم و چون هر ربات تنها می‌تواند به یک ایستگاه تخصیص یابد ستون مربوط به آن را صفر می‌کنیم و در گام دوم همین مراحل برای سطر دوم تکرار می‌شود که نتیجه تخصیص ربات سوم به ایستگاه دوم است.

---

۱. Harmony Memory Size

شکل ۲. نحوه نمایش جواب تخصیص ربات‌ها به ایستگاه‌های کاری



بخش بعدی نمایش جواب مربوط به تخصیص فعالیت‌ها به ایستگاه‌های کاری است. طول این بخش از نمایش جواب  $(1 - (\text{تعداد ایستگاه} * 2) + \text{تعداد فعالیت})$  است. برای تخصیص، با مرتب‌سازی این بخش، به یک جایگشت با همان طول می‌رسیم. در این جایگشت، اعداد مساوی شماره فعالیت است. به‌طور مثال برای ۹ فعالیت و ۳ ایستگاه کاری که خود شامل دو بخش ورودی و خروجی است طول نمایش جواب  $(9 - 1) + (2 * 3)$  است، در شکل ۳ مرتب‌سازی به این صورت است که در رشته اعداد اعشاری کوچک‌ترین مقدار مربوط به  $0.12$  است که یازدهمین خانه از سمت چپ است. در جایگشت نیز ترتیب کوچک به بزرگ را از سمت چپ آغاز می‌کنیم پس خانه اول مقدار ۱۱ می‌گیرد و به همین ترتیب ادامه می‌دهیم، در صورتی که دو عدد تصادفی ایجاد شده یکسان باشند اولویت با عدد سمت چپ است و در آخر بعد از مشخص کردن ۹ خانه اول با رنگ سبز که شماره فعالیت است بقیه خانه‌ها، به‌عنوان جداکننده با رنگ صورتی مشخص می‌شود چون تعداد بخش‌های قابل تخصیص ۶ خانه است (تعداد ایستگاه‌ها ۳ و هر ایستگاه شامل دو بخش ورودی و خروجی است)، تعداد جداکننده‌ها شامل ۵ خانه می‌شود.

شکل ۳. نحوه تبدیل یک رشته اعداد اعشاری به جواب مسئله برای تخصیص فعالیت به ایستگاه کاری





در این مرحله برای اعمال محدودیت‌های پیش‌نیازی خانه‌های مربوط به فعالیت‌ها، با ترتیب پیش‌نیازی مرتب می‌گردند. پس از این مرتب‌سازی، هم تأثیر ترتیب اولیه و هم تأثیر ترتیب با پیش‌نیازی، دیده می‌شود، در شکل ۳ اولین خانه، مربوط به فعالیت ۷ است اما از آنجایی که فعالیت‌های پیش‌نیاز آن یعنی فعالیت ۴ و ۳ و ۲ و ۱ انجام نشده است، اول شروع به تخصیص این فعالیت‌ها به ترتیب پیش‌نیازی می‌کنیم. پس از تخصیص ربات‌ها و فعالیت‌ها به ایستگاه‌های کاری، با استفاده از نمایش چگونگی قرارگیری فعالیت‌ها در خط مونتاژ، زمان انجام فعالیت‌ها، توالی فعالیت و به دنبال آن آماده‌سازی بین فعالیت‌ها) به دست می‌آید.

### یافته‌های پژوهش

با گسترش ابعاد مسئله حل مسائل از طریق نرم‌افزار گمز بسیار زمان‌بر است و برای رفع این مشکل از نرم‌افزار متلب برای حل مسائل در سایزهای مختلف استفاده می‌شود الگوریتم پیشنهادی این تحقیق الگوریتم جستجوی هارمونی چندهدفه است. یافته‌های پژوهش در این بخش ارائه شده است و مسئله‌های مختلفی در سه سایز کوچک، متوسط و بزرگ، برای ارزیابی عملکرد الگوریتم پیشنهادی مورد مطالعه قرار گرفته است. نمودار پیش‌نیازی، تعداد فعالیت‌ها، زمان‌های آماده‌سازی بین فعالیت‌ها و تعداد ایستگاه‌ها از سایت [www.assembly-line-balancing.de](http://www.assembly-line-balancing.de) دریافت می‌شوند؛ اما از آنجاکه در مثال‌های موجود در این پایگاه تمام جزئیات مورد نیاز این مقاله، نظیر تعداد روبات، هزینه روبات، زمان تعمیرات و نگهداری پیشگیرانه موجود نبود این مقادیر به شکل تصادفی و متناسب با اندازه مسئله ایجاد شدند. پارامترهای ورودی الگوریتم پیشنهادی و سطوح انتخابی آنها در جدول ۱ تعیین می‌شود.

جدول ۱. پارامترها و سطوح آنها برای الگوریتم MOHS

الگوریتم MOHS			سطح
زیاد	متوسط	کم	پارامتر
۳۰۰	۲۰۰	۱۰۰	حداکثر تکرار
۷۰	۵۰	۳۰	اندازه حافظه هارمونی
۰/۹۵	۰/۸۵	۰/۷۵	نرخ انتخاب از حافظه هارمونی
۰/۵	۰/۳	۰/۱	نرخ تنظیم گام

به منظور تعیین کاراترین سطوح هر پارامتر به روش تاگوچی، با استفاده از نرم افزار minitab 18 تعدادی آزمایش را تعیین می کنیم. با توجه به جبهه های پارتوی حاصل از اجرای هر آزمایش، شاخص هایی را همان طور که در جدول ۱ آورده ایم برای هر آزمایش حساب می کنیم. نتایج نرمالایز شده و به منظور ایجاد یک خروجی از هر آزمایش با استفاده از روش وزن دهی تمامی شاخص ها به یک پاسخ تبدیل خواهد شد. در نهایت پس از انجام تنظیمات، خروجی نرم افزار برای الگوریتم جستجوی هارمونی به صورت شکل ۴ خواهد بود. همان طور که در شکل ۴ می بینیم پارامتر حداکثر تکرار بهتر است در سطح اول خود تنظیم شود و سطح دوم برای پارامتر HMS، سطح اول برای پارامتر HMCR و سطح دوم برای پارامتر PAR، کاراترین سطوح این پارامترهاست.

شکل ۴. میانگین میانگین ها و نسبت S/N برای پارامترهای الگوریتم MOHS



جدول ۲ مقادیر زمان سیکل و هزینه را در سطوح مختلف عدم قطعیت با استفاده از الگوریتم جستجوی هارمونی با رویکرد استوار و همچنین روش اپسیلون محدودیت تقویت شده نشان می‌دهد. برای ارزیابی پاسخ‌ها حداکثر زمان ۵۰۰ ثانیه برای هر دو روش در نظر گرفته شد. همان‌طور که در این جدول مشاهده می‌شود، با توجه به شرایط و ویژگی‌های مسئله روش اپسیلون محدودیت تقویت شده تنها توانست برای ابعاد خیلی کوچک (دو مسئله از ۸ مسئله) و در سقف زمان ۵۰۰ ثانیه به پاسخ برسد و برای ابعاد بالاتر، امکان دستیابی به پاسخ فراهم نبود. چنین چیزی نیاز استفاده از الگوریتم‌های فرا ابتکاری، نظیر الگوریتمی که در این مقاله ارائه شده است را برای این مسئله ضروری تر می‌کند. مقادیر حداقل، حداکثر و متوسط توابع هدف و متوسط زمان حل گزارش شده در این جداول نتیجه ۵ بار تکرار برای هر مسئله در هر یک از سطوح عدم قطعیت است.

جدول ۲. نتایج مقادیر توابع هدف برای مسائل و سطوح مختلف  $\alpha$  با استفاده از الگوریتم MOHS و اپسیلون محدودیت تقویت شده

