

# بهینه‌سازی انتخاب و ترکیب وب‌سرویس‌ها بر اساس ویژگی‌های کیفی با در نظر گرفتن وابستگی، ناسازگاری و همبستگی بین وب‌سرویس‌ها

مهدی فرزندی و فریدون شمس علیئی

یکی از فناوری‌های نوظهور در این زمینه استفاده از وب‌سرویس‌ها است، بنابراین سازمان‌ها برای کاهش هزینه‌های مطابقت خود با نیازمندی‌های متغیر مشتریان از وب‌سرویس‌ها استفاده می‌کنند و به این منظور وب‌سرویس‌هایی با عملکرد مورد نظر خود را شناسایی و انتخاب می‌کنند. با گسترش اینترنت و تمایل سازمان‌ها به استفاده از وب‌سرویس‌ها، به مرور زمان تعداد وب‌سرویس‌هایی با عملکرد یکسان و ویژگی‌های کیفی<sup>۱</sup> (QoS) متفاوت افزایش یافت و مسئله انتخاب وب‌سرویس با بهترین ویژگی کیفی برای سازمان‌ها با اهمیت شد. از طرفی دیگر وب‌سرویس‌های موجود تنها برای یک عملکرد ساخته می‌شوند و تک‌منظوره هستند، بنابراین سازمان‌ها نمی‌توانند برای درخواست‌های پیچیده و چندمنظوره مشتریان که نیازمند ترکیب مناسبی از چندین وب‌سرویس هستند، از یک وب‌سرویس که تک‌منظوره است استفاده کنند. بنابراین مسئله انتخاب و ترکیب وب‌سرویس‌ها بر اساس ویژگی‌های کیفی آنها برای سازمان‌ها به منظور سریع‌تر پاسخ‌دادن به نیازمندی‌های پیچیده مشتریان بسیار حایز اهمیت شد.

شکل‌های مختلفی از ترکیب وب‌سرویس‌ها از جمله تریبی، موازی، شرطی و حلقوی وجود دارد. روابط مختلفی بین وب‌سرویس‌ها و ویژگی‌های کیفی وب‌سرویس‌ها وجود دارد، از جمله وابستگی و ناسازگاری بین وب‌سرویس‌ها و همبستگی بین ویژگی‌های کیفی وب‌سرویس‌ها که با در نظر گرفتن این روابط می‌توانیم وب‌سرویس‌های مناسب‌تری را برای ترکیب بیابیم. تاکنون روشی که تمام این روابط بین وب‌سرویس‌ها را در شکل‌های مختلف ترکیب وب‌سرویس‌ها اعمال کند و وب‌سرویس مرکب مناسبی را به دست آورد، ارائه نشده است. هدف ما در این مقاله ارائه روشی برای حل این مسئله است. روش ارائه‌شده در این مقاله بر اساس الگوریتم ژنتیک می‌باشد که از این الگوریتم تاکنون برای حل مسایل بهینه‌سازی مختلفی استفاده شده و در بیشتر موارد نتایج نسبتاً خوبی به همراه داشته است. در این روش برای پیدا کردن بهترین وب‌سرویس مرکب، حالت‌های مختلف ترکیب وب‌سرویس‌های کاندید بر اساس مجموع ویژگی‌های کیفی و روابط بین آنها از جمله وابستگی، ناسازگاری و همبستگی بررسی می‌شوند.

بخش‌های بعدی مقاله به صورت زیر سازمان‌دهی شده‌اند: در بخش دوم کارهای پیشین و مرتبط، در بخش سوم مبانی مسئله روش پیشنهادی، در بخش چهارم روش پیشنهادی برای بررسی وابستگی، ناسازگاری و همبستگی بین وب‌سرویس‌ها ارائه می‌شود. سپس در بخش پنجم به نتایج پیاده‌سازی و ارزیابی روش پیشنهادی و در بخش ششم به نتیجه‌گیری نهایی می‌پردازیم و در پایان در بخش هفتم اشاره‌ای به کارهای آینده می‌کنیم.

چکیده: امروزه تغییرات مداوم در نیازمندی‌های مشتریان به عنوان اصلی‌ترین چالش پیش روی سازمان‌ها است، معماری سرویس‌گرا به عنوان یک راه‌حل عملی برای رفع این مشکل برای سازمان‌های سرویس‌گرا مطرح می‌شود. در معماری سرویس‌گرا انتخاب و ترکیب سرویس‌ها برای پاسخ‌گویی سریع به نیازمندی‌های پیچیده مشتریان در دسترس سازمان‌های سرویس‌گرا قرار می‌گیرد. سازمان‌ها برای پاسخ‌گویی سریع‌تر به نیازمندی‌های پیچیده و متغیر مشتریان از سرویس‌های آماده و برون‌سازمانی استفاده می‌کنند که یکی از فناوری‌های نوظهور در این زمینه وب‌سرویس‌ها هستند. با گسترش تمایل سازمان‌ها به استفاده از وب‌سرویس‌ها، به مرور زمان تأمین‌کنندