مقاله پژوهشی

ارائه الگوریتم چندهدفه بهبودیافته به منظور انتخاب بهینه در ترکیب وب‌سرویس‌های آگاه به کیفیت در اینترنت اشیا

نرجس ظهیری، فرشته دهقانی و سلمان گلی بیدگلی

چکیده: با ظهور اینترنت اشیا، مسئله ترکیب وب‌سرویس‌ها و برآورده‌کردن نیازهای متعدد و پیچیده از سوی کاربران بیش از پیش مورد توجه قرار گرفته است. به منظور ارائه خدمت به برنامه‌های کاربردی سیستم‌های مبتنی بر اینترنت اشیا، کاندیداهای متفاوتی با ویژگی‌های کیفی گوناگون وجود دارند. بنابراین یک چالش اساسی، انتخاب یک ترکیب بهینه از میان این کاندیداها به عنوان یک مسئله NP-hard است. در این مقاله، راه‌حل نزدیک به بهینه برای حل مسئله ترکیب وب‌سرویس در اینترنت اشیا و یافتن جبهه بهینه پارتو با استفاده از الگوریتم جستجوی فراابتکاری چندهدفه NSGA-III ارائه شده و سپس به منظور افزایش کیفیت و تنوع راه‌حل‌ها، الگوریتم بهبودیافته‌ای با ترکیب الگوریتم NSGA-III و تابع برازندگی جدید پیشنهاد گردیده است. به منظور بهینه‌سازی ترکیب سرویس‌ها در الگوریتم پیشنهادی از ۹ پارامتر کیفی استفاده شده و در ادامه برای عملکرد بهتر به سه هدف اصلی تبدیل شده‌اند. نتایج آزمایش‌ها نشان می‌دهند که رویکرد پیشنهادی از نظر میانگین دو هدف از سه هدف در مقایسه با الگوریتم NSGA-III نتیجه بهتری دارد. همچنین از نظر شاخص‌های عملکردی توانسته به طور میانگین به ۱۱ درصد پوشش بیشتر دست یابد و هم از لحاظ توزیع راه‌حل‌ها و هم از لحاظ پراکندگی نسبت به سایر الگوریتم‌ها عملکرد بهتری داشته باشد.

کلیدواژه: **الگوریتم تکاملی، اینترنت اشیا، بهینه‌سازی چندهدفه، ترکیب و انتخاب بهینه وب‌سرویس‌ها، وب‌سرویس‌های آگاه به کیفیت.**

۱- مقدمه

در سال‌های اخیر، رایانش ابری به‌عنوان یکی از مهم‌ترین الزامات اکوسیستم اینترنت اشیا (IoT)[1] پیشرفت چشم‌گیری داشته است. سرویس‌های مختلف با عملکرد یکسان توسط ارائه‌دهندگان سرویس‌های ابری (CSP)[2] در سراسر جهان ارائه می‌گردند [۱]. از یک طرف به دلیل افزایش تعداد سرویس‌های ابری که از نظر عملکرد یکسان هستند، استفاده از روش‌های دقیق انتخاب سرویس ضروری است و از طرفی دیگر با توجه به درخواست‌های ترکیبی و پیچیده کاربران و محدودیت منابع یک ارائه‌دهنده سرویس، نیاز به ترکیب این سرویس‌های ابری بیش از پیش احساس می‌شود [۲]. ترکیب سرویس به عنوان یک نیاز مهم به درخواست‌های مختلف اجازه می‌دهد تا ابرها و منابع را به اشتراک بگذارند و سرویس‌های ترکیبی کارآمدی را در محیط‌های اینترنت اشیا دریافت کنند. ارتباطات و داده‌های به اشتراک گذاشته‌شده از طریق اشیا در اینترنت اشیا، نقش مهمی در تولید برنامه‌های کاربردی مختلف دارند [۳]. اساساً اینترنت اشیا به عنوان مجموعه‌ای از اشیای فیزیکی در نظر گرفته می‌شود؛ به نحوی که از طریق اینترنت به یکدیگر متصل شده‌اند تا شبکه‌ای از دستگاه‌ها را تشکیل دهند. بیشتر برنامه‌های کاربردی در اینترنت اشیا برای تصمیم‌گیری در مورد بهترین راه‌حل از الگوریتم‌های بهینه‌سازی چندهدفه استفاده می‌کنند [۴]. به عنوان مثال، سرویس‌ها در اینترنت اشیا باید بهینه شوند به نحوی که حداقل زمان اجرا و حداکثر قابلیت اطمینان را داشته باشند؛ یعنی زمان اجرا را به حداقل و قابلیت اطمینان را به حداکثر برسانند [۵] و [۶]. با این حال رویکردهای زیادی برای پیاده‌سازی بهینه‌سازی چندهدفه وجود دارند. از آنجا که در مسئله ترکیب سرویس‌ها، فضای جستجو به دلیل تعداد زیاد ارائه‌دهندگان سرویس‌های ابری بسیار بزرگ است، انتخاب یک ترکیب بهینه از میان این تعداد سرویس کاندیدا که هر ارائه‌کننده فراهم می‌کند، یک مسئله NP-hard است [۷]. از این رو برای یافتن اهداف نزدیک به بهینه، استفاده از الگوریتم‌های چندهدفه تکاملی پیشنهاد می‌شوند. به طور کلی درخواست کاربر برای دریافت پاسخ مناسب به ارائه‌دهنده سرویس تحویل داده می‌شود؛ سپس مؤلفه ترکیب سرویس به‌عنوان یک واسط[3] در حوزه اینترنت اشیا کمک می‌کند [۸].

معماری سرویس‌گرا (SOA)[4] سبک جدیدی از نرم‌افزار است که برای ایجاد برنامه‌های کاربردی مبتنی بر سرویس توسعه یافته است [۹]. سیستم مرتبط با این معماری به عنوان ترکیب سرویس‌های وب شناخته می‌شود. یک مدل ترکیبی در سیستم‌های سرویس‌گرا از وب‌سرویس‌هایی با عملکردهای متفاوت تشکیل گردیده که به آنها وب‌سرویس انتزاعی می‌گویند. برای سهولت در درک این مدل ترکیبی می‌توان آن را به صورت یک گراف نشان داد که هر گره، نشان‌دهنده یک وب‌سرویس انتزاعی و یال‌ها نشان‌دهنده ارتباط بین وب‌سرویس‌ها هستند. برای هر وب‌سرویس انتزاعی، مجموعه‌ای از سرویس‌ها با عملکرد یکسان، اما کیفیت متفاوت وجود دارد که به آن سرویس کاندیدا می‌گویند [۱۰]. این کاندیداها که با ویژگی‌های کیفیت سرویس (QoS)[5] از یکدیگر متمایز

این مقاله در تاریخ ۱۵ خرداد ماه ۱۴۰۲ دریافت و در تاریخ ۱۶ اسفند ماه ۱۴۰۲ بازنگری شد.

نرجس ظهیری، گروه مهندسی نرم‌افزار، دانشکده برق و کامپیوتر، دانشگاه کاشان، کاشان، ایران، (email: nargess.zahiri@gmail.com).

فرشته دهقانی (نویسنده مسئول)، گروه هوش مصنوعی، دانشکده برق و کامپیوتر، دانشگاه کاشان، کاشان، ایران، (email: fdehghani@kashanu.ac.ir).

سلمان گلی بیدگلی، گروه مهندسی نرم‌افزار، دانشکده برق و کامپیوتر، دانشگاه کاشان، کاشان، ایران، (email: salmangoli@kashanu.ac.ir).

1. Internet of Things
2. Cloud Service Provider
3. Broker
4. Service Oriented Architecture
5. Quality of Service

جدول ۱: روابط توابع تجمعی بر اساس ویژگی و ساختار الگوی گراف [۱۲].

ویژگی‌های کیفی	واحد	توالی m تا گره	حلقه با k بار تکرار	شرطی با احتمال p	موازی با احتمال p
زمان پاسخ (T)	میلی‌ثانیه	$\sum_{i=1}^{m} T(s_i)$	$k\sum_{i=1}^{n} T(s_i)$	$\sum_{i=1}^{m} P_i \times T(s_i)$	$\max_{i=1}^{p} T(s_i)$
دسترس‌پذیری (A)	–	$\prod_{i=1}^{m} A(s_i)$	$(\prod_{i=1}^{n} A(s_i))^k$	$\sum_{i=1}^{m} P_i \times A(s_i)$	$\prod_{i=1}^{p} A(s_i)$
قابلیت اطمینان (R)	–	$\prod_{i=1}^{m} R(s_i)$	$(\prod_{i=1}^{n} R(s_i))^k$	$\sum_{i=1}^{m} P_i \times R(s_i)$	$\min_{i=1}^{p} R(s_i)$
بازدهی (G)	فراخوانی/ ثانیه	$\min_{i=1}^{m} G(s_i)$	$\min_{i=1}^{m} \frac{G(s_i)}{k}$	$\sum_{i=1}^{m} P_i \times G(s_i)$	$\min_{i=1}^{p} G(s_i)$
تأخیر (L)	میلی‌ثانیه	$\sum_{i=1}^{m} L(a_i)$	$k.\sum_{i=1}^{n} L(a_i)$	$\sum_{i=1}^{m} P_i \times L(s_i)$	$\min_{i=1}^{p} L(s_i)$
موفقیت (U)	–	$\prod_{i=1}^{m} U(s_i)$	$(\prod_{i=1}^{n} U(s_i))^k$	$\sum_{i=1}^{m} P_i \times U(s_i)$	$\frac{\sum_{i=1}^{m} U(s_i)}{n}$
انطباق (C)	–	$\frac{\sum_{i=1}^{m} C(s_i)}{n}$	$\frac{\sum_{i=1}^{m} C(s_i)}{n}$	$\sum_{i=1}^{m} P_i \times C(s_i)$	$\frac{\sum_{i=1}^{m} C(s_i)}{n}$
بهترین عملکرد (B)	–	$\frac{\sum_{i=1}^{m} B(s_i)}{n}$	$\frac{\sum_{i=1}^{m} B(s_i)}{n}$	$\sum_{i=1}^{m} P_i \times B(s_i)$	$\frac{\sum_{i=1}^{p} B(s_i)}{n}$
مستندسازی (D)	–	$\frac{\sum_{i=1}^{m} D(s_i)}{n}$	$\frac{\sum_{i=1}^{m} D(s_i)}{n}$	$\frac{\sum_{i=1}^{m} D(s_i)}{n}$	$\frac{\sum_{i=1}^{p} D(s_i)}{n}$