مسئله	$\alpha$	MOHS											
		زمان سیکل					زمان (ثانیه)	اپسیلون محدودیت تقویت شده					
		حداقل	حداکثر	متوسط	حداقل	حداکثر		متوسط	حداقل	حداکثر	متوسط	حداقل	حداکثر
۴-۷	۰/۱	۱۶۵/۹	۵۲۷/۶	۳۷۵/۱	۱۵۵۰	۲۰۹۳۰	۱۷۶۷۵	۲۵/۳۳	۳۷۶,۳	۱۱	۱۵۶۱۵,۸	۱۴۹۲۹,۸	۴۰,۲
	۰/۲	۱۸۵/۱	۵۴۵/۱	۳۸۶/۹	۱۶۳۱/۰	۲۱۵۴۰	۱۷۲۸۹	۳۷۶,۸	۱۱	۱۶۰۷۰,۶	۱۵۶۳۴,۷		
	۰/۳	۱۸۹/۱	۵۴۶/۷	۳۹۵/۸	۱۶۷۷۰	۲۲۱۵۰	۱۸۷۰۵	۳۸۵,۷	۱۱	۱۶۵۲۵,۴	۱۵۷۹۹,۵		
	۰/۴	۱۹۸/۳	۵۶۲/۳	۴۳۰/۴	۱۷۲۳۰	۲۱۳۸۰	۱۸۹۰۱	۳۹۴,۶	۱۱	۱۶۹۸۰,۳	۱۶۲۳۴,۴		
	۰/۵	۲۰۰	۵۵۷/۶	۳۹۱/۳	۱۷۶۹۰	۲۳۳۷۰	۱۹۴۹۵	۴۰۳,۵	۱۱	۱۷۴۳۵,۱	۱۶۶۶۹,۲		
	۰/۶	۲۰۲/۵	۵۸۹/۵	۴۳۰/۵	۱۸۱۵۰	۲۳۹۸۰	۲۰۵۸۲	۴۱۲,۴	۱۱	۱۷۸۸۹,۹	۱۷۱۰۴,۱		
	۰/۷	۱۹۹/۲	۶۰۹/۴	۴۳۷/۲	۱۸۶۱۰	۲۴۵۹۰	۲۰۷۸۸	۴۲۱,۳	۱۱	۱۸۳۴۴,۸	۱۷۵۳۸,۹		
	۰/۸	۲۱۳	۶۱۵/۶	۴۳۷	۱۹۰۸۰	۲۵۲۰۰	۲۰۹۸۸	۴۳۰,۲	۱۱	۱۸۷۹۹,۶	۱۷۹۷۳,۸		
	۰/۹	۲۱۳/۲	۶۳۹	۴۳۸	۱۹۵۴۰	۲۵۸۱۰	۲۱۷۰۴	439.1	۱۱	۱۹۲۵۴,۴	۱۸۴۰۸,۶		
۹-۱۱	۰/۱	۲۰۰/۳	۳۵۵/۱	۲۷۹/۵	۲۰۵۲۰	۲۲۲۴۰	۲۱۰۶۵	۲۷/۶۹	۳۸۱,۲	۱۷۲	۱۹۹۶۱,۴	۱۹۴۸۷,۶	۱۵۳,۲
	۰/۲	۲۱۰/۵	۳۶۸/۱	۲۸۷/۹	۲۱۱۲۰	۲۲۸۹۰	۲۱۵۶۷	۳۹۰,۴	۱۷۶,۲	۲۰۵۴۲,۸	۲۰۰۵۲,۲		
	۰/۳	۲۱۴/۸	۳۸۱/۲	۲۹۹/۴	۲۱۷۱۰	۲۳۵۳۰	۲۲۲۸۸	۳۹۹,۶	۱۸۰,۳	۲۱۱۲۴,۲	۲۰۶۲۲,۸		
	۰/۴	۲۱۳	۳۷۹/۳	۳۰۵/۱	۲۲۳۱۰	۲۴۱۸۰	۲۲۷۸۴	۴۰۸,۸	۱۸۴,۵	۲۱۷۰۵,۶	۲۱۱۹۰,۴		
	۰/۵	۲۲۳/۳	۳۹۲/۴	۳۱۲/۱	۲۲۹۱۰	۲۴۸۳۰	۲۳۴۸۹	418	۱۸۸,۶	۲۲۲۸۷	۲۱۷۵۸		
	۰/۶	۲۲۱/۵	۴۰۰/۴	۳۱۷/۷	۲۳۵۱۰	۲۵۴۸۰	۲۴۱۸۴	427.2	۱۹۲,۷	۲۲۸۶۸,۴	۲۲۳۲۵,۶		
	۰/۷	۲۲۵/۸	۴۰۸/۵	۳۳۵/۹	۲۴۱۰۰	۲۴۸۳۰	۲۴۵۲۵	436.4	۱۹۶,۸	23449.8	۲۲۸۹۳,۲		
	۰/۸	۲۳۶	۴۱۶/۶	۳۴۳/۱	۲۴۷۰۰	۲۵۴۴۰	۲۵۰۰۸	445.6	۲۰۱	۲۴۰۳۱,۲	۲۳۴۶۰,۸		
	۰/۹	۲۲۸/۳	۴۲۴/۷	۳۵۹/۴	۲۵۳۰۰	۲۷۴۲۰	۲۵۸۳۸	454.8	۲۰۵,۱	۲۴۶۱۱,۶	۲۴۰۲۸,۴		
۱۱	۰/۱	۳۸۴/۹	۴۱۰/۳	۳۹۱/۶	۱۹۷۰۰	۲۰۹۰۰	۲۰۰۴۰	۲۳/۸۵					
	۰/۲	۳۸۹/۸	۴۰۴/۶	۳۹۸/۸	۲۰۲۸۰	۲۱۵۱۰	۲۰۴۵۵						
	۰/۳	۳۹۶/۶	۴۲۵/۸	۴۰۷/۶	۲۰۸۵۰	۲۰۸۵۰	۲۰۸۵۰						



