

# یک مدل EOQ برای کالاهای فاسدشدنی با رویکرد الگوریتم ژنتیک

جعفر رضایی (دانشجوی دکتری)

مدیریت تکنولوژی، دانشگاه صنعتی دلفند

سید محمدتقی فاطمی قمی (استاد)

دانشکده مهندسی صنایع، دانشگاه صنعتی امیرکبیر

منصور داودی منفرد (دانشجوی دکتری)

دانشکده علوم کامپیوتر، دانشگاه صنعتی امیرکبیر

فاطمه رضایی (دانشجوی کارشناسی ارشد)

دانشکده مهندسی صنایع، دانشگاه علامه طباطبائی

در این نوشتار توسعه‌ی یک مدل کنترل موجودی برای کالاهای فاسدشدنی که در آن، نرخ تقاضا و کمبود به‌طور خطی افزایش می‌یابد، بررسی شده است. از یک تابع توزیع ویبول سه‌پارامتری برای نشان دادن زمان خرابی کالاهای فاسدشدنی استفاده می‌شود. با توجه به این که معمولاً تقاضای یک محصول مصرفی با گذشت زمان تغییر می‌کند، تقاضا وابسته به زمان در نظر گرفته شده است. همچنین معمولاً به‌دلایلی در موجودی کمبود رخ می‌دهد. با در نظر گرفتن عوامل یادشده، یک مدل برنامه‌ریزی غیرخطی ارائه می‌شود. از سوی دیگر با توجه به ابهامی که معمولاً در تخمین هزینه‌های موجودی وجود دارد، در برآورد آنها از مقادیر بازه‌ی استفاده می‌شود. بدین ترتیب، یک مدل برنامه‌ریزی غیرخطی با تابع هدف بازه‌ی حاصل می‌شود که برای حل آن از الگوریتم ژنتیک استفاده می‌شود. در پایان نیز یک مسئله‌ی واقعی برای کاربرد نتایج مدل ارائه، و نتایج آن تحلیل حساسیت می‌شود.

j.rezaei@tudelft.nl  
Fatemi@aut.ac.ir  
mdmonfared@aut.ac.ir  
fatemehrezaei23@yahoo.com

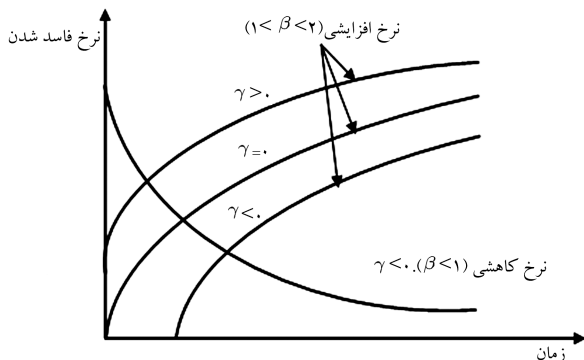
واژگان کلیدی: کنترل موجودی، کالاهای فاسدشدنی، مقدار سفارش اقتصادی، مقادیر بازه‌ی، الگوریتم ژنتیک.

## مقدمه

در یک سازمان مدرن، کنترل موجودی کالاهای فاسدشدنی از اهمیت شایانی برخوردار است. معمولاً فاسدشدن<sup>[۱]</sup> به‌عنوان خرابی، ضایعات، فروپاشی، کهنه‌شدن و تبخیر تعریف می‌شود. در این راستا، منطقی است محصولی را بررسی کنیم که با رسیدن مطلوبیت آن به صفر، دوره‌ی مصرف آن تمام‌شده تلقی شود. برای کالاهایی مثل فلزات سخت، ظروف شیشه‌یی و اسباب‌بازی سرعت خراب‌شدن خیلی پایین است و در تعیین مقدار سفارش آنها در نظر گرفتن میزان خراب‌شدن کالا از اهمیت کم‌تری برخوردار است. اما در نظر گرفتن زمان فاسدشدن تعدادی از کالاها مثل خون، ماهی، توت‌فرونگی، الکل، رادیواکتیو، مواد شیمیایی و دارو غالباً از اهمیت بسیار بالایی برخوردار است. در این زمینه تنی چند از محققین، چند مدل ریاضی برای کنترل موجودی ارائه داده‌اند که در اینجا به تعدادی از مهم‌ترین آنها اشاره می‌شود.

در این نوشتار با استفاده از یک توزیع ویبول سه‌پارامتری مبتنی بر مدل ارائه‌شده توسط محققین<sup>[۱۹]</sup> به توسعه‌ی مدل EOQ برای کالاهای فاسدشدنی با نرخ خرابی متغیر خواهیم پرداخت. این مدل ارائه‌شده<sup>[۱۹]</sup> یک مدل غیرخطی نامعین است که برای حل آن از روش‌های عددی استفاده شده است. در این پژوهش با در نظر گرفتن فضای محدود انبار، یک مدل غیرخطی معین فرموله می‌شود. همچنین به‌منظور در نظر گرفتن ابهام موجود در برآورد هزینه‌های نگهداری، سفارش، کمبود و خرید، از

در این راستا از دور خارج شدن کالاهای مد روز در پایان دوره‌ی ذخیره بررسی شده است.<sup>[۱]</sup> همچنین یک مدل EOQ<sup>۲</sup> برای کالاهای فاسدشدنی و دارای تقاضای معین ارائه شده است.<sup>[۲]</sup> سپس یک مدل EOQ برای کالاهای فاسدشدنی که از توزیع



شکل ۲. رابطه‌ی بین زمان و نرخ فاسدشدن برای تابع توزیع ویبول سه پارامتری.

که در آن  $\gamma$  پارامتر مکان است،  $(t \geq \gamma)$ ،  $\alpha$ ،  $\beta$  و  $t$  نیز قبلاً تعریف شده است. نرخ آبی خرابی مواد فاسدشدنی در زمان  $t$ ،  $Z(t)$ ، از رابطه‌ی ۳ به دست می‌آید:

$$Z(t) = \frac{f(t)}{1 - F(t)} \quad (3)$$

که در آن  $F(t)$  تابع توزیع ویبول تجمعی سه پارامتری بوده و برابر با  $1 - e^{-\alpha(t-\gamma)^\beta}$  است. با جایگذاری  $f(t)$  و  $F(t)$  در رابطه ۳ ساده کردن آن داریم:

$$Z(t) = \alpha\beta(t - \gamma)^{\beta-1} \quad (4)$$

رابطه‌ی بین زمان و نرخ خرابی برای توزیع ویبول سه پارامتری در شکل ۲ نشان داده شده است. از این شکل مشخص می‌شود که توزیع ویبول سه پارامتری برای کالاهایی با هر نرخ خرابی و نیز برای کالاهایی که خرابی آنها بعد از دوره‌ی خاصی شروع می‌شود مناسب است.