می‌شوند، نقش کلیدی در فرایند انتخاب و ترکیب سرویس‌ها دارند. این ویژگی‌های کیفی شامل ۹ ویژگی از جمله در دسترس بودن[1]، زمان پاسخگویی[2]، قابلیت اطمینان[3]، توان عملیاتی[4]، تأخیر[5]، انطباق[6]، موفقیت[7]، موفقیت[7]، مستندسازی[8] و بهترین عملکرد[9] هستند. دسته‌ای از این ویژگی‌ها مانند زمان پاسخ و تأخیر باید کمینه شوند. دسته دیگر خصوصیاتی مانند توان عملیاتی و دسترس‌پذیری هستند که باید بیشینه شوند [۱۱]. سرویس‌های انتزاعی در این مدل ترکیبی بر اساس الگوهای مختلفی با یکدیگر در ارتباط هستند (انواع این الگوها در بخش ۲ آمده‌اند) که این الگوها را می‌توان با روش‌های مختلف از جمله روش تجمعی ساده‌سازی کرد [۱۲]. از این رو یافتن بهترین راه‌حل یعنی بهترین کاندیدا برای ترکیب سرویس‌ها چالشی است که با استفاده از الگوریتم‌های تکاملی انجام می‌شود.

در بیشتر مقالات ارائه‌شده، تنها الگوی توالی در گراف در نظر گرفته شده است [۱۳] تا [۱۶]. در برخی مقالات نیز مسئله را با وزن‌دهی به پارامترها به صورت تک‌هدفه حل کرده‌اند [۱۶] و [۱۷] که به دلیل ماهیت متضاد معیارهای کیفیت سرویس (QoS) مختلف، انجام این کار صحیح نیست. بنابراین بهتر است از الگوریتم‌های بهینه‌سازی چندهدفه برای رسیدگی به این مسئله استفاده شود. مقالات [۱۳] تا [۱۹] از الگوریتم تکاملی NSGA-II استفاده کرده‌اند که بخشی از آنها مشکلات گفته‌شده در بالا را دارند. همچنین [۲۰] تا [۲۴] از الگوریتم تکاملی NSGA-III یا بهبودیافته آن استفاده کرده‌اند. این الگوریتم، یک الگوریتم بهینه‌سازی چندهدفه تکاملی است که با استفاده از مجموعه‌ای از نقاط مرجع بر اساس مرتب‌سازی نامغلوب مبتنی بر سطح عمل می‌کند. از آنجا که در الگوریتم NSGA-III کیفیت راه‌حل‌ها به موقعیت مرجع بستگی دارد، در [۲۰] تا [۲۲] از تصمیم‌گیرنده خواسته شده تا نقطه مرجع را مشخص کنند و چون معیار دقیقی برای تعیین یک نقطه مرجع یا یک منطقه هدف وجود ندارد، برای مشتریان غیرممکن است که مقدار دقیقی از هر معیار مورد انتظار ارائه کنند؛ بنابراین بهتر است این عمل به صورت خودکار انجام شود.

در این مقاله برای حل مسئله تشریح‌شده، ابتدا با استفاده از روش تجمعی و روابط موجود برای ساده‌سازی انواع الگوها (جدول ۱)، ساده‌سازی گراف انجام شده است. سپس با استفاده از الگوریتم تکاملی NSGA-III بهبودیافته، کاندیدای برتر برای هر وب‌سرویس به گونه‌ای انتخاب شده که کیفیت نهایی حاصل از ساده‌سازی گراف ترکیب بهینه باشد. در این الگوریتم ارائه‌شده علاوه بر بهبود کیفیت راه‌حل‌ها، تنوع آنها نیز در نظر گرفته شده است.

این مقاله شامل بخش‌های زیر است: در بخش ۲ مروری بر مفاهیم مربوط به الگوها و همچنین توصیفی از الگوریتم بهینه‌ساز چندهدفه NSGA-III شده است. در بخش ۳ کارهای گذشته بیان گردیده و الگوریتم NSGA-III بهبودیافته در بخش ۴ آمده است. بخش ۵ به ارزیابی عملکرد و نتایج روش ارائه‌شده پرداخته و در بخش ۶ نتیجه‌گیری آمده است.

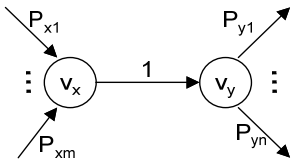
۲- مفاهیم اولیه

در این بخش به مفاهیم اولیه مربوط به انواع الگوها و توضیح مختصری از الگوریتم NSGA-III پرداخته شده است.

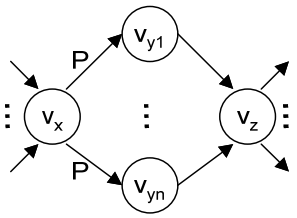
۲-۱ انواع الگوهای ساختاری

الگوهای ساختاری می‌توانند به چند دسته توالی، موازی، شرطی و حلقه تقسیم شوند که در ادامه به شرح هر یک از آنها پرداخته می‌شود [۱۲].

1. Availability
2. Response Time
3. Reliability
4. Throughput
5. Latency
6. Compliance
7. Successability
8. Documentation
9. Best Practices



شکل ۱: الگوی توالی.



شکل ۲: الگوی موازی.

توالی

دو گره v_x و v_y را متوالی گویند اگر و فقط اگر گره v_x تنها دارای یک خروجی به گره v_y باشد و ورودی از گره v_y نداشته باشد و گره v_y تنها دارای یک ورودی از گره v_x باشد. این الگو در شکل ۱ نشان داده شده است.

موازی

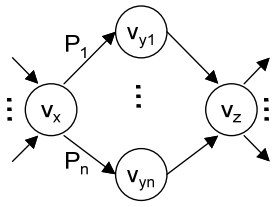
گره‌های v_{y1} تا v_{yn} را موازی گویند اگر و فقط اگر همگی یک ورودی مشترک با گذر احتمال یکسان و حداقل یک گره خروجی مشترک داشته باشند. در صورت داشتن خروجی غیرمشترک، آن خروجی نباید خود گره باشد. این الگو در شکل ۲ نشان داده شده است.

شرطی

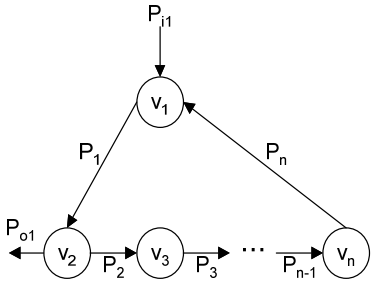
گره‌های v_{y1} تا v_{yn} را شرطی گویند اگر و تنها اگر همگی گره ورودی مشترک با احتمال گذر غیریکسان و یک گره خروجی مشترک داشته باشند. این الگو در شکل ۳ نشان داده شده است.

حلقه

گره‌های v_1 تا v_n تشکیل حلقه می‌دهند اگر و فقط اگر گره اولی با احتمالی به گره دومی، گره دومی با یک احتمالی به گره سومی، ...، گره $n-1$ با احتمالی به گره nام و گره nام با احتمالی به گره اول متصل باشد. این الگو در شکل ۴ نشان داده شده است.



شکل ۳: الگوی شرطی.



شکل ۴: الگوی حلقه.