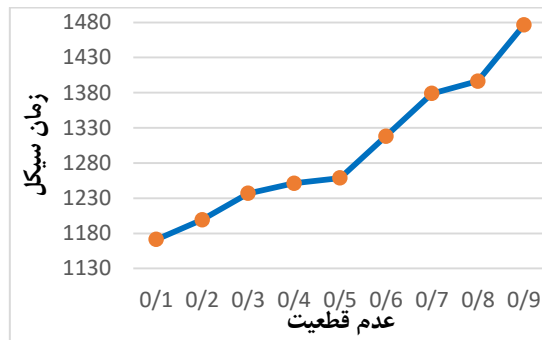
۰/۳	۱۸۵۲۰	۲۱۰۹۰	۱۹۷۴۰	۵۰۰۱۰	۵۲۹۰۰	۵۰۶۰۱	---	---	---	---
۰/۴	۱۸۹۰۰	۲۲۲۲۰	۱۹۶۳۰	۵۰۰۱۰	۵۱۲۴۰	۵۰۳۰۲	---	---	---	---
۰/۵	۱۸۶۹۰	۲۰۵۰۰	۱۹۷۰	۵۰۰۱۰	۵۰۱۳۰	۵۰۰۴۰	---	---	---	---
۰/۶	۱۸۶۲۰	۲۰۷۵۰	۱۸۷۸۰	۵۰۰۱۰	۵۰۲۴۰	۵۰۱۰۲	---	---	---	---
۰/۷	۱۸۷۳۰	۲۱۸۱۰	۱۹۱۱۰	۵۰۰۱۰	۵۱۸۹۰	۵۰۵۶۸	---	---	---	---
۰/۸	۲۰۲۴۰	۲۲۵۶۰	۲۰۷۸۰	۵۰۰۱۰	۵۲۱۳۰	۵۰۷۵۵	---	---	---	---
۰/۹	۱۸۶۱۰	۲۱۳۸۰	۲۰۹۲۰	۵۰۰۱۰	۵۲۰۱۰	۵۰۴۷۳	---	---	---	---

با توجه به جدول ۲ می توان به نتایج زیر رسید:

