

بهینه‌سازی شبیه‌سازی در خطوط تولید با استفاده از SA¹

دکتر بهروز زارعی، عضو هیات علمی دانشگاه تهران

فرشید بابا اکبری، کارشناسی ارشد مدیریت صنعتی (fb_farshid@yahoo.com)

عظیم زارعی، دانشجوی دوره دکتری مدیریت دانشگاه تربیت مدرس (zare_i_az@modares.ac.ir)

چکیده

در حال حاضر پیچیدگی و پویایی روزافزون سیستم‌های تولیدی، کاربرد مدل‌های تحلیلی را در ارزیابی و تصمیم‌گیری آنها با محدودیتهای قابل توجهی روبه‌رو نموده است. لذا استفاده از شبیه‌سازی کامپیوتری به عنوان ابزاری که قابلیت گسترده‌ای در فرموله نمودن سیستم‌های فوق دارد، به طور وسیع مورد استقبال قرار گرفته است. با وجود این، ارائه بهترین راه‌حل همواره یکی از چالش‌های اصلی این حوزه می‌باشد. بهینه‌سازی شبیه‌سازی پاسخی به این چالش است. در این راستا، مقاله حاضر به چگونگی پیاده‌سازی الگوریتم SA در بهینه‌سازی شبیه‌سازی مساله تعیین ظرفیت مناسب انبارهای میانگیر² خط تولید می‌پردازد. در این خصوص پس از تلفیق مفهومی رویکرد SA و شبیه‌سازی، نرم‌افزاری برای اجرای آن طراحی و تهیه شده است و به کمک آن حالات مختلف سیستم آزمایش گردید. تجزیه و تحلیل نتایج بدست آمده از آزمایشها، به خوبی دلالت بر کارآمدی پیاده‌سازی SA در شبیه‌سازی این گونه سیستم‌ها دارد.

واژگان کلیدی: بهینه‌سازی شبیه‌سازی، تعیین ظرفیت انبارهای میانگیر، SA، شبیه‌سازی وقایع گسسته، روشهای فرا ابتکاری

¹ Simulated Annealing

² Buffer

(۱) مقدمه

امروزه پیچیدگی، پویایی روزافزون و ماهیت تصادفی کارکرد سیستمها بیش از پیش بررسی و تحلیل آنها را مشکل نموده است. از این نقطه نظر بکارگیری مدل‌های تحلیلی به خاطر ساده‌سازی و عدم توجه به تمامی ابعاد مترتب بر سیستم، با محدودیت‌های قابل توجهی روبرو شده است. این موضوع، بستر پیدایش و گسترش استفاده از رویکرد شبیه‌سازی کامپیوتری را فراهم آورد. با وجود این، ارائه بهترین راه‌حل همواره یکی از چالش‌های اصلی در این رویکرد است. اگرچه می‌توان به کمک شبیه‌سازی رفتار سیستم را تقلید نمود، لیکن تعیین مقادیر بهینه متغیرها به وسیله این روش امکان‌پذیر نیست. بهینه‌سازی شبیه‌سازی پاسخی برای برطرف نمودن این محدودیت است. خوشبختانه روشهای متنوع و گسترده بهینه‌سازی شبیه‌سازی امکان تحلیل مسائل پیچیده را فراهم نموده است. مزیت بارز این روشها در آن است که بدون جستجوی کلیه نقاط فضای موجه، سعی در یافتن جواب بهینه دارند. الگوریتم SA یکی از این روشها است که می‌تواند در بهینه‌سازی شبیه‌سازی مورد استفاده قرار گیرد. این روش در زمره روشهای فرا ابتکاری بهینه‌سازی شبیه‌سازی قرار دارد که ایده اصلی خود را از فرآیند ذوب فلزات دریافت نموده و با تطبیق مفاهیم طبیعی فرآیند ذوب فلزات و بهینه‌سازی به صورت یک الگوریتم جستجوی موثر شناخته شده است. این روش در حل طیف گسترده‌ای از مسائل بهینه‌سازی بکار می‌رود [۴، ۳، ۱، ۲]. در مقاله حاضر به چگونگی بکارگیری این روش در بهینه‌سازی شبیه‌سازی مساله تعیین ظرفیت بهینه انبارهای میانگیر خط تولید پرداخته شده است. نتایج بدست آمده به خوبی دلالت بر کارآمدی تلفیق SA با شبیه‌سازی در چارچوب مساله بررسی شده، دارد. در ادامه مقاله ابتدا مروری مختصر بر روشهای بهینه‌سازی شبیه‌سازی ارائه گردیده، سپس مفاهیم اصلی روش SA تشریح شده است. بخش بعدی به توصیف یک مسئله ساده تولید و پیاده‌سازی روش SA در حل آن پرداخته و نتایج بدست آمده مورد تجزیه و تحلیل قرار گرفته، و سرانجام پیشنهاداتی در مورد تحقیقات آتی ارائه شده است.

(۲) بهینه‌سازی شبیه‌سازی

بطور کلی بهینه‌سازی شبیه‌سازی فرآیند تعیین مقادیر بهینه متغیرهای ورودی سیستم (X_1, X_2, \dots, X_n) بمنظور حداکثر نمودن متغیرهای عملکردی است. مدل‌های بهینه‌سازی شبیه‌سازی دارای تابع هدف (یعنی، بهینه نمودن پاسخ مدل شبیه‌سازی) و مجموعه‌ای از محدودیتها می‌باشند. به عبارت دیگر یک مدل بهینه‌سازی را می‌توان به صورت زیر نشان داد.

$$\text{Min } f[X]$$

$$X \in \Theta$$

به طوریکه X متغیرهای تصمیم مدل و Θ بیانگر ناحیه موجه مدل است. برای بهینه کردن چنین مساله‌ای پنج دسته اصلی از روشهای وجود دارد: روشهای جستجوی مبتنی بر گرادیان، بهینه‌سازی تصادفی، متدولوژی پاسخ سطح، روشهای آماری و روشهای ابتکاری.

(۱-۲) روشهای جستجوی مبتنی بر گرادیان

این دسته از روشها با بکارگیری فنون برنامه‌ریزی ریاضی به تخمین گرادیان تابع خروجی شبیه‌سازی جهت ارزیابی تابع هدف می‌پردازند. در این زمینه چندین روش ارائه شده است. روش برآورد تقاضات محدود [۵]. بعنوان ابتدایی‌ترین روش تخمین گرادیان، به صورت زیر عمل می‌کند:

$$g(x) = \nabla f(x)$$

$$\hat{g}(x) = (\hat{g}_1(x), \hat{g}_2(x), \dots, \hat{g}_d(x))^T$$

$$\hat{g}_i(X) = \begin{cases} \frac{\hat{f}(X + ce_i) - \hat{f}(X)}{c} & i=1,2,\dots,d \\ \text{روش تفاضلات پیشرو} \\ \frac{\hat{f}(X + ce_i) - \hat{f}(X - ce_i)}{2c} & i=1,2,\dots,d \\ \text{روش تفاضلات مرکزی} \end{cases}$$

که در آن $e_i = (0, 0, \dots, 1, 0, 0, \dots, 0)$ بردار مختصات و c ، عددی، ثابت، مثبت و کوچک.

