

مقاله پژوهشی

ارائه روش حل بازگشتی جدید برای استراتژی مختلط در مسائل بهینه‌سازی قابلیت اطمینان

مصطفی ابویی اردکان* و سجاد تلخابی

گروه مهندسی صنایع، دانشکده فنی و مهندسی، دانشگاه خوارزمی، تهران

(دریافت مقاله: ۱۳۹۸/۰۳/۰۴ - دریافت نسخه نهایی: ۱۳۹۸/۰۹/۱۹)

چکیده - یکی از روش‌های رایج برای بهبود قابلیت اطمینان، استفاده از قطعات مازاد موازی در زیرسیستم‌ها است. این مسئله که تحت عنوان مسئله تخصیص افزونگی شناخته می‌شود، شامل انتخاب اجزای مازاد جهت حداکثر کردن قابلیت اطمینان سیستم با توجه به محدودیت‌های از پیش تعیین شده‌ای نظیر هزینه، وزن و حجم سیستم است. به‌طور سنتی دو استراتژی قطعات مازاد فعال و ذخیره برای حل این‌گونه مسائل وجود دارد. به‌تازگی استراتژی جدیدی تحت عنوان استراتژی مختلط ارائه شده است که عملکرد بهتری نسبت به هر دو استراتژی قبلی داشته است. مشکلی اصلی در استفاده از استراتژی مختلط، پیچیدگی معادلات مربوط به آن است. از این‌رو، در این مقاله یک روش جدید برای محاسبه قابلیت اطمینان استراتژی مختلط ارائه می‌شود که نیازی به محاسبات پیچیده انتگرالی روش قبل را ندارد و همچنین باعث کاهش زمان حل مسائل می‌شود. روش حل پیشنهادی به کمک روابط بازگشتی، یک حالت کلی از فرمول استراتژی مختلط ارائه می‌دهد که با افزایش یا کاهش قطعات هر زیرسیستم، فرمول محاسباتی تغییر چندانی نمی‌کند. این انعطاف‌پذیری و ثبات در فرمول، در سیستم‌هایی با تعداد قطعات بالا، بسیار کارآمد خواهد بود. به‌منظور ارزیابی روش حل جدید و مقایسه آن با روش پیشین، مسئله الگوی سری- موازی با چهارده زیرسیستم در نظر گرفته شده و نتایج دو روش با یکدیگر مقایسه شده است.

واژه‌های کلیدی: بهینه‌سازی قابلیت اطمینان، تخصیص افزونگی، استراتژی مختلط، توابع بازگشتی.

A New Recursive Formulation for the Mixed Redundancy Strategy in Reliability Optimization Problems

M. Abouei Ardakan* and S. Talkhabi

Department of Industrial Engineering, Faculty of Engineering, Kharazmi University, Tehran, Iran.

Abstract: One of the common approaches for improving the reliability of a specific system is to use parallel redundant components in subsystems. This approach, which is known as the redundancy allocation problem (RAP), includes the simultaneous selection of the component type and its level for each subsystem in order to maximize the system reliability.

*: مسئول مکاتبات، پست الکترونیکی: mabouei2001@gmail.com

Traditionally, there are two redundancy strategies, namely active and standby, for the redundant components. Recently, a new powerful strategy called mixed strategy has been developed. It has been proved that the mixed strategy has a better performance when compared to both previous strategies. The main issue in utilizing the mixed strategy is its complicated formulation and sophisticated calculations, leading to a time-consuming procedure for solving the problems. Hence, in this paper, a new formulation based on the recursive approach is introduced to ease the calculation of the mixed strategy. In the new formulation, the complex double integral calculations are removed and the calculation times is reduced. The proposed recursive formulation provides a general statement for the mixed strategy formula which is not changed by altering the number of components in each subsystem. This flexibility and stability in the formula can be very important, especially for large scale cases. In order to evaluate the new approach and to compare its performances with the previous formulation, a benchmark problem with 14 subsystems is considered and the results of the two formulation are compared with each other.

Keywords: Reliability optimization, Redundancy allocation problem, Mixed strategy, Recursive functions.

فهرست علائم

n_{si} مجموعه	n_s	مجموعه‌ای از تمام زیرسیستم‌ها که از استراتژی فعال استفاده می‌کنند	A
حد بالا برای n_i	$n_{max,i}$	حد بالای محدودیت‌های سیستم برای وزن و هزینه	C, W
قابلیت اطمینان سیستم در زمان t و برای بردارهای طراحی Z, n_A و n_s	$R(t, Z, n_A, n_s)$	هزینه سوئیچ استفاده شده در زیرسیستم i ام	$C_{switch,i}$
قابلیت اطمینان زیرسیستمی که از استراتژی مختلط با n_{Ai} قطعه فعال و n_{Si} قطعه مازاد ذخیره استفاده می‌کند	$R_s^{n_{Ai}, n_{Si}}(t)$	تابع چگالی احتمال برای Z -امین رخداد خرابی در زیرسیستم i ام وقتی از قطعه نوع Z_i استفاده شده است	$f_{Z_i}^{(j)}$
قابلیت اطمینان قطعه نوع Z_i در زیرسیستم i ام	$r_{i,Z_i}(t)$	تابع توزیع تجمعی $f_{Z_i}^{(j)}$	$F_{Z_i}^{(j)}$
تعداد زیرسیستم‌ها	S	تابع چگالی احتمال برای رخداد خرابی در آخرین قطعه فعال در زیرسیستم i ام وقتی از قطعه نوع Z_i استفاده شود	$f_{Z_i}^{Max, n_{Ai}}$
مجموعه‌ای از تمام زیرسیستم‌ها که از استراتژی ذخیره استفاده می‌کنند	S	تابع چگالی احتمال خرابی n_{Ai} قطعه فعال و Z امین قطعه مازاد ذخیره در زیرسیستم i ام وقتی از قطعه نوع Z_i استفاده شود	$f_{Z_i}^{n_{Ai}, Z, \dots, Z}$
زمان مأموریت	T	مجموعه‌ای از تمام زیرسیستم‌ها که از استراتژی مختلط استفاده می‌کنند	M
$c_{i,Z}$ وزن مربوط به Z امین انتخاب موجود برای قطعه مورد استفاده در زیرسیستم i ام	$w_{i,Z}$	تعداد نوع قطعات موجود برای انتخاب کردن در زیرسیستم i ام	m_i
وزن سوئیچ استفاده شده در زیرسیستم i ام	$w_{switch,i}$	مجموعه‌ای از تمام زیرسیستم‌ها که از هیچ استراتژی استفاده نمی‌کنند (یک قطعه استفاده شده است)	N
اندیس نشان‌دهنده نوع قطعه استفاده شده در زیرسیستم i ام	Z_i	تعداد قطعات استفاده شده در زیرسیستم i ام	n_i

n_{Ai}	تعداد قطعات فعال در زیرسیستم i ام	Z	مجموعه Z_i ها
n_{si}	تعداد قطعات ذخیره-سرد در زیرسیستم i ام	$\rho_i(t)$	قابلیت اطمینان سیستم سوئیچینگ در زمان t برای حالت پیوسته
n_A	مجموعه n_{Ai}		

۱- مقدمه

امروزه با پیشرفت فناوری و پیچیده‌تر شدن سیستم‌ها، مفهوم قابلیت اطمینان و بهینه‌سازی آن به‌طور گسترده‌ای توسط محققان بررسی شده است. کاربرد این مفهوم در جهان واقعی را می‌توان در بسیاری از سیستم‌های صنعتی همچون سیستم‌های الکتریکی، سیستم‌های ارتباطات، کاوشگرهای فضایی و سیستم‌های ماهواره‌ای یافت [۱]. به‌طور کلی، چهار روش برای بهبود قابلیت اطمینان یک سیستم وجود دارد که عبارتند از الف) افزایش قابلیت اطمینان قطعات، ب) استفاده از قطعات مازاد موازی در زیرسیستم‌ها، ج) ترکیب افزایش قابلیت اطمینان قطعات و استفاده از قطعات مازاد و د) باز طراحی ساختار سیستم [۲]. روش دوم که به مسئله تخصیص اجزای مازاد^۱ (RAP) معروف است، به‌علت داشتن حوزه وسیع و کاربردی‌تر، بیش از سایر موارد استفاده شده است. همچنین ثابت شده است که مسائل RAP در دسته مسائل NP-Hard قرار دارند [۳]؛ بنابراین حل این مسائل به‌خصوص در ابعاد بزرگ و یا با اضافه شدن فرض‌های جدید و پیچیدگی بیشتر مسئله دشوار خواهد بود.