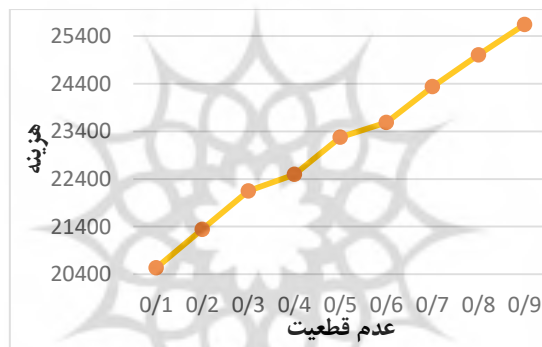
- در صورتی که مسائل با ابعاد بزرگ توسط MOHS حل شوند و مستلزم صرف زمان زیادی نمی باشد. این در حالی است که روش اسپیلون محدودیت های تقویت شده، تنها قادر است، مسائلی با این شرایط را تنها با ابعاد بسیار کوچک و در زمان معقولی حل کند.
- عملکرد رویکرد استوار نشان دهنده این است که در شرایط واقعی و در نظر گرفتن احتمال رویداد خرابی مقادیر زمان سیکل مقدار قابل توجهی تغییر می کند که این خود نشان دهنده ضرورت در نظر گرفتن عدم قطعیت به خصوص خرابی در خطوط مونتاژ رباتیک است.
- در مسائلی که در آنها شاهد خرابی و یا انجام عملیات نگهداری و تعمیر ربات هستیم با افزایش عدم قطعیت، زمان نگهداری و تعمیر پس از خرابی نیز افزایش و در نتیجه مقادیر زمان سیکل نیز روند رو به افزایشی دارد، دلیل این امر غیرقطعی بودن پارامترهای زمان نگهداری و زمان تعمیر ربات بود.
- به دلیل غیرقطعی بودن هزینه احداث ربات با افزایش عدم قطعیت، مجموع هزینه های احداث ربات ها نیز افزایش می یابند.

شکل ۵ و ۶ جزئیات تغییرات متوسط زمان سیکل و هزینه در سطوح مختلف عدم قطعیت را برای مثال چهارم نشان می دهد (با اختلافات جزئی قابل تعمیم به مسائل دیگر نیز هست).

شکل ۵. متوسط زمان سیکل برای مسئله ۲۱ فعالیت



شکل ۶. متوسط هزینه برای مسئله ۲۱ فعالیت



همان‌طور که در شکل ۵ و ۶ مشاهده می‌شود، با افزایش سطح عدم قطعیت به‌طور کلی مقادیر متوسط توابع هدف (این الگو برای مقادیر حداقل، حداکثر نیز صدق می‌کند) روند رو به افزایشی دارد.

### نتیجه‌گیری و پیشنهادها

در سال‌های اخیر، خط مونتاژ رباتیک II-شکل توجه زیادی را به خود جلب کرده و در بسیاری از کشورهای توسعه‌یافته کاربرد دارد. تغییرات سریع در تکنولوژی و تقاضای مشتریان برای انواع محصولات بیشتر، الزامات انعطاف‌پذیری تولید را افزایش می‌دهد. خطوط مونتاژ رباتیک نقش مهمی در توسعه انعطاف‌پذیری مطلوب ایفا می‌کند. این تحقیق

مسئله بالانس و توالی خط مونتاژ رباتیک II- شکل مدل ترکیبی را با در نظر گرفتن زمان آماده‌سازی بین فعالیت‌ها و زمان نگهداری و تعمیرات و زمان خرابی مربوط به هر ربات، مورد بررسی قرار می‌دهد، این مدل برنامه‌ریزی مختلط عدد صحیح به منظور حداقل کردن زمان سیکل و به حداقل رساندن هزینه‌های ربات توسعه یافته است. این مدل پس از اعتبار سنجی، به کمک روش اپسیلون محدودیت تقویت شده حل گردید. اما نتایج نشان می‌داد که این روش، تنها قادر است، مسائلی با ابعاد بسیار کوچک را در زمان معقولی حل کند و در مسائلی که ابعاد مقداری افزایش می‌یافت این روش قادر به پاسخگویی نبود. این در حالی است که در اغلب خطوط روباتیک به دلیل هزینه‌های بالای ربات‌ها، مدیران کارخانجات تمایل دارند خطوط بهره‌وری داشته باشند و اگر اتفاق غیر منتظره‌ای پیش آمد بتوانند در سریع‌ترین زمان ممکن واکنش لازم را داشته باشند. از این رو برای جلوگیری از زمان محاسباتی طولانی مسائل سائز بزرگ، از الگوریتم جستجوی هارمونی چندهدفه به عنوان روش فرا ابتکاری برای حل در نرم‌افزار متلب استفاده شد.