در روش تجزیه و تحلیل اغتشاش و روش نسبت احتمال تنها با یکبار اجرای شبیه‌سازی به تخمین گرادیان تابع خروجی شبیه‌سازی می‌پردازند [6]. در این روش‌ها فرض می‌شود متغیر تصادفی $x(X)$ دارای تابع توزیع تجمعی $F(x)$ و از طریق تبدیل معکوس به صورت $x = F_x^{-1}(u)$ می‌باشد بطوریکه $X \in \Theta$ و U متغیر تصادفی یکنواخت در بازه $[0, 1]$ است. در روش تجزیه و تحلیل اغتشاش، تخمین گرادیان تابع خروجی شبیه‌سازی به صورت:

$$g(X) = \sum_{i=1}^{i=N} \nabla F_x^{-1}(u) / N$$

است. حال آنکه در روش نسبت احتمال فرض می‌شود متغیر تصادفی $x(X)$ دارای تابع چگالی f_x است. از این رو توزیع F_x گرادیان تابع به صورت:

$$g(X) = \sum_{i=1}^N x_i(X) \nabla f_x(x_i(X)) / f_x(x_i(X)) / N$$

می‌باشد. روش آزمایشات با دامنه نوسانی که یک روش تجزیه و تحلیل حساسیت و بررسی عوامل می‌باشد [7] یکی از روشهای دیگر تخمین گرادیان است. در این روش متغیر ورودی انتخاب شده در خلال اجرای شبیه‌سازی در تکرارهای مختلف تحت تاثیر نوسانات سینوسی قرار گرفته و با تجزیه و تحلیل طیفی به تخمین گرادیان تابع مدل می‌پردازد. بردار متغیر ورودی مدل شبیه‌سازی در این روش به صورت:

$$X(t) = X_0 + \alpha \sin(\omega t)$$

تولید می‌گردد که در آن X_0 ، بردار پارامتر ورودی مدل، α ، بردار دامنه نوسان و ω ، بردار فرکانس تحریک است.

۲-۲) بهینه‌سازی تصادفی

این روش در زمره رویه‌های برگشتی، مبتنی بر گرادیان و تکرارپذیر است [8]. روش مزبور ابتدا به تخمین تابع رگرسیون جواب پرداخته و سپس با حل معادله گرادیان $\nabla g(x) = 0$ متغیرهای ورودی را بهینه می‌نماید. فرمول اصلی بازگشتی پارامتر ورودی مدل شبیه‌سازی به صورت

$$X_{n+1} = \Pi_{\Theta}(X_n - a_n \hat{\nabla} g_n(X_n))$$

تولید می‌گردد که X_n ، مقدار پارامتر تصمیم سیستم در ابتدای تکرار n ام، $\nabla g(X_n)$ ، برآورد $\nabla g(X)$ در تکرار n ام، a_n ، یک دنباله مثبت و Π_{Θ} ، تصویر نقطه بدست آمده بر فضای Θ (نزدیکترین نقطه در فضای موجود) است [۹].

۳-۲) متدولوژی پاسخ سطح

رویکرد این روش در بهینه‌سازی شبیه‌سازی، تعیین ارتباط میان متغیرهای ورودی و خروجی شبیه‌سازی صرف‌نظر از چگونگی پردازش مدل است [۱۰]. به این روشها فرامدل^۳ می‌گویند که این امر در روش حاضر به صورت عددی و با استفاده از فنون آماری، انجام می‌پذیرد [۱۱]. یعنی $E(f(X)) = \Theta(x_1, x_2, \dots, x_d)$ که در ادامه ابتدا تابع Θ بوسیله یک مدل رگرسیون درجه اول در دامنه بخصوصی از X به صورت زیر تقریب زده می‌شود:

$$E(F(X)) = \sum_{i=1}^p \beta_i + x_d \beta_d + \dots + \beta_0 + \beta_1 x_1$$

چنانچه قابلیت اعتماد مدل بدست آمده کم باشد، تابع Θ مجدداً در نقطه بهینه تقریب فوق، با مدل‌های درجه دوم رگرسیون تخمین زده می‌شود. این رویه تا حصول قابلیت اطمینان مورد نظر ادامه می‌یابد. مقدار بهینه مدل رگرسیون به عنوان مقدار بهینه متغیر ورودی مدل شبیه‌سازی خواهد بود.

۴-۲) روشهای آماری

زمانی از این روشها در مسائل بهینه‌سازی شبیه‌سازی استفاده می‌شود که مسئله انتخاب بهترین متغیر ورودی از میان چند مجموعه از متغیرهای ورودی مد نظر باشد. رتبه‌بندی و انتخاب، انتخاب زیرمجموعه و مقایسات چندگانه برخی از این روشهای است.

روش رتبه‌بندی و انتخاب زمانی مورد استفاده قرار می‌گیرد که تعداد ورودی کم باشد و در انجام این امر از دو طریق یعنی انتخاب زیرمجموعه و ناحیه بی‌تفاوتی^۴، عمل می‌کند. در صورتیکه انتخاب زیرمجموعه‌هایی از متغیرهای ورودی که دربرگیرنده بهترین پارامتر ورودی است، مطرح باشد از طریق انتخاب زیرمجموعه و زمانی که انتخاب بهترین متغیر ورودی مد نظر باشد، از رتبه‌بندی ناحیه بی‌تفاوتی استفاده می‌شود. از سوی دیگر، روش انتخاب زیرمجموعه، از طریق ساده‌سازی و کوچک کردن فضای جواب به بهینه‌سازی کمک می‌کند. این روش به عنوان یک راهنمای عمل در روش رتبه‌بندی و انتخاب نیز بکار می‌رود. نهایتاً از روش مقایسات چندگانه که در آنها انتخاب بهترین پارامتر ورودی مطرح می‌باشد، بعنوان جایگزینی برای روش رتبه‌بندی و انتخاب استفاده می‌شود با این تفاوت که اطلاعاتی نیز در خصوص رابطه میان متغیرهای ورودی و پاسخ مدل نیز ارائه می‌نماید [۱۳، ۱۱].

۵-۲) روشهای ابتکاری

این روشها بر خلاف روشهای تحلیلی فضای مسئله را به صورت تصادفی (مبتنی بر یک ایده ابتکاری) بررسی و به دلیل سادگی، کاربرد وسیعی در مسائل بهینه‌سازی دارند [۱۴]. ایده اولیه آنها برگرفته از تکامل زیستی، هوش مصنوعی، علوم ریاضی، فیزیک و

³ Meta model

⁴ Indifference zone

سیستمهای عصبی است. الگوریتم ژنتیک که یکی از این روشها می‌باشد، ایده خود را از انتقال صفات موروثی بوسیله ژن‌ها از یک نسل به نسل دیگر دریافت کرده است. در بکارگیری الگوریتم مزبور در مسائل بهینه‌سازی شبیه‌سازی، ابتدا مجموعه‌ای از متغیرهای ورودی به طور تصادفی تولید و با در نظر گرفتن پاسخ مدل شبیه‌سازی، تلفیق و ترکیبی بین این مجموعه صورت گرفته تا متغیرهای ورودی بعدی تولید گردد [۱۵، ۱۶، ۱۷]. این رویه تا زمانی ادامه پیدا می‌کند که میزان بهبود در جواب، از یک آستانه از قبل تعیین شده فراتر نرود. دیگر روش در این حوزه، روش جستجوی ممنوع است. این الگوریتم بر اساس اصول هوش مصنوعی در سال ۱۹۹۸ توسط فردگلاور مطرح شد که در آن مسیر جستجو از یک نقطه به سوی بهترین نقطه همسایگی در فضای جواب، دنبال می‌شود و با بهره‌گیری از مکانیزمی تحت عنوان لیست تابو^۵ اولاً از بررسی مجدد نقاط اجتناب و در ثانی امکان پذیرش جوابهای نامطلوب را به امید دستیابی به جواب بهتر (فرار از دام نقطه بهینه موضعی) می‌پذیرد. با افزایش تکرار الگوریتم، فرایند جستجو با دقت بیشتری انجام می‌گیرد [۱۸].

استراتژیهای تکاملی از جمله دیگر روشهای ابتکاری است. خواستگاه این روش قواعد تکامل طبیعی، می‌باشد. در محاسبات تکاملی این روش، دو طرح برای تولید جواب جدید وجود دارد:

۱- $(\lambda + \mu)$ که λ و μ به ترتیب بیانگر والدین و فرزندان می‌باشند. لذا مجموعه متغیرهای ورودی، تنوعی به اندازه $(\lambda + \mu)$ خواهد داشت و μ موجود به عنوان والدین برای تولید نسل بعد انتخاب و با حذف بدترین مقدار، اندازه جمعیت فعلی حفظ می‌شود.

