Archive of SID

Mazandaran University of Science and Technology

Available online at www.iiec2017.com

13th International Conference on Industrial Engineering
(IIEC 2017)

Iran Institute of Industrial Engineering

بکارگیری الگوریتم فراابتکاری خفاش و خفاش پویا با استفاده از جستجوی همسایگی متغیر برای حل مساله چیدمان تک سطری تسهیلات (IIEC 2017)

راضیه موسوی[۱]، دکتر سید مصطفی خرمی‌زاده[۲]. دکتر مرتضی کاظمی[۳]. سید امیرحسین صالحی امیری[۴]

[۱] دانشجوی کارشناسی ارشد رشته مهندسی صنایع دانشگاه صنعتی شیراز؛ Razieh.msv@gmail.com

[۲] استاد مشاور، دانشکده ریاضی دانشگاه صنعتی شیراز؛ Mkhorramizadh@sutech.ac.ir

[۳] استاد راهنما، گروه مهندسی صنایع دانشگاه صنعتی شیراز؛ Kaazemi@sutech.ac.ir

[۴] دانشجوی کارشناسی ارشد رشته مهندسی صنایع دانشگاه صنعتی شیراز؛ Amirhossein.salehi.ie@gmail.com

چکیده

مساله چیدمان تک‌سطری تسهیلات در زمره مسائل NP- سخت طبقه‌بندی می‌گردد و هدف آن قرار دادن n تسهیل در یک خط مستقیم است به‌طوری‌که مجموع وزنی جریان بین تسهیلات حداقل گردد. حل مساله چیدمان تک‌سطری تسهیلات با استفاده از الگوریتم‌های دقیق در یک زمان چندجمله‌ای ممکن نیست و با افزایش تعداد تسهیلات، زمان حل بصورت نمایی افزایش می‌یابد، بنابراین در این مقاله با استفاده از الگوریتم‌های فراابتکاری به حل مساله می‌پردازیم تا در زمانی مناسب به جوابی خوب دست یابیم. در این مقاله برای نخستین بار با استفاده از الگوریتم فراابتکاری خفاش و الگوریتم فراابتکاری جستجوی همسایگی متغیر به حل مساله چیدمان تک‌سطری می‌پردازیم. در این مقاله ابتدا الگوریتم خفاشی متناسب با مساله چیدمان تک‌سطری پیشنهاد داده و در محیط نرم‌افزار سی‌شارپ پیاده‌سازی می‌کنیم. پس از آن مساله را با استفاده از الگوریتم فراابتکاری خفاش پویا حل کرده و جواب بدست آمده از هر دو روش را با یکدیگر مقایسه می‌کنیم.

کلمات کلیدی

چیدمان تک‌سطری تسهیلات، الگوریتم فراابتکاری خفاش، الگوریتم فراابتکاری خفاش پویا، جستجوی همسایگی متغیر.

Application of Bat Algorithm and Dynamic Bat Algorithm Using Variable Neighborhood Search for Solving Single Row Facility Layout Problem

Razieh Mousavi, Dr Morteza Kazemi, Dr Seyyed Mosftafa Khoramizadeh, Seyyed AmirHossein Salehi

Master of Science student of Industrial Engineering, Industrial Engineering Department, Mathematics Department, Master of Science student of Industrial Engineering

ABSTRACT

Single Row Facility Layout is the NP-hard problem and its purpose is to put n facilities in a straight line to minimize the weighted sum of the distances between facility pairs. Solving Single Row Facility Layout Problem using exact algorithms in a polynomial time is impossible and by increasing the number of facilities, solution time increases exponentially. Therefore, in this paper, we solve the problem using meta-heuristic algorithms to achieve a good answer in an appropriate time. For the first time in this paper, we solved Single Row Facility Layout Problem using Bat Algorithm and Variable Neighborhood Search therefore we first proposed an algorithm corresponding to single row facility layout and implement it in C# software environment. Then, we solved the problem using Dynamic Bat Algorithm and the answer from both methods compared with each other.