در مسائل RAP، بهبود قابلیت اطمینان سیستم از طریق اضافه کردن قطعات موازی به زیرسیستم‌های یک سیستم اصلی رخ می‌دهد. در این مسئله، نحوه استفاده یا ورود قطعات مازاد به زیرسیستم، در میزان قابلیت اطمینان زیرسیستم و بالطبع آن در قابلیت اطمینان کل سیستم اثرگذار است. نحوه استفاده از قطعات مازاد را استراتژی قطعات مازاد^۲ می‌نامند [۴]. به‌طور سنتی دو استراتژی کلی برای قطعات مازاد وجود دارد که به‌ترتیب عبارتند از استراتژی‌های

قطعات فعال^۳ و قطعات ذخیره^۴. پژوهشگران زیادی به بررسی و مقایسه عملکرد این استراتژی‌ها در شرایط و ساختارهای مختلف پرداخته‌اند. به‌تازگی در سال ۲۰۱۴، یک استراتژی جدید تحت عنوان استراتژی مختلط^۵ ارائه شده است که کارایی بسیار بالایی در بهبود قابلیت اطمینان داشته است [۵]. هر دو استراتژی فعال و ذخیره حالت‌های خاصی از استراتژی مختلط به‌شمار می‌آیند. به عبارت دیگر، استراتژی مختلط می‌تواند در مواقع لزوم به استراتژی‌های فعال و ذخیره نیز تبدیل شود؛ بنابراین استفاده از این استراتژی نسبت به دو استراتژی دیگر در اولویت قرار دارد.

مسائل بهینه‌سازی قابلیت اطمینان، بیشتر با در نظر گرفتن استراتژی فعال مدل‌سازی شده است. برای حل این دسته از مسائل روش‌های مختلفی همچون، روش‌های دقیق از جمله برنامه‌ریزی پویا [۶ و ۷]، برنامه‌ریزی عدد صحیح [۸]، ضرایب لاگرانژ [۹] و انواع متفاوتی از الگوریتم‌های فراابتکاری همانند الگوریتم ژنتیک [۱۰ و ۱۱]، کلونی مورچه‌ها [۱۲]، جستجوی ممنوعه [۱۳]، الگوریتم انبوه ذرات [۱۴] و شبیه‌سازی تبرید [۱۵] به‌کار برده شده است.

استراتژی ذخیره یا آماده به‌کار، خود دارای سه حالت متفاوت با نام‌های ذخیره-سرد، ذخیره-گرم و ذخیره-داغ است. در حالت ذخیره سرد، قطعات از تنش‌های ناشی از عملکرد سیستم در امان هستند و در هنگام عدم کارکرد سیستم، آسیبی به آن وارد نمی‌شود. در حالت ذخیره گرم، امکان تأثیرگذاری این تنش‌ها روی قطعات ذخیره، وجود دارد و در حالت ذخیره داغ، خرابی قطعه ارتباطی به کارکرد یا عدم کارکرد قطعه ندارد و امکان خرابی قطعات در هر

لحظه با نرخ یکسان وجود دارد [۱۶]. در این مقاله هر جا از قطعات ذخیره نام برده می‌شود منظور حالت ذخیره-سرد است. در استراتژی ذخیره یک قطعه (قطعه اصلی) به صورت فعال کار می‌کند و سایر قطعات به صورت ذخیره در سیستم قرار می‌گیرند. قطعات ذخیره به هنگام خرابی قطعه فعال بر اساس یک اولویت از قبل تعیین شده و به ترتیب وارد سیستم می‌شوند. در این دسته از استراتژی‌ها، به هنگام خرابی یک قطعه فعال، یک سوئیچ الکترونیکی یا مکانیکی، قطعه ذخیره را وارد مدار می‌کند. برای استراتژی ذخیره در مقایسه با استراتژی فعال مطالعات کمتری صورت گرفته است. رابینسون و همکاران [۱۷] روی این استراتژی با در نظر گرفتن قطعات غیرقابل تعمیر مطالعه‌ای را انجام داده‌اند. برای مسئله سری-موازی، کویت مسئله RAP را با در نظر گرفتن استراتژی ذخیره سرد برای کل سیستم و استفاده از روش برنامه‌ریزی عدد صحیح مطالعه کرده است. همچنین ابویی و همدانی [۱۸] با در نظر گرفتن این استراتژی برای حل مسئله تخصیص افزونگی-قابلیت اطمینان (RRAP) که نوع پیچیده‌تری از مسئله RAP است، از الگوریتم ژنتیک استفاده کرده‌اند.

در اکثر طراحی‌ها، به دلیل وجود توابع هدف چندگانه متضاد، محاسبه قابلیت اطمینان دشوار است. ساکاو و همکاران [۱۹] مدل‌سازی تخصیص قابلیت اطمینان چندهدفه با هدف بیشینه‌سازی قابلیت اطمینان و کمینه‌سازی هزینه سیستم را بررسی و برای حل آن روش موازنه ارزش جانیشینی را پیشنهاد داده‌اند. در تحقیقات دیگر از توابع چندهدفه برای بهینه کردن قابلیت اطمینان و طول عمر [۲۰] و قابلیت اطمینان و هزینه سیستم [۲۱] نیز استفاده شده است.

در جدیدترین مطالعات، فیض‌آبادی و جهرمی [۲۲] استراتژی مختلط را برای مسئله RAP با اجزای غیرهمگن توسعه دادند. به بیان واضح‌تر، آنها حالتی را در نظر گرفتند که در آن اجزای آماده به کار متفاوت از اجزای فعال باشند اما همه اجزای فعال و آماده به کار باید یکسان باشند. کیم [۲۳]،

مسئله تخصیص افزونگی با اجزای مختلط را با در نظر گرفتن سوئیچ غیر ایده‌آل طراحی کرد. او نشان داد که توالی قطعات در سیستم آماده به کار، بر قابلیت اطمینان سیستم تأثیرگذار است. کیم با معرفی مفهوم توالی بهینه برای زیرسیستم‌های آماده به کار، توانست قابلیت اطمینان سیستم را بهبود دهد. در سال ۲۰۱۸، ابویی و رضوان [۲۴]، مسئله RRAP را با فرض استفاده از قطعات ذخیره سرد و با در نظر گرفتن دو هدف بهینه‌سازی قابلیت اطمینان و کاهش هزینه‌های سیستم بررسی کردند. در این مطالعه، برای تعیین میزان دقیق قابلیت اطمینان استراتژی ذخیره از مدل مارکوف جدیدی استفاده شده است. هانگ و همکاران [۲۵] سعی کردند تا تئوری اثر بقا را در مدل بهینه‌سازی قابلیت اطمینان-افزونگی پیاده‌سازی کنند. در این پژوهش اطلاعات مربوط به ساختار یک سیستم را با اثر بقا بررسی کردند و از یک تابع جریمه تطبیقی جدید، برای انتقال مسئله بهینه‌سازی با محدودیت به حالت بدون محدود استفاده کردند. آنها به کمک الگوریتم اکتشافی جستجوی فرکتال تصادفی مسئله بهینه‌سازی بدون محدودیت را حل کردند.

همان‌طور که اشاره شد، استراتژی مختلط اولین بار در سال ۲۰۱۴ توسط ابویی و همدانی [۵] برای بهبود مسئله تخصیص افزونگی پیشنهاد شده است. این استراتژی شامل استفاده هم‌زمان از قطعات فعال و ذخیره در یک زیرسیستم است. در این استراتژی تعداد قطعات فعال و ذخیره برای هر زیرسیستم یک متغیر تصمیم در نظر گرفته می‌شود به طوری که پس از خرابی تمامی قطعات فعال، قطعات ذخیره یک به یک توسط یک سوئیچ وارد سیستم می‌شوند. این استراتژی ترکیبی از استراتژی‌های فعال و ذخیره است و نشان داده شده است که نسبت به استراتژی‌های قبلی خود (فعال و ذخیره) به مقادیر قابلیت اطمینان بالاتری برای سیستم نتیجه می‌شود. برای نشان دادن میزان برتری این استراتژی نسبت به استراتژی‌های قبلی، این استراتژی روی یک مسئله الگوی معروف در حالت دوهدفه بررسی شده است [۲۶]. نتایج این

یک بخش کوچک، می توان به فرمول مدنظر رسید. این فرمول بازگشتی علاوه بر کاهش پیچیدگی و زمان حل مسئله، از انعطاف پذیری بالایی برخوردار است. از آنجا که با افزایش تعداد قطعات ذخیره، قابلیت اطمینان زیرسیستم‌ها افزایش می یابد، با استفاده از انعطاف پذیری فرمول بازگشتی می توانیم برای هر تعداد قطعات ذخیره، در زیرسیستم فرمول بازگشتی مناسب را با اندکی تغییر به دست آورد.