۲- (λ, μ) که در آن تعداد فرزندان بیشتر از والدین بوده و از آنها در تولید مجموعه بعدی استفاده می‌شود [۱۹]. روش دیگر SA می‌باشد که به لحاظ تمرکز مقاله بر آن در ادامه با جزئیات بیشتری مطرح می‌گردد.

۳) الگوریتم SA

در برخی از مسائل بهینه‌سازی ترکیبی بدلیل فضای گسترده جواب و زمان محاسبه بالا استفاده از SA مؤثر است. مفاهیم اصلی این الگوریتم برگرفته از قواعد فیزیکی و ترمودینامیکی مربوط به ذوب فلزات است. بدین ترتیب که دمای یک جسم جامد، T ، در اثر حرارت افزایش یافته تا به حالت مایع درآید. سپس دمای جسم بتدریج کاهش می‌یابد. چنانچه کاهش دما به آهستگی صورت پذیرد آرایش اتمهای جسم در پایین‌ترین سطح انرژی خود قرار می‌گیرند. از منظر مهندسی متالوژی این فرآیند سعی دارد اتمهای جسم را به نحوی در کنار هم قرار دهد تا حالت فیزیکی جسم در بهترین شکل ممکن قرار گیرد. ارتباط مفاهیم فیزیکی با بهینه‌سازی ترکیبی بدین ترتیب است که جوابهای مختلف در مسئله بهینه‌سازی ترکیبی با موقعیتهای فیزیکی مختلف یک جسم معادل است. همچنین هزینه هر یک از جوابها در مسائل بهینه‌سازی ترکیبی با سطوح مختلف انرژی، E ، مترادف می‌باشد [۲۰]. نتایج پژوهشهای ارائه شده دلالت بر آن دارد که روش SA توانایی قابل ملاحظه‌ای در حل مسائل بهینه‌سازی شبیه‌سازی دارد [۲۱].

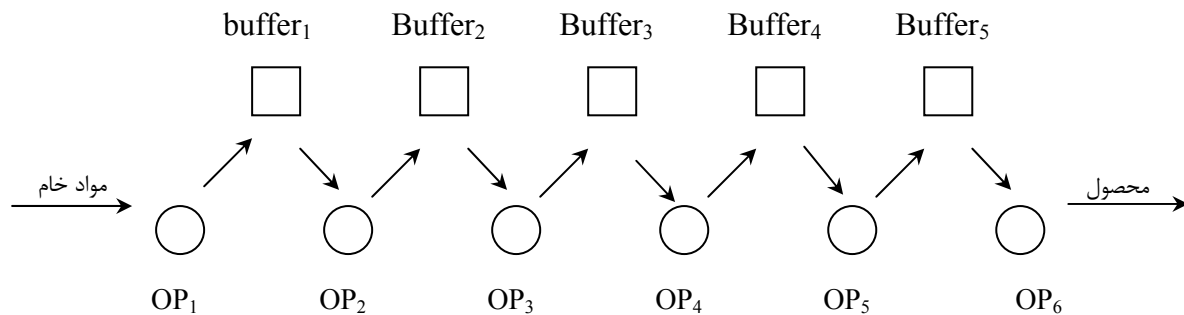
الگوریتم SA در حل مسائل بهینه‌سازی ترکیبی معمولاً با انتخاب یک جواب اولیه تصادفی آغاز می‌شود. این الگوریتم تلاش می‌کند با انتخاب جوابهای جدید بوسیله یک مکانیزم مناسب و محاسبه اختلاف هزینه متناظر با هر جواب، ΔE ، جوابها را بهبود دهد. اگر مقایسه جواب موجود و جواب جدید بیانگر بهبود باشد ($\Delta E \leq 0$) جواب جدید پذیرفته می‌شود و فرآیند جستجو تا رسیدن به یک جواب رضایت‌بخش ادامه می‌یابد. چنانچه این مقایسه بیانگر بهبود جواب نباشد ($\Delta E > 0$) جواب جدید با احتمال $\exp\left(\frac{-\Delta E}{T}\right)$ هنوز شانس پذیرش را دارد. بدین صورت که یک عدد تصادفی ($0 \leq R \leq 1$) تولید و چنانچه این عدد تصادفی از احتمال یاد شده کوچکتر باشد پذیرفته و در غیر این صورت رد می‌شود [۲۲].

⁵ Tabu list

۴) تشریح و مفروضات مسئله

در این مقاله چگونگی استفاده از روشهای بهینه‌سازی شبیه‌سازی در تعیین بهینه ظرفیت انبارهای میانگیر به کمک الگوریتم SA مورد بحث قرار گرفته است. خط تولید بررسی شده به صورت سری است که در آن ایستگاههای کاری بوسیله یک انبار میانگیر از هم جدا شده و قطعات از ایستگاه اول با یک تابع توزیع نمایی وارد سیستم شده و به ترتیب از تمامی ایستگاههای و فضاها انبار میانگیر بین آنها عبور می‌کنند. در صورت خالی بودن انبار میانگیر پس از انجام عملیات بر روی یک قطعه به انبار میانگیر انتقال داده می‌شود. فضای قبل از ایستگاه اول بعنوان یک انبار میانگیر محسوب نمی‌شود و همچنین مدت زمان انجام عملیات در هر ایستگاه بصورت متغیرهای تصادفی نرمال است. از این رو، بخاطر ماهیت تصادفی زمان سرویس‌دهی در ایستگاهها، انبارهای میانگیر بین آنها تعبیه شده است. ایستگاه پایانی نیز محصول نهایی را به انبار کالاهای ساخته شده تحویل می‌دهد که در این مورد نیز فرض می‌شود انبار کالاهای ساخته شده فاقد محدودیت ظرفیت است.

در برخی ایستگاهها ممکن است عملیات به علت تکمیل ظرفیت انبار میانگیر بعدی متوقف شود. در این حالت نوعی هزینه فرصت از دست رفته وجود دارد که در این مقاله از آن تحت عنوان هزینه توقف عملیات یاد می‌شود. همچنین افزایش هر واحد ظرفیت انبارها نیز هزینه مخصوص به خود را در پی دارد. خط تولید مورد بررسی دارای ۶ ایستگاه (OP_1, OP_2, \dots, OP_5) و ۵ انبار میانگیر ($buffer_1, \dots, buffer_5$) می‌باشد که در شکل ۱ نشان داده شده است.



شکل ۱. نمایش خط تولید بررسی

مسئله تصمیم‌گیری تعیین ظرفیت بهینه هر یک از انبارهای میانگیر است. اندازه‌های مختلف ظرفیت این انبارها سطوح نتایج مختلفی را در خط تولید ایجاد می‌کند. ظرفیت انبارها بر میزان تولید محصول (خروجی سیستم) و هزینه سیستم، تأثیر می‌گذارد. بنحویکه قرار دادن یک انبار میانگیر با ظرفیت بالا هزینه توقف عملیات را کاهش ولی در عوض هزینه انبارهای میانگیر را افزایش می‌دهد. از سوی دیگر، تعیین بهینه ظرفیت انبارها به عواملی همچون مدت زمان عملیات هر ایستگاه کاری و تابع هزینه تولید بستگی دارد. چنانچه در ایستگاهی زمان عملیات کم و زمان عملیات ایستگاه بعدی زیاد باشد، انبار میانگیر با ظرفیت بالا مورد نیاز است. همچنین زمانی که هزینه مربوط به توقف عملیات نسبت به هزینه افزایش ظرفیت انبارها زیاد باشد، می‌بایست تلاش شود که با افزایش ظرفیت انبارها از بروز توقف عملیات، اجتناب شود.