مطالعه ارائه شده در این مقاله اولین مطالعه در زمینه خط مونتاژ رباتیک II- شکل با استفاده از حل استوار می‌باشد. این تحقیق نقطه شروع خوبی برای مطالعات آینده است که بر روی حل استوار تمرکز دارد. برای تحقیقات آینده، می‌توان از دیگر روش‌های فرا ابتکاری (الگوریتم فاخته، جستجوی ممنوعه، الگوریتم رقابت استعماری، و غیره) برای بدست آوردن نتایج احتمالاً بهتری استفاده کرد. یا پیشنهاد دیگری که می‌توان داد این است که طرح‌های مختلف خط مونتاژ، مانند خطوط مونتاژ دوطرفه و یا عدم قطعیت در تقاضا مورد توجه قرار گیرد. علاوه بر این، در مورد داده‌های فازی و احتمالی، نیز می‌توان این کار را گسترش داد.



## ORCID

Parvaneh Samouei

<http://orcid.org/0000-0003-0867-7617>

## منابع

- Aase, G.R., Olson, J.R., Schniederjans M.J (2004), “*U-shaped assembly line layouts and their impact on labor productivity: an experimental study*”, European Journal of Operational Research, 156(3), 698–711.
- Akpınar, S., Baykasoglu, A (2014), “*Modeling and solving mixed-model assembly line balancing problem with setups. Part I: a mixed integer linear programming model*”, Journal of Manufacturing Systems, 33 (1), 177–187.
- Ben-Tal, A., Ghaoui, L.E., Nemirovski, A (2009), “*Robust optimization*”, Princeton series in applied mathematics, Princeton University press.
- Hazir, Ö., Dolgui, A (2013), “*Assembly line balancing under uncertainty: Robust optimization models and exact solution method*”, Computers & Industrial Engineering, 65, 261–267.
- Hazir, Ö., Dolgui, A (2015), “*A decomposition-based solution algorithm for utype assembly line balancing with interval data*”, Computers & Operations Research, 59, 126–131.
- Hwang, R.K., Katayama, H., Gen, M (2008), “*U-shaped assembly line balancing problem with genetic algorithm*”, International Journal of Production Research, 46(16), 4637-4649.
- Jin, W., He, Z., Wu, Q. (2021), “*Robust optimization of resource-constrained assembly line balancing problems with uncertain operation times*”, Engineering Computations, Vol. ahead-of-print No. ahead-of-print. <https://doi.org/10.1108/EC-01-2021-0061>.
- Li, Z., Ding, R., Floudas, C.A. (2011), “*A Comparative Theoretical and Computational Study on Robust Counterpart Optimization: I. Robust Linear Optimization and Robust Mixed Integer Linear Optimization*”, Industrial & Engineering Chemistry Research, 50(18), 10567–10603.
- Liu, X., Yang, X., Lei, M. (2021). “*Optimisation of mixed-model assembly line balancing problem under uncertain demand*”. Journal of Manufacturing Systems, 59, 214-227.
- Lu, Z., Cui, W., Han, X (2015), “*production and preventive maintenance scheduling for a single machine with failure uncertainty*”, Computers & Industrial Engineering, 80, 236–244.
- Miltenburg, J (2001), “*U-shaped production lines: a review of theory and practice*”, International Journal of Production Economics, 70(3), 201–214.
- Nilakantan, J.M., Nielsen, I., Ponnambalam, S.G., Venkataramanaiah, S (2016), “*Differential evolution algorithm for solving RALB problem using cost and time-based models*”, International Journal of Advanced Manufacturing Technology, 89, 311–332.
- Nilakantan, J.M., Ponnambalam, S.G. (2015), “*Robotic U-shaped assembly line balancing using particle swarm optimization*”, Engineering Optimization, 48(2), 231-252.
- Nilakantan, J.M., Ponnambalam, S.G., Jawahar, N (2016), “*Design of energy efficient RAL system using evolutionary algorithms*”, Engineering Computations, 33(2), 580-602.