KEYWORDS

Single Row Facility Layout, Bat Algorithm, Dynamic Bat Algorithm, Variable Neighborhood Search.

www.SID.ir

Archive of SID

۱- مقدمه

چیدمان تسهیلات یکی از مسائل مهمی است که امروزه در صنایع تولیدی و غیرتولیدی مورد توجه قرار گرفته‌است. بطور کلی در چیدمان تسهیلات ترتیب قرارگیری اجزای فیزیکی یک سیستم تولیدی یا خدماتی شامل ماشین‌آلات، تجهیزات تولید، امکانات خدماتی، دپارتمان‌ها، انبار و ... و نحوه تعامل بین این واحدها مشخص می‌گردد. چیدمان مناسب موجب روان بودن جریان مواد در کارخانه می‌شود، کارهای نیمه تمام و زمان کل تولید را کاهش داده و کارایی کل سیستم تولیدی را افزایش می‌دهد. بر طبق نظریه فرانسیس[۱]، هزینه جریان مواد ۲۰ تا ۵۰ درصد کل هزینه های عملیاتی و هم چنین ۱۵ تا ۷۰ درصد هزینه یک محصول را در بردارد؛ چیدمان مناسب می‌تواند ۱۰ تا ۳۰ درصد در این هزینه صرفه‌جویی نماید [۱]. مساله چیدمان تک‌سطری تسهیلات یکی از مسائل مهمی است که توجه محققان زیادی را به خود جلب کرده و به علت کاربرد وسیعش مورد استفاده قرار می‌گیرد. از جمله کاربردهای این مساله می توان به چیدن اتاق ها در یک بیمارستان، چیدن دپارتمان ها در یک ساختمان اداری، قراردادن کتاب‌های یک کتابخانه در یک قفسه، تخصیص هواپیماها به گیت‌های ورودی و چیدن ماشین‌آلات تولیدی در یک خط اشاره کرد [۲].

۲- مرور ادبیات

برای حل مساله چیدمان تک‌سطری تسهیلات روش‌های مختلفی پیشنهاد شده است. اولین بار سیمونز در سال ۱۹۶۹ روش شاخه و کران را پیشنهاد داد که مسائلی با حداکثر ابعاد یازده تسهیل را حل کرد [۳]. در سال ۱۹۷۰ هال مساله چیدمان تک سطری تجهیزات را با روش برنامه ریزی پویا حل کرد و پس از آن پیکارد و کویران در سال ۱۹۸۱ از روش برنامه‌ریزی پویا استفاده کرد [۴]. این روش نیز توانایی حل مساله با ابعاد حداکثر ۱۵ تجهیز را دارد. آنجوس و همکاران در سال ۲۰۰۵ از برنامه‌ریزی نیمه معین استفاده کردند و با بدست آوردن یک حد پایین برای جواب بهینه مساله، آن را حل کردند. این روش مسائلی با اندازه حداکثر ۳۰ تجهیز را حل کرد [۵]. آمارال در سال۲۰۰۹ ابتدا مساله را به صورت صفر و یک خطی مدل‌سازی کرد و سپس نامساوی‌های معتبری را پیشنهاد کرد و از آن‌ها به عنوان صفحات برش، برای حل مساله چیدمان تک سطری استفاده کرد. جواد نعمتیان در سال ۲۰۱۴ مساله چیدمان تک سطری را بر اساس متغیرهای جدید و با استفاده از روش شاخه و کران حل کرد [۶]. برای حل مساله چیدمان تک‌سطری روش‌های فراابتکاری زیادی نیز ارائه شده است. رومرو و سنچز فلورز اولین بار در سال ۱۹۹۰ از الگوریتم عملیات حرارتی شبیه سازی شده برای حل مساله چیدمان تک سطری استفاده کردند [۷]. کومار و همکاران در سال ۱۹۹۵ یک الگوریتم با رهیافت آزمندی را برای حل مساله چیدمان خطی توسعه دادند [۸]. دی آلوارنگا و همکارانش در سال ۲۰۰۰ نخستین بار از الگوریتم جستجوی ممنوع برای حل مساله تک سطری با ابعاد کوچک استفاده کردند [۹]. این الگوریتم با یک جواب اولیه تصادفی و لیست ممنوع خالی شروع می‌شود. در این الگوریتم یک حد پایین و یک حد بالا برای مساله انتخاب می‌شود و در هر گام با جابجا کردن زوجی تجهیزات، جواب‌های جدید پیدا می‌شوند. در سال ۲۰۰۴ فیکو و همکاران از روش ژنتیک استفاده کردند. در روش پیشنهادی فرض شده است که هر جواب موجه در واقع یک ژن را تشکیل می‌دهد سپس از ژن‌هایی که در طول فرآیند حل بدست آمده‌اند به عنوان والد استفاده شده و رویه‌های مخصوص الگوریتم ژنتیک بر روی آن‌ها تعریف شده است [۱۰]. سلیمانپور و همکاران در سال ۲۰۰۵ برای حل مساله چیدمان تک سطری تجهیزات از الگوریتم بهینه‌سازی کولونی مورچگان استفاده کرده‌اند [۱۱]. در سال ۲۰۱۶ گنج لین و جیان گوان از ترکیب جستجوی همسایگی متغیر و الگوریتم مورچگان جهت حل مساله استفاده کردند. الگوریتم مورچگان جهت تنوع‌سازی و جستجوی همسایگی متغیر جهت تمرکزسازی استفاده می‌شوند. همسایگی‌های استفاده شده در این مقاله شامل روش جابجایی دوتایی، درجی و جابجایی تجهیزات همسایه می‌باشد [۱۲].