یک راه‌حل برای یافتن بهترین ترکیب انبارها آن است که تمام ترکیبات مختلف ظرفیت‌ها آزمایش شوند. اما این راهکار نه تنها مقرون به صرفه نیست، بلکه چندان عملی نیز نمی‌باشد. زیرا تعداد ترکیبهای مربوط به تخصیص u واحد ظرفیت کل به n انبار میانگیر چنانچه

مجموع ظرفیتها در هر ترکیب مساوی u باشد، برابر با $\binom{u+(n-1)}{u}$ و چنانچه مجموع انبارها کوچکتر یا مساوی u باشد، تعداد ترکیبها

برابر $\binom{u+n}{u}$ خواهد بود [۲۳]. به عنوان مثال در حالت دوم، تعداد ترکیبهای تخصیص ۱۰ و ۱۵ واحد ظرفیت به ۵ انبار میانگیر به ترتیب

برابر با ۳۰۰۳ و ۱۵۵۰۴ است. بنابراین با افزایش ظرفیت کل و تعداد انبارها، میزان ترکیبهای تخصیص به شدت افزایش می‌یابد. بعلاوه از آنجائیکه مدت زمان عملیات تابع توزیع احتمال می‌باشد، بررسی سیستم تنها به کمک شبیه‌سازی کامپیوتری میسر بوده و برای رسیدن به نتایج قابل اطمینان در هر ترکیب مشخص، اجراهای مختلف مورد نیاز است. لذا در بهینه‌سازی این سیستمها، استفاده از روشهای کارآمدی نظیر SA ضروری است.

تعیین ظرفیت هر انبار میانگیر بگونه‌ای که بیشترین سطح عملکردی را موجب شود از طریق دو روش امکان‌پذیر است:

- جستجوی تعویضی^۶
- جستجوی فراگیر^۷

در روش اول، تنها انبارهای میانگیری که مجموع ظرفیت آنها ثابت باشد مورد بررسی قرار می‌گیرند. به طور مثال، یک آرایش از انبارهای میانگیر که دارای ترکیب ظرفیتی ۱-۳-۳-۳-۵ (با جمع فضای ذخیره انبار ۱۵ واحد) می‌تواند با آرایش ۳-۳-۳-۳-۳ که ۲ فضا از انبار دوم کاسته و ۲ فضا به انبار پنجم اضافه شده تعویض گردد. روش دوم به دنبال شناسایی انبارهای میانگیری با بالاترین سطح از نتایج و با کمترین سطح ممکن از ظرفیت می‌باشد.

هـ) پیاده‌سازی الگوریتم

بکارگیری الگوریتم SA در مساله فوق به صورت زیر خواهد بود:

قدم ۱. یک آرایش اولیه از خط تولید (C_0) و یک دمای اولیه (T_0) انتخاب شود.

قدم ۲. تابع هزینه C_0 محاسبه شود.

قدم ۳. یک آرایش جدید از خط تولید (C_n) انتخاب شود.

قدم ۴. اختلاف تابع هزینه (ΔE) بین C_0 و C_n محاسبه گردد.

قدم ۵. اگر آرایش جدید دارای کارایی بیشتری از آرایش قبلی باشد ($\Delta E \leq 0$) پذیرفته شود.

قدم ۶. اگر آرایش جدید دارای کارایی کمتری باشد ($\Delta E > 0$) یک عدد تصادفی (R) تولید و چنانچه $R < \exp\left(\frac{-\Delta E}{T}\right)$ باشد آن را پذیرفته، و در غیر این صورت رد شود.

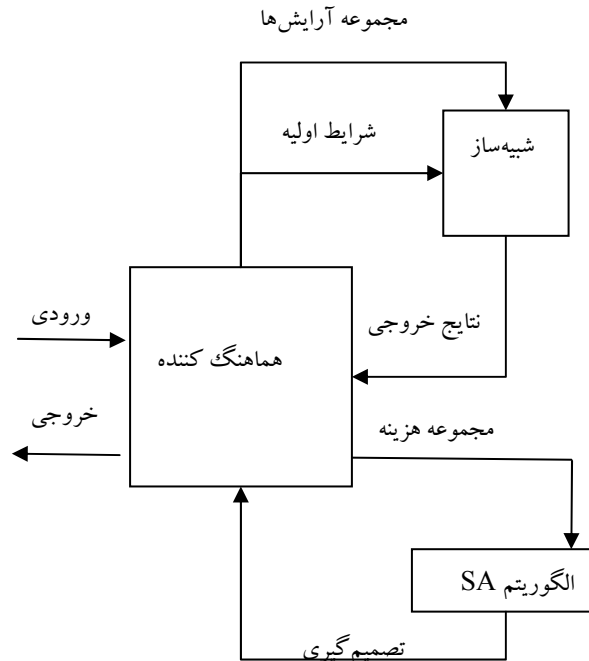
قدم ۷. اگر شرط تعادل برآورده شود، فرایند جستجو متوقف، در غیر این صورت، دما کاهش و به قدم ۳ برگردید.

برای اجرای بهینه‌سازی شبیه‌سازی مسئله فوق، یک برنامه کامپیوتری در محیط VISUAL C++ به صورت شیء‌گرا و با رعایت اصول طراحی نرم‌افزار تهیه گردید. ایجاد معماری دقیق، موجب ارتقاء عملکرد، قابلیت استفاده و نگهداری نرم‌افزار گردید. کلاسهای طراحی شده به گونه‌ای است که برای تعمیم مسئله و یا استفاده در مسائل مشابه (مسائلی که نیاز به شبیه‌سازی گسسته و یا بهینه‌سازی با الگوریتم

⁶ Swap search

⁷ Global search

SA دارند) کارایی لازم را با کمترین تغییرات دارد. سرعت بالا برای نرم‌افزارهای بهینه‌سازی شبیه‌سازی ضروری است که در این نرم‌افزار محیط پیاده‌سازی و الگوریتم‌های بکار رفته این خواسته را به خوبی برآورده نموده است. معماری کلی نرم‌افزار در شکل ۲ نشان داده شده است.



شکل ۲. معماری نرم‌افزار بهینه‌سازی شبیه‌سازی به کمک الگوریتم SA

بخش همهانگ کننده، اطلاعاتی نظیر زمان شبیه‌سازی، تعداد اقلام تولید شده در شبیه‌سازی، اطلاعات مربوط به وضعیت سیستم شامل تعداد ایستگاهها، اندازه اولیه انبارهای میانگیر، نرخ ورود زمان سرویس‌دهی، زمان عملیات ایستگاهها و اطلاعات مربوط به تنظیمات الگوریتم SA مشتمل بر دمای اولیه، تعداد تکرارها در هر دم، مقدار اِپسِلِن برای خروج از برنامه و پارامترهای هزینه شامل هزینه توقف عملیات و هزینه افزایش ظرفیت را از فایل ورودی دریافت می‌کند. همهانگ‌کننده با توجه به اطلاعات فوق یک آرایش از سیستم را ساخته و با انجام تنظیمات مربوط به بخش شبیه‌سازی، تقاضای شبیه‌سازی را می‌دهد. بخش شبیه‌ساز که خود از اجزای کوچکتری مانند executor، scheduler و clock تشکیل شده است بر مبنای روش شبیه‌سازی سه مرحله‌ای^۸، سیستم را شبیه‌سازی نموده، گزارش کاملی از نتایج شبیه‌سازی را شامل موارد زیر تهیه می‌کند:

زمان پایان شبیه‌سازی، تعداد ورودی‌های تولید شده، تعداد ورودی‌ها در صف، متوسط طول صف، میانگین و حداکثر تعداد قطعات در هر کدام از انبارهای میانگیر، میزان استفاده، زمان بیکاری و زمان توقف عملیات هر یک از ایستگاهها بر حسب درصد و تعداد قطعات خارج شده از سیستم. قسمت شبیه‌ساز این گزارش‌ها را در یک فایل ذخیره و آنرا به قسمت همهانگ کننده برمی‌گرداند. همهانگ کننده، تابع هزینه را از شیء نتایج فراخوانده و آرایش فعلی را ارزیابی می‌نماید.

توابع مختلفی می‌تواند برای ارزیابی آرایشها مورد استفاده قرار گیرد. دو نسخه مختلف از این تابع طراحی و پیاده‌سازی شده است. همهانگ کننده، هزینه آرایش را به الگوریتم SA ارجاع می‌دهد. الگوریتم SA با توجه به دما و هزینه جواب فعلی و هزینه جواب قبلی تصمیمی منجر به قبول یا رد آن و نیز ضرورت بررسی آرایشهای بیشتر اتخاذ می‌کند و آن را به قسمت همهانگ کننده اعلام می‌دارد. همهانگ کننده با توجه به تصمیم SA در صورتیکه احتیاج به بررسی آرایشهای جدید باشد، تابع تغییر آرایش را از شیء آرایش فراخوانده و یک

⁸ Three-Phase method

آرایش همسایه جدید می‌سازد. این تابع تغییر آرایش می‌تواند دارای مکانیزم‌های مختلفی باشد. در این نرم‌افزار سه نسخه از این تابع که یکی منطقی و دو مورد دیگر تصادفی است طراحی و پیاده‌سازی شده است. هماهنگ‌کننده، آرایش جدید را به شبیه‌ساز ارجاع داده و از نتایج تولید شده، هزینه آرایش جدید را محاسبه آن را برای تصمیم‌گیری در اختیار الگوریتم SA قرار می‌دهد. این عملیات بطور متوالی تکرار می‌شود. در هر تکرار شبیه‌سازی به تعداد تکرار تنظیم شده در ابتدای برنامه اجرا و سپس دما کاهش می‌یابد. مکانیزم تغییر دما به صورت هندسی و با ضریب ۰.۹۵ درصد صورت می‌پذیرد. هرگاه شرط خروج حاصل شد (برای مثال مقدار بهبود در n جواب پی در پی کمتر از اپسیلن باشد) الگوریتم SA تصمیم خروج را به هماهنگ‌کننده اعلام و هماهنگ‌کننده نیز آخرین آرایش پذیرفته شده را به همراه نتایج آن به عنوان خروجی نمایش می‌دهد.

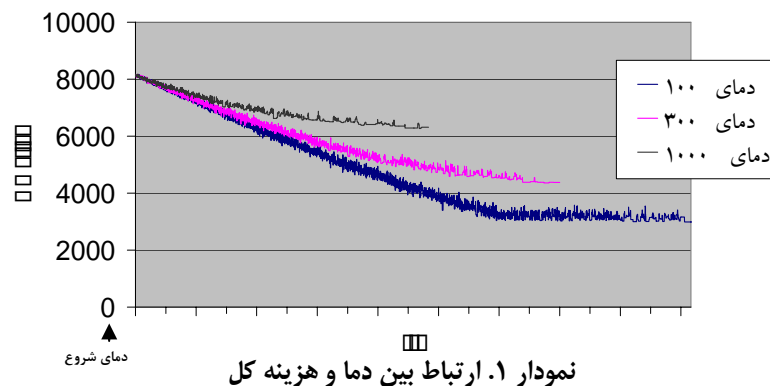
برای آنالیز خروجی‌ها، فایل خروجی به نحوی تنظیم شده است که قابل ورود به نرم‌افزار EXCEL باشد. در برنامه EXCEL هم تعدادی ماکرو^۹ به زبان VISUAL BASIC نوشته شده که می‌تواند تحلیل‌های آماری مختلفی را روی داده‌ها انجام دهد. این برنامه به راحتی می‌تواند با پیاده‌سازی توابع جدید در حالت‌های مختلفی از SA مورد آزمایش قرار گیرد. همچنین می‌توان روش‌های دیگر بهینه‌سازی شبیه‌سازی مانند الگوریتم‌های ژنتیک یا جستجوی ممنوع را به آن افزود و نتایج روش‌های مختلف را با هم مقایسه کرد.

۶) تجزیه و تحلیل نتایج

با توجه به آزمایشات انجام شده پنجاه دسته نتایج به شرح زیر استخراج و تجزیه و تحلیل گردیده است. در این آزمایشات دمای شروع شبیه‌سازی ۱۰۰۰، اندازه اولیه هر یک از انبارها ۱۰، مقدار میانگین و انحراف معیار تابع احتمال (نرمال) عملیات هر یک از ایستگاه‌های کاری به ترتیب (۲۰۱)، (۱۰۲)، (۵۰۱)، (۷۰۱)، (۲۰۱) و (۸۰۲)، تعداد تکرار شبیه‌سازی در هر دما ۲۰۰، مقدار اپسیلن ۲، نرخ ورود قطعات به سیستم تابع توزیع نمایی با $\lambda = 2$ ، هزینه توقف عملیات ۱۰۰ و هزینه افزایش ظرفیت انبارها ۱ واحد تنظیم گردیده است. در پارامترهای از آزمایشات، برخی از مقادیر فوق تغییر یافته که در متن به آنها اشاره گردیده است.

۶-۱) تاثیر دما بر تابع هزینه کل

نمودار شماره ۱ روند بهبود مقدار هزینه کل با تغییر دمای شبیه‌سازی را نشان می‌دهد. نمودار دلالت بر آن دارد که میزان بهبود جواب در دمای بالا، زیاد و در دمای پایین، کاهش یافته است به طوری که در آخرین دمای بررسی شده هیچگونه بهبودی در جواب مشاهده نگردیده است. همین امر موجب توقف ادامه بهینه‌سازی شبیه‌سازی شده است. با توجه به این موضوع می‌توان دریافت که کارایی محاسباتی الگوریتم SA در دماهای بالا، زیاد و در دماهای پایین، کمتر می‌شود.