۱-۱- مدل ریاضی مسئله تخصیص افزونگی

به منظور حل مسئله تک هدفه تخصیص افزونگی، لازم است مدل ریاضی این مسئله ارزیابی شود. برای این منظور لازم است که ابتدا متغیرها و پارامترهای این مسئله و سپس مدل مورد نظر معرفی شود. در این مسئله فرض شده است که در هر زیرسیستم تعداد m_i نوع قطعه مختلف که از نظر کارکردی یکسان هستند وجود دارد. همچنین در هر زیرسیستم فقط یکی از این نوع قطعات را می توان استفاده کرد. هر کدام از این نوع قطعات دارای هزینه، وزن، قابلیت اطمینان و سایر ویژگی های مختص به خود هستند و از هر کدام از این قطعات به تعداد نامحدودی در دسترس است. علاوه بر این، برای تعداد قطعاتی که در هر زیرسیستم استفاده شده نیز محدودیتی در نظر گرفته می شود. هدف از حل این مسئله، تعیین بهترین نوع قطعه، تعداد قطعات مازاد فعال و تعداد قطعات مازاد ذخیره برای هر زیرسیستم به منظور حداکثر کردن قابلیت اطمینان کل سیستم است. پس از تعیین تعداد قطعات مازاد فعال و ذخیره، نوع استراتژی بهینه برای هر زیرسیستم نیز قابل تعیین است. در نهایت، فرض شده است که برای نظارت بر خرابی قطعات، یک سیستم سوئیچینگ غیر ایده آل وجود دارد که این سیستم عملکرد ساختار طراحی شده را به طور پیوسته تحت کنترل دارد.

از آنجا که معروف ترین مسئله مورد بررسی در این حوزه یک سیستم سری- موازی است، در این بخش این سیستم بررسی شده است. مدل ریاضی و نمادهای به کار گرفته شده

پژوهش نشان می دهد که استفاده از استراتژی مختلط منجر به ۱۴ ساختار جدید برای سیستم سری- موازی می شود که در تمامی آنها میزان قابلیت اطمینان از بهترین جواب های مطالعات قبلی با استراتژی فعال و یا ذخیره بهتر است. علاوه بر این، عملکرد استراتژی مختلط در مسائل RRAP نیز تست شده است و نتایج نشان دهنده برتری کامل استراتژی مختلط در تمامی مسائل بنچمارک نسبت به استراتژی های قبلی است [۲۷].

به تازگی استراتژی جدیدی به نام «استراتژی K-Mixed» معرفی شده است که شکل کلی تری از استراتژی مختلط است. تفاوت این استراتژی نسبت به استراتژی مختلط در زمان روش کردن قطعات ذخیره است. این استراتژی با فرض سیستم سوئیچ گسسته مدل سازی شده و روی مسئله الگوی سری- موازی پیاده سازی شده است. نتایج نشان می دهد که در بعضی از حالت ها، استراتژی جدید عملکرد بهتری نسبت به استراتژی مختلط دارد ولی هیچ کدام از این دو استراتژی عملکرد مطلقاً بهتری نیست به دیگری ندارد [۲۸].

با همه این تفاسیر و برتری های استراتژی مختلط نسبت به استراتژی های سنتی فعال و ذخیره، این استراتژی از فرمول های پیچیده و زمان بر برای محاسبه قابلیت اطمینان استفاده می کند که حل آن را دشوار کرده است. این پیچیدگی، به دلیل استفاده زیاد از انتگرال های چندگانه در فرمول محاسباتی استراتژی مختلط است. با افزایش تعداد قطعات در زیرسیستم ها، پیچیدگی این فرمول ها چندین برابر شده و بالطبع زمان حل به صورت نمایی افزایش می یابد. در این مقاله، برای رفع این مشکل، برای نخستین بار، یک روش محاسباتی جدید مبتنی بر روابط بازگشتی ارائه شده است که پیچیدگی و زمان حل را تا حدود زیادی بهبود می بخشد. این فرمول جدید، فقط از انتگرال یگانه برای محاسبه قابلیت اطمینان استفاده می کند. از آنجا که مسئله با رویکرد بازگشتی بررسی می شود با افزایش تعداد قطعات در زیرسیستم ها، فرمول پیشنهادی تغییر چندانی نداشته و تنها با اضافه کردن

$$R(t; Z, n_A, n_S) = \prod_{i \in A} \left(1 - (1 - F_{iZ_i}(t))^{n_i} \right) \times \prod_{i \in S} \left(F_{iZ_i}(t) + \sum_{j=1}^{n_i-1} \int_0^t \rho_i(u) F_{iZ_i}(t-u) f_{iZ_i}^{(j)}(u) du \right) \times \prod_{i \in N} F_{iZ_i}(t) \times \prod_{i \in M} R_{iZ_i, Mix}(t) \quad (5)$$

که در این رابطه داریم؛

$$R_{iZ_i, Mix}(t) = \left(1 - (1 - F_{iZ_i}(t))^{n_{Ai}} \right) + \int_0^t \rho_i(u) F_{iZ_i}(t-u) \times f_{iZ_i}^{Max, n_{Ai}}(u) du + \sum_{j=1}^{n_{sz_i}-1} \int_0^t \int_{t_1}^t \rho_i(u) F_{iZ_i}(t-u) \times f_{iZ_i}^{(j)}(u-t_1) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) du dt_1 \quad (6)$$

$$f_{iZ_i}^{Max, n}(t) = n [F_{iZ_i}(t)]^{n-1} f_{iZ_i}(t) \quad (7)$$

قسمت اول رابطه (5) قابلیت اطمینان را در زیرسیستم‌هایی که از استراتژی فعال استفاده می‌کنند، محاسبه می‌کند. قسمت دوم، قابلیت اطمینان زیرسیستم‌هایی با استراتژی ذخیره-سرد را محاسبه و قسمت سوم برای زیرسیستم‌هایی است که هیچ استراتژی یا به عبارتی هیچ قطعه مازادی ندارند. قسمت چهارم نیز مربوط به محاسبه قابلیت اطمینان برای زیرسیستم‌هایی است که از استراتژی مختلط استفاده می‌کنند. از آنجا که سیستم سری- موازی است تمامی مقادیر به دست آمده برای قابلیت اطمینان زیرسیستم‌ها در یکدیگر ضرب می‌شوند. در رابطه (7)، تابع چگالی احتمال برای رخداد خرابی در آخرین (n_A امین) قطعه فعال محاسبه شده است [5].

۱-۲-۱-۱- روش حل بازگشتی

همان‌طور که اشاره شد فرمول محاسباتی استراتژی مختلط، دارای پیچیدگی زیاد و زمان حل طولانی است. در این مقاله با استفاده از روابط بازگشتی، فرمول استراتژی مختلط بازنویسی شده است تا به یک رابطه ساده‌تر برای این استراتژی دست یابیم. برای این منظور، همان‌طور که در فرمول مربوط به استراتژی مختلط (رابطه 6) مشاهده می‌شود با افزایش تعداد

در این مدل، برای یک ساختار سری- موازی با S زیرسیستم و با لحاظ کردن دو محدودیت وزن و هزینه و با در نظر گرفتن استراتژی مختلط برای قطعات در ادامه مقاله ارائه شده است.

۱-۲-۱-۱- مدل ریاضی

مدل ریاضی مسئله تخصیص افزونگی به صورت زیر است.

$$\text{Max } R(t; Z, n_A, n_S) \quad (1)$$

s.t:

$$\sum_{i=1}^s c_{i,j} n_i + \sum_{i \in \{S, M\}} c_{\text{switch}, i} \leq C, n_i \in \{1, 2, \dots, n_{\text{max}, i}\} \quad (2)$$

$$\sum_{i=1}^s w_{i,j} n_i + \sum_{i \in \{S, M\}} w_{\text{switch}, i} \leq W, j \in \{1, 2, \dots, m_j\} \quad (3)$$

$$n_i = n_{Ai} + n_{Si} \quad \forall i = 1, 2, \dots, s. \quad (4)$$

در رابطه (1)، تابع هدف شامل نوع قطعه، تعداد قطعات فعال و ذخیره سرد در هر زیرسیستم است که با هدف حداکثر کردن میزان قابلیت اطمینان سیستم، تعریف شده است. نوع استراتژی بهینه برای هر زیرسیستم، پس از تعیین تعداد قطعات مازاد فعال و ذخیره، تعیین می‌شود. محدودیت‌های ارائه شده در روابط (2) و (3) به ترتیب محدودیت‌های مربوط به هزینه و وزن هستند.