۳- تعریف مسئله

در چیدمان تک سطری تجهیزات، ، هدف تعیین جانمایی n تجهیز به صورت خطی و به گونه ای است که هزینه حمل و نقل بین تجهیزات کمینه گردد. هزینه حمل و نقل بین هر زوج تجهیز به صورت حاصلضرب میزان جریان در فاصله بین مراکز آن دو تجهیز تعریف می‌شود. در چیدمان تک سطری، شکل هر تجهیز به صورت یک مستطیل با عرض یکسان و طول مشخص است. طول تجهیز j ام با l_j نشان داده می‌شود. میزان جریان بین تجهیزات i و j ، c_{ij} در نظر گرفته می‌شود.

هر جواب موجه از چیدمان تک سطری را می توان متناظر با جایگشتی از اعداد ۱، ۲، ...، n در نظر گرفت. هزینه متناظر با چیدمان π به صورت زیر محاسبه می‌شود:

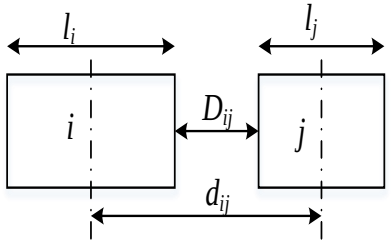
$$\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} c_{ij}d_{ij} \qquad (۱)$$

[۱] Francis

www.SID.ir

Archive of SID

در این رابطه d_{ij} فاصله میان مراکز ثقل تجهیزات i و j می‌باشد و برابر با $d_{ij} = D_{ij} + \frac{l_j}{2} + \frac{l_i}{2}$ است، که D_{ij} فاصله بین دو انتهای تجهیزات در چیدمان متناظر است (شکل ۱). هدف از حل مساله یافتن جایگشت با کمترین هزینه است.



شکل ۱- فاصله مرکز ثقل تسهیلات

۱-۳- الگوریتم فراابتکاری خفاش

الگوریتم فراابتکاری خفاش نخستین بار در سال ۲۰۱۰ توسط اکسین شه یانگ[۲] مطرح شد [۱۳]. این الگوریتم بر اساس ویژگی انعکاس صدا در خفاش‌ها طراحی شده‌است. بدین صورت که ابتدا خفاش‌ها با صدای بلند (A) و نرخ انتشار صوت پایین (r) شروع به حرکت می‌کنند و هرچه به طعمه نزدیک می‌شوند بلندی صدا کاهش یافته و نرخ انتشار صوت افزایش می‌یابد. سرعت(v) ، مکان (x) و فرکانس (f) حرکت خفاش نیز با توجه به فرمول‌های (۲) و (۳) و (۴) تعیین می‌گردند.

$$f_i = f_{min} + (f_{max} - f_{min})\beta \qquad (۲)$$

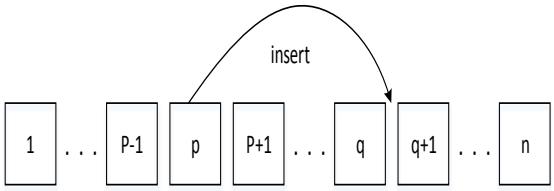
$$v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i \qquad (۳)$$

$$x_i^t = x_i^{t-1} + v_i^t \qquad (۴)$$

در فرمول‌های فوق، f_i عددی تصادفی در بازه $[0,1]$ است و x_* نیز بهترین جواب (مکان) فعلی از بین جواب‌های موجود است. و نیز گام-های الگوریتم را متغیر t مشخص می‌کند و i نیز شماره‌ی خفاش را تعیین می‌کند.