- Nourmohammadi, A., Zandieh, M., Tavakkoli-Moghaddam, R. (2013), "An imperialist competitive algorithm for multi-objective U-type assembly line design", *Journal of Computational Science*.
- Oksuz, M.K., Buyukozkan, K., Satoglu, S.I. (2017), "U-shaped Assembly Line Worker Assignment and Balancing Problem: A Mathematical Model and Two Meta-heuristics", *Computers & Industrial Engineering*, 112, 246-263.
- Özcan, U. (2019). "Balancing and scheduling tasks in parallel assembly lines with sequence-dependent setup times", *International Journal of Production Economics*, *International Journal of Production Economics*, 213, 81-96.
- Pereira, J. (2018), "The Robust (minmax regret) assembly line worker assignment and balancing problem", *Computers and Operations Research*, 93, 27-40.
- Pereira, J., Miranda, E.A. (2017), "An exact approach for the robust assembly line balancing problem", *Omega*, 78, 85-98.
- Pereira, J., Ritt, M., Vasquez, O.C. (2018), "A memetic algorithm for the Cost-oriented Robotic Assembly Line Balancing Problem", *Computers and Operations Research*, 99, 249-261.
- Purnomo, H.D., Wee, H.M. (2014), "Maximizing production rate and workload balancing in a two-sided assembly line using Harmony Search", *Computers & Industrial Engineering*, 76, 222-230.
- Rabbani, M., Moghaddam, M., Manavizadeh, N (2012), "Balancing of mixed model two-sided assembly lines with multiple u-shaped layout", *The International Journal of Advanced Manufacturing Technology*, 59, 1191-1210.
- Rabbani, M., Mousavi, Z. Farrokhi-Asl, H (2016), "Multi-objective metaheuristics for solving a type II robotic mixed-model assembly line balancing problem", *Journal of Industrial and Production Engineering*, 33(7), 472-484.
- Rubinovitz, J., Bukchin, J. (1991). "Design and balancing of robotic assembly lines". In: *Proceedings of the fourth world conference on robotics research*, Pittsburgh, PA.
- Samouei, P. & Ashayeri, J (2019), "Developing Optimization & Robust Models for a Mixed-Model Assembly Line Balancing Problem with Semi-Automated Operations", *Applied Mathematical Modelling*, 72, 259-275.
- Sirovetnukul, R. & Chutima, P (2010), "The impact of walking time on U-shaped assembly line worker allocation problems", *Engineering Journal*, 14(2), 53-78.
- Sivasubramani, S. & Swarup, K.S. (2011), "Multi-objective harmony search algorithm for optimal power flow problem", *Electrical Power and Energy Systems* 33(3), 745-752.
- Sobaszek Ł., Gola A., Świć A. (2022), "The Algorithms for Robust Scheduling of Production Jobs Under Machine Failure and Variable Technological Operation Times", In: Machado J., Soares F., Trojanowska J., Ivanov V. (eds) *Innovations in Industrial Engineering. icieng 2021. Lecture Notes in Mechanical Engineering*. Springer, Cham. [https://doi.org/10.1007/978-3-030-78170-5\\_6](https://doi.org/10.1007/978-3-030-78170-5_6).
- Thomopoulos, N.T. (2014) "Assembly line planning and control", *Stuart School of Business Illinois Institute of Technology, Chicago, IL, USA*
- Toksari, M.D., İşleyen, S.K., Güner, E. & Baykoç, Ö.F. (2008), "Simple and U-type assembly line balancing problems with a learning effect", *Applied Mathematical Modelling*, 32(12), 2954-2961.

- Yılmaz, Ö. F. (2020). "Robust optimization for U-shaped assembly line worker assignment and balancing problem with uncertain task times". Croatian Operational Research Review, 229-239.
- Zhang, Z., Tang, Q. & Zhang, L (2019), "Mathematical model and grey wolf optimization for low-carbon and low-noise U-shaped robotic assembly line balancing problem", Journal of Cleaner Production, 215, 744-756.
- Zhang, Z., Tang, Q., Chica, M. (2021). "A robust MILP and gene expression programming based on heuristic rules for mixed-model multi-manned assembly line balancing". Applied Soft Computing, 107513.
- Zhou, B., Wu, Q. (2020). "Decomposition-based bi-objective optimization for sustainable robotic assembly line balancing problems". Journal of Manufacturing Systems, 55, 30-43.