### مدل‌سازی مسئله

مفروضات و نمادهای مدل عبارت است از:

۱. نرخ تقاضا،  $f(t)$ ، در زمان  $t$  به وسیله‌ی  $f(t) = a + bt$  معین می‌شود که در آن  $a$  و  $b$  ثابت‌های غیرمنفی‌اند.
  ۲. جایگزینی آبی است و زمان تأخیر صفر است.
  ۳. کمبود در موجودی مجاز است.
  ۴. توزیع زمان خرابی از یک توزیع ویبول سه پارامتری پیروی می‌کند.
  ۵. هزینه‌ی نگهداری موجودی هر واحد در واحد زمان  $(C_1)$ ، هزینه‌ی کمبود هر واحد در واحد زمان  $(C_2)$ ، هزینه‌ی خرید هر واحد  $(C_3)$  و هزینه‌ی سفارش در هر بار  $(A)$  به صورت ثابت‌هایی شناخته شده‌اند. این هزینه‌ها در مدل‌سازی نهایی به صورت اعداد بازه‌ی منظور می‌شوند.
  ۶. حداکثر ظرفیت انبار برابر با  $W$  و فضای اشغال شده توسط هر واحد موجودی برابر با  $\eta$  است.
- یک دوره‌ی موجودی در شکل ۳ نشان داده شده است. که در آن:

$$\gamma' = \gamma, \quad \gamma \geq 0; \quad \gamma' = 0, \quad \gamma \leq 0$$

اعداد بازه‌ی در مدل‌سازی مسئله استفاده می‌شود. بنابراین، مدل حاصله یک مدل غیرخطی مقید با اعداد بازه‌ی خواهد بود. از آنجا که در اغلب کاربردهای واقعی، تصمیم‌گیرنده قادر به تخمین پارامترهای هزینه نیست، استفاده از اعداد بازه‌ی مدل را به دنیای واقعی نزدیک‌تر و آن را قابل قبول‌تر می‌سازد. لازمه‌ی حل مدل حاصل، استفاده از روش‌های تقریب عددی، ابتکاری و شبه‌ابتکاری است. الگوریتم ژنتیک، به عنوان یک الگوریتم شبه‌ابتکاری، نه تنها قادر به حل چنین مسائلی است، بلکه به تصمیم‌گیرنده انتخاب‌های بیشتری ارائه می‌کند، به نحوی که تصمیم‌گیرنده قادر خواهد بود براساس اطلاعات سطح بالاتر (که عمدتاً کیفی‌اند و در مدل‌سازی مسئله وارد نمی‌شوند) تصمیم نهایی را اتخاذ کند و از میان مجموعه‌ی از جواب‌ها، یک جواب را انتخاب کند.<sup>[۲۰]</sup> براین اساس، در حل مدل از یک الگوریتم ژنتیک استفاده می‌شود. این مدل برای یک محصول لبنی در کارخانه‌ی تولید محصولات لبنی به کار گرفته شده است که نتایج آن در انتهای نوشتار ارائه می‌شود. همچنین حساسیت مدل نسبت به تغییر در پارامترهای هزینه تحلیل می‌شود.

در این نوشتار ابتدا توابع توزیع ویبول را مورد بحث قرار خواهیم داد و سپس به مدل‌سازی مسئله خواهیم پرداخت. در ادامه با بررسی حساب بازه‌های متناهی، به تبدیل مدل با اعداد بازه‌ی می‌پردازیم. در بخش بعدی الگوریتم ژنتیک و نحوه‌ی حل مسئله با استفاده از این الگوریتم تشریح می‌شود. کاربرد واقعی مدل برای یکی از محصولات یک کارخانه‌ی تولید محصولات لبنی منظور شده و بررسی می‌شود.

### توابع توزیع ویبول

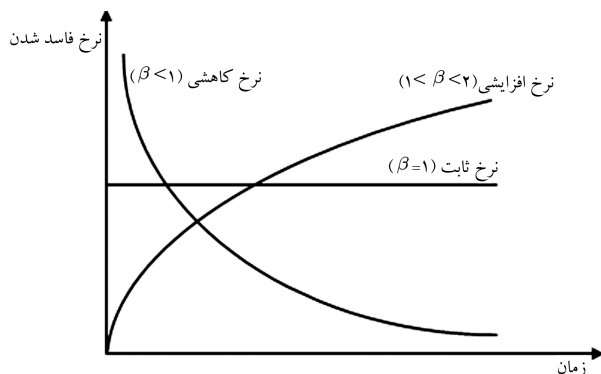
تابع نرخ آبی  $Z(t)$  برای تابع ویبول دو پارامتری مطابق رابطه‌ی ۱ است:

$$Z(t) = \alpha\beta t^{\beta-1} \quad (1)$$

که در آن  $\alpha$  پارامتر مقیاس  $(\alpha > 0)$ ،  $\beta$  پارامتر شکل  $(\beta > 0)$  و  $t$  زمان خرابی  $(t > 0)$  است. رابطه‌ی بین زمان و نرخ خرابی برای این توزیع در شکل ۱ نشان داده شده است. رابطه‌ی ۱ و شکل ۱ نشان می‌دهد که توزیع ویبول دو پارامتری برای کالاهایی مناسب است که از نرخ کاهش خرابی برخوردار است و نرخ خرابی ابتدایی آنها بسیار بالا است. همچنین این توزیع ممکن است برای کالاهایی با نرخ افزایشی خرابی استفاده شود به شرطی که نرخ خرابی ابتدایی آنها نزدیک صفر باشد.

برای رفع این کاستی‌ها، یک مدل کلی EOQ با توزیع ویبول سه پارامتری برای بیان زمان خرابی توسعه داده شده است.<sup>[۲۱]</sup> تابع چگالی  $f(t)$  این توزیع چنین است:

$$f(t) = \alpha\beta(t - \gamma)^{\beta-1} e^{-\alpha(t-\gamma)^\beta} \quad (2)$$



شکل ۳. رابطه‌ی بین زمان و نرخ فاسدشدن برای تابع توزیع ویبول دو پارامتری.

با استفاده از شرط مرزی قید شده در رابطه‌ی ۶ داریم:

$$I_0 = a\gamma' + \frac{1}{\gamma} b\gamma'^t + \frac{\int_{\gamma'}^{t_1} (a+bt)e^{\alpha(t-\gamma)^\beta} dt}{e^{\alpha(\gamma'-\gamma)^\beta}} \quad (11)$$

همچنین از رابطه‌های ۱۰ و ۱۱ داریم:

$$I_1(t) = \frac{\int_t^{t_1} (a+bt)e^{\alpha(t-\gamma)^\beta} dt}{e^{\alpha(\gamma'-\gamma)^\beta}}, \quad \gamma' \leq t \leq t_1 \quad (12)$$

پاسخ معادله‌ی دیفرانسیل (رابطه‌ی ۷)، بعد از استفاده از شرط مرزی قید شده در رابطه‌ی ۸ عبارت خواهد بود از:

$$I_2(t) = -a(t-t_1) - \frac{1}{\gamma} b(t_1-t_1), \quad t_1 \leq t \leq T \quad (13)$$

حال تقاضای واقعی در فاصله‌ی  $(0, T)$  عبارت است از:

$$\int_0^{t_1} (a+bt)dt = at_1 + \frac{1}{2}bt_1^2 \quad (14)$$

از این رو مقدار خرابی در فاصله‌ی مذکور معادل  $I_0 - at_1 - \frac{1}{2}bt_1^2$  خواهد بود. با این فرض که  $0 < K < 1$ ,  $t_1 = KT$  به صورت سری‌های نامحدود و سپس انتگرال‌گیری جزء به جزء داریم:

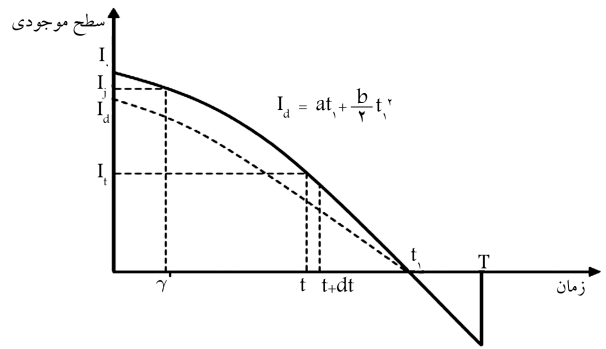
$$I_0 = a\gamma' + \frac{1}{\gamma} b\gamma'^t + e^{-\alpha(\gamma'-\gamma)^\beta} \left[ (a+b\gamma) \sum_{n=0}^{\infty} \frac{\alpha^n (KT-\gamma)^{n\beta+1}}{n!(n\beta+1)} + b \sum_{n=0}^{\infty} \frac{\alpha^n (KT-\gamma)^{n\beta+2}}{n!(n\beta+2)} - (a+b\gamma) \sum_{n=0}^{\infty} \frac{\alpha^n (\gamma'-\gamma)^{n\beta+1}}{n!(n\beta+1)} - b \sum_{n=0}^{\infty} \frac{\alpha^n (\gamma'-\gamma)^{n\beta+2}}{n!(n\beta+2)} \right] \quad (15)$$

کل هزینه سیستم برای دوره‌ی زمانی  $T$  مجموع هزینه‌ی نگهداری موجودی، هزینه‌ی خرابی، هزینه‌ی کمبود و هزینه‌ی سفارش خواهد بود. از این رو متوسط هزینه‌ی سیستم  $C(T, t_1)$  در هر واحد زمان مطابق رابطه‌ی ۱۶ به دست می‌آید:

$$C = \frac{1}{T} C_1 I_0 t_1 + \frac{C_r}{T} (I_0 - at_1 - \frac{1}{2}bt_1^2) - \frac{C_r}{T} \int_{t_1}^T I_2(t)dt + \frac{A}{T} \quad (16)$$

که در آن  $\frac{1}{T} C_1 I_0 t_1$  هزینه‌ی نگهداری موجودی،  $C_r \int_{t_1}^T I_2(t)$  هزینه‌ی کمبود و  $C_r (I_0 - at_1 - \frac{1}{2}bt_1^2)$  هزینه‌ی خرابی است. گرچه منحنی اتمام موجودی در اینجا یک خط راست نیست، متوسط هزینه‌ی موجودی در اینجا به‌طور تقریبی مشابه شکل مورد استفاده در مدل‌های EOQ با نرخ ثابت تقاضا و بدون خرابی است. بدون این تقریب، حل مدل بسیار پیچیده می‌شود. با ساده‌سازی مختصر در رابطه‌ی ۱۶ داریم:

$$C(K, T) = \left( \frac{1}{\gamma} C_1 K + \frac{C_r}{T} \right) \left[ a\gamma' + \frac{1}{\gamma} b\gamma'^t + e^{-\alpha(\gamma'-\gamma)^\beta} \left[ (a+b\gamma) \sum_{n=0}^{\infty} \frac{\alpha^n (KT-\gamma)^{n\beta+1}}{n!(n\beta+1)} + b \sum_{n=0}^{\infty} \frac{\alpha^n (KT-\gamma)^{n\beta+2}}{n!(n\beta+2)} - (a+b\gamma) \sum_{n=0}^{\infty} \frac{\alpha^n (\gamma'-\gamma)^{n\beta+1}}{n!(n\beta+1)} - b \sum_{n=0}^{\infty} \frac{\alpha^n (\gamma'-\gamma)^{n\beta+2}}{n!(n\beta+2)} \right] - aC_r K - \frac{1}{\gamma} bK^t TC_r + \frac{1}{\gamma} aC_r T(1-K)^t - \frac{1}{\gamma} bC_r T^t(1-K)(2K^t - K - 1) + \frac{A}{T} \right] \quad (17)$$



شکل ۳. مدل EOQ با در نظر گرفتن کمبود و نرخ فاسد شدن.

فرض می‌شود که  $\gamma'$  کم‌تر از زمان دوره‌ی  $T$  است، در غیر این صورت جوابی معمولی به دست می‌آید چرا که در این صورت در دوره‌ی موجودی هیچ خرابی نخواهیم داشت. سطح ابتدایی انبار  $I_0$  در نظر گرفته می‌شود. موجودی به تدریج و با گذشت زمان تمام می‌شود که قسمتی از آن صرف تأمین تقاضا می‌شود و قسمتی دیگر به علت خرابی کاهش می‌یابد و نهایتاً در زمان  $t_1 < T$  سطح موجودی به صفر می‌رسد.  $(t_1, T)$  فاصله‌ی کمبود است. فرض کنید  $I_1(t)$  سطح موجودی در زمان  $t$  باشد. بنابراین وضعیت آنی  $I_1(t)$  در زمان  $t$  به وسیله‌ی معادله‌ی دیفرانسیل ۵ به دست می‌آید:

$$\frac{dI_1(t)}{dt} = -I_1(t)Z(t) - f(t), \quad \gamma' \leq t \leq t_1 \quad (5)$$

با شرط مرزی

$$I_1(0) = I_0 \quad (6)$$

موجودی به تدریج با گذشت زمان به علت دو عامل تقاضا و خرابی تمام می‌شود. عبارت اول سمت راست رابطه‌ی ۵ کاهش موجودی در هر واحد زمان ناشی از خرابی را محاسبه می‌کند، در حالی که عبارت دوم کاهش موجودی ناشی از تقاضا را بیان می‌کند. علامت منفی نشان می‌دهد که سطح موجودی در طول زمان از بابت این دو عامل کاهش می‌یابد. معادله‌ی دیفرانسیل مربوط به نرخ آنی کمبود  $I_2(t)$  در فاصله‌ی  $(t_1, T)$  عبارت است از:

$$\frac{dI_2(t)}{dt} = f(t), \quad t_1 \leq t \leq T \quad (7)$$

با شرط مرزی

$$I_2(t_1) = 0 = I_1(t_1) \quad (8)$$

$$Z(t) = \alpha\beta(t-\gamma)^{\beta-1} \text{ و } f(t) = a+bt$$

چنانچه نرخ خرابی در فاصله‌ی  $(0, \gamma')$  صفر باشد، سطح موجودی  $I_{\gamma'}$  در پایان دوره‌ی  $\gamma'$  را می‌توان چنین نوشت:

$$I_{\gamma'} = I_0 - \int_0^{\gamma'} (a+bt)dt = I_0 - a\gamma' - \frac{b\gamma'^t}{2} \quad (9)$$

جواب معادله‌ی دیفرانسیل ۵ عبارت است از:

$$I_1(t) = \frac{I_{\gamma'} e^{\alpha(\gamma'-\gamma)^\beta} - \int_{\gamma'}^t (a+bt)e^{\alpha(t-\gamma)^\beta} dt}{e^{\alpha(t-\gamma)^\beta}}, \quad \gamma' \leq t \leq t_1 \quad (10)$$

که در آن  $\lambda$  یک عدد حقیقی است.  
 تعریف ۲: برای مسئله‌ی کمینه‌سازی رابطه‌ی  $\leq_P$  برای دو عدد بازه‌ی  $A = [a_L, a_R]$  و  $B = [b_L, b_R]$  به صورت زیر تعریف می‌شود:

$$A \leq_P B \Leftrightarrow a_C \leq b_C \quad (25)$$

$$A <_P B \Leftrightarrow A \leq_P B \wedge A \neq B \quad (26)$$

بر اساس این تعریف می‌توان به سادگی به تعریف رابطه‌ی ترتیبی  $\geq_P$  بین دو عدد بازه‌ی زیر پرداخت:

$$A \geq_P B \Leftrightarrow a_C \geq b_C \quad (27)$$

$$A >_P B \Leftrightarrow A \geq_P B \wedge A \neq B \quad (28)$$

### تبدیل مدل با اعداد بازه‌ی

با استفاده از مفاهیم قسمت قبل برای مقادیر بازه‌ی  $C_1, C_2, C_r$  و  $A$  داریم:

$$C(K, L) \in [C_L(K, T), C_R(K, T)] \quad (29)$$

که در آن:

$$C_L(K, T) = \left( \frac{1}{\gamma} C_{\setminus L} K + \frac{C_{rL}}{T} \right) [a\gamma' + \frac{1}{\gamma} b\gamma'^r + e^{-\alpha(\gamma'-\gamma)\beta} [(a+b\gamma) \sum_{n=0}^{\infty} \frac{\alpha^n (KT-\gamma)^{n\beta+1}}{n!(n\beta+1)} + b \sum_{n=0}^{\infty} \frac{\alpha^n (KT-\gamma)^{n\beta+2}}{n!(n\beta+2)} - (a+b\gamma) \sum_{n=0}^{\infty} \frac{\alpha^n (\gamma'-\gamma)^{n\beta+1}}{n!(n\beta+1)} - b \sum_{n=0}^{\infty} \frac{\alpha^n (\gamma'-\gamma)^{n\beta+2}}{n!(n\beta+2)}] - aC_{rR}K - \frac{1}{\gamma} bK^r TC_{rR} + \frac{1}{\gamma} aC_{rL}T(1-K)^r - \frac{1}{\phi} bC_{rR}T^r(1-K)(2K^r - K - 1) + \frac{A_L}{T}$$

$$s.t. \quad \frac{1}{\gamma} \eta \frac{I_0 t_1}{T} \leq W \quad (30)$$

$$C_R(K, T) = \left( \frac{1}{\gamma} C_{\setminus R} K + \frac{C_{rR}}{T} \right) [a\gamma' + \frac{1}{\gamma} b\gamma'^r + e^{-\alpha(\gamma'-\gamma)\beta} [(a+b\gamma) \sum_{n=0}^{\infty} \frac{\alpha^n (KT-\gamma)^{n\beta+1}}{n!(n\beta+1)} + b \sum_{n=0}^{\infty} \frac{\alpha^n (KT-\gamma)^{n\beta+2}}{n!(n\beta+2)} - (a+b\gamma) \sum_{n=0}^{\infty} \frac{\alpha^n (\gamma'-\gamma)^{n\beta+1}}{n!(n\beta+1)} - b \sum_{n=0}^{\infty} \frac{\alpha^n (\gamma'-\gamma)^{n\beta+2}}{n!(n\beta+2)}] - aC_{rL}K - \frac{1}{\gamma} bK^r TC_{rL} + \frac{1}{\gamma} aC_{rR}T(1-K)^r - \frac{1}{\phi} bC_{rL}T^r(1-K)(2K^r - K - 1) + \frac{A_R}{T}$$

$$s.t. \quad \frac{1}{\gamma} \eta \frac{I_0 t_1}{T} \leq W \quad (31)$$

مسئله‌ی فوق یک مسئله‌ی بهینه‌یابی غیرخطی مقید با تابع هدف بازه‌ی است که برای حل آن از یک الگوریتم ژنتیک استفاده می‌کنیم.

از آنجا که متوسط موجودی برابر با  $\frac{1}{\gamma} \frac{I_0 t_1}{T}$  و فضای ذخیره‌ی هر واحد موجودی برابر با  $\eta$  است، کل فضای اشغال شده  $\frac{1}{\gamma} \eta \frac{I_0 t_1}{T}$  خواهد بود. فضای کل انبار برابر با  $W$  فرض می‌شود و از این رو محدودیت انبار که ایجاد می‌شود عبارت است از:

$$\frac{1}{\gamma} \eta \frac{I_0 t_1}{T} \leq W \quad (18)$$

در نتیجه مسئله‌ی غیرخطی مقید حاصل، با در نظر گرفتن محدودیت انبار، عبارت است از:

$$Min \ C(K, T) = \left( \frac{1}{\gamma} C_{\setminus} K + \frac{C_r}{T} \right) [a\gamma' + \frac{1}{\gamma} b\gamma'^r + e^{-\alpha(\gamma'-\gamma)\beta} [(a+b\gamma) \sum_{n=0}^{\infty} \frac{\alpha^n (KT-\gamma)^{n\beta+1}}{n!(n\beta+1)} + b \sum_{n=0}^{\infty} \frac{\alpha^n (KT-\gamma)^{n\beta+2}}{n!(n\beta+2)} - (a+b\gamma) \sum_{n=0}^{\infty} \frac{\alpha^n (\gamma'-\gamma)^{n\beta+1}}{n!(n\beta+1)} - b \sum_{n=0}^{\infty} \frac{\alpha^n (\gamma'-\gamma)^{n\beta+2}}{n!(n\beta+2)}] - aC_rK - \frac{1}{\gamma} bK^r TC_r + \frac{1}{\gamma} aC_rT(1-K)^r - \frac{1}{\phi} bC_rT^r(1-K)(2K^r - K - 1) + \frac{A}{T}$$

$$s.t. \quad \frac{1}{\gamma} \eta \frac{I_0 t_1}{T} \leq W \quad (19)$$

### حساب بازه‌های متناهی

یک عدد بازه‌ی یک بازه‌ی بسته است که طبق رابطه‌ی ۲۰ تعریف می‌شود: [۲۲]

$$A = [a_L, a_R] = \{x : a_L \leq x \leq a_R, x \in R\} \quad (20)$$

که در آن  $a_L$  و  $a_R$  به ترتیب حدود چپ و راست  $A$  هستند. می‌توان با استفاده از مرکز و شعاع، نیز به تعریف یک عدد بازه‌ی پرداخت:

$$A = \langle a_C, a_W \rangle = \{x : a_C - a_W \leq x \leq a_C + a_W, x \in R\} \quad (21)$$

که در آن مرکز و شعاع به ترتیب برابرند با:  $(a_L + a_R)/2$  و  $(a_R - a_L)/2$ .

به طور خاص، هر عدد حقیقی را می‌توان به صورت یک بازه نشان داد. به عبارت دیگر، به ازاء هر  $x \in R$ ، می‌توان  $x$  را به صورت بازه  $[x, x]$  با طول صفر بیان کرد.

حال تعریف چهار عمل اصلی بر بازه‌ها را ارائه می‌دهیم. [۲۲]

تعریف ۱:  $\{+, -, \dots\} \in *$  را به عنوان یک عملگر دوتایی بر مجموعه‌ی اعداد حقیقی در نظر بگیرید. اگر  $A$  و  $B$  دو بازه بسته باشند،  $A * B = \{a * b : a \in A, b \in B\}$  به عنوان یک عملگر دوتایی بر مجموعه بازه‌های بسته تعریف می‌شود. در مورد تقسیم فرض بر این است که  $0 \notin B$ .

با استفاده از تعریف ۱ برای دو عدد بازه‌ی  $A = [a_L, a_R]$  و  $B = [b_L, b_R]$  داریم:

$$A + B = [a_L + b_L, a_R + b_R] \quad (22)$$

$$A - B = [a_L - b_L, a_R - b_R] \quad (23)$$

$$\lambda A = \begin{cases} [\lambda a_L, \lambda a_R] & \text{if } \lambda \geq 0 \\ [\lambda a_R, \lambda a_L] & \text{if } \lambda < 0 \end{cases} \quad (24)$$

## الگوریتم ژنتیک

الگوریتم ژنتیک یکی از روش‌های جست‌وجو در فضا‌های پیچیده و چندبعدی است که غالباً برای حل مسائلی که راه حل خاصی ندارند، یا در صورت داشتن راه حل از پیچیدگی زمانی فراتر از چندجمله‌ی برخوردارند، استفاده می‌شود. [۲۰ و ۲۳]

از آنجا که مبنای کار الگوریتم ژنتیک، جست‌وجوی تصادفی و احتمالی در فضای جست‌وجو است، بهینه‌بودن جواب یا جواب‌های نهایی را تضمین نمی‌کند ولی با احتمال بسیار بالا مشخص شده است که در بسیاری از مسائل جواب نهایی الگوریتم همان جواب بهینه‌ی عمومی است. [۲۴، ۲۵] یکی از ویژگی‌های مهم الگوریتم‌های ژنتیک این است که می‌توانند طیفی از جواب‌های بهینه یا بسیار نزدیک به بهینه را پیدا کنند. [۲۶]

الگوریتم ژنتیک همانند سایر الگوریتم‌های تکاملی ابتدا جمعیتی از جواب‌ها را به صورت تصادفی انتخاب، و سپس با استفاده از عملگرهای انتخاب و عملگرهای ژنتیک - نظیر ترکیب و جهش - و در بیشتر موارد عملگر نخبه‌گرایی، سعی می‌کند این جواب‌ها یا اصطلاحاً کروموزوم‌ها را تکامل بخشد (بهینه کند). در ادامه به تشریح نحوه‌ی عملکرد الگوریتم ژنتیک و عملگرهای آن می‌پردازیم.