۲-۳- جستجوی همسایگی متغیر

جستجوی همسایگی متغیر، یک روش فراابتکاری برای حل مسائل بهینه سازی ترکیبی است. ایده اصلی این روش، تغییر همسایگی است که به وسیله آن می‌توان جواب بهتری را تولید کرد. در این مقاله از دو ساختار همسایگی درج کردن[۳] و جابجایی دو تجهیز[۴] به صورت تصادفی استفاده می‌شود. برای ایجاد همسایگی جدید از یک جایگشت به روش درج کردن می‌بایست دو تسهیل به صورت تصادفی انتخاب شود و جابجایی مطابق با شکل ۲ صورت گیرد. در روش جابجایی دو تسهیل تصادفی نیز پس از انتخاب دو تسهیل، مکان آن‌ها با یکدیگر عوض می‌شود.

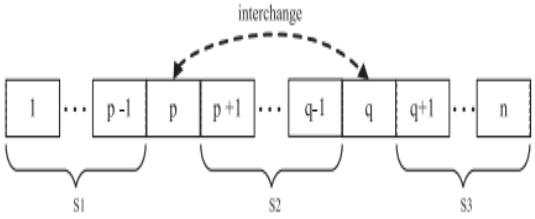


شکل ۲ - ایجاد همسایگی با استفاده از روش درج کردن (انتخاب تسهیل p و q به صورت تصادفی)

برای افزایش سرعت الگوریتم، تابع هزینه مربوط به همسایه‌ها را می‌توان با استفاده از تابع هزینه جواب فعلی بدست آورد. فرض می‌کنیم در جایگشت π تجهیزی که در مکان p قرار دارد را حذف کرده و در مکان q وارد می‌کنیم بنابراین یک همسایه جدید π^* برای π بدست می‌آید($p < q$). با توجه به شکل ۳، تجهیزات را به سه مجموعه s_1, s_2, s_3 تقسیم می‌کنیم [۱۲].

مجموعه s_1 : همه تجهیزاتی در جایگشت π قبل از تجهیز p قرار می‌گیرند. تعداد تجهیزات مجموعه s_1 را با N_1 و مجموع طول تجهیزاتی که در این مجموعه قرار گرفته اند را با D_1 نشان می‌دهیم.

$$D_1 = \sum_{i \in s_1} l_i \qquad D_2 = \sum_{i \in s_2} l_i \qquad D_3 = \sum_{i \in s_3} l_i$$



شکل ۳- تقسیم بندی جایگشت ها به سه مجموعه

تابع هزینه مربوط به جایگشت π^* از فرمول زیر محاسبه می‌شود:

---

[۲] Xin- She Yang

[۳] insertion

[۴] interchange

www.SID.ir

Archive of SID

$$(\Pi^*) = z(\Pi) + \sum_{i\in S1} c_{ip}D2 - \sum_{i\in S1}\sum_{j\in S2} c_{ij}l_p + \sum_{i\in S2} c_{pi}(D2 + l_p - 2d_{pi}) - \sum_{i\in S3} c_{pi}D2 + \sum_{i\in S2}\sum_{j\in S3} c_{ij}l_p$$

اگر $q < p$ باشد با توجه به روشی که ارائه شد نتایج زیر بدست می‌آید:

$$z(\Pi^*) = z(\Pi) - \sum_{i\in S1} c_{ip}D2 + \sum_{i\in S1}\sum_{j\in S2} c_{ij}l_p - \sum_{i\in S2} c_{pi}(D2 + l_p - 2d_{pi}) + \sum_{i\in S3} c_{pi}D2 - \sum_{i\in S2}\sum_{j\in S3} c_{ij}l_p$$