نکته اساسی، نحوه محاسبه تابع هدف $R(t; Z, n_A, n_S)$ است. مسئله طراحی شده باید قادر باشد هر سه نوع استراتژی موجود را مدنظر قرار دهد. به عبارت دیگر، هر زیرسیستم از سیستم اصلی ممکن است یکی از این سه استراتژی را داشته باشد و معادله ریاضی باید بتواند قابلیت اطمینان کل سیستم که متشکل از این زیرسیستم‌ها است را محاسبه کند. برای این منظور، معادله ارائه شده در رابطه (5) برای محاسبه قابلیت اطمینان کل سیستم برای حالت سیستم سوئیچینگ پیوسته ارائه شده است.

بازگشتی این است که برای محاسبه قابلیت اطمینان زیرسیستم در هر مرحله از روابط مرحله قبل استفاده می‌کنیم. به‌عنوان مثال در رابطه (۱۱) برای محاسبه قابلیت اطمینان زیرسیستم با n_{Ai} قطعه فعال و دو قطعه ذخیره، از رابطه (۹) استفاده شده است. این روش بازگشتی جدید از محاسبات تکراری و غیرضروری پرهیز کرده و پیچیدگی روابط را کاهش می‌دهد. در ادامه با افزایش تعداد قطعات مازاد ذخیره در زیرسیستم (n_{Si}) تابع چگالی احتمال خرابی n_{Ai} قطعه فعال و n_{Si} قطعه مازاد ذخیره به‌کمک روابط بازگشتی (۱۲) محاسبه می‌شود و در نهایت رابطه (۱۳) نحوه محاسبه بازگشتی قابلیت اطمینان زیرسیستم‌های که از استراتژی مختلط با n_{Ai} قطعه فعال و n_{Si} قطعه ذخیره استفاده می‌کنند را نشان می‌دهد. همان‌طور که در نحوه محاسبه قابلیت اطمینان با روابط بازگشتی گفته شد با اضافه شدن هر قطعه ذخیره جدید، فقط یک انتگرال یگانه به معادله قبلی اضافه می‌شود. این در حالتی است که در فرمول پیشین، با اضافه شدن هر قطعه ذخیره جدید، یک انتگرال دوگانه به فرمول اضافه می‌شود که با اضافه شدن انتگرال دوگانه، پیچیدگی و زمان حل مسئله به‌طور چشمگیری افزایش می‌یابد. توضیحات بیشتری از روش بازگشتی پیشنهادی در پیوست مقاله ارائه شده است.

در ادامه برای مقایسه فرمول پیشین و فرمول بازگشتی پیشنهادی، یک مسئله الگو که یک ساختار سری- موازی با چهارده زیرسیستم است بررسی شده است و با دو روش مدنظر حل شده و زمان حل و جواب‌های با یکدیگر مقایسه شده‌اند. داده‌های مسئله الگو در جدول (۱) نشان داده شده است.

۱-۲-۲-۱-۲- محاسبات عددی

برای حل مسئله الگو مورد نظر در این بخش، از الگوریتم ژنتیک استفاده شده است. ساختار کروموزوم طراحی شده همانند شکل (۱)، یک ساختار $3 \times S$ است که سطر اول

قطعات زیرسیستم‌ها، تعداد جملات آن افزایش یافته و انتگرال‌های دوگانه زیادی در فرمول ظاهر می‌شوند که پیچیدگی و زمان حل فرمول را به‌طور چشمگیری افزایش می‌دهد. در ادامه با در نظر گرفتن پارامترهای بالا، فرمول بازگشتی پیشنهادی به‌صورت زیر معرفی می‌شود.

$$R_S^{n_{Ai}}(t) = 1 - (1 - r_{iZ_i}(t))^{n_{Ai}} \quad (8)$$

$$R_S^{n_{Ai},1}(t) = R_S^{n_{Ai}}(t) + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{Max, n_{Ai}}(u) du \quad (9)$$

رابطه (۸) قابلیت اطمینان یک زیرسیستم با n_{Ai} قطعه فعال را نشان می‌دهد. رابطه (۹) قابلیت اطمینان زیرسیستم‌هایی که از استراتژی مختلط با n_{Ai} قطعه فعال و یک قطعه ذخیره استفاده می‌کنند را نشان می‌دهد که قسمت اول معادله همان قابلیت اطمینان n_{Ai} قطعه فعال و در قسمت دوم احتمال خرابی آخرین قطعه از n_{Ai} قطعه فعال $(f_{iZ_i}^{Max, n_{Ai}})$ و کارکرد صحیح قطعه ذخیره تا پایان زمان T است. نحوه محاسبه تابع چگالی احتمال خرابی برای حالتی که n_{Ai} قطعه فعال و یک قطعه مازاد ذخیره در زیرسیستم وجود دارد در رابطه (۱۰) ارائه شده است.

$$f_{iZ_i}^{n_{Ai},1}(t) = \int_0^t f_{iZ_i}^{(1)}(t-u) \times f_{iZ_i}^{Max, n_{Ai}}(u) du \quad (10)$$

$$R_S^{n_{Ai},1,2}(t) = R_S^{n_{Ai},1}(t) + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{n_{Ai},1}(u) du \quad (11)$$

$$f_{iZ_i}^{n_{Ai},1,2,\dots,n_{Si}}(t) = \int_0^t f_{iZ_i}^{(j)}(t-u) \times f_{iZ_i}^{n_{Ai},1,2,\dots,(n_{Si}-1)}(u) du \quad (12)$$

$$R_S^{n_{Ai},1,2,\dots,n_{Si}}(t) = R_S^{n_{Ai},1,2,\dots,(n_{Si}-1)}(t) + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{n_{Ai},1,2,\dots,(n_{Si}-1)}(u) du \quad (13)$$

رابطه (۱۱) قابلیت اطمینان زیرسیستم‌های که از استراتژی مختلط با n_{Ai} قطعه فعال و دو قطعه ذخیره استفاده می‌کنند را به‌صورت بازگشتی محاسبه می‌کند. منظور از رابطه

جدول ۱- اطلاعات قطعات برای سیستم سری- موازی

انتخاب ۴ (j=۴)			انتخاب ۳ (j=۳)			انتخاب ۲ (j=۲)			انتخاب ۱ (j=۱)			ردیف
R _{ij}	c _{ij}	w _{ij}	R _{ij}	c _{ij}	w _{ij}	R _{ij}	c _{ij}	w _{ij}	R _{ij}	c _{ij}	w _{ij}	
۰/۹۵	۲	۵	۰/۹۱	۲	۲	۰/۹۳	۱	۴	۰/۹۰	۱	۳	۱
-	-	-	۰/۹۳	۱	۹	۰/۹۴	۱	۱۰	۰/۹۵	۲	۸	۲
۰/۹۲	۴	۴	۰/۸۷	۱	۶	۰/۹۰	۳	۵	۰/۸۵	۲	۷	۳
-	-	-	۰/۸۵	۵	۴	۰/۸۷	۴	۶	۰/۸۳	۳	۵	۴
-	-	-	۰/۹۵	۳	۵	۰/۹۳	۲	۳	۰/۹۴	۲	۴	۵
۰/۹۶	۲	۴	۰/۹۷	۲	۵	۰/۹۸	۳	۴	۰/۹۹	۳	۵	۶
-	-	-	۰/۹۴	۵	۹	۰/۹۲	۴	۸	۰/۹۱	۴	۷	۷
-	-	-	۰/۹۱	۶	۶	۰/۹۰	۵	۷	۰/۸۱	۳	۴	۸
۰/۹۱	۳	۸	۰/۹۶	۴	۷	۰/۹۹	۳	۹	۰/۹۷	۲	۸	۹
-	-	-	۰/۹۰	۵	۶	۰/۸۵	۴	۵	۰/۸۳	۴	۶	۱۰
-	-	-	۰/۹۶	۵	۶	۰/۹۵	۴	۶	۰/۹۴	۳	۵	۱۱
۰/۹۰	۵	۷	۰/۸۵	۴	۶	۰/۸۲	۳	۵	۰/۷۹	۲	۴	۱۲
-	-	-	۰/۹۷	۲	۶	۰/۹۹	۳	۵	۰/۹۸	۲	۵	۱۳
۰/۹۹	۶	۹	۰/۹۵	۵	۶	۰/۹۲	۴	۷	۰/۹۰	۴	۶	۱۴

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	2	3	3	4	1	6	4	1	2	3	2	2	1	4
2	1	3	2	2	3	1	1	3	2	2	1	2	2	3
3	2	0	0	1	2	1	3	1	2	1	3	2	2	1

نوع قطعه انتخاب شده
تعداد قطعه فعال
تعداد قطعه ذخیره

شکل ۱- ساختار کروموزوم

الگوریتم ژنتیک طراحی شده برای هر دو روش (محاسبات با فرمول پیشین و روش بازگشتی) به صورت یکسان در نظر گرفته شده است. پس از اجرا مسئله الگو، نتایج و زمان حل مسئله الگو با استفاده از هر دو روش مقایسه شده است.