۲-۲ الگوریتم بهینه‌ساز NSGA-III

چارچوب اصلی الگوریتم NSGA-III با تغییرات قابل توجهی در مکانیسم آن مشابه الگوریتم NSGA-II است [۲۴] که با یک سری نقاط مرجع برای بهبود تنوع و همگرایی NSGA-III معرفی شده است. از آنجا که الگوریتم پیشنهادی بر پایه الگوریتم NSGA-III است، شرحی از این الگوریتم در این بخش داده می‌شود. الگوریتم NSGA-III با یک جمعیت اولیه به اندازه N و یک سری نقاط مرجع Gبعدی با توزیع گسترده شروع می‌شود. P نشان‌دهنده تقسیم است و توسط کاربر تعیین می‌شود. تعداد کل نقاط مرجع (H) در مسئله Mهدفه طبق (۱) به دست می‌آید

$$H = \binom{G+P-1}{P} \qquad (1)$$

الگوریتم NSGA-III از مجموعه‌ای از جهت‌های مرجع برای حفظ تنوع بین راه‌حل‌ها استفاده می‌کند. جهت مرجع پرتویی است که از نقطه مبدأ شروع شده و از نقطه مرجع عبور می‌کند. فرض کنید جمعیت والد در نسل t، P_t نامیده شود و دارای N عضو باشد. جمعیت فرزندان Q_t دارای N عضو با ادغام و جهش جمعیت والد P_t به دست می‌آید. C_t ترکیبی از جمعیت والدین و فرزندان به اندازه $2N$ است. برای حفظ اعضای برتر، اعضای C_t به سطوح مختلف غالب $(L_1, L_2, \ldots, L_l)$ طبقه‌بندی گردیده؛ سپس اعضا از هر سطح غالب برای ساختن جمعیت جدیدی انتخاب می‌شوند که البته از اولین سطح غالب L_1 شروع می‌شود تا به جمعیت با اعضای N برسیم. اگر اعضای موجود در سطح اول که بهترین اعضا هستند از جمعیت N بیشتر باشند، برای انتخاب بهترین‌ها از تنوع استفاده می‌شود. NSGA-II فاصله ازدحامی اعضا را محاسبه می‌کند، اما NSGA-III آن را با روش مبتنی بر جهت مرجع جایگزین می‌کند. ابتدا تعدادی نقطه مرجع انتخاب می‌شود که این نقاط می‌توانند سیستمی انتخاب شوند و یا خود تصمیم‌گیرنده آنها را انتخاب کند. سپس نقطه ایده‌آلی از ترکیب جواب‌های اکسترمم هر تابع هدف تعیین می‌شود. نقطه ایده‌آل تعریف‌شده، جواب‌های یافت‌شده را نرمال می‌کند و هر جواب به نقاط مرجع تخصیص داده می‌شود. چگالی هر نقطه مرجع نسبت به طبقه جواب‌های مختص خود محاسبه و رتبه‌بندی می‌شود که هرچه این چگالی کمتر باشد بهتر است. سپس به تعداد اعضای جمعیت اولیه از جمعیت جدید رتبه‌بندی‌شده انتخاب گردیده و با جمعیت جدید، دوباره جمعیت ثانویه تشکیل شده و ادامه مراحل تا وقتی شرط توقف برقرار گردد، تکرار خواهند شد. قبل از عملیات فوق، اهداف با (۲) تا (۴) نرمال می‌شوند [۲۴]

$$f'_g(x) = f_g(x) - z_g^{\min} \qquad (2)$$

$$ASF(x,w) = \frac{\max_{g=1}^{G} f'_g(x)}{w_g} \qquad (3)$$

$$f_g^n(x) = \frac{f'_g(x)}{a_g - z_g^{\min}} \qquad (4)$$

نقطه ایده‌آل جمعیت S_t با تعریف‌کردن کمترین مقدار $(z_g^{\min})$ برای هر تابع هدف $g = 1,2,\ldots,G$ و با ساختن نقطه ایده‌آل $z = (z_1^{\min}, z_2^{\min}, \ldots, z_G^{\min})$ مشخص می‌شود. $f'_g(x)$ نشان‌دهنده توابع هدف ترجمه‌شده $(x \in S_t)$ است. $ASF(x,w)$ نشان‌دهنده مقادیر نقاط اکسترمم در هر محور هدف و w_g بردار وزن هر هدف است (وقتی $w_g = 0$، با عدد کوچک 10^{-6} جایگزین می‌شود). $f_g^n(x)$ توابع هدف نرمال‌شده و a_g حائل اندیس هدف gام است.

پس از عملیات نرمال‌سازی، نقاط مرجع اصلی محاسبه‌شده روی این ابرصفحه نرمال‌شده قرار می‌گیرند. به منظور مرتبط‌کردن هر عضو جمعیت با یک نقطه مرجع، کوتاه‌ترین فاصله عمودی بین هر عضو از S_t و هر خط مرجع محاسبه می‌شود. نهایتاً یک استراتژی حفظ نیچه برای انتخاب افراد از سطح lام که با هر نقطه مرجع مرتبط هستند، استفاده می‌شود. α_h نشان‌دهنده تعداد اعضای جمعیت از $P_{t+1} = S_t / L_l$ متصل به hامین نقطه مرجع است. استراتژی خاص حفظ نیچه به صورت زیر نشان داده شده است:

۱) مجموعه نقاط مرجع $J_{\min} = \{h : \arg\min \alpha_h\}$ با $\min \alpha_h$ مشخص می‌شود. وقتی $|J_{\min}^h| > 1$، یک $h \in J_{\min}$ به طور تصادفی انتخاب می‌شود.

۲) با توجه به مقدار α_h، دو مورد بحث می‌شود

$$\alpha_h = 0 \quad \text{(i)}$$

الف) اگر یک یا چند عضو در جلوی lامین سطح با نقطه مرجع hام مرتبط شوند، نقطه‌ای که کوتاه‌ترین فاصله عمودی از hامین خط مرجع را دارد به P_{t+1} اضافه می‌شود. تعداد α_h نیز یک واحد اضافه می‌شود.

ب) اگر یک عضو در سطح lام به نقطه مرجع h مرتبط باشد، نقطه مرجع h برای نسل فعلی نادیده گرفته می‌شود

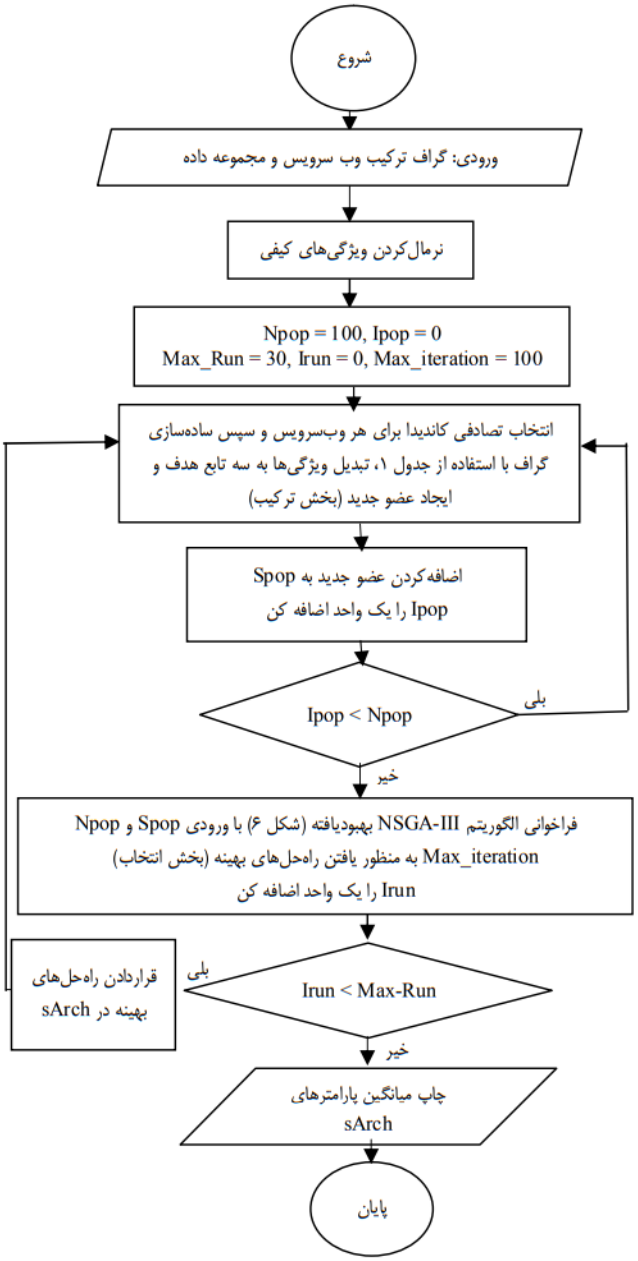
$$\alpha_h \geq 1 \quad \text{(ii)}$$

یک عضو به طور تصادفی انتخاب‌شده از سطح lام که با نقطه مرجع h مرتبط است به P_{t+1} اضافه می‌شود و سپس α_h یک واحد افزایش می‌یابد. پس از به‌روزرسانی شمارش، روش فوق برای پرکردن کل P_{t+1} تکرار می‌شود.