۳-۳- الگوریتم خفاش پیشنهادی برای مسئله چیدمان تک سطری

در ابتدا الگوریتم خفاش را با مساله چیدمان تک سطری تطبیق می‌دهیم، بنابراین هر یک از جایگشت‌ها را با متغیر x نمایش می‌دهیم و معیار فرکانس را حذف می‌کنیم. سرعت هر یک از جواب‌ها را نیز با استفاده از مفهوم فاصله همینگ[۵] تعریف می‌کنیم. برای بدست آوردن فاصله همینگ دو جایگشت می‌بایست تعداد عناصری که در هر دو جایگشت یکسان نمی‌باشند را جمع نمود.

بنابراین در مدل پیشنهادی سرعت خفاش‌ها (جواب) در هر گام طبق فرمول (۵) بدست می‌آید:

$$\boldsymbol{v_i^t = Random[1, (x_i^t, x_*)]} \quad (۵)$$

نحوه‌ی تغییر r و A نیز بر حسب فرمول (۶) تعیین می‌گردد.

$$\boldsymbol{A_i^{t+1} = \alpha A_i^t, \quad r_i^{t+1} = r_i^0[1 - \exp(-\gamma t)]} \quad (۶)$$

شبه کد الگوریتم به صورت جدول ۱ می‌باشد.

جدول ۳- شبه کد الگوریتم فراابتکاری خفاش برای مساله SRFLP

Bat algorithm for Single Row Facility Layout Problem: n is the number of facilities

Define the objective function f(x)
Initialize the bat population X=x_1, x_2, . . . ,x_n
for** each bat i in the population **do
Initialize the pulse rate r_i, velocity v_i and loudness A_i
end
repeat
for** each bat x_i in the population **do
Generate new solution
If** v_i^t<n/2 **then
Generate neighbors by insertion neighborhood search
Else
Generate neighbors by interchange neighborhood search
End
If** (rand>r_i) **then
Select one solution among the best solutions
Generate new bat selecing the best neighbor around the chosen bat
Use insertion neighborhood search
End
If** (rand <A_i && f(x_i)<f(x_*)) **Then
Accept the new solution
Increase ri and reduce Ai
End
End
Until Termination criterion not reached
Rank the bats and return the best bat

۴-۳- الگوریتم فراابتکاری خفاش پویا

الگوریتم فراابتکاری خفاش پویا نخستین بار در سال ۲۰۱۶ توسط علی اوسمان و همکارش ارائه شد. در این الگوریتم فقط از دو خفاش برای جستجوی جواب بهینه استفاده می‌شود.در هر مرحله یکی از خفاش برای تنوع‌سازی و دیگری برای تمرکزسازی استفاده می‌شوند، بدین معنی که یکی از خفاش‌ها فضای بیشتری را را جستجو می‌کند و دیگری با استفاده از روش‌های ایجاد همسایگی فضا را به طور دقیق‌تر جستجو می‌کند.

۵-۳- الگوریتم فراابتکاری خفاش پویا برای حل مسئله چیدمان تک سطری

با در نظر گرفتن ۳۰ گام برای الگوریتم و تولید پارامترهای اولیه به

[۵] Hamming Distance

www.SID.ir

Archive of SID

صورت تصادفی، الگوریتم فراابتکاری خفاش پویا را در محیط نرم‌افزار سی شارپ برای مساله چیدمان تک سطری پیاده‌سازی کردیم. شبه کد الگوریتم به صورت جدول (۲) می‌باشد:

جدول ۲- شبه کد الگوریتم فراابتکاری خفاش پویا برای مساله SRFLP

Dynamic Bat algorithm for Single Row Facility Layout Problem

Define the objective function f(x)
Initialize the bat population X=x_1, x_2
for** each bat i in the population **do
Initialize the pulse rate r_i, velocity v_i and loudness A_i
end
while** (t<number of iterations)**do
If** f(x_1)<f(x_2) **then
Generate neighbors by insertion neighborhood search for Bat1
Improve solution by interchange neighborhood search
Set the new solution as Bat1
Else
Generate neighbors by insertion neighborhood search for Bat2
Improve solution by interchange neighborhood search
Set the new solution as Bat2
End if
Set min(f(x1),f(x2))as the best solution
End while

۶-۳- نتایج محاسبات

برای پیاده‌سازی الگوریتم‌های حل مساله، از نرم‌افزار سی‌شارپ استفاده می‌کنیم و با استفاده از داده‌های مقاله‌ی آنجوس و آمارال هر یک از روش‌های BA و DBA را حل کرده می‌کنیم [۱۴]. تعداد تکرارهای الگوریتم برای هر دو روش ۳۰ در نظر گرفته می‌شود و تعداد جمعیت اولیه برای روش BA نیز ۲۰ خفاش در نظر گرفته می‌شود.