### نمایش کروموزوم‌ها و مقاردهی اولیه‌ی آنها

چگونگی نمایش یک کروموزوم اولین گام پیاده‌سازی الگوریتم‌های ژنتیک و در عین حال گامی بسیار مهم در موفقیت و عملکرد صحیح الگوریتم است، به گونه‌ی که گام‌های بعدی و عملگرهای ژنتیک متأثر از این نمایش‌اند. به طور کلی می‌توان دو گونه نمایش دودویی از کروموزوم‌ها، یا نمایش حقیقی از آنها ارائه کرد. همچنین یک کروموزوم از الگوریتم می‌تواند نشان‌دهنده‌ی یک جواب کامل یا بخشی از یک جواب کامل باشد و با گذر نسل علاوه بر بهینگی از نظر برآزش، از این حیث نیز تکامل یابد. از آنجا که در مدل ارائه شده متغیرهای تصمیم مسئله عبارت‌اند از  $k$  و  $T$ ، در پیاده‌سازی الگوریتم هر کروموزوم به شکل ساده یک زوج متغیر حقیقی، بازه‌ی از زمان که کمبود وجود ندارد ( $k$ ) و طول دوره ( $T$ ) تشکیل می‌شود. همچنین برای ساخت جمعیت اولیه از جواب‌ها، از توزیع تصادفی یکنواخت با توجه به محدوده‌ی پایین و بالای متغیرهای تصمیم ( $a_L$  و  $a_R$ ) استفاده شده است. در طول اجرای الگوریتم نیز، پس از اعمال عملگرهای ترکیب و جهش، محدوده‌ی مجاز متغیرها بررسی می‌شوند و در صورت نیاز آن را تغییر می‌دهیم.

از دیگر پارامترهای مؤثر در مسئله می‌توان به اندازه جمعیت یا همان تعداد کروموزوم‌های موجود در هر نسل اشاره کرد. با توجه به تصادفی بودن جست‌وجوی الگوریتم ژنتیک، و به طور کلی الگوریتم‌های تکاملی، روشن است که احتمال یافتن جواب بهتر در یک جمعیت بزرگ‌تر بیشتر از یک جمعیت کوچک‌تر است. بنابراین هرچه اندازه‌ی جمعیت بزرگ‌تر باشد احتمال موفقیت الگوریتم در دست‌یابی به جواب بهینه بیشتر می‌شود. تنها محدودیتی که در این مورد وجود دارد بحث منابع رایانه‌ی است که پیچیدگی فضایی و زمانی الگوریتم را شامل می‌شود. از این رو اندازه‌ی جمعیت را می‌توان با توجه به این دو بحث به صورت کارآیی انتخاب کرد. در پیاده‌سازی این نوشتار از اندازه جمعیت  $10^5$  استفاده شده است.

### تابع برآزش و عملگر انتخاب

یکی از مهم‌ترین ویژگی‌های الگوریتم‌های ژنتیک این است که قادرند در مورد مسائلی که تابع هدف مشخص و صریحی<sup>۲</sup> ندارند، اقدام به بهینه‌یابی کنند. در واقع تنها

کافی است که بتوان بین دو جواب مسئله یک مقایسه‌ی تقریبی از نظر «بهتر بودن» انجام داد. همچنین در مسائل بهینه‌یابی همراه با محدودیت، ارضاء محدودیت‌ها به عنوان یک مسئله‌ی چالش‌انگیز مطرح بوده و روش‌های بسیار مختلفی برای این منظور ارائه شده است که از محبوب‌ترین آنها می‌توان به روش تابع جریمه و انتخاب بر پایه‌ی موجه‌بودن اشاره کرد. [۲۰] از آنجا که در مدل ارائه شده در این نوشتار، رابطه‌ی ۱۸ فضای جواب را به دو بخش موجه و ناموجه تقسیم می‌کند بنابراین باید بتوان در مرحله‌ی اول به سمت جواب‌های موجه و سپس به سمت جواب‌های بهینه حرکت کرد، و یا این دو هدف را هم‌زمان در روند اجرای الگوریتم در نظر گرفت.

برای پیاده‌سازی الگوریتم ژنتیک ارائه شده در این نوشتار با استفاده از شیوه‌ی «انتخاب مسابقی محدود»<sup>۳</sup> سعی می‌شود به طور هم‌زمان نسبت به ارضاء محدودیت‌ها، عملگر ترکیب و نحوه‌ی برآزش جواب‌ها اقدام شود. به طور کلی عملگرهای انتخاب مسابقی به این نحو عمل می‌کنند که در هر بار دو یا چند جواب را از بین جمعیت انتخاب کرده و با یک معیار مشخص «بهتر بودن» میان آنها مسابقی برگزار می‌شود، برنده‌ی این مسابقه به عنوان والد برای تولید نسل بعدی انتخاب می‌شود. عملگر انتخاب مسابقی محدود بین دو جواب فرضی  $p$  و  $q$  از جمعیت به شکل زیر انجام می‌شود:

- اگر هر دو جواب  $p$  و  $q$  ناموجه باشند، جوابی که نقض محدودیت کم‌تری دارد، برنده خواهد بود.
  - اگر یکی از جواب‌ها موجه و دیگری ناموجه باشد، برنده جواب موجه است.
  - اگر  $p$  و  $q$  هر دو موجه باشند، برنده جوابی است که تابع هدف بهتری دارد.
- قانون اول بر جواب‌های موجه تأکید دارد و این حالت به ویژه در نسل‌های اولیه‌ی الگوریتم روی می‌دهد؛ مشخص است که با استفاده از آن می‌توان به سمت نواحی موجه حرکت کرد. قانون دوم و سوم نیز به ترتیب با تأکید بر جواب‌های موجه در مقابل جواب‌های ناموجه و جوابی با برآزش بهتر ارائه شده‌اند. روشن است که این امکان وجود دارد که در یک نسل هر سه قانون مورد استفاده قرار گیرد، اما عموماً قوانین اول، دوم و سوم در نسل‌های ابتدایی و میانی و انتهای الگوریتم بیشتر مورد استفاده قرار می‌گیرند. به کارگیری هم‌زمان این قوانین کمک می‌کند که جواب‌های ناموجه به سمت نواحی موجه، و جواب‌های موجه به سمت جواب‌های (های) بهینه سوق داده شوند.