۳- مروری بر کارهای گذشته

دنیلا کلارو و همکاران (۲۰۰۵)، الگوریتم ژنتیک مرتب‌سازی نامغلوب چندهدفه[1] را به منظور یافتن راه‌حل‌های بهینه با در نظر گرفتن قیود محلی و کلی به کار برده‌اند. پارامترهای کیفی مورد استفاده در ترکیب وب‌سرویس‌ها زمان پاسخ، دسترس‌پذیری و اعتبار[2] است. آنها در این مقاله تکامل مدل خود را بر اساس تعداد راه‌حل‌های نامغلوب نشان داده‌اند [۱۳]. لی و همکاران (۲۰۱۰) از الگوریتم ژنتیک مرتب‌سازی نامغلوب به منظور یافتن راه‌حل‌های بهینه در ترکیب وب‌سرویس‌ها با در نظر گرفتن ۱۰ تابع هدف (پارامتر) استفاده کرده‌اند و با گزارش تعداد راه‌حل‌های غالب و مغلوب به ارزیابی الگوریتم مورد نظر پرداخته‌اند [۱۴]. یوجی یاو و همکاران (۲۰۱۰) به مقایسه الگوریتم ژنتیک مرتب‌سازی نامغلوب چندهدفه با اعمال قوانین[3] و بدون قاعده و قانون پرداخته‌اند. در الگوریتم ژنتیک بر اساس قاعده، راه‌حل‌های بهینه مرتب می‌شوند و ده تا از بهترین جواب‌های منطبق بر قیود در نظر گرفته می‌شوند و بر اساس این تعداد عضو، الگوریتم ادامه می‌یابد. نتایج نشان می‌دهند که این الگوریتم در مقایسه با الگوریتم ژنتیک بی‌قاعده منجر به یافتن راه‌حل‌های بهینه‌ای می‌شود که رضایت بیشتر کاربر را به همراه دارد [۱۵]. خیام هشمی و همکاران (۲۰۱۳)، الگوریتم ژنتیک مرتب‌سازی نامغلوب چندهدفه را ارائه داده‌اند که علاوه بر در نظر گرفتن معیارهای کیفی به صورت پیش‌فرض، از مذاکره نیز به منظور بهبود ویژگی‌های کیفی بهره برده است. این الگوریتم مقادیر کیفی هر وب‌سرویس را با مقادیر به‌دست‌آمده از تابعی که به مقادیر گذشته و مقادیر جدید وزن می‌دهد، جایگزین می‌کند تا بدین طریق بتوان مقدار برازندگی را بهبود بخشید [۱۶]. یوجی یاو و همکاران (۲۰۰۹) از الگوریتم ژنتیک مرتب‌سازی نامغلوب برای یافتن راه‌حل‌های بهینه در گراف با الگوهای پیچیده استفاده کرده‌اند و آن را با الگوریتم ژنتیک از نظر سرعت همگرایی و تولید جواب‌های بهینه در یک زمان خاص مقایسه کرده‌اند. میانگین انحراف کمتر و در نتیجه سرعت همگرایی کمتر این الگوریتم که سبب انتخاب گسترده‌تر مشتریان می‌شود نشان‌دهنده عملکرد بهتر آن نسبت به ژنتیک است [۱۷]. پروین شریف‌آرا و همکاران (۲۰۱۴) روشی ارائه داده‌اند که این روش علاوه بر رسیدن به راه‌حل بهینه، سرعت اجرای بالایی نیز دارد. در روش ارائه‌شده ابتدا مسئله به یک مسئله تک‌هدفه تبدیل می‌شود. در این حالت به منظور افزایش سرعت، دقت و قابلیت اطمینان ترکیبی از الگوریتم ژنتیک و روش فازی به کار گرفته شده است. سپس مسئله به یک مسئله چندهدفه تبدیل شده که با الگوریتم ژنتیک مرتب‌سازی نامغلوب جدیدی حل شده است. این الگوریتم هر ویژگی در هر عضو از جمعیت را مرتب می‌کند و به آن ویژگی یک اندیس می‌دهد و در پایان، اندیس‌های به‌دست‌آمده از تمام ویژگی‌های یک عضو را با یکدیگر جمع می‌کند و اعضای جمعیت را بر اساس مقدار آن اندیس مرتب می‌کند و به این ترتیب پیچیدگی زمانی الگوریتم را کم می‌کند [۱۸]. لی لیو و همکاران (۲۰۱۵)، ابتدا دو الگوریتم چندهدفه مرتب‌سازی نامغلوب و بهینه‌سازی ازدحام ذرات را مقایسه و با اندازه‌گیری شاخص‌های عملکردی، برتری الگوریتم مرتب‌سازی نامغلوب را ثابت کرده‌اند. سپس الگوریتم مرتب‌سازی نامغلوب و بهینه‌سازی ازدحام ذرات را با روش فرایند تحلیل سلسله‌مراتبی[4] ادغام کرده‌اند. در این الگوریتم‌ها از فرایند تحلیل سلسله‌مراتبی به جای فاصله ازدحامی استفاده شده و با الگوریتم‌های چندهدفه مرتب‌سازی نامغلوب و بهینه‌سازی ازدحام ذرات که در انتها روی جواب‌هایشان از تحلیل سلسله‌مراتبی استفاده شده است، مقایسه گردیده است. نتایج نشان داده که این رویکرد قادر است به طور دقیق‌تر ترجیحات کاربران را در نظر بگیرد [۱۹]. اکوروت و همکاران (۲۰۱۷) روشی را بر اساس ترکیب الگوریتم تکاملی NSGA-III با جستجوی تابو پیشنهاد داده‌اند و عملکرد آن را از نظر پارامترهای هزینه و دسترس‌پذیری و زمان اجرا با سایر الگوریتم‌ها مقایسه کرده‌اند. نتایج شبیه‌سازی نشان داده که اگرچه هزینه در الگوریتم پیشنهادی نسبت به سایر الگوریتم‌ها بالاست، ولی از لحاظ دو پارامتر دیگر یعنی زمان اجرا و نرخ رد[5] (عدم دسترس‌پذیری) بسیار خوب عمل می‌کند [۲۳]. لی و همکاران (۲۰۱۸) به منظور شبیه‌سازی یک محیط میدان جنگ واقعی، مدل جدید سه‌هدفه شامل به حداکثر رساندن خسارت مورد انتظار دشمن، به حداقل رساندن هزینه موشک و حداکثرکردن ارزش جنگنده را ارائه داده‌اند و برای یافتن مقادیر بهینه این اهداف، الگوریتم NSGA-III بهبودیافته پیشنهاد شده که به طور پیوسته یک سری نقاط مرجع با عملکرد خوب در همگرایی و توزیع با توجه به جمعیت فعلی برای هدایت تکامل تولید می‌شوند و نقاط مرجع بی‌فایده حذف می‌گردند. نتایج شبیه‌سازی نشان می‌دهد که الگوریتم NSGA-III بهبودیافته، راه‌حل‌های پرتو بهتری نسبت به سه الگوریتم دیگر ارائه می‌دهد و همچنین از نظر عملکرد زمانی مناسب‌تر است [۲۴]. سنگاراج و همکاران (۲۰۲۱)، الگوریتم

1. Multi Objective Non-Dominated Sorting Genetic Algorithm
2. Reputation
3. Rule Based NSGA2
4. Analytical Hierarchy Process
5. Rejection Rate



شکل ۵: جریان کار کلی روش پیشنهادی.

ترکیبی ژنتیک و جستجوی تابو[1] را برای یافتن بهترین کاندیدا بر اساس پارامترهای زمان پاسخ، هزینه و قابلیت اطمینان ارائه کرده‌اند. در این روش با ارزیابی مقدار آستانه، بهترین مجموعه کاندیدای قابل همکاری با حداکثر قابلیت اطمینان و توان عملیاتی با استفاده از جستجوی تابو به کاربر نهایی پیشنهاد می‌شود. آزمایش‌ها نشان می‌دهند که روش پیشنهادی به طور متوسط ۰٫۵٪ بهبود در برازندگی و حدود ۰٫۲۵٪ کاهش خطا داشته است [۲۵]. داهان و همکاران (۲۰۲۱)، یک الگوریتم ترکیبی را معرفی کردند که ترکیبی از زنبور عسل و ژنتیک است. الگوریتم ژنتیک برای تنظیم خودکار پارامترهای زنبور عسل استفاده می‌شود و این الگوریتم عملکرد خود را بر اساس تنظیم پارامترها تطبیق می‌دهد. نتایج تجربی نشان می‌دهد که روش پیشنهادی در مقایسه با سایر روش‌ها اگرچه به زمان بیشتری نیاز دارد، اما از نظر هزینه، زمان پاسخ، قابلیت اطمینان و در دسترس بودن بهتر است [۲۶]. اصغری و همکاران (۲۰۲۲) یک مدل مفهومی مبتنی بر اینترنت اشیا را برای حل مسئله ترکیب وب‌سرویس‌ها ارائه داده‌اند و برای یافتن راه‌حل‌های نزدیک به بهینه در این مدل از الگوریتم تکاملی SFLAGA که از ترکیب دو الگوریتم جهش قورباغه درهم‌ریخته (SFLA) و ژنتیک (GA) تشکیل شده است، بهره برده‌اند. در این تحقیق ۹ ویژگی کیفی زمان پاسخ، دسترس‌پذیری، قابلیت اطمینان، بازدهی، تأخیر، نرخ موفقیت، کاربرد، بهترین عملکرد و مستندات در نظر گرفته شده است. روش پیشنهادی بر اساس برازندگی و میزان شباهت با سه الگوریتم دیگر مقایسه شده است. نتایج تجربی نشان‌دهنده برتری الگوریتم پیشنهادی نسبت به الگوریتم‌های ژنتیک، فرهنگ و جهش قورباغه است [۱۲]. سنتشکومار و همکاران (۲۰۲۳)، روشی جدید را به‌جای استفاده از مرتب‌سازی نامغلوب در الگوریتم NSGA-III ارائه دادند. در این روش به‌جای استفاده از روش رایج برای ایجاد بردارهای مرجع، از روشی مبتنی بر انرژی استفاده شده که اجازه می‌دهد تا هر اندازه از جمعیت برای هر بعد هدف استفاده شود. نتایج شبیه‌سازی روی مسائل دو تا ده هدفه، نشان‌دهنده عملکرد بهتر الگوریتم NSGA-III بهبودیافته در مقایسه با الگوریتم NSGA-III است [۲۷]. همچنین نظیف و همکاران (۲۰۲۳) از ترکیب الگوریتم بهینه‌سازی ازدحام ذرات (PSO)[2] و منطق فازی به منظور پیداکردن ترکیب نزدیک به بهینه سرویس‌ها در محیط ابری بهره بردند. نتایج این تحقیق نشان داد که استفاده از توابع فازی باعث بهبود قابل توجهی در الگوریتم PSO می‌شود و از افتادن در نقاط بهینه محلی و یا کاهش سرعت همگرایی این الگوریتم جلوگیری می‌کند [۲۸]. برهمی و همکاران (۲۰۲۳) برای ترکیب بهینه سرویس‌ها در محیط ابری، دو معیار QoS و اعتماد را در نظر گرفتند. در این مقاله برای جلوگیری از افشای اطلاعات حساس، سرویس‌هایی با اعتماد متقابل بالا به همراه QoS خوب انتخاب می‌شوند. برای این منظور یک الگوریتم اکتشافی جدید به نام CWS-SMA ارائه شد و توانست ترکیب بهینه سرویس‌ها را مبتنی بر QoS و اعتماد بر اساس سیستم مختصات ریاضی محاسبه نماید و زمان پاسخ را بهبود دهد [۲۹].