برای مقایسه‌ی الگوریتم‌های پیشنهادی مقادیر تابع هدف بدست آمده از هر روش را با یکدیگر مقایسه می‌کنیم. این مقادیر در جدول (۳) مشخص شده‌اند.

جدول۳- مقایسه تابع هدف دو روش پیشنهادی خفاش و خفاش پویا برای مساله SRFLP

درصد اختلاف	روش پیشنهادی BA	روش پیشنهادی DBA	تعداد تسهیلات
٪۲۱,۲۲	۵۵۹۸	۴۶۱۸	۲۵
٪۲۳,۲۶	۴۵۷۴۸	۳۷۱۱۶,۵	۲۵
٪۳,۷۰	۲۵۲۰۱	۲۴۳۰۱	۲۵
٪۲,۲۲	۴۹۳۶۵	۴۸۲۹۱,۵	۲۵
٪۴,۷۲	۱۶۳۶۰	۱۵۶۲۳	۲۵
٪۳۰,۷۱	۱۰۷۸۰	۸۲۴۷	۳۰
٪۲۵,۴۷	۲۷۰۸۰,۵	۲۱۵۸۲,۵	۳۰
٪۳,۸۸	۴۷۲۱۴	۴۵۴۴۹	۳۰
٪۰,۴۱	۵۷۱۰۸	۵۶۸۷۳,۵	۳۰
٪۱۰,۰۴	۱۲۶۸۴۰	۱۱۵۲۶۸	۳۰
٪۱,۰۳	۱۴۹۳۰۶۱	۱۴۷۷۸۳۴	۶۰
٪۳,۶۶	۸۷۲۵۴۶	۸۴۱۷۷۶	۶۰
٪۱,۰۵	۱۴۵۶۲۱۴	۱۴۴۱۰۲۸	۷۰
٪۳,۳۵	۱۵۶۹۸۹۰	۱۵۱۸۹۹۳,۵	۷۰
٪۴,۹۰	۲۱۷۰۴۱۲	۲۰۶۹۰۹۷	۸۰
٪۱,۹۷	۱۹۵۸۹۱۵	۱۹۲۱۱۳۶	۸۰

با توجه به جدول ۳ در تمامی مثال‌ها روش پیشنهادی DBA برای مساله چیدمان تک سطری تسهیلات جواب بهتری را بدست آورده است و عملکرد بسیار بهتری نسبت به روش BA داشته است.

جدول ۴- مقایسه زمان حل دو روش پیشنهادی خفاش و خفاش پویا برای مساله SRFLP

روش پیشنهادی BA	روش پیشنهادی DBA	تعداد تسهیلات
۲,۰۶	۰,۹۲	۲۵
۱,۹۸	۱,۰۳	۲۵
۲,۶۸	۰,۸۹	۲۵
۲,۹۱	۱,۰۹	۲۵
۲,۰۴	۰,۹۵	۲۵
۶,۵۱	۲,۵۴	۳۰
۸,۶۴	۲,۶۲	۳۰
۷,۲۵	۱,۹۵	۳۰
۵,۶۱	۱,۷۴	۳۰
۷,۸۹	۲,۰۳	۳۰
۳۲,۲۵	۱۹,۰۶	۶۰
۲۸,۷۸	۲۱,۰۱	۶۰

www.SID.ir

Archive of SID

۸۴,۳۲	۵۰,۲	۷۰
۷۵,۱۲	۵۸,۰۶	۷۰
۹۰,۱۱	۶۵,۹۸	۸۰
۱۰۶,۲	۷۴,۰۲	۸۰

با توجه به جدول فوق در تمامی مثال‌ها الگوریتم خفاش پویا جواب بهتری بدست می‌دهد.