کروموزوم نوع قطعه استفاده شده و دو سطر بعدی، تعداد قطعات فعال و ذخیره مورد نیاز برای هر زیرسیستم است. سایر ویژگی‌های الگوریتم همچون، عملگرهای تقاطع و جهش، شرایط توقف و نحوی جریمه کردن تابع هدف در

<table style="width: 100%; border-collapse: collapse;"> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td></tr> <tr><td>1</td><td>4</td><td>1</td><td>2</td><td>2</td><td>3</td><td>2</td><td>1</td><td>4</td><td>6</td><td>1</td><td>4</td><td>3</td><td>3</td><td>2</td></tr> <tr><td>2</td><td>1</td><td>3</td><td>2</td><td>2</td><td>3</td><td>1</td><td>1</td><td>3</td><td>2</td><td>2</td><td>1</td><td>2</td><td>1</td><td>3</td></tr> <tr><td>3</td><td>2</td><td>0</td><td>0</td><td>1</td><td>2</td><td>1</td><td>3</td><td>1</td><td>2</td><td>1</td><td>3</td><td>2</td><td>3</td><td>1</td></tr> </table>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	1	4	1	2	2	3	2	1	4	6	1	4	3	3	2	2	1	3	2	2	3	1	1	3	2	2	1	2	1	3	3	2	0	0	1	2	1	3	1	2	1	3	2	3	1	<table style="width: 100%; border-collapse: collapse;"> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td></tr> <tr><td>2</td><td>4</td><td>1</td><td>4</td><td>4</td><td>2</td><td>3</td><td>1</td><td>5</td><td>6</td><td>2</td><td>5</td><td>1</td><td>4</td></tr> <tr><td>1</td><td>3</td><td>2</td><td>2</td><td>1</td><td>2</td><td>3</td><td>2</td><td>2</td><td>2</td><td>1</td><td>3</td><td>2</td><td>1</td></tr> <tr><td>2</td><td>0</td><td>2</td><td>3</td><td>1</td><td>1</td><td>1</td><td>0</td><td>3</td><td>3</td><td>0</td><td>3</td><td>1</td><td>3</td></tr> </table>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	2	4	1	4	4	2	3	1	5	6	2	5	1	4	1	3	2	2	1	2	3	2	2	2	1	3	2	1	2	0	2	3	1	1	1	0	3	3	0	3	1	3
1	2	3	4	5	6	7	8	9	10	11	12	13	14																																																																																																							
1	4	1	2	2	3	2	1	4	6	1	4	3	3	2																																																																																																						
2	1	3	2	2	3	1	1	3	2	2	1	2	1	3																																																																																																						
3	2	0	0	1	2	1	3	1	2	1	3	2	3	1																																																																																																						
1	2	3	4	5	6	7	8	9	10	11	12	13	14																																																																																																							
2	4	1	4	4	2	3	1	5	6	2	5	1	4																																																																																																							
1	3	2	2	1	2	3	2	2	2	1	3	2	1																																																																																																							
2	0	2	3	1	1	1	0	3	3	0	3	1	3																																																																																																							
⇓																																																																																																																				
<table style="width: 100%; border-collapse: collapse;"> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td></tr> <tr><td>1</td><td>4</td><td>1</td><td>2</td><td>2</td><td>4</td><td>2</td><td>3</td><td>1</td><td>5</td><td>6</td><td>4</td><td>3</td><td>3</td><td>2</td></tr> <tr><td>2</td><td>1</td><td>3</td><td>2</td><td>2</td><td>1</td><td>2</td><td>3</td><td>2</td><td>2</td><td>2</td><td>1</td><td>2</td><td>1</td><td>3</td></tr> <tr><td>3</td><td>2</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>3</td><td>3</td><td>3</td><td>2</td><td>3</td><td>1</td></tr> </table>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	1	4	1	2	2	4	2	3	1	5	6	4	3	3	2	2	1	3	2	2	1	2	3	2	2	2	1	2	1	3	3	2	0	0	1	1	1	0	3	3	3	2	3	1	<table style="width: 100%; border-collapse: collapse;"> <tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td></tr> <tr><td>2</td><td>4</td><td>1</td><td>4</td><td>3</td><td>2</td><td>1</td><td>4</td><td>6</td><td>1</td><td>2</td><td>5</td><td>1</td><td>4</td></tr> <tr><td>1</td><td>3</td><td>2</td><td>2</td><td>3</td><td>1</td><td>1</td><td>3</td><td>2</td><td>2</td><td>1</td><td>3</td><td>2</td><td>1</td></tr> <tr><td>2</td><td>0</td><td>2</td><td>3</td><td>2</td><td>1</td><td>3</td><td>1</td><td>2</td><td>1</td><td>0</td><td>3</td><td>1</td><td>3</td></tr> </table>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	2	4	1	4	3	2	1	4	6	1	2	5	1	4	1	3	2	2	3	1	1	3	2	2	1	3	2	1	2	0	2	3	2	1	3	1	2	1	0	3	1	3	
1	2	3	4	5	6	7	8	9	10	11	12	13	14																																																																																																							
1	4	1	2	2	4	2	3	1	5	6	4	3	3	2																																																																																																						
2	1	3	2	2	1	2	3	2	2	2	1	2	1	3																																																																																																						
3	2	0	0	1	1	1	0	3	3	3	2	3	1																																																																																																							
1	2	3	4	5	6	7	8	9	10	11	12	13	14																																																																																																							
2	4	1	4	3	2	1	4	6	1	2	5	1	4																																																																																																							
1	3	2	2	3	1	1	3	2	2	1	3	2	1																																																																																																							
2	0	2	3	2	1	3	1	2	1	0	3	1	3																																																																																																							

شکل ۲- عملگر تقاطع دونقطه‌ای

جهش تعداد F_M درصد از والدین به صورت تصادفی انتخاب می‌شوند و مقادیر ژن‌های آنها با احتمال p_M درصد تغییر پیدا می‌کنند. پس از این تغییر میزان تابع هدف برای کروموزوم جهش‌یافته محاسبه می‌شود و در صورتی که این میزان از تابع هدف کروموزوم اولیه بیشتر باشد، کروموزوم جهش‌یافته جایگزین کروموزوم قبلی (والد) می‌شود و در غیر این صورت کروموزوم والد به‌عنوان کروموزوم برتر باقی می‌ماند. اطلاعات مربوط به الگوریتم ژنتیک چنین است:

$$p_M = 0.3,$$

$$n_{Pop} = 300,$$

$$\text{Iterations} = 150,$$

$$F_M = 0.2$$

همان‌طور که در جدول (۲) نشان داده شده است، هنگامی که فقط یک قطعه ذخیره در زیرسیستم موجود باشد فرمول پیشین و فرمول بازگشتی دقیقاً یکسان و زمان حل هر دو روش مساوی است؛ اما با افزایش تعداد قطعات ذخیره، $(n_s \geq 2)$ در هر زیرسیستم، برتری فرمول بازگشتی نسبت به فرمول پیشین، مشخص می‌شود. در این حالت‌ها، علاوه بر کاهش پیچیدگی محاسباتی، زمان حل تا حد زیادی کاهش می‌یابد. همچنین به دلیل استفاده از روابط بازگشتی، فرمول پیشنهادی انعطاف‌پذیر بوده و برای هر تعداد قطعه ذخیره و فعال، فرمول محاسباتی مناسب به راحتی به دست می‌آید. در جدول (۳) نتایج و زمان حل مسئله الگو با روش حل پیشین و روش حل بازگشتی نشان داده شده است. همان‌طور که

همان‌طور که از نتایج مشخص است، زمان حل روش بازگشتی بسیار کمتر از روش پیشین است که این اختلاف زمان حل با افزایش تعداد قطعات ذخیره، بیشتر می‌شود. به منظور نشان دادن برتری روش حل بازگشتی در مقایسه با روش حل پیشین، فرمول محاسباتی هر دو روش، بر اساس تعداد قطعات ذخیره در جدول (۲) ارائه شده است.