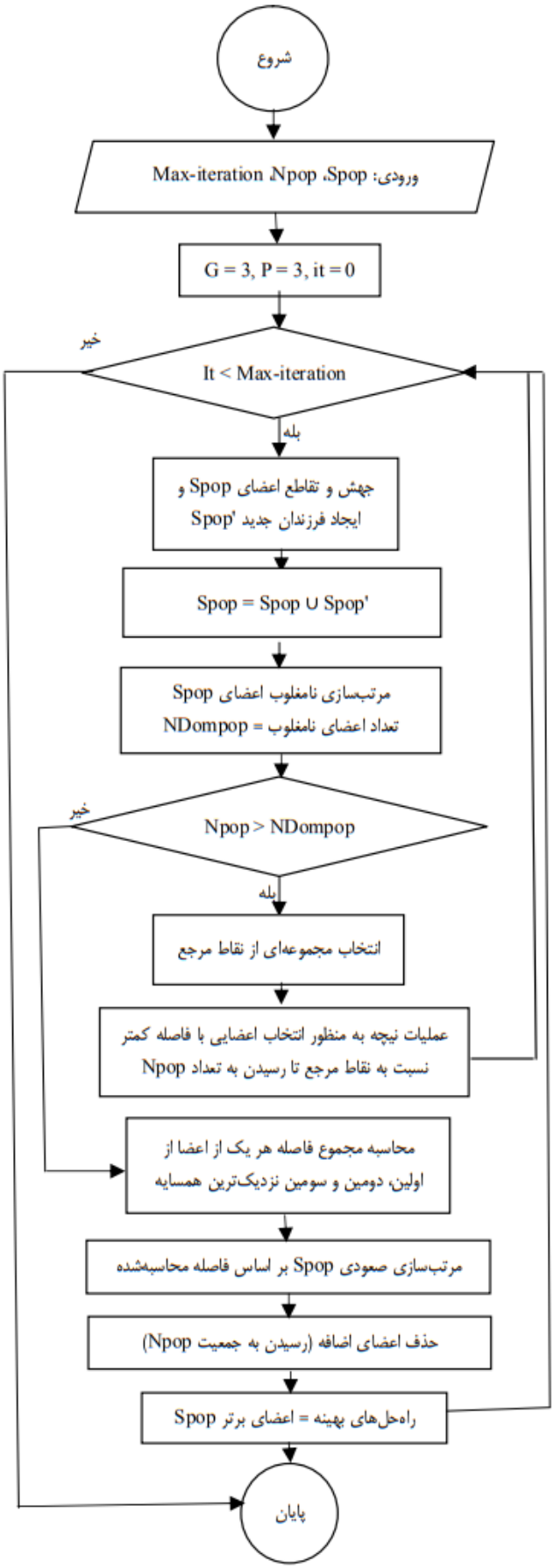
٤- روش پیشنهادی

روش مورد استفاده در این مقاله به دو بخش ترکیب و انتخاب تقسیم می‌شود که در شکل ۵ آمده است. در بخش ترکیب با توجه به الگوهای ارائه‌شده در بخش ۲-۲ و روابط جدول ۱ به ساده‌سازی گراف پرداخته می‌شود. در بخش انتخاب از الگوریتم NSGA-III بهبودیافته (شکل ۶) برای انتخاب بهینه وب‌سرویس‌ها استفاده شده است. متغیرهای Ipop، Npop، Irun، Max_Run، Spop، Max_iteration و sArch به ترتیب شمارشگر تعداد اعضای جمعیت Spop، شمارشگر تعداد دفعات اجرای الگوریتم، بیشینه دفعات اجرای الگوریتم، مجموعه اعضای اولیه الگوریتم، بیشینه تعداد تکرار الگوریتم برای رسیدن به راه‌حل بهینه و آرشیوی برای حفظ راه‌حل‌های بهینه هستند. در این بخش ابتدا نحوه نگاشت الگوریتم NSGA-III به مسئله ترکیب وب‌سرویس به طور خلاصه گفته شده و سپس با در نظر گرفتن الگوریتم NSGA-III در بخش ۲-۲، چگونگی بهبود الگوریتم NSGA-III در شکل ۶ آمده است.

هر عضو از جمعیت در این الگوریتم یک راه‌حل و هر راه‌حل در مسئله ترکیب وب‌سرویس برداری با d بعد است که d تعداد سرویس‌های انتزاعی (تعداد گره‌های گراف) می‌باشد. مقدار هر بعد اندیس کاندیدای منتخب برای آن وب‌سرویس است. اعضای اولیه جمعیت همان طور که در

1. Tabu Search

2. Particle Swarm Optimization



شکل ۶ جریان کاری الگوریتم NSGA-III بهبودیافته.

مقدمه اشاره شد با جایگزینی تصادفی کاندیدا برای هر وب‌سرویس انتزاعی و سپس ساده‌کردن گراف ترکیب به دست می‌آید. از آنجا که هر کاندیدا در این مقاله، حاوی ۹ ویژگی است که در جدول ۱ آمده است، بنابراین هر یک از اعضای جمعیت که پس از ساده‌سازی گراف حاصل می‌شوند نیز دارای ۹ ویژگی کیفی هستند. در جریان کاری کلی الگوریتم (شکل ۵) ابتدا هر یک از ۹ ویژگی موجود در دیتاست با هدف بیشینه‌سازی و همچنین قرارگرفتن در یک مقیاس، نرمال‌سازی شده‌اند (با استفاده از (۵) و (۶)). پس از آن برای هر سرویس انتزاعی یک کاندیدای تصادفی انتخاب و پس از ساده‌سازی ترکیب گراف (با استفاده از جدول ۱) یک عضو از جمعیت ایجاد می‌شود. قبل از فراخوانی الگوریتم NSGA-III بهبودیافته به منظور افزایش کارایی الگوریتم مورد نظر در تعیین اعضای نامغلوب، ۹ ویژگی کیفی که از خلاصه‌سازی گراف برای هر عضو از جمعیت جدید به دست آمده، به ۳ هدف اصلی تبدیل می‌شوند؛ به گونه‌ای که اهداف اول، دوم و سوم به ترتیب مجموع وزن‌دار ویژگی‌های زمان پاسخ، تأخیر و دسترس‌پذیری، بازدهی، موفقیت، قابلیت اطمینان و انطباق، بهترین عملکرد و مستندسازی هستند. در مرحله بعد، الگوریتم بهبودیافته NSGA-III فراخوانی می‌شود. در این الگوریتم (شکل ۶) تا زمانی که تعداد اعضای نامغلوب جمعیت به اندازه Npop نیستند، از عملیات نیچه به منظور انتخاب اعضا استفاده می‌شود که تا اینجا الگوریتم اصلی NSGA-III اجرا می‌شود. بهبود این الگوریتم برای زمانی است که تعداد اعضای نامغلوب از Npop بزرگ‌تر شوند. در این صورت برای حذف اعضای اضافی، مجموع فاصله هر یک از اعضا از اولین، دومین و سومین نزدیک‌ترین همسایه محاسبه شده و اعضای جمعیت (Spop) بر اساس این فاصله به صورت نزولی مرتب شده و اعضای اضافه (اعضایی با فاصله کمتر) حذف می‌گردند و مابقی اعضا، راه‌حل‌های بهینه در نظر گرفته می‌شوند.

به منظور نرمال‌سازی ویژگی‌های کیفی منفی (زمان پاسخ و تأخیر) که کاهش آنها مطلوب‌تر است، از (۵) و برای نرمال‌سازی ویژگی‌های کیفی مثبت (دسترس‌پذیری، نرخ موفقیت و ...) که افزایش آنها مطلوب‌تر است از (۶) استفاده شده است. q مقدار ویژگی کیفی و k تعداد ویژگی‌های کیفی مورد استفاده در مسئله می‌باشد؛ بنابراین $k = ۱,۲,۳,\ldots,۹$ است

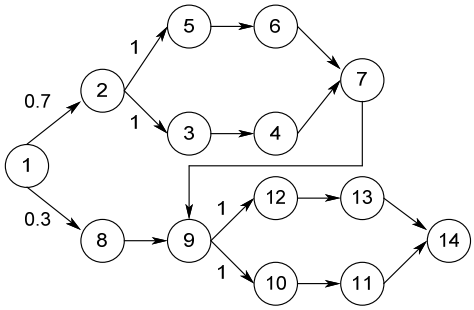
$$\begin{cases} \dfrac{q_k^{\max} - q_k}{q_k^{\max} - q_k^{\min}} & , \ q_k^{\max} - q_k^{\min} \neq ۰ \\ ۱ & , \ \text{otherwise} \end{cases} \quad (۵)$$

$$\begin{cases} \dfrac{q_k - q_k^{\min}}{q_k^{\max} - q_k^{\min}} & , \ q_k^{\max} - q_k^{\min} \neq ۰ \\ ۱ & , \ \text{otherwise} \end{cases} \quad (۶)$$

۵- آزمایش‌ها و نتایج

محیط پیاده‌سازی، سیستمی با پردازنده Corei۵، ۴ گیگابایت RAM و سیستم عامل ویندوز ۱۰ می‌باشد و نیز برای پیاده‌سازی الگوریتم‌ها از MATLAB ۲۰۱۶ استفاده شده است. به منظور ارزیابی روش پیشنهادی از گراف گردش کار BPMN[1] (شکل ۷) استفاده شده [۱۲] که فرایند خرید کالا را با استفاده از الگوهای ترکیب وب‌سرویس‌ها نشان می‌دهد. این گردش کار نمونه‌ای از یک مدل فرایند کسب و کار است که توسط ۱۴ نوع وظیفه انتزاعی (۱، ۲، ۳، ... ، ۱۴) ساخته شده است. این گراف، ارتباط سرویس‌ها را در یک مدل ترکیبی نشان می‌دهد. برای هر وظیفه انتزاعی، کاندیدایی در نظر گرفته شده است. به عنوان مثال برای وظیفه

1. Business Process Model and Notation



شکل ۷: گراف ترکیب وب سرویس با ۱۴ گره.