۴- نتیجه‌گیری و پیشنهادهای آتی

در این مقاله، برای اولین بار با ایده گرفتن از الگوریتم‌های فراابتکاری خفاش، خفاش پویا و جستجوی همسایگی متغیر به حل مساله چیدمان تک‌سطری تسهیلات پرداخته‌شد و دو روش حل جدید را پیشنهاد دادیم. در الگوریتم اول از ترکیب الگوریتم خفاش و جستجوی همسایگی متغیر و در روش دوم از الگوریتم فراابتکاری خفاش پویا و جستجوی همسایگی متغیر استفاده می‌شود. نتایج بدست آمده نشان داد که الگوریتم پیشنهادی DBA کارایی بسیار بالاتری نسبت به الگوریتم BA دارد و در همه‌ی مثال‌ها جواب بهینه‌ی مساله را بدست می‌دهد و زمان حل مناسب تری دارد.

در ادامه‌ی پژوهش حاضر می‌توان برای افزایش کارایی روش‌های پیشنهادی، الگوریتم‌های ترکیبی جدیدی بر اساس روش‌های ابتکاری و فراابتکاری ارائه نمود. پیشنهاد می‌شود از سایر روش‌های ایجاد همسایگی و ترکیب آن‌ها با یکدیگر نیز استفاده شود. با توجه به اینکه الگوریتم فراابتکاری خفاش تاکنون با سایر اگوریتم‌های فراابتکاری مقایسه نشده است بنابراین می‌تواند به عنوان موضوعی جهت تحقیقات آتی در نظر گرفته شود.

۵- مراجع

[۱] Richard, L. F., & White, J. A. (1974). Facility layout and location: an analytical approach: Prentice Hall.

[۲] Picard, J.-C., & Queyranne, M. (1981). On the one-dimensional space allocation problem. *Operations Research, 29*(2), 371-391 .

[۳] Simmons, D. M. (1969). One-dimensional space allocation: an ordering algorithm. *Operations Research, 17*(5), 812-826.

[۴] Hall, K. M. (1970). An r-dimensional quadratic placement algorithm. *Management science, 17*(3), 219-229.*Manufacturing, 11*(4), 421-430.

[۵] Anjos, M. F., Kennings, A., & Vannelli, A. (2005). A semidefinite optimization approach for the single-row layout problem with unequal dimensions. *Discrete Optimization, 2*(2), 113-122 .

[۶] Nematian, J. (2014). A robust single row facility layout problem with fuzzy random variables. *The International Journal of Advanced Manufacturing Technology, 72*(1-4), 255

[۷] Romero, D., & Sánchez-Flores, A. (1990). Methods for the one-dimensional space allocation problem. *Computers & Operations Research, 17*(5), 465-473.

[۸] Kumar, K. R., Hadjinicola, G. C., & Lin, T.-l. (1995). A heuristic procedure for the single-row facility layout problem. *European Journal of Operational Research, 87*(1), 65-73 .

[۹] De Alvarenga, A. G., & Negreiros-Gomes, F. J. (2000). Metaheuristic methods for a class of the facility layout problem. *Journal of Intelligent Manufacturing, 11*(4), 421-430.

[۱۰] Ficko, M., Brezocnik, M., & Balic, J. (2004). Designing the layout of single-and multiple-rows flexible manufacturing system by genetic algorithms. *Journal of materials processing technology, 157*, 150-158 .

[۱۱] Solimanpur, M., Vrat, P., & Shankar, R. (2005). An ant algorithm for the single row layout problem in flexible manufacturing systems. *Computers & Operations Research, 32*(3), 583-598.

[۱۲] Guan, J., & Lin, G. (2016). Hybridizing variable neighborhood search with ant colony optimization for solving the single row facility layout problem. *European Journal of Operational Research, 248*(3), 899-909.

[۱۳] Yang, X.-S. (2010). A new metaheuristic bat-inspired algorithm *Nature inspired cooperative strategies for optimization (NICSO 2010)* (pp. 65-74): Springer.

[۱۴] Amaral, A. R. (2008). Enhanced local search applied to the single-row facility layout problem.

www.SID.ir