### عملگر ترکیب

این عملگر ژنتیک از ترکیب کروموزوم‌ها یا همان جواب‌ها، کروموزوم‌های جدیدی را ایجاد می‌کند. در اصطلاح، کروموزوم‌های به دست آمده را «فرزند» و کروموزوم‌های اولیه را «والدین» می‌نامیم. طبق نظریه‌ی تکامل داروین احتمالاً فرزند ژن‌های برتری را از دو والد خود گرفته و از هر دوتای آنها کامل‌تر (یعنی بهتر) می‌شود و به سمت بهینه سوق می‌یابد. نوع ترکیب معمولاً به ماهیت مسئله و نحوه‌ی نمایش کروموزوم‌ها بستگی دارد. گرچه روش‌های بسیار متنوعی برای پیاده‌سازی عملگر ترکیب وجود دارد، در اینجا دو نمونه‌ی ساده از آنها به اختصار ارائه شده است. روش ترکیب دودویی تک‌نقطه‌ی یا دونقطه‌ی از روش‌های ساده‌ی عمل ترکیب هستند. در این شیوه ضروری است ابتدا کروموزوم‌های والد به عدد دودویی متناظر کد شوند و سپس به صورت بیتی با یکدیگر ترکیب شده و کروموزوم (یا کروموزوم‌های) فرزند را به وجود آورند، و در نهایت کروموزوم (یا کروموزوم‌های) فرزند به صورت پارامتر حقیقی تبدیل شود. یکی دیگر از روش‌های ترکیب که در اینجا استفاده شده است روش ترکیب خطی است. در این روش پس از انتخاب کروموزوم‌های والد و با استفاده از یک عدد تصادفی،  $\lambda$ ، که معمولاً بین صفر و یک است، ترکیب خطی حاصل از آنها

انجام می‌شود و دو فرزند به وجود می‌آید.<sup>[۲۷]</sup> این عمل به تعداد لازم تکرار می‌شود تا جمعیت جدید ساخته شود. از آنجا که کروموزوم‌ها را در پیاده‌سازی الگوریتم ژنتیک به صورت حقیقی کد کرده‌ایم، برای ترکیب آنها نیز از ترکیب خطی استفاده می‌کنیم. اما برای افزایش قدرت جست‌وجوی الگوریتم از مقداری تصادفی بین صفر و ۱/۲ برای  $\lambda$  استفاده شده است؛ این کار باعث می‌شود که گاهی فرزند خارج از ترکیب خطی ولی در همان راستا تولید شود.<sup>[۲۸]</sup>

همچنین در بعضی از موارد از عملگر «نخبه‌گرایی»<sup>۵</sup> نیز استفاده می‌شود. در این عمل درصدی مشخص از بهترین‌های نسل قدیم بدون هیچ تغییری در نسل جدید کپی می‌شوند. نخبه‌گرایی در همگرایی جواب‌ها به سمت ناحیه‌ی بهینه بسیار مؤثر است؛ بنابراین رعایت دقت در انتخاب درصد مناسب برای آن لازم است.<sup>[۲۹،۳۰]</sup> در پیاده‌سازی الگوریتم ما در هر نسل ۱۰ درصد از بهترین‌ها را به نسل بعدی منتقل می‌کنیم و مابقی نسل بعدی را با استفاده از عملگر ترکیب (و البته جهش) ایجاد می‌کنیم. در واقع یک الگوریتم ژنتیک موفق باید قادر به برقراری تعادل بین میزان جست‌وجو و میزان بهره‌برداری باشد. تأکید بیشتر به دو عملگر ترکیب و نخبه‌گرایی به نوعی تأکید بر میزان بهره‌برداری از بهترین جواب‌های به دست آمده است. تأکید بیش از حد بر این دو عملگر باعث همگرایی احتمالی جمعیت به سمت نزدیک‌ترین بهینه - که معمولاً یک بهینه‌ی محلی است (نه عمومی) - می‌شود. از این رو باید به نوعی بر میزان قدرت جست‌وجوی الگوریتم افزود. اولین روش همان عملگر ترکیب بود که جواب‌های جدید تولید می‌کند، اما یک گام مؤثر در این جهت به‌کارگیری عملگر جهش است که در ادامه آورده شده است.

### عملگر جهش

در تمام الگوریتم‌های تکاملی شیوه‌ی برای گریختن از بهینه‌های محلی وجود دارد که عموماً به صورت تصادفی عمل می‌کند. در الگوریتم ژنتیک عملگر جهش عهده‌دار این وظیفه است و از همگرایی سریع جمعیت به سمت بهینه‌های محلی جلوگیری می‌کند. یکی از شیوه‌های پیاده‌سازی جهش هنگامی که از کد دودویی استفاده می‌شود این است که یک یا چند ژن یا بیت از یک کروموزوم انتخاب شده و به صورت تصادفی تغییر داده می‌شود، یا جای دو ژن از یک کروموزوم با یکدیگر عوض می‌شود. در مواردی که از پارامترهای حقیقی برای نگه‌داری مقدار متغیرها استفاده می‌شود نیز می‌توان برای پیاده‌سازی جهش، آن‌ها را با توجه به مقدار دامنه‌ی متغیرها در یک عدد تصادفی ضرب کرد. در حالت کلی شکل کروموزوم، نحوه‌ی پیاده‌سازی جهش را تعیین می‌کند، ولی همانند نخبه‌گرایی باید در میزان استفاده از جهش نیز دقت شود.<sup>[۳۱،۳۲]</sup> به دست آوردن میزان مؤثر برای جهش و به‌طور کلی برقراری تعادل میان جست‌وجو و بهره‌برداری عملی است که به شکل فضای جست‌وجو بسیار وابسته است و گاهی در فضاهای پیچیده که اطلاعات خاصی از آن در دسترس نیست، کاری ناممکن است. به هر حال، یکی از رایج‌ترین روش‌ها برای این کار، انتخاب تجربی و آزمایش و خطا است. در اینجا، احتمال انجام جهش پنج درصد در نظر گرفته شده است، و برای جهش در هر بار یکی از دو متغیر تصمیم مسئله به صورت تصادفی انتخاب، و در عددی تصادفی بین ۰/۵ و ۱/۵ ضرب می‌شود. در هر دو مورد عملگرهای ترکیب و جهش محدودی متغیرها کنترل می‌شود.

### شرط پایان الگوریتم

عملگرهای فوق به صورت متوالی تکرار می‌شوند و جمعیت در شروع با جواب‌های تصادفی اولیه نسل به نسل به سمت نواحی بهینه حرکت می‌کند. مانند اندازه

جمعیت بیشتر، اثبات می‌شود که با تکرار بیشتر چرخه‌ی الگوریتم، احتمال یافتن جواب بهینه نیز افزایش می‌یابد.<sup>[۳۰]</sup> برای اتمام این روند شروط مختلفی ارائه شده است که به نوعی متأثر از منابع رایانه‌ی - به‌ویژه زمان - هستند؛ از آن جمله می‌توان به همگرایی درصد زیادی از جمعیت به یک جواب خاص، تعداد تکرار مشخص از پیش تعیین شده، مدت زمان مشخص، یافتن یک جواب با ارزش خاص از پیش تعیین شده، یا ترکیبی از شروط فوق اشاره کرد. عموماً در الگوریتم‌های ژنتیک با توجه به گستردگی فضای جست‌وجو و همچنین محدودیت زمانی، تعداد تکرار مشخصی برای خاتمه‌ی الگوریتم استفاده می‌کنند. همچنین برای اطمینان بیشتر از کیفیت جواب‌های (های) الگوریتم می‌توان چندین بار آن را با جمعیت‌های اولیه‌ی مختلف اجرا، و در پایان بهترین جواب‌ها را گزارش کرد. در اینجا، از تعداد تکرار ۵۰۰ استفاده می‌شود. در شبیه‌سازی‌های انجام شده که با زبان برنامه‌نویسی C# پیاده‌سازی و بر روی یک رایانه‌ی خانگی معمولی انجام شده است، الگوریتم قادر است در زمانی در کم‌تر از یک ثانیه به جواب مورد قبولی دست یابد.