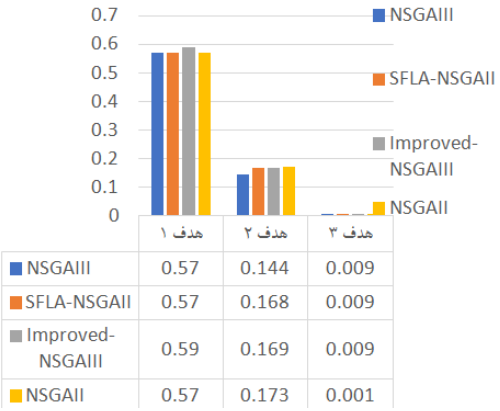
جدول ۲: مقادیر پارامترهای مورد استفاده در الگوریتم‌ها.

پارامتر	مقدار
اندازه جمعیت اولیه	۱۰۰
تعداد Memplex (الگوریتم SFLA-NSGAII)	۵
اندازه هر Memplex (الگوریتم SFLA-NSGAII)	۲۰
اندازه آرشیو	۱۰۰
بیشترین تکرار الگوریتم‌ها	۱۰۰
تعداد دفعات اجرای الگوریتم‌ها	۳۰
احتمال جهش الگوریتم‌ها	۰٫۲
احتمال تقاطع الگوریتم‌ها	۰٫۸

۱، m کاندیدا به صورت $(ca_{1,1}, ca_{1,2}, \ldots, ca_{1,m})$ در نظر گرفته می‌شود که کاندیداها دارای ویژگی‌های کیفی متفاوتی هستند. در این مثال هر گره بدین شرح است: گره ۱: چک‌کردن موجودی انبار، گره ۲: چک‌کردن موجودی موارد خام، گره ۳: درخواست مواد خام از تهیه‌کننده ۲، گره ۴: به دست آوردن مواد خام از تهیه‌کننده ۲، گره ۵: درخواست مواد خام از تهیه‌کننده ۱، گره ۶: به دست آوردن مواد خام از تهیه‌کننده ۱، گره ۷: تولید محصول در کارخانه، گره ۸: بازیابی محصول از انبار، گره ۹: تأیید سفارش، گره ۱۰: فرستادن فاکتور، گره ۱۱: دریافت پول، گره ۱۲: دریافت آدرس، گره ۱۳: فرستادن محصول و گره ۱۴: ذخیره‌سازی سفارش و ارسال آن.

در این گراف پس از دریافت سفارش خرید و چک‌کردن موجودی انبار، دو حالت (الگوی شرطی) در نظر گرفته شده است. در صورت عدم وجود کالا در انبار (احتمال ۰٫۷) به وظیفه ۲ و در غیر این صورت (احتمال ۰٫۳) به وظیفه ۸ می‌رسد. اگر به وظیفه ۲ رسید بر اساس این که مواد خام توسط تهیه‌کننده ۱ یا ۲ تولید شود، به وظایف ۵ و سپس ۶ (الگوی توالی) و یا ۳ و سپس ۴ رسیده است. در صورت موجودبودن کالا در انبار و رفتن به وظیفه ۸ و سپس ۹ به منظور تأیید کالا، وظایف ۱۲ و ۱۳ (گرفتن آدرس و سپس فرستادن کالا) و وظایف ۱۰ و ۱۱ (ارسال فاکتور و پرداخت پول) به صورت همزمان انجام خواهند شد (الگوی موازی).

برای هر وظیفه در گراف ترکیب وب‌سرویس‌ها، کاندیدایی با استفاده از مجموعه داده QWS[2] [1] انتخاب می‌شود. این مجموعه داده شامل ۲۵۰۷ وب‌سرویس کاندیدا به‌همراه مقادیر ویژگی‌های کیفی این کاندیداست و در طول سال ۲۰۰۸ با استفاده از چارچوب کارگزار وب‌سرویس[3] اندازه‌گیری شده است. بنابراین مجموعه داده شامل ۲۵۰۷ ردیف است و ۹ پارامتر اول در هر ردیف مربوط به ویژگی‌های کیفی ذکرشده در جدول ۱ است که



	هدف ۱	هدف ۲	هدف ۳
NSGAIII	0.57	0.144	0.009
SFLA-NSGAII	0.57	0.168	0.009
Improved-NSGAIII	0.59	0.169	0.009
NSGAII	0.57	0.173	0.001

شکل ۸: مقایسه الگوریتم‌ها بر اساس میانگین اهداف.

مقادیر این پارامترها با کاما از یکدیگر جدا شده‌اند و دو پارامتر آخر نشان‌دهنده نام سرویس و ارجاع به سند WSDL[4] است. پس از تشخیص و شناسایی انواع الگوها در گراف و انتساب هر کاندیدا به یک وظیفه مشخص با استفاده از روش تجمعی به ساده‌سازی گراف می‌پردازیم. در این روش پس از جایگذاری کاندیدای منتخب هر وب‌سرویس، الگوی توالی، موازی، شرطی و حلقه به ترتیب چک و در صورت وجود ساده می‌شوند. برای رسیدن به این هدف از جدول ۱ استفاده شده است. پس از ساده‌سازی کل گراف، یک گره باقی می‌ماند. این گره باقیمانده که گره خلاصه‌سازی‌شده گراف است، یک عضو از جمعیت الگوریتم‌های مورد استفاده است. گره خلاصه دارای ۹ ویژگی کیفی است و بر اساس مطالب گفته‌شده در بخش ۴-۱ به سه پارامتر هدف تبدیل شده است.

پس از تعیین جمعیت اولیه، از الگوریتم NSGA-III بهبودیافته برای انتخاب بهینه ترکیب گراف استفاده شده است (شکل ۶). به منظور ارزیابی الگوریتم پیشنهادی، الگوریتم‌های مطرح NSGA-III، NSGA-II و SFLA-NSGAII که ورژن چندهدفه الگوریتم پیشنهادی در [۱۲] است پیشنهاد شده‌اند. مقادیر پارامترهای مورد استفاده این الگوریتم‌ها در جدول ۲ آمده‌اند. پس از ۳۰ بار اجرای الگوریتم‌ها و ۱۰۰ تکرار به ازای هر اجرا، الگوریتم‌ها بر اساس میانگین اهداف و شاخص‌های ارزیابی نرخ پوشش‌دهی، پراکندگی، توزیع و زمان اجرا ارزیابی می‌شوند.

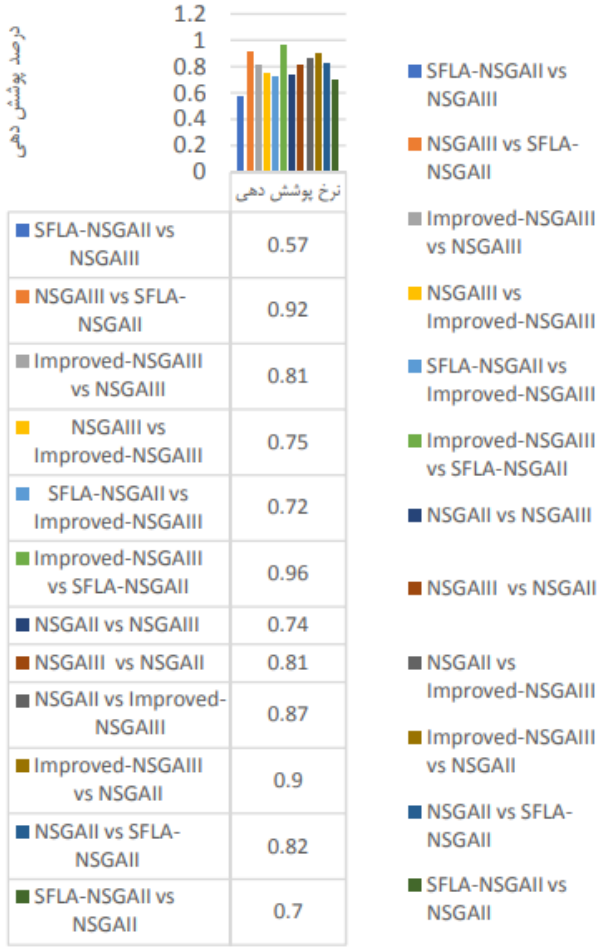
تجزیه و تحلیل نتایج بر اساس اهداف

با استفاده از روش پیشنهادی، جبهه پارتو (مجموعه‌ای از راه‌حل‌های نزدیک به بهینه) با تعداد اعضای Npop (شکل ۶) تولید می‌شود. بنابراین جبهه پارتو حاوی Npop راه‌حل بوده که هر راه‌حل حاوی سه هدف است. برای تعیین کیفیت اعضای یک جبهه پارتو، میانگین هر یک از اهداف در ۳۰ بار اجرا به دست آمده است. شکل ۸ نتایج مقایسه الگوریتم پیشنهادی با ۳ الگوریتم ذکرشده را بر اساس میانگین اهداف نشان می‌دهد. الگوریتم NSGA-III بهبودیافته از نظر هدف اول بهتر از سه الگوریتم دیگر است. از نظر هدف دوم از همه به‌جز الگوریتم NSGA-II بهتر است و از لحاظ هدف سوم با دو الگوریتم NSGA-III و SFLA-NSGAII برابری می‌کند و از الگوریتم NSGA-II بهتر است.