۴-۱- عملگر تقاطع

عملگر تقاطع در الگوریتم ژنتیک برای تولید جواب‌های جدید از بهترین جواب‌های یافته شده در یک نسل از جواب‌ها است. عملگر تقاطع دونقطه‌ای، یکی از رایج‌ترین عملگرهای تقاطع در الگوریتم ژنتیک است. در این عملگر، دو کروموزوم به صورت تصادفی انتخاب می‌شوند و سپس دونقطه تصادفی از این کروموزوم‌ها انتخاب شده و ژن‌های مربوط به قسمتی از این کروموزوم‌ها با یکدیگر جابه‌جا می‌شوند. شکل (۲) نمونه‌ای از تقاطع دونقطه‌ای را نمایش می‌دهد.

در این مقاله علاوه بر عملگر تقاطع دونقطه‌ای از عملگرهای تقاطع ماکس-مین و ماسک نیز استفاده شده است [۵].

۴-۲- عملگر جهش

به منظور افزایش تنوع در جواب‌ها و فرار از نقاط بهینه محلی، از عملگر جهش استفاده شده است. برای به کار بردن عملگر

جدول ۲- مقایسه فرمول روش پیشین و روش بازگشتی

روش حل پیشین برای استراتژی مختلط	روش حل بازگشتی برای استراتژی مختلط	تعداد قطعات ذخیره
$1 - (1 - r_{iZ_i}(t))^{n_{Ai}} + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{Max, n_{Ai}}(u) du$	$1 - (1 - r(t))^{n_{Ai}} + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{Max, n_{Ai}}(u) du$	۱
$1 - (1 - r_{iZ_i}(t))^{n_{Ai}} + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{Max, n_{Ai}}(u) du + \int_0^t \int_{t_1}^t \rho_i(u) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) \times f_{iZ_i}^{(1)}(u-t_1) \times r_{iZ_i}(t-u) du dt_1$	$R_S^{n_{Ai}, 1}(t) + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_S^{n_{Ai}, 1}(u) du$	۲
$1 - (1 - r_{iZ_i}(t))^{n_{Ai}} + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{Max, n_{Ai}}(u) du + \int_0^t \int_{t_1}^t \rho_i(u) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) \times f_{iZ_i}^{(1)}(u-t_1) \times r_{iZ_i}(t-u) du dt_1 + \int_0^t \int_{t_1}^t \rho_i(u) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) \times f_{iZ_i}^{(r)}(u-t_1) \times r_{iZ_i}(t-u) du dt_1$	$R_S^{n_{Ai}, 1, r}(t) + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{n_{Ai}, 1, r}(u) du$	۳
$1 - (1 - r_{iZ_i}(t))^{n_{Ai}} + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{Max, n_{Ai}}(u) du + \int_0^t \int_{t_1}^t \rho_i(u) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) \times f_{iZ_i}^{(1)}(u-t_1) \times r_{iZ_i}(t-u) du dt_1 + \int_0^t \int_{t_1}^t \rho_i(u) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) \times f_{iZ_i}^{(r)}(u-t_1) \times r_{iZ_i}(t-u) du dt_1 + \int_0^t \int_{t_1}^t \rho_i(u) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) \times f_{iZ_i}^{(r)}(u-t_1) \times r_{iZ_i}(t-u) du dt_1$	$R_S^{n_{Ai}, 1, r, \gamma}(t) + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{n_{Ai}, 1, r, \gamma}(u) du$	۴
$1 - (1 - r_{iZ_i}(t))^{n_{Ai}} + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{Max, n_{Ai}}(u) du + \int_0^t \int_{t_1}^t \rho_i(u) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) \times f_{iZ_i}^{(1)}(u-t_1) \times r_{iZ_i}(t-u) du dt_1 + \int_0^t \int_{t_1}^t \rho_i(u) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) \times f_{iZ_i}^{(r)}(u-t_1) \times r_{iZ_i}(t-u) du dt_1 + \int_0^t \int_{t_1}^t \rho_i(u) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) \times f_{iZ_i}^{(r)}(u-t_1) \times r_{iZ_i}(t-u) du dt_1 + \int_0^t \int_{t_1}^t \rho_i(u) \times f_{iZ_i}^{Max, n_{Ai}}(t_1) \times f_{iZ_i}^{(r)}(u-t_1) \times r_{iZ_i}(t-u) du dt_1$	$R_S^{n_{Ai}, 1, r, \gamma, \delta}(t) + \int_0^t \rho_i(u) \times r_{iZ_i}(t-u) \times f_{iZ_i}^{n_{Ai}, 1, r, \gamma, \delta}(u) du$	۵

جدول ۳- مقایسه نتایج و زمان حل مسئله الگو به روش پیشین و روش بازگشتی

قابلیت اطمینان	زمان حل		حداکثر بهبود امکان پذیر (MPI)	قابلیت اطمینان سیستم	
	روش حل پیشین	روش حل بازگشتی		روش حل پیشین	روش حل بازگشتی
سوئیچ ۰/۳۰	۸ ساعت و ۴۰ دقیقه	۱ ساعت ۱۰ دقیقه	٪ ۳/۱۴	۰/۹۶۲۶۷۲۴۶	۰/۹۶۳۸۴۵۶۹
۰/۴۰	۸ ساعت و ۵۲ دقیقه	۱ ساعت ۲۱ دقیقه	٪ ۲/۹۶	۰/۹۶۳۵۲۳۴۶	۰/۹۶۴۶۰۳۶۹
۰/۵۰	۹ ساعت و ۱۱ دقیقه	۱ ساعت ۵ دقیقه	٪ ۳/۱۸	۰/۹۶۴۳۶۲۱۴	۰/۹۶۵۴۹۶۵۳
۰/۶۰	۸ ساعت و ۳۵ دقیقه	۱ ساعت ۲۶ دقیقه	٪ ۴/۰۱	۰/۹۶۵۲۰۱۴۳	۰/۹۶۶۵۹۹۷۳۴
۰/۷۰	۸ ساعت و ۴۵ دقیقه	۱ ساعت ۲ دقیقه	٪ ۳/۳۴	۰/۹۶۶۰۵۱۲۳	۰/۹۶۷۱۸۷۴۵
۰/۸۰	۹ ساعت و ۵ دقیقه	۱ ساعت ۳۰ دقیقه	٪ ۲/۲۳	۰/۹۶۶۸۹۳۲۴	۰/۹۶۷۶۳۳۶۸
۰/۸۵	۸ ساعت و ۵۶ دقیقه	۱ ساعت ۱۷ دقیقه	٪ ۳/۶۱	۰/۹۶۷۳۱۱۸۵	۰/۹۶۸۴۹۳۲۱۴
۰/۹۰	۸ ساعت و ۳۹ دقیقه	۱ ساعت ۸ دقیقه	٪ ۳/۰۴	۰/۹۶۸۵۳۷۱۲	۰/۹۶۹۴۹۴۸۷
۰/۹۵	۸ ساعت و ۵۷ دقیقه	۱ ساعت ۳ دقیقه	٪ ۲/۸۲	۰/۹۷۲۹۷۰۳۲	۰/۹۷۳۷۳۴۸۲
۰/۹۹	۸ ساعت و ۲۶ دقیقه	۱ ساعت ۹ دقیقه	٪ ۲/۴۶	۰/۹۸۱۹۴۶۸۹	۰/۹۸۲۳۹۲۳۲

یک زیرسیستم (سیستم) نحوه استفاده از قطعات مازاد حائز اهمیت است که تحت عنوان استراتژی قطعات مازاد شناخته می‌شود. به‌طور سنتی دو استراتژی فعال و ذخیره برای قطعات مازاد در نظر گرفته می‌شود. به‌تازگی استراتژی جدیدی تحت عنوان استراتژی مختلط ارائه شده است که عملکردی به‌مراتب بهتر نسبت به استراتژی‌های سنتی داشته است. در این مقاله یک روش حل جدید ارائه شده است که با استفاده از روابط بازگشتی به محاسبه قابلیت اطمینان استراتژی مختلط می‌پردازد. این روش جدید علاوه بر کاهش پیچیدگی فرمول محاسباتی استراتژی مختلط زمان حل آن را نیز تا حد زیادی بهبود می‌بخشد. در این مقاله، ابتدا فرمول محاسباتی مختلط بررسی شد و سپس با استفاده از روابط بازگشتی فرمول جدیدی برای این استراتژی ارائه شد و در نهایت یک مسئله الگوی سری- موازی با استفاده از هر دو فرمول ارزیابی شد. جواب‌ها و زمان حل مسئله بررسی و مقایسه شد. نتایج نشان می‌دهد که روش حل بازگشتی، از لحاظ پیچیدگی و زمان حل، نسبت به روش پیشین بسیار کارآمدتر است و قابلیت اطمینان سیستم را افزایش می‌دهد. از این رو پیشنهاد می‌شود تا برای پژوهش‌های آتی، از روش حل بازگشتی برای مسائلی که در