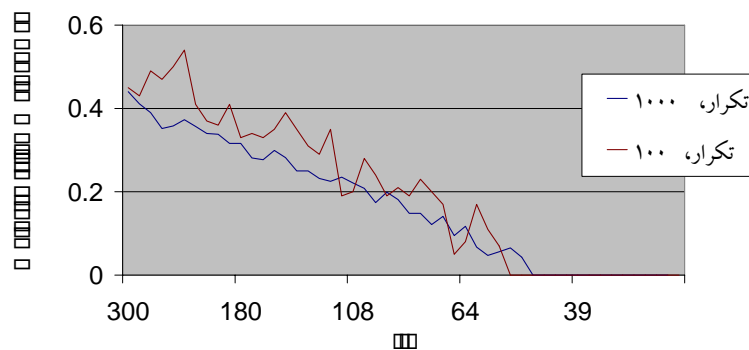
⁹ Macro

۶-۲) ارتباط دما با نسبت پذیرش جوابهای نامطلوب

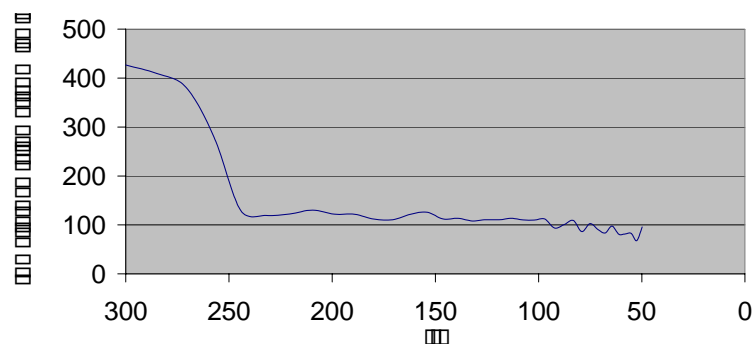
یکی از نقاط قوت الگوریتم SA اجتناب از انتخاب جواب بهینه موضعی است. در این راستا الگوریتم، جوابهای نامطلوب را به بهای گریز از چنین جوابهایی می‌پذیرد. نمودار شماره ۲ روند پذیرش جوابهای نامطلوب را به ازای دماهای مختلف در الگوریتم SA و برای تکرارهای شبیه‌سازی ۱۰۰ و ۱۰۰۰ در هر دما نشان می‌دهد. همانگونه که مشاهده می‌گردد با تغییر دمای شبیه‌سازی، نسبت پذیرش جوابهای نامطلوب، کاهش یافته است. با توجه به نتایج بدست آمده، می‌توان چنین نتیجه گرفت که در بهینه‌سازی شبیه‌سازی به کمک SA، با افزایش تعداد تکرار و کاهش دما، الگوریتم نسبت به پذیرش جواب نامطلوب، سخت‌گیری بیشتری اعمال کرده و انتظار دستیابی به جواب رضایت بخش بیشتر خواهد بود. مشاهده تاثیر افزایش تکرارها بر درصد پذیرش جوابهای نامطلوب نشان می‌دهد که در یک دمای بخصوص (مثلاً ۲۷۵) این نسبت در تکرارهای بیشتر (مثلاً ۱۰۰۰ تکرار) کوچکتر خواهد بود. زیرا در تکرارهای بالاتر، تعداد آرایشهای بیشتری از جوابها مورد بررسی قرار گرفته و امید ریاضی نتایج شبیه‌سازی پایدارتر می‌باشد.

۶-۳) ارتباط دما با انحراف معیار جوابهای پذیرفته شده

در نمودار شماره ۳ انحراف معیار جوابهای پذیرفته شده در هر دما نشان داده شده است. همانگونه که مشاهده می‌شود انحراف معیار جوابها در دماهای بالا بیشتر و با کاهش دما کمتر می‌شود. همچنین در شروع، نمودار دارای شیب تند است اما به تدریج شیب نمودار کمتر می‌شود که این امر بیانگر ارتباط مستقیم انحراف معیار جوابها و دما است. همچنین انحراف معیار در دماهای پایین به سمت یک مقدار ثابت میل نموده که نشان دهنده رسیدن الگوریتم به یک جواب مناسب است. بطور کلی از مطالب فوق می‌توان نتیجه گرفت که در دماهای بالا بدلیل سخت‌گیری کمتر در پذیرش جوابها و فرار الگوریتم از نقاط بهینه موضعی انحراف معیار بیشتری بین جوابها در یک دما مشاهده می‌شود. در دماهای پایین بدلیل رسیدن الگوریتم به جوابهای مناسب، انحراف کمتری بین دو جواب مشاهده می‌گردد که بیانگر رسیدن الگوریتم به نقطه تعادل است.



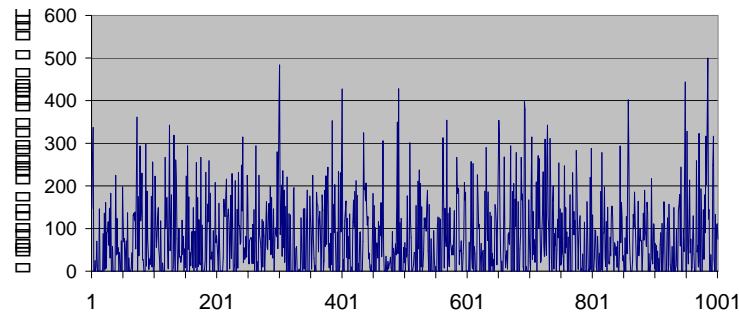
نمودار ۲. ارتباط بین دما و درصد پذیرش جوابهای نامطلوب



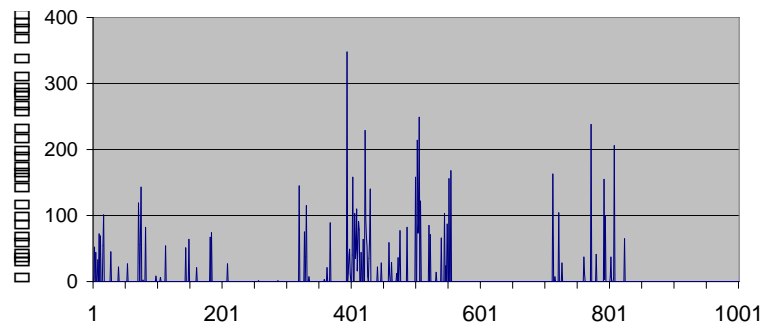
نمودار ۳. ارتباط بین دما و انحراف معیار جوابهای پذیرفته شده

۴-۶) ارتباط بین دما و تفاوت جوابهای پذیرفته شده متوالی

یکی از موضوعاتی که در نتایج مورد بررسی قرار گرفت، پراکنش هزینه کل آرایشهای پذیرفته شده در دماهای مختلف است. نمودارهای شماره ۴ و ۵ این موضوع را به ترتیب در دماهای ۲۴۴ و ۵۰ نشان می‌دهد. همانگونه که مشاهده می‌شود میزان پراکنش در دمای ۲۴۴ بسیار زیاد و در دمای ۵۰ کاهش یافته است به گونه‌ای در دمای پایین بسیاری از تکرارها هیچگونه پراکنشی نداشته است. نتایج بدست آمده در این نمودارها، استنباط ارائه شده در خصوص ارتباط بین دما و انحراف معیار تابع هزینه کل را مورد تأیید قرار می‌دهد.



نمودار ۴. ارتباط بین دما و پراکنش جوابهای پذیرفته شده در دمای ۲۴۴

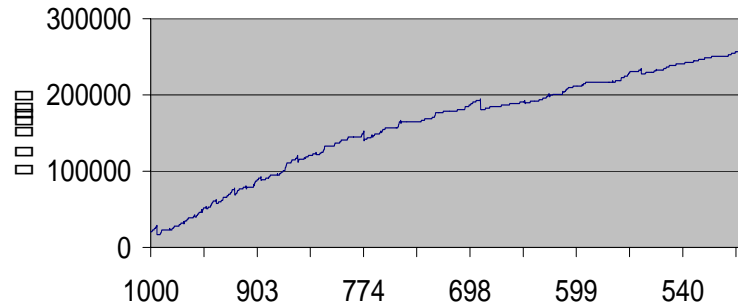


نمودار ۵. ارتباط بین دما و پراکنش جوابهای پذیرفته شده در دمای ۵۰

۵-۶) ارتباط بین هزینه کل و نحوه تغییر آرایش ظرفیت انبارهای میانگیر

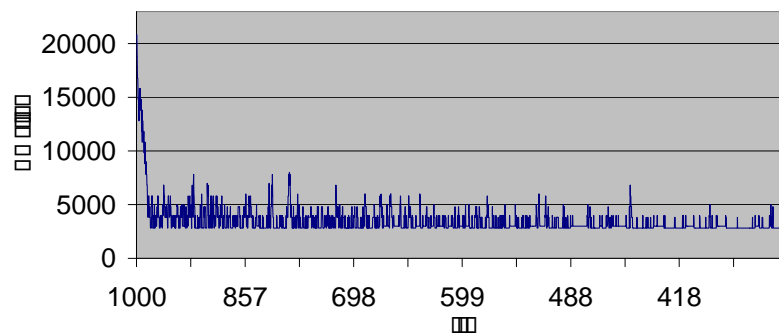
در یکی از حالت‌های بررسی شده سعی گردید تا تغییر آرایش ظرفیت انبارها بطور منطقی انجام پذیرد. بدین ترتیب که تغییر آرایش با در نظر گرفتن هزینه توقف عملیات و هزینه افزایش ظرفیت انبارهای میانگیر صورت گیرد. یعنی در تغییر آرایش، ظرفیت انباری که موجب توقف ماشینها شده، یک واحد افزایش یافت. از سوی دیگر، در صورت عدم استفاده از حداکثر ظرفیت انبارها، ظرفیت آنها به میزان حداکثر تعداد استفاده شده کاهش می‌یافت. انجام این آزمایش در شرایطی که نسبت هزینه افزایش ظرفیت به هزینه توقف کوچک بود، دستیابی به جواب رضایت‌بخش را ممکن می‌ساخت. در آزمایشاتی که نسبت هزینه افزایش ظرفیتها به هزینه توقف عملیات افزایش داشته و منطق تغییر آرایش همچنان به صورت فوق حفظ گردیده، نتیجه بدست آمده دلالت بر آن داشت که الگوریتم SA در این وضعیت از یافتن جواب مناسب به شدت دور گردیده و به سمت جوابهای نامناسب میل نموده است. علت این امر رویکرد الگوریتم SA در اجتناب از انتخاب نقطه بهینه موضعی می‌باشد. زیرا در این وضعیت بخاطر اجتناب از هزینه توقف عملیات و ظرفیت انبارها، افزایش و صرفه‌جویی ناشی از هزینه توقف عملیات کمتر از هزینه افزایش ظرفیت انبارها می‌باشد. این موضوع بر بدتر شدن جوابها تاثیر خواهد گذاشت. زمانیکه یک جواب

نامناسب به صورت احتمالی پذیرفته شود، جواب مدل از یک حالت بد به حالت بدتر تغییر می‌کند. این حالت نیز در پی پذیرش یک جواب بدتر دیگر به صورت پی‌درپی تکرار می‌گردد که تا ثیر آن بدتر شدن تدریجی جواب خواهد بود. نتایج بدست آمده از این آزمایش در نمودار شماره ۶ مشاهده می‌گردد.



نمودار ۶. ارتباط بین دما و هزینه کل (تغییر آرایش منطقی)

بنابراین، روش منطقی تغییر آرایش ظرفیت انبارها تنها برای مواردی کاربرد دارد که هزینه توقف عملیات بیشتر از هزینه افزایش ظرفیت انبارها باشد. در این راستا بررسی‌های بعمل آمده نشان می‌دهد که روش تصادفی تغییر آرایش ظرفیت انبارها به عنوان یک گزینه می‌تواند جواب مناسبی برای شرایطی که هزینه توقف عملیات کمتر از هزینه افزایش ظرفیت انبارها باشد، ارائه نماید. علت این امر آن است که تغییر تصادفی آرایش ظرفیت انبارها، مستقل از ساختار تابع هزینه مدل عمل می‌کند. نتایج آزمایشات با روش تصادفی تغییر آرایشها در نمودار شماره ۷ نشان داده شده است.



نمودار ۷. ارتباط بین دما و هزینه کل (تغییر آرایش تصادفی)

در ارتباط با موضوع فوق، آزمایشات بیشتری انجام که نتایج بدست آمده در جدول شماره ۱ نمایش داده شده است. دمای شروع شبیه‌سازی این آزمایشات ۱۰۰۰ و تعداد تکرار در هر دما ۲۰۰ می‌باشد. همانگونه که مشاهده می‌گردد، عملکرد روش تصادفی تحت تاثیر ساختار هزینه قرار ندارد، اما با توجه به دمای تعادل می‌توان نتیجه گرفت که با افزایش ضرایب هزینه (صرفنظر از نوع آن) دمای تعادل نیز افزایش می‌یابد. در روش منطقی، زمانیکه نسبت هزینه توقف عملیات بیشتر از هزینه افزایش ظرفیت باشد، این موضوع نیز صادق است. بنابراین با حرکت الگوریتم SA در جهت بهبود تابع هدف شبیه‌سازی، با افزایش ضرایب هزینه، دمای تعادل شبیه‌سازی نیز افزایش می‌یابد. همچنین همانطور که اشاره گردید، عملکرد روش تغییر منطقی کاملا تحت تاثیر ساختار هزینه است. نتایج مندرج در جدول به خوبی این موضوع را نشان می‌دهد.

دماي تعادل		هزینه کل		پارامتر هزینه افزایش ظرفیت، توقف عملیات)
تصادفی	منطقی	تصادفی	منطقی	
۲۲,۴۷	۲۴,۸۹	۱۸۱	۱۵۲۳۰۳	(۱, ۵۰)
۲۹,۰۳	۵۱,۰۵	۲۹۹	۳۰۲۷۰۴	(۱, ۱۰۰)
۱۱۰,۱۸	۱۴۲,۳۹	۲۹۹	۶۷۸۰۲۲	(۱, ۳۰۰)
۱۵۷,۷۸	۳۴۰,۵۶	۲۹۹	۴۶۹۲۶۱	(۱, ۶۰۰)
۳۴۰,۵۶	۵۹۸,۰۷	۲۹۹	۲۲۴۰۷۵	(۱, ۱۰۰۰)
۹۵۰	۹۵۰	۲۹۹	۲۰۰۰۸۱	(۱, ۱۰۰۰۰)
۴۱,۰۵۷	۴۳,۷۷	۴۲۰۷۴	۲۹۹۱۱	(۱۰۰, ۱۰)
۲۰,۳	۳۵۸	۷۲۴۵۰	۵۴۰۲۶	(۱۰۰۰, ۱۰)
۳۹,۵۰	۵۹,۵۴	۲۹۹	۳۰۱۳۰۳	(۵۰, ۵۰)
۵۹,۵۴	۸۹,۷۵	۸۳۰۵	۲۹۶۸۱۷	(۱۰۰, ۱۰۰)
۲۶,۲۰	۱۹,۲۶	۵۵۱۹	۳۰۲۳	(۵۰, ۱)
۴۳,۷۷	۶۳,۵۹	۳۷۵۲	۳۰۰۸	(۱۰۰, ۱)
۱۱۰,۱۸	۱۵۷,۷۸	۱۶۳۸۶	۳۰۱۶	(۳۰۰, ۱)
۲۱۴,۶۴	۲۷۷,۳۹	۳۵۹۶۹	۱۳۴۵۸	(۶۰۰, ۱)
۲۵۰,۳۴	۳۰۷,۳۶	۶۲۷۲۳	۳۶۷۲۱	(۱۰۰۰, ۱)
۷۳۵,۰۹	۸۵۷,۳۷	۳۹۳۱۲۷	۳۸۶۰۳۸	(۵۰۰۰, ۱)
۷۵۷,۳۷	۸۱۴,۵۱	۸۰۱۷۷۵	۷۸۸۰۸۵	(۱۰۰۰۰, ۱)