تجزیه و تحلیل نتایج بر اساس شاخص‌های عملکرد

نرخ پوشش‌دهی[5]: این اندیکاتور[6] یکی از مهم‌ترین شاخص‌های مقایسه جبهه‌های پرتو بین دو الگوریتم است و نشان‌دهنده آن است که

1. Dataset
2. https://qwsdata.github.io/qws2.html
3. Web Service Broker
4. Web Service Description Language
5. Coverage Ratio
6. Indicator



SFLA-NSGAII vs NSGAIII	0.57
NSGAIII vs SFLA-NSGAII	0.92
Improved-NSGAIII vs NSGAIII	0.81
NSGAIII vs Improved-NSGAIII	0.75
SFLA-NSGAII vs Improved-NSGAIII	0.72
Improved-NSGAIII vs SFLA-NSGAII	0.96
NSGAII vs NSGAIII	0.74
NSGAIII vs NSGAII	0.81
NSGAII vs Improved-NSGAIII	0.87
Improved-NSGAIII vs NSGAII	0.9
NSGAII vs SFLA-NSGAII	0.82
SFLA-NSGAII vs NSGAII	0.7

شکل ۹: مقایسه الگوریتم‌ها بر اساس نرخ پوشش‌دهی.

چه درصدی از راه‌حل‌های الگوریتم دوم، مغلوب راه‌حل‌های الگوریتم اول شده‌اند.

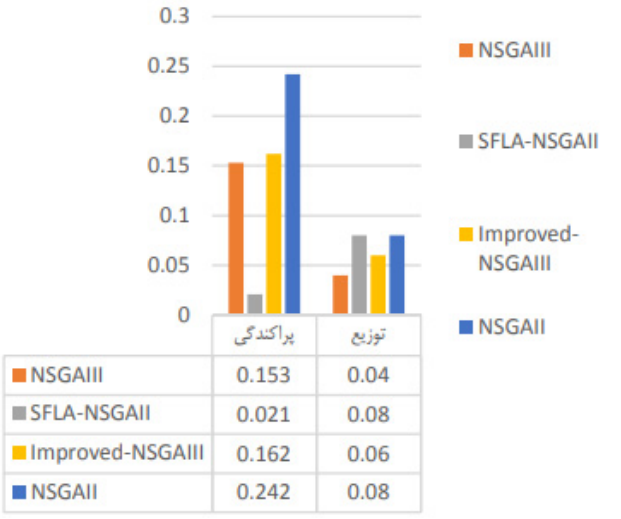
شکل ۹ نرخ پوشش‌دهی هر دو الگوریتم را نسبت به یکدیگر نشان می‌دهد. الگوریتم پیشنهادی نسبت به الگوریتم‌های NSGA-III، NSGA-II و SFLA-NSGAII به ترتیب ۶ درصد، ۳ درصد و ۲۴ درصد نرخ پوشش‌دهی بیشتری دارد؛ یعنی به طور میانگین ۱۱ درصد بیشتر راه‌حل‌های جبهه پرتو این سه الگوریتم را مغلوب می‌کند.

توزیع‌شدگی[1]: این شاخص میانگین فاصله راه‌حل‌های به‌دست‌آمده در جبهه پارتو را از یکدیگر نشان می‌دهد. هرچه میانگین فاصله راه‌حل‌ها از یکدیگر کمتر باشد نشان‌دهنده توزیع یکسان راه‌حل‌هاست. بر اساس نتایج ارائه‌شده در شکل ۱۰، الگوریتم پیشنهادی پس از الگوریتم NSGA-III بهترین عملکرد را دارد.

بیشترین پراکندگی[2]: این شاخص فاصله بین راه‌حل‌های مرزی را نشان می‌دهد و مقدار بیشتر این شاخص، نشان‌دهنده پوشش بهتر راه‌حل‌های مرزی است. شکل ۱۰ علاوه بر توزیع‌شدگی، مقدار این شاخص را نیز برای الگوریتم‌ها نشان می‌دهد. شکل ۱۰ نشان می‌دهد که الگوریتم NSGA-II بهترین مقدار پراکندگی را در بین الگوریتم‌ها دارد و پس از آن الگوریتم پیشنهادی قرار دارد.

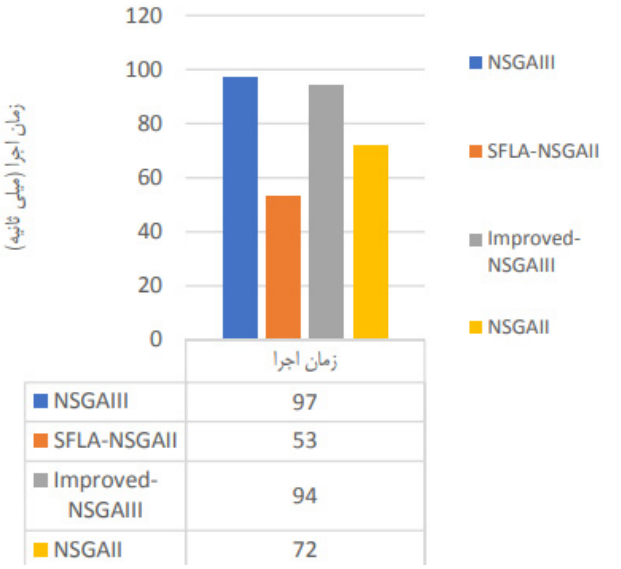
زمان اجرا[3]: زمان اجرای الگوریتم، زمان تولید جواب‌های نزدیک به بهینه (جبهه پرتو) را توسط الگوریتم نشان می‌دهد (شکل ۱۱). الگوریتم

1. Distance Based Distribution
2. Maximum Spread
3. Execution Time



	پراکندگی	توزیع
NSGAIII	0.153	0.04
SFLA-NSGAII	0.021	0.08
Improved-NSGAIII	0.162	0.06
NSGAII	0.242	0.08

شکل ۱۰: مقایسه الگوریتم‌ها بر اساس توزیع‌شدگی و پراکندگی.



	زمان اجرا
NSGAIII	97
SFLA-NSGAII	53
Improved-NSGAIII	94
NSGAII	72

شکل ۱۱: مقایسه الگوریتم‌ها بر اساس زمان اجرا.

NSGA-III بهبودیافته هرچند از نظر کیفیت بهتر از سایر الگوریتم‌ها است و دقتی بالاتر در انتخاب راه‌حل‌های جبهه پرتو دارد، اما زمان اجرای بیشتری نسبت به هر دو الگوریتم NSGA-II و SFLA-NSGAII دارد.

۶- نتیجه‌گیری

مسئله ترکیب وب‌سرویس‌ها، یک حوزه تحقیقاتی است که ترکیب چندین سرویس مستقل را به یک سرویس باکیفیت بهینه برای برآوردن درخواست‌های پیچیده مشتری در نظر می‌گیرد. برای حل این دسته از مسائل که جزء مسائل NP-hard است، از الگوریتم‌های تکاملی استفاده شده است. اکثر این الگوریتم‌ها بر تبدیل چند ویژگی به یک تابع تک‌هدفه متمرکز شده‌اند و این مسئله از دیدگاه چندهدفه به‌ندرت مورد توجه قرار گرفته است. در این مقاله، مسئله ترکیب وب‌سرویس با یک گراف گردش کار تعریف شده است. همچنین کیفیت کاندیدای هر وب‌سرویس به همراه روابط موجود بین وب‌سرویس‌ها در هر الگو و برای هر ویژگی کیفی به منظور ساده‌سازی الگو آمده است. این مقاله ابتدا به ساده‌سازی گراف ترکیب وب‌سرویس پرداخته و پس از آن با نگاشت مسئله به الگوریتم تکاملی NSGA-III، مجموعه راه‌حل‌های پارتو را به دست آورده و نهایتاً با اضافه‌کردن تابعی به الگوریتم NSGA-III به‌منظور تنوع بیشتر

راه‌حل‌ها و دستیابی به کیفیت بهتر آن را بهبود داده است. برای ارزیابی کارایی الگوریتم NSGA-III بهبودیافته از مجموعه داده‌های شناخته‌شده QWS استفاده گردیده و با الگوریتم‌های NSGA-III، NSGA-II و SFLA-NSGAII مقایسه شده است. آزمایش‌ها ۳۰ بار اجرا شده که نشان‌دهنده برتری الگوریتم پیشنهادی در حداقل دو هدف از ۳ هدف نسبت به سه الگوریتم دیگر است. همچنین در شاخص عملکردی نرخ پوشش‌دهی به‌طور میانگین ۱۱ درصد بهبود حاصل شده و در شاخص‌های توزیع و پراکندگی راه‌حل‌ها نیز نسبتاً خوب عمل کرده است.