مشخص است در سوئیچ‌های مختلف، روش حل بازگشتی زمان حل کمتری نسبت به روش حل پیشین داشته است. همچنین با کمک روش حل بازگشتی نتایج قابلیت اطمینان سیستم بهبود پیدا کرده است. به‌عنوان مثال در سوئیچ ۰/۹۹ قابلیت اطمینان سیستم به روش حل پیشین ۰/۹۸۱۹۴۶۸۹ است که این مقدار قابلیت اطمینان با روش حل بازگشتی به ۰/۹۸۲۳۹۲۳۲ افزایش پیدا کرده است. به‌منظور نمایش میزان بهبود حاصل شده نسبت به روش حل پیشین، از شاخص حداکثر بهبود امکان‌پذیر^۶ (MPI) استفاده شده است. این شاخص در پژوهش‌هایی همچون [۵، ۷، ۱۸ و ۲۳] استفاده شده است که به‌صورت زیر تعریف می‌شود.

$$MPI(\%) = \frac{\text{قابلیت اطمینان روش حل پیشین} - \text{قابلیت اطمینان با روش حل بازگشتی}}{\text{قابلیت اطمینان روش حل پیشین} - ۱}$$

۴- نتیجه‌گیری

در بحث بهینه‌سازی قابلیت اطمینان، استفاده از قطعات مازاد یک روش بسیار رایج است. در زمان استفاده از قطعات مازاد در

مقاله، فرمول حل بازگشتی مناسب برای استراتژی K-Mixed که به تازگی معرفی شده است، نیز استخراج و آزمایش شد.

آنها قطعات ذخیره گرم وجود دارد، استفاده شود. علاوه بر این، پیشنهاد می شود تا با استفاده از روش توضیح داده شده در این

واژه نامه

- | | | |
|--|---------------------|---------------------------------|
| 1. redundancy allocation problem (rap) | 3. active strategy | 6. maximum possible improvement |
| 2. redundancy strategy | 4. standby strategy | |
| | 5. mixed strategy | |

مراجع

- Chambari, A., Rahmati, S. H. A. and Najafi, A. A., "A Bi-Objective Model to Optimize Reliability and Cost of System with a Choice of Redundancy Strategies", *Computers and Industrial Engineering*, Vol. 63, No. 1, pp. 109-119, 2012.
- Kuo, w. and Prasad, V. R., "An Annotated Overview of System-Reliability Optimization", *IEEE Transactions on Reliability*, Vol. 49, No. 2, pp. 176-187, 2000.
- Chern, M. S., "On the Computational Complexity of Reliability Redundancy Allocation in a Series System", *Operations Research Letters*, Vol. 11, No. 5, pp. 309-315, 1992.
- Fyffe, D. E., Hines W. W. and Lee, N. K., "System Reliability Allocation and a Computational Algorithm", *IEEE Transactions on Reliability*, Vol. 2, pp. 64-69, 1968.
- Abouei Ardakan, M. and Zeinal Hamadani, A., "Reliability Optimization of Series- Parallel Systems with Mixed Redundancy Strategy in Subsystems", *Reliability Engineering and System Safety*, Vol. 130, pp. 132-139, 10// 2014.
- Ng, K. Y. and Sancho, N., "A hybrid 'Dynamic Programming/ Depth-First Search' algorithm, with an Application to Redundancy Allocation", *IIEE Transactions*, Vol. 33, No. 12, pp. 1047-1058, 2001.
- Coit, D. W., "Maximization of System Reliability with a Choice of Redundancy Strategies", *IIEE Transactions*, Vol. 35, No. 6, pp. 535-543, 2003.
- Misra, K. B. and Sharma, U., "An Efficient Algorithm to Solve Integer-Programming Problems Arising in System-Reliability Design", *IEEE Transactions on Reliability*, Vol. 40, No. 1, pp. 81-91, 1991.
- Misra, K. B., "Reliability Optimization of a Series-Parallel System", *IEEE Transactions on Reliability*, Vol. 21, No. 4, pp. 230-238, 1972.
- Coit, D. W. and Smith, A. E., "Reliability Optimization of Series- Parallel Systems using a Genetic Algorithm", *IEEE Transactions on Reliability*, Vol. 45, No. 2, pp. 254-260, 266, 1996.
- Eslami, A. A., Seyed-esfahani, M., Farsi, M. A. and Mahmoudi, M., "Redundancy Optimization in Series- Parallel Systems and Choice of Redundancy Strategies", *Tenth International Industrial Engineering Conference*, 27-28 January, 2014. (In Farsi).
- Liang, Y.-C. and Smith, A. E., "An Ant Colony Optimization Algorithm for the Redundancy Allocation Problem (RAP)", *IEEE Transactions on reliability*, Vol. 53, No. 3, pp. 417-423, 2004.
- Ouzineb, M., Nourelfath, M. and Gendreau, M., "Tabu Search for the Redundancy Allocation Problem of Homogenous Series-Parallel Multi-State Systems", *Reliability Engineering and System Safety*, Vol. 93, No. 8, pp. 1257-1272, 2008.
- Beji, N., Jarboui, B., Eddaly, M. and Chabchoub, H., "A Hybrid Particle Swarm Optimization Algorithm for the Redundancy Allocation Problem", *Journal of Computational Science*, Vol. 1, No. 3, pp. 159-167, 2010.
- Seyed Esfahani, M., Hajian Heidary, M. and Jaberi, S., "A Simulated Annealing Algorithm for Fuzzy Reliability Optimization Model in Series-Parallel and Standby Systems", *International Journal of Industrial Engineering and Production Management*, Vol. 24, No. 4, pp. 413-422, January 2014. (In Farsi).
- Coit, D. W., "Cold-Standby Redundancy Optimization for Nonrepairable Systems", *Iie Transactions*, Vol. 33, No. 6, pp. 471-478, 2001.
- Robinson, D. and Neuts, M., "Standby Redundancy in Reliability: A Review", *IEEE Transactions on Reliability*, Vol. 38, No. 4, pp. 430-435, 1989.
- Abouei Ardakan, M. and Zeinal Hamadani, A., "Reliability-Redundancy Allocation Problem with Cold-Standby Redundancy Strategy", *Simulation Modelling Practice and Theory*, Vol. 42, pp. 107-118, 2014.
- Sakawa, M., "Multi-Objective Optimization by the Surrogate Worth Trade-Off Method", *IEEE Transaction on Reliability*, Vol. 27, No 5, pp. 311-314, 1978.
- Azizi Boroujerdi, S., Amiri, M. and Garshasbi, D., "A Mathematical Model for Solving a Multi

- Objective Choice of Redundancy Allocation Strategy Problem by Optimizing MTTF for K out Of N systems”, 13th *International Conference on Industrial Engineering*, IIEC 2017, Babol, Mazandaran. (In Farsi).
21. Jamshidi, H., and Hejaz, T. H., “A Bi-Objective Mathematical Programming Model to Solve Redundancy Allocation Problems in a Series-Parallel System with Mixed Strategy and Price Discount Policy”, 13th *International Conference on Industrial Engineering*, IIEC 2017, Babol, Mazandaran (In Farsi).
22. Feizabadi, M. and Jahromi A. E., “A New Model for Reliability Optimization of Series-Parallel Systems with Non-Homogeneous Components”, *Reliability Engineering and System Safety*, Vol. 157, pp. 101-112, 2017.
23. Kim, H., “Maximization of System Reliability with the Consideration of Component Sequencing”, *Reliability Engineering and System Safety*, Vol. 170: 64-72, 2018.
24. Ardakan, M. A. and Rezvan, M. T., “Multi-Objective Optimization of Reliability-Redundancy Allocation Problem with Cold-Standby Strategy Using NSGA-II”, *Reliability Engineering and System Safety*, Vol. 172, pp. 225-238. 2018.
25. Huang X, Coolen FP, Coolen-Maturi T. “A Heuristic Survival Signature based Approach for Reliability-Redundancy Allocation”, *Reliability Engineering and System Safety*, Vol. 185, pp. 511-517, 2019.
26. Zoufaghari, H., Hamadani, A. Z. and Ardakan, M. A., “Bi-Objective Redundancy Allocation Problem for a System with Mixed Repairable and Non-Repairable Components,” *ISA Transactions*, Vol. 53, No. 1, pp. 17-24, 2014.
27. Ardakan Abouei, M., Sima, M., Zeinal Hamadani, A. and Coit, D. W., “A Novel Strategy for Redundant Components in Reliability-Redundancy Allocation Problems”, *IIE Transactions*, accepted, 2016.
28. Peiravi, A., Karbasian, M., Ardakan, M. A. and Coit, D. W., “Reliability Optimization of Series-Parallel Systems with K-Mixed Redundancy Strategy”, *Reliability Engineering and System Safety*, Vol. 183, pp. 17-28. 2019.