جدول ۱. تاثیر آرایش بر دمای تعادل و هزینه کل

(۷) نتیجه گیری

پيچيدگي و ماهيت تصادفي کارکرد سيستم‌ها به طور عام و سيستم‌هاي توليدي به طور خاص، واقعي است که بيش از بيش در بررسي و تصميم‌گيري آنها نمايان گرديده است. در اين راستا، کمبود منابع و امکانات و رقابت فزاينده جهاني ضرورت استفاده از روشهاي کارآمد را ايجاد مي‌نمايد. لذا بکارگيري مدلهاي تحليلي به خاطر ساده‌سازي بيش از اندازه و نادیده انگاشتن برخي از ابعاد سيستمها، با محدوديتهاي قابل توجهي روبه‌رو است. در اين فضا شبیه‌سازي کامپيوتري به عنوان پاسخي به محدوديتهاي مدلهاي تحليلي به طور وسيع مورد استفاده قرار گرفت. اما به زودي بزرگترين ضعف اين رويکرد نيز مشخص شد؛ عدم ارائه جواب بهينه سيستم. در پاسخ به اين نقطه ضعف روشهاي متنوع بهينه‌سازي شبیه‌سازي ارائه گرديدند. الگوريتم SA، يکي از اين روشهاست که در زمره روشهاي ابتکاري آن قرار دارد. ایده اصلي اين روش از فرآيند ذوب فلزات اخذ شده و با اعمال آن در مفاهيم بهينه‌سازي، الگوريتم جستجوي کارايي را پديد آورده است. نقطه قوت SA، گريز از نقطه بهينه موضعي و ارائه جواب رضايتمند بدون جستجوي کليه نقاط فضاي موجه مسئله است. در اين مقاله، چگونگي پياده‌سازي SA در شبیه‌سازي مساله تعيين بهينه ظرفيت انبارهاي ميانگير خطوط توليد، مورد بررسي قرار گرفت. هدف مساله فوق عبارت بود از تعيين ظرفيت بهينه هر يک از انبارها به گونه‌اي که بيشترين بازده سيستم بدست آيد. در اين زمينه يک خط توليد با شش ايستگاه کاري و پنج انبار ميانگير به عنوان نمونه انتخاب گرديد. تابع هزينه مشتمل بر هزينه توقف عمليات (به علت تکميل بودن ظرفيت انبارهاي ميانگير) و هزينه افزايش هر واحد ظرفيت انبار ميانگير است. پس از تلفيق مفهومي SA و شبیه‌سازي براي انجام آزمايشات اقدام به طراحي و تهيه يک نرم‌افزار گرديد. به کمک نرم‌افزار، حالات مختلف خط توليد شامل ضرايب متفاوت تابع هزينه، شرايط متفاوت شروع شبیه‌سازي،

شرایط توقف شبیه‌سازی، تعداد تکرارهای شبیه‌سازی و دو نوع تغییر آرایش تصادفی و منطقی سیستم بررسی گردید. تجزیه و تحلیل نتایج بدست آمده نشان می‌دهد که با افزایش تعداد تکرار و کاهش دما الگوریتم نسبت به پذیرش جواب نامطلوب سخت‌گیری بیشتری اعمال می‌کند. روند بهبود مقدار هزینه کل و تغییر دمای شبیه‌سازی دلالت بر آن داشت که میزان بهبود (در دمای بالا) قابل توجه و در دمای پایین، کمتر است از این موضوع استنباط گردید که رابطه مستقیمی بین کارایی محاسباتی SA و تغییرات دما وجود دارد.

بررسی ارتباط بین دما و انحراف معیار جوابهای پذیرفته شده نشان می‌دهد که به علت پذیرش آسانتر جواب در دمای بالا و فرار الگوریتم از نقاط بهینه موضعی، رابطه مستقیمی بین دما و میزان انحراف معیار جواب وجود دارد. همچنین نتیجه حاصل از بررسی رفتار پراکنش هزینه کل آرایشهای پذیرفته شده در دماهای مختلف، نتیجه فوق را تایید نمود. پراکنش در دمای بالا بسیار زیاد و در دمای پایین کاهش نشان داد. از بررسی تاثیر تغییر آرایش تصادفی و منطقی بر هزینه کل سیستم معلوم گردید که عملکرد روش تصادفی تحت تاثیر چگونگی تنظیم ضرایب هزینه قرار نداشته در حالیکه روش منطقی کاملاً تحت تاثیر پارامترهای هزینه‌ای است. در مواردی که نسبت هزینه توقف عملیات به هزینه افزایش ظرفیت انبار افزایش یافته، روش منطقی بهتر از روش تصادفی عمل می‌کند و برعکس.

بطور کلی نتایج بدست آمده در این تحقیق، دلالت بر کارآمدی SA در بهینه‌سازی شبیه‌سازی مسئله تعیین ظرفیت انبارهای میانگیر خط تولید دارد. برای تحقیقات آتی مقایسه SA با سایر روشهای بهینه‌سازی شبیه‌سازی پیشنهاد می‌گردد. همچنین می‌توان روشهای مختلف تابع تغییر آرایشها را مورد بررسی قرار داد. علاوه تلفیق الگوریتم SA با روشهای هوشمندی که بتواند از بررسی مجدد نقاط اجتناب ورزد به عنوان یکی دیگر از زمینه‌های تحقیق در این حوزه پیشنهاد می‌گردد.

References:

1. Eglese, R., "Scheduling in a cellular manufacturing system: A simulated annealing approach", *International journal of production research*, Vol. 46, pp. 271-281, 1993.
2. Lai, K., Chan, W., "Developing a simulated algorithm for the cutting stock problem", *Computers indo engng*, Vol. 32, pp. 115-127, 1997.
3. Tam, K., "A simulated annealing algorithm for allocating space to manufacturing cells", *International journal of production reseach*, Vol. 30, pp. 63-87, 1991.
4. Van laarhoven, P., et. al., "Job shop scheduling by simulated annealing", *Operation research*, Vol. 40, pp. 113-125, 1992.
5. Sigrun, A., " A review of simulation optimization techniques", *proceedings of the 1998 winter simulation conference*, pp. 151-157, 1998.
6. Carson, Y. and Meria, A., "Simulation Optimization: Methods and applications", *proceedings of the 1997 winter simulation conference*, pp. 118-126, 1997.
7. Sheldon, H., "Analyzing the M/ M/ 1 queue in frequency domain experiments", *Mathematics and Computation*, Vol. 69 pp.185-194 ,1995.
8. Arsham H., "Stochastic optimization of discrete event systems simulation", *Microelectron Reliable*, Vol.36, pp.1357-1368, 1996.
9. Azadivar, F. and Lee Y., "Optimization of discrete variable stochastic systems by computer simulation", *Mathematics and Computers in Simulation*, Vol.30, pp. 331-345, 1998.



10. Fu, M., " Optimization via simulation: A review", *Annals of Operations Research*, Vol. 53, pp. 199-247, 1994.
11. Neddermeijer G. et. al., " A framework for response surface methodology for simulation optimization", *proceedings for the 2000 winter simulation conference*, 2000.
12. James, R. and Sheldon H., "A survey of ranking, selection, and multiple comparison procedures for discrete-event simulation", *proceedings of the 1999 winter simulation conference*, pp. 492-501, 1999.
13. Olafsson, S., "Simulation optimization", *proceedings of the 2002 winter simulation conference*, pp. 79- 84, 2002.
14. Digalskis J. and Margaritis k., "Meta-heuristics Algorithms", *THESSALONIKI*, 2001.
15. Goldberg D., "Genetic algorithms in search, optimization and Machine Learning", *Addison-Wesley, Boston*, 1989.
16. Tanka, H., and Youshimoto, K., "Genetic algorithm Applied to the facility layout problem department of industrial engineering and management", *Waseda University, Tokyo*, 1993.
17. Tate, D. and Smit, A., "A genetic approach to the quadratic assignment problem", *Computers and Operations Research*, Vol. 22, pp. 73-83, 1995.
18. Glover, F. and et al, "A users guide to tabu search", *Annals of operation Research*, Vol. 91, pp. 3-28, 1993.
19. Kendall, G. and et al, "Evolutionary strategies; A new macroeconomic policy tool?", www.cs.nott.ac.uk/~gxm/papers/ifac01div.pdf.
20. Kirkpatric, S., et. al., " Optimization by simulated annealing", *Science*, Vol. 220, pp. 671-680, 1993.
21. Henderson, D., et. al., "Solving the shortest route cut anf fill problem using simulated annealing", Vol. 145, pp. 72-48, 2003.
22. Wang, T. and Wu K., " A parameter set design procedure for the simulated annealing algorithm under the computational time constraint", *Computer and Operations Research*, Vol. 26, pp. 665-678, 1999.
23. Christian, M. and et al, "Determining buffer location and size in production lines using tabu search", *Operational research*, Vol. 106, pp. 301-316, 1998.