مراجع

[1] D. Prajapati and K. Bhargavi, "Old-age health risk prediction and maintenance via IoT devices and artificial neural network," in *Proc. of the 6th Int. Conference on FICTA*, pp. 373-381 Bhubaneswar, India, 14-16 Oct. 2017.

[2] Y. Wu, W. Jin, J. Ren, and Z. Sun, "A multi-perspective architecture for high-speed train fault diagnosis based on variational mode decomposition and enhanced multi-scale structure," *Applied Intelligence*, vol. 49, no. 11, pp. 3923-3937, 2019.

[3] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Service composition approaches in IoT: a systematic review," *J. of Network and Computer Applications*, vol. 120, pp. 61-77, Oct. 2018.

[4] N. Kashyap, A. C. Kumari, and R. Chhikara, "Multi-objective optimization using NSGA II for service composition in IoT," *Procedia Computer Science*, vol. 167, pp. 1928-1933, 2020.

[5] A. C. Kumari, K. Srinivas, and M. P. Gupta, "Multi-objective test suite minimisation using quantum-inspired multi-objective differential evolution algorithm," in *Proc. IEEE Int. Conf. on Computational Intelligence and Computing Research*, 7 pp., Coimbatore, India, 18-20 Dec. 2012.

[6] M. E. Khanouche, Y. Amirat, A. Chibani, M. Kerkar, and A. Yachir, "Energy-centered and QoS-aware services selection for Internet of Things," *IEEE Trans. on Automation Science and Engineering*, vol. 13, no. 3, pp. 1256-1269, Jul. 2016.

[7] Q. Li, R. Dou, F. Chen, and G. Nan, "A QoS-oriented web service composition approach based on multi-population genetic algorithm for Internet of Things," *International J. of Computational Intelligence Systems*, vol. 7, no. sup. 2, pp. 26-34, Jul. 2014.

[8] A. Souri, A. M. Rahmani, N. J. Navimipour, and R. Rezaei, "Formal modeling and verification of a service composition approach in the social customer relationship management system," *Information Technology & People*, vol. 32, no. 6, pp. 1591-1607, Nov. 2019.

[9] L. J. Zhang, J. Zhang, and H. Cai, "Service-oriented architecture," *In: Services Computing,* pp. 89-113, Springer, Berlin, Heidelberg, 2007.

[10] A. Strunk, "QoS-aware service composition: a survey," in *Proc. 8th IEEE European Conf. on Web Services*, pp. 67-74, Ayia Napa, Cyprus, 1-3 Dec. 2010.

[11] Z. Brahmi and M. M. Gammoudi, "QoS-aware automatic web service composition based on cooperative agents," in *Proc. Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pp. 27-32, Hammamet, Tunisia, 17-20 Jun. 2013.

[12] P. Asghari, A. M. Rahmani, and H. H. S. Javadi, "Privacy-aware cloud service composition based on QoS optimization in Internet of Things," *J. of Ambient Intelligence and Humanized Computing*, vol. 13, no. 11, pp. 5295-5320, 2022.

[13] D. B. Claro, P. Albers, and J. K. Hao, "Selecting web services for optimal composition," in *Proc. Second Int. Workshop on Semantic and Dynamic Web Processes*, 14 pp., Orlando, FL, USA, 11-11 Jul. 2005.

[14] L. Li, P. Yang, L. Ou, Z. Zhang, and P. Cheng, "Genetic algorithm-based multi-objective optimisation for QoS-aware web services composition," in *Proc. 4th Int. Conf. on Knowledge Science, Engineering and Management*, pp. 549-554, Belfast, Northern Ireland, UK, 1-3 Sept. 2010.

[15] Y. Yao and H. Chen, "A rule-based web service composition approach," in *Proc. 6th Int. Conf. on Autonomic and Autonomous Systems*, pp. 150-155, Cancun, Mexico, 7-13 Mar. 2010.

[16] K. Hashmi, A. Alhosban, E. Najmi, and Z. Malik, "Automated web service quality component negotiation using NSGA-2," in *Proc. ACS Int. Conf. on Computer Systems and Applications,* 6 pp., Ifrane, Morocco, 27-30 May 2013.

[17] Y. Yao and H. Chen, "QoS-aware service composition using NSGA-II_1," in *Proc. of the 2nd Int. Conf. on Interaction Sciences: Information Technology, Culture and Human*, pp. 358-363, Seoul, Korea, 24-26, Nov. 2009.

[18] P. Sharifara, A. Yari, and M. M. R. Kashani, "An evolutionary algorithmic based web service composition with quality of service," in *Proc. 7th Int. Symp. on Telecommunications*, pp. 61-65, Tehran, Iran, 9-11 Sept. 2014.

[19] L. Liu and M. Zhang, "Multi-objective optimization model with AHP decision-making for cloud service composition," *KSII Trans. on Internet and Information Systems*, vol. 9, no. 9, pp. 3293-3311, Sept. 2015.

[20] L. B. Said, S. Bechikh, and K. Ghédira, "The r-dominance: a new dominance relation for interactive evolutionary multicriteria decision making," *IEEE Trans. on Evolutionary Computation*, vol. 14, no. 5, pp. 801-818, Oct. 2010.

[21] J. Molina, L. V. Santana, A. G. Hernández-Díaz, C. A. C. Coello, and R. Caballero, "g-dominance: reference point based dominance for multiobjective metaheuristics," *European J. of Operational Research*, vol. 197, no. 2, pp. 685-692, Sept. 2009.

[22] J. H. Zheng and Z. Z. Xie, "A study on how to use angle information to include decision maker's preferences," *Acta Electonica Sinica*, vol. 42, no. 11, pp. 2239-2246, 2014.

[23] T. Ecarot, D. Zeghlache, and C. Brandily, "Consumer-and-provider-oriented efficient IaaS resource allocation," in *Proc. IEEE Int. Parallel and Distributed Processing Symp. Workshops*, pp. 77-85, Lake Buena Vista, FL, USA. 29 May- 2 Jun. 2017.

[24] Y. Li, Y. Kou, and Z. Li, "An improved nondominated sorting genetic algorithm III method for solving multiobjective weapon-target assignment part i: the value of fighter combat," *International J. of Aerospace Engineering*, Article ID: 8302324, 23 pp., 2018.

[25] P. Thangaraj and P. Balasubramanie, "Meta heuristic QoS based service composition for service computing," *J. of Ambient Intelligence and Humanized Computing*, vol. 12, no. 5, pp. 5619-5625, May 2021.

[26] F. Dahan, W. Binsaeedan, M. Altaf, M. S. Al-Asaly, and M. M. Hassan, "An efficient hybrid metaheuristic algorithm for QoS-aware cloud service composition problem," *IEEE Access*, vol. 9, pp. 95208-95217, 2021.

[27] B. Santoshkumar, K. Deb, and L. Chen, "Eliminating non-dominated sorting from NSGA-III," in *Proc. 12th Int. Conf. on Evolutionary Multi-Criterion Optimization*, pp. 71-85, Leiden, the Netherlands, Mar. 2023, 20-24 Mar. 2023.

[28] H. Nazif, M. Nassr, H. M. R. Al-Khafaji, N. Jafari Navimipour, and M. Unal, "A cloud service composition method using a fuzzy-based particle swarm optimization algorithm," *Multimedia Tools and Applications*, vol. 83, no. 19, pp. 1-28, Dec. 2023.

[29] Z. Brahmi and A. Selmi, "Coordinate system-based trust-aware web services composition in edge and cloud environment," *The Computer J.*, vol. 66, no. 9, pp. 2102-2117, Sept. 2023.

نرجس ظهیری در سال ۱۳۹۳ مدرک کارشناسی مهندسی نرم‌افزار کامپیوتر خود را از دانشگاه صنعتی اصفهان و در سال ۱۳۹۸ مدرک کارشناسی ارشد مهندسی نرم‌افزار خود را از دانشگاه کاشان دریافت نمود. در سال ۱۴۰۰ به عنوان دانشجوی استعداد درخشان در دانشگاه کاشان پذیرش شد و اکنون نیز دانشجوی دکترای مهندسی نرم‌افزار در همان دانشگاه هستند. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: مهندسی نرم‌افزار، محاسبات تکاملی، سیستم‌های توزیع‌شده، محاسبات ابری و مه.

فرشته دهقانی در سال ۱۳۸۶ مدرک کارشناسی مهندسی کامپیوتر خود را از دانشگاه کاشان، در سال ۱۳۸۹ مدرک کارشناسی ارشد و در سال ۱۳۹۸ مدرک دکتری مهندسی کامپیوتر خود را از دانشگاه اصفهان دریافت نمود. ایشان هم‌اکنون استادیار دانشکده مهندسی برق و کامپیوتر گرایش هوش مصنوعی دانشگاه کاشان می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان یادگیری تقویتی، یادگیری عمیق و بهینه‌سازی می‌باشد.

سلمان گلی بیدگلی در سال ۱۳۸۵ مدرک کارشناسی مهندسی کامپیوتر خود را از دانشگاه کاشان، در سال ۱۳۸۹ مدرک کارشناسی ارشد و در سال ۱۳۹۶ مدرک دکتری مهندسی کامپیوتر خود را از دانشگاه اصفهان دریافت نمود. ایشان هم‌اکنون استادیار دانشکده مهندسی برق و کامپیوتر دانشگاه کاشان می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: شبکه‌های کامپیوتری، اینترنت اشیا و بلاک چین.