زیرسیستم، از گام‌های بازگشتی استفاده می‌کنیم، گام‌های بازگشتی بر اساس تعداد قطعات ذخیره استفاده شده در زیرسیستم تعیین می‌شوند. نحوه محاسبه گام‌ها به صورت زیر تشریح شده است:

گام اول روش بازگشتی: در این گام قطعه ذخیره نداریم و تمام قطعات زیرسیستم فعال هستند که در این حالت قابلیت اطمینان زیرسیستم برابر است با: حداقل یکی از n_{Ai} قطعه فعال تا پایان زمان T درست کار کند. (حالت اول)

$$R_S^{n_{Ai}}(t) = 1 - (1 - r_{iZ_1}(t))^{n_{Ai}} \quad (پ-۱)$$

در گام دوم علاوه بر n_{Ai} قطعه فعال، یک قطعه ذخیره در زیرسیستم استفاده شده است که قابلیت اطمینان زیرسیستم به صورت زیر محاسبه می‌شود:

$$R_S^{n_{Ai}+1}(t) =$$

p + (حداقل یکی از n_{Ai} قطعه فعال تا زمان T کار کند) p (خرابی آخرین قطعه از n_{Ai} قطعه فعال و کارکرد صحیح قطعه اول ذخیره تا پایان زمان T)

$$= R_S^{n_{Ai}}(t) + \int_0^t \rho_i(u) \times r_{iZ_1}(t-u) \times f_{iZ_1}^{Max, n_{Ai}}(u) du \quad (پ-۲)$$

پیوست- تشریح روش حل بازگشتی پیشنهادی

به منظور محاسبه قابلیت اطمینان در ساختار ذخیره سرد، می‌بایست حالت‌هایی که در آن، یک زیرسیستم می‌تواند به طور صحیح کار کند را محاسبه کنیم. در استراتژی مختلط زیرسیستم، کار خود را با n_{Ai} قطعه فعال آغاز می‌کند و بعد از خراب شدن n_{Ai} قطعه فعال، قطعات ذخیره به کمک سیستم سوئیچینگ یک به یک فعال و وارد سیستم می‌شوند. به طور کلی $n_{Si} + 1$ حالت وجود داد که در آن زیرسیستم می‌تواند به طور صحیح کار کند. این حالت‌ها به شرح زیر است:

حالت اول: حداقل یکی از n_{Ai} قطعه فعال تا پایان زمان T درست کار کند.

حالت دوم: خرابی آخرین قطعه از n_{Ai} قطعه فعال و کارکرد صحیح قطعه ذخیره اول تا پایان زمان T.

حالت $n_{Si} - 1$ ام: خرابی آخرین قطعه از n_{Ai} قطعه فعال و همچنین خرابی $(n_{Si} - 1)$ امین قطعه ذخیره و کارکرد صحیح قطعه ذخیره n_{Si} ام تا پایان زمان T.

اطمینان زیرسیستم از حاصل جمع این حالت‌ها به دست می‌آید. در روش بازگشتی به منظور محاسبه قابلیت اطمینان

می‌کند. در ادامه نحوه محاسبه بازگشتی تابع چگالی f توضیح داده شده است:

$$F_{IZ_i}^{n_{Ai},1}(t) = 1 - R_{IZ_i}^{n_{Ai},1}(t) = \int_0^t F_{IZ_i}^{(j)}(t-u) f_{IZ_i}^{Max, n_{Ai}}(u) du \quad (4-پ)$$

$$f_{IZ_i}^{n_{Ai},1}(t) = F_{IZ_i}^{(j)}(t-t) f_{IZ_i}^{Max, n_{Ai}}(t) + \int_0^t f_{IZ_i}^{(l)}(t-u) f_{IZ_i}^{Max, n_{Ai}}(u) du = \int_0^t f_{IZ_i}^{(j)}(t-u) f_{IZ_i}^{Max, n_{Ai}}(u) du \quad (5-پ)$$

$$f_{IZ_i}^{n_{Ai},1,2,\dots,n_{Si}}(t) = \int_0^t f_{IZ_i}^{(l)}(t-u) \times f_{IZ_i}^{n_{Ai},1,2,\dots,(n_{Si}-1)}(u) du \quad (6-پ)$$

همان‌طور که در این بخش مشاهده می‌شود تابع چگالی $f_{IZ_i}^{n_{Ai},1}(t)$ با انتگرال‌گیری از حاصل ضرب $f_{IZ_i}^{(l)}(t-u)$ و $f_{IZ_i}^{Max, n_{Ai}}(u)$ به دست می‌آید. برای درک بهتر فرمول بازگشتی، باید به مفهوم تابع چگالی f توجه کنیم. تابع چگالی f احتمال خرابی n_{Ai} قطعه فعال و $(n_{Si}-1)$ قطعه ذخیره تا قبل از پایان زمان T را محاسبه می‌کند که بر اساس تعداد قطعات ذخیره به شکل $f_{IZ_i}^{n_{Ai},1,2,\dots,n_{Si}}(t)$ تعریف شده است.

در ادامه با محاسبه گام سوم برای زیرسیستم با دو قطعه ذخیره، فرمول بازگشتی را برای گام n ام برای زیرسیستم با $(n_{Si}-1)$ قطعه ذخیره بسط داده می‌شود:

$$R_S^{n_{Ai},1,2}(t) = p + (حداقل یکی از n_{Ai} قطعه فعال تا زمان T کار کند) p (خرابی آخرین قطعه از n_{Ai} قطعه فعال و کارکرد صحیح قطعه اول ذخیره تا پایان زمان T) $+p$ (خرابی n_{Ai} قطعه فعال و اولین قطعه ذخیره قبل از زمان T و کارکرد صحیح قطعه دوم ذخیره تا پایان زمان T) $(7-پ)$$$

همان‌طور که مشاهده می‌شود گام دوم از حاصل جمع گام اول با یک انتگرال به دست آمده است. این انتگرال احتمال روشن شدن قطعه ذخیره و کارکرد صحیح آن تا پایان زمان T به شرطی که آخرین قطعه از n_{Ai} قطعه فعال قبل از پایان زمان T خراب شده باشند را محاسبه می‌کند.

برای حالتی که در زیرسیستم از دو قطعه ذخیره استفاده می‌شود، گام سوم استفاده می‌شود. گام سوم از حاصل جمع گام دوم با یک مقدار جدید حاصل می‌شود که در این گام قابلیت اطمینان زیرسیستم به صورت زیر محاسبه می‌شود:

$$R_S^{n_{Ai},1,2}(t) = p + (حداقل یکی از n_{Ai} قطعه فعال تا زمان T کار کند) p (خرابی آخرین قطعه از n_{Ai} قطعه فعال و کارکرد صحیح قطعه اول ذخیره تا پایان زمان T) $+p$ (خرابی n_{Ai} قطعه فعال و اولین قطعه ذخیره قبل از زمان T و کارکرد صحیح قطعه دوم ذخیره تا پایان زمان T) $= R_S^{n_{Ai}}(t) + \int_0^t \rho_i(u) \times r_{IZ_i}(t-u) \times f_{IZ_i}^{Max, n_{Ai}}(u) du + \int_0^t \rho_i(u) \times r_{IZ_i}(t-u) \times f_{IZ_i}^{n_{Ai},1}(u) du = R_S^{n_{Ai},1}(t) + \int_0^t \rho_i(u) \times r_{IZ_i}(t-u) \times f_{IZ_i}^{n_{Ai},1}(u) du \quad (3-پ)$$$

همان‌طور که در گام‌های دوم و سوم مشاهده می‌کنید هر گام جدید از حاصل جمع گام قبلی با یک انتگرال به دست می‌آید. با افزایش تعداد قطعات ذخیره، این روند نیز به همین شکل ادامه می‌یابد، یعنی قابلیت اطمینان هر گام، از حاصل جمع قابلیت اطمینان گام قبل با یک مقدار جدید به دست می‌آید. این مقدار جدید در هر گام، احتمال روشن شدن آخرین قطعه ذخیره در زیرسیستم و کارکرد صحیح آن تا پایان زمان T به شرطی که n_{Ai} قطعه فعال و $(n_{Si}-1)$ قطعه ذخیره تا قبل از پایان زمان T خراب شوند را محاسبه می‌کند. با توجه به توابع داخل انتگرال، در هر گام فقط مقدار تابع چگالی f تغییر می‌کند. که این تابع چگالی نیز از یک روش بازگشتی پیروی