### کاربرد واقعی

مدل ارائه‌شده در این پژوهش برای یکی از محصولات یک شرکت تولید فرآورده‌های لبنی به کار گرفته شده است. با توجه به ابهام موجود در اندازه‌گیری هزینه‌های موجودی، این هزینه‌ها به صورت مقادیر بازه‌ی تخمین زده شد. بر این اساس، هزینه‌ی نگه‌داری هر واحد موجودی در واحد زمان ( $C_1$ ) در دامنه‌ی [۰/۶۹۴۰، ۰/۷۵۵]، هزینه‌ی کمبود هر واحد موجودی ( $C_2$ ) در دامنه‌ی [۲/۹۳۲، ۳/۲۲۴]، هزینه‌ی خرید هر واحد موجودی ( $C_3$ ) در دامنه‌ی [۰/۱۳۱، ۰/۱۵۵] و هزینه‌ی سفارش هر بار ( $A$ ) در دامنه‌ی [۲۰، ۲۵۰] تخمین زده می‌شود. همچنین براساس داده‌های تاریخی مربوط به خرابی این محصول، پارامترهای مقیاس  $(\alpha)$ ، شکل  $(\beta)$  و مکان  $(\gamma)$  به ترتیب برابر با ۰/۵، ۱/۲۵ و ۱/۶۸۲ تخمین زده شد. نرخ تقاضای این محصول  $f(t)$  در زمان  $t$  طبق  $f(t) = a + bt$  معین می‌شود که در آن  $a$  و  $b$  بر پایه‌ی اطلاعات موجود مربوط به تقاضای این محصول، به ترتیب برابر با ۱۷/۹۳ و ۲۵/۱۳ تخمین زده شد. فضای اشغال‌شده توسط هر واحد موجودی (با احتساب فضای مورد نیاز برای حمل و جابه‌جایی از جمله راهروها) برابر با ۰/۱۲ متر مکعب و فضای اختصاص‌یافته از کل انبار به این قلم موجودی مجموعاً برابر ۴ متر مکعب است. افق زمانی برنامه‌ریزی ماهیانه است. داده‌های مربوط به این قلم موجودی به‌اختصار در جدول ۱ ارائه شده است. با به‌کارگیری الگوریتم ژنتیک توصیف شده، جواب‌های به دست آمده در جدول ۲ آمده است.

### تحلیل حساسیت مدل

در این بخش به تحلیل حساسیت مدل نسبت به پارامترهای هزینه می‌پردازیم. بنا به تعریف یک عدد بازه‌ی با استفاده از مرکز و شعاع (رابطه‌ی ۲۱) داریم:  $A = \langle a_C, a_W \rangle = \{x : a_C - a_W \leq x \leq a_C + a_W, x \in R\}$  که در آن مرکز و شعاع به ترتیب برابرند با:  $(a_L + a_R)/2$  و  $(a_R - a_L)/2$ . هرچه در تخمین یک پارامتر ابهام بیشتری موجود باشد، شعاع عدد بازه‌ی مربوط به آن بزرگ‌تر خواهد بود و بر عکس. بر این اساس، با معرفی درصد ابهام در پارامترهای هزینه به صورت  $\theta$  عدد بازه‌ی را به صورت:  $A = \langle a_C, a_W \rangle = \{x : a_C - \theta a_W \leq x \leq a_C + \theta a_W, x \in R, \theta \in [0, 1]\}$  تعریف می‌کنیم.

### نتیجه‌گیری

در این نوشتار با استفاده از یک توزیع ویبول سه پارامتری مبتنی بر تحقیق‌های انجام شده<sup>[۱۹]</sup> به توسعه‌ی مدل EOQ برای کالاهای فاسدشدنی با نرخ خرابی متغیر پرداخته شد. در این راستا علاوه بر در نظر گرفتن فضای محدود انبار، برای در نظر گرفتن ابهام موجود در برآورد هزینه‌های نگهداری، سفارش، کمبود و خرید، از اعداد بازه‌ی در مدل‌سازی مسئله و برای حل مدل از یک الگوریتم ژنتیک استفاده شد. این مدل خصوصیات واقعی متعددی از جمله خرابی، کمبود، محدودیت انبار و ابهام در برآورد پارامترها را که تا حدی مربوط به انواع موجودی‌های است، ترکیب می‌کند. از این مدل برای یک محصول لبنی در یک کارخانه‌ی تولید محصولات لبنی استفاده شد. مدل یادشده قابل کاربرد در بسیاری از موقعیت‌های موجودی است که مؤلفه‌ی اصلی محصول فسادپذیری آن باشد. تحلیل حساسیت نتایج، بیان‌گر کارایی بالای مدل در تصمیم‌گیری‌های دنیای واقعی است. براین اساس، استفاده از مقادیر بازه‌ی در مدل‌سازی سایر مدل‌های موجودی، و همچنین استفاده از الگوریتم ژنتیک به‌عنوان یک الگوریتم ابتکاری قوی، در حل مدل‌های پیچیده موجودی پیشنهاد می‌شود.

$$A = \langle 225, 25 \rangle =$$

$$\{x : 225 - 25\theta \leq x \leq 225 + 25\theta, x \in R, \theta \in [0, 1]\}$$

جواب ارائه‌شده در جدول ۲ با استفاده از  $\theta = 1$  به دست آمده است. در جدول ۳ مقادیر جواب، به‌ازای مقادیر مختلف  $\theta$  (برای کلیه‌ی هزینه‌های نگهداری، کمبود، خرید و سفارش) ارائه شده است. نتایج به‌دست آمده در جدول ۳ به‌خوبی نشان می‌دهد که با کاهش مقادیر  $\theta$ ، حدود چپ و راست مقادیر جواب به یکدیگر نزدیک‌تر می‌شوند تا جایی‌که به‌ازای  $\theta = 0$ ، حدود چپ و راست برابر شده و جواب یگانه‌ی به‌دست می‌آید. به هر حال، مقادیر به‌دست آمده به‌ازای  $\theta = 0$ ، یا به‌عبارت دیگر جواب‌های به‌دست آمده در حالت قطعی در دامنه‌ی جواب‌های بازه‌ی قرار می‌گیرند. این امر کارایی استفاده از مقادیر بازه‌ی را نشان می‌دهد. بنابراین در مواقعی که تخمین قطعی پارامترهای موجودی بسیار پیچیده و هزینه‌بر است (شبه آنچه که در مورد مثال واقعی این نوشتار صادق است)، استفاده از مدل یادشده بسیار معقول و مناسب به نظر می‌رسد.

جدول ۱. داده‌های مثال واقعی.

مقدار	پارامترهای تابع ویبول	مقدار	پارامترهای تقاضا و انبار	حد راست (R)	حد چپ (L)	پارامترهای هزینه
۰٫۰۰۵	$\alpha$	۱۷٫۹۳	$a$	۰٫۷۵۵	۰٫۶۹۴	$C_1$
۱٫۲۵	$\beta$	۲۵٫۱۳	$b$	۳٫۲۲۴	۲٫۹۳۲	$C_2$
۱٫۶۸۲	$\lambda$	۰٫۱۲	$\eta$	۰٫۱۵۵	۰٫۱۳۱	$C_3$
		۴٫۰۰	$W$	۲۵۰	۲۰۰	$A$

جدول ۲. نتایج کاربرد مثال واقعی.

	$K^*$	$T^*$	$C(K, T)^*$	$t_1^*$	$I^*$
حد چپ (L)	۰٫۷۴۷۲۰۶	۲٫۵۹۱۰۱۰۸	۱۱۵٫۱۳	۱٫۹۹۳۷۴۱	۸۵٫۷۰۴
حد راست (R)	۰٫۷۶۹۴۶۲	۲٫۶۸۳۶۷۰۶	۱۳۷٫۳۸	۲٫۰۰۵۲۵	۸۶٫۴۹۰

جدول ۳. نتایج تحلیل حساسیت مدل به‌ازای مقادیر مختلف  $\theta$ .

$\theta$		$K^*$	$T^*$	$C(K, T)^*$	$t_1^*$	$I^*$
۰٫۰۰	حد چپ (L)	۰٫۷۷۰۷۸	۲٫۶۰۱۵۵	۱۲۶٫۰۵۹۵۹	۲٫۰۰۵۲۲	۸۶٫۴۸۸
	حد راست (R)	۰٫۷۷۰۷۸	۲٫۶۰۱۵۵	۱۲۶٫۰۵۹۵۹	۲٫۰۰۵۲۲	۸۶٫۴۸۸
۰٫۲۵	حد چپ (L)	۰٫۷۶۱۷۳	۲٫۵۱۸۸۸	۱۲۳٫۴۷۶۶	۱٫۹۷۵۵۵	۸۴٫۴۶۹۲۳
	حد راست (R)	۰٫۷۸۴۳۰	۲٫۶۳۳۴۶	۱۲۸٫۹۷۲	۲٫۶۳۳۴۶	۸۶٫۵۴۰۴
۰٫۵۰	حد چپ (L)	۰٫۷۶۷۵۷	۲٫۵۲۱۴	۱۲۰٫۶۲۶۵	۱٫۹۶۰۲۹	۸۳٫۴۴۰۱۲
	حد راست (R)	۰٫۷۸۲۶۰	۲٫۵۵۳۸۸	۱۳۲٫۲۴۹۹۳	۱٫۹۷۳۲۵	۸۴٫۳۱۳۸۳
۰٫۷۵	حد چپ (L)	۰٫۷۵۴۴۲	۲٫۵۸۲۹	۱۱۸٫۶۸۳۴	۱٫۹۸۹۵	۸۵٫۴۱۵۶
	حد راست (R)	۰٫۷۷۰۲۵	۲٫۶۸۱۸	۱۳۳٫۶۱۶	۲٫۰۲۳۲	۸۷٫۷۲۲۱۱
۱٫۰۰	حد چپ (L)	۰٫۷۴۷۲۰۶	۲٫۵۹۱۰۱۰۸	۱۱۵٫۱۳	۱٫۹۹۳۷۴۱	۸۵٫۷۰۴
	حد راست (R)	۰٫۷۶۹۴۶۲	۲٫۶۸۳۶۷۰۶	۱۳۷٫۳۸	۲٫۰۰۵۲۵	۸۶٫۴۹۰

1. deterioration
2. economic order quantity
3. Ill-behave objective functions
4. constrained tournament selection
5. elitist

## پانوش

## منابع

1. Whitin, T.M. *Theory of Inventory Management*, Princeton University Press, Princeton, NJ (1957).
2. Ghare, P.M., and Schrader, G.P. "A model for exponentially decaying inventory", *Journal of Industrial Engineering*, **14**, pp. 238-243 (1963).
3. Covert, R.P., and Philip, G.C. "An EOQ model for items with Weibull distribution deterioration", *AIIE Transactions*, **5**, pp. 323-326 (1973).
4. Weiss, H. J. "Economic order quantity models with non-linear holding cost", *European Journal of Operational Research*, **9**, pp. 56-60 (1982).
5. Shah, Y.K., and Jaiswal, M.C. "An order-level inventory model for a system with constant rate of deterioration", *Operations Research*, **14**, pp. 174-184 (1977).
6. Mishra, R.B. "Optimum production lot-size model for a system with deteriorating inventory", *International Journal of Production Research*, **13**, pp. 495-505 (1975).
7. Bahari-Kashani, H. "Replenishment schedule for deteriorating items with time proportional demand", *Journal of Operational Research Society*, **40**, pp. 75-81 (1989).
8. Goswami, A., and Chaudhuri, K.S. "Variations of order-level inventory models for deteriorating items", *International Journal of Production Economics*, **27**, pp. 111-117 (1992).
9. Dave, U. "An order-level inventory model for deteriorating items with variable instantaneous demand and discrete opportunities for replenishment", *Operations Research*, **23**, pp. 244-249 (1986).
10. Dave, U., and Patel, L.K. "(T, Si) policy inventory model for deteriorating items with time proportional demand", *Journal of Operational Research Society*, **32**, pp. 137-142, (1981).
11. Dave, U. "On a discrete-in-time order-level inventory model for deteriorating items", *Operations Research*, **Q-30**, pp. 349-354 (1979).
12. Heng, K.J.; Labban J., and Linn, R.L. "An order-level lot-size inventory model for deteriorating items with finite replenishment rate", *Computers and Industrial Engineering*, **20**, pp. 187-197 (1991).
13. Elsayed, E.A., and Teresi, C. "Analysis of inventory systems with deteriorating items", *International Journal of Production Research*, **21**, pp. 449-460 (1983).
14. Kang, S., and Kim, I. "A study on the price and production level of the deteriorating inventory system", *International Journal of Production Research*, **21**, pp. 449-460 (1983).
15. Mak, K.L. "A production lot size inventory model for deteriorating items", *Computers and Industrial Engineering*, **6**, pp. 309-317 (1982).
16. Raffat, F.; Wolf, P.M., and Eldin, H.K. "An inventory model for deteriorating items", *Computers and Industrial Engineering*, **20**, pp. 89-94 (1991).
17. Aggarwal, V., and Bahari-Kashani, H. "Synchronized production policies for deteriorating items in a declining market", *IIE Transactions*, **23**, pp. 185-197 (1991).
18. Wee, H.M. "A deterministic lot-size inventory model for deteriorating items with shortages on a declining market", *Computers & Operations Research*, **22**, pp. 345-356 (1995).
19. Chakrabarty, T.; Giri, B.C., and Chadhuri, K.S. "An EOQ model for items with Weibull distribution deterioration, shortage and trends demand: an extension of Philip's model", *Computers and Operations Research*, **25**(7-8), pp. 649-657 (1998).
۲۰. کالیانمودی، دب، الگوریتم‌های ژنتیک با رویکرد بهینه‌یابی چندهدفه، مترجمان: جعفر رضایی و منصور داودی منفرد، چاپ اول، تهران، انتشارات پلک، (۱۳۸۷).
21. Philip, G.C. "A generalized EOQ model for items with Weibull distribution deterioration", *AIIE Transactions*, **6**, pp. 159-162 (1974).
22. Gupta, R.K.; Bhunia, A.K., and Goyal, S.K. "An application of genetic algorithm in a marketing oriented inventory model with interval valued inventory costs and three-component demand rate dependent on displayed stock level", *Applied Mathematics and Computation*, **192**(2), pp. 466-478 (2007).
23. Goldberg, D.E. *Genetic Algorithm for Search, Optimization, and Machine Learning*, Reading, MA: Adding-Wesley, (1989).
24. Rudolph, G. "Convergence analysis of canonical genetic algorithms", *IEEE Transactions on Neural Network*, **5**(1), pp. 96-101 (1994).
25. Vose, M.D. *Simple Genetic Algorithm: Foundation and Theory*. Ann Arbor, MI: MIT Press, (1999).
26. Augusto, O.B.; Rabeau, S.; Depince, Ph., and Bennis, F. "Multi-objective genetic algorithms: A way to improve the convergence rate", *Engineering Applications of Artificial Intelligence*, **19**, pp. 501-510 (2006).
27. Deb, K., and Kumar, A. "Real-coded genetic algorithm with simulated binary crossover: Studies on multi-modal and multi-objective problems", *Complex Systems*, **9**(6), pp. 431-454 (1995).
28. Deb, K., and Goel, T. "Controlled elitist non-dominated sorting genetic algorithms for better convergence", *In Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, EMO, pp. 385-399 (2001).
29. Rodolph, G. "Convergence of evolutionary algorithms in general search space", *In Proceedings of the Third IEEE Conference on Evolutionary Computation*, pp. 50-54 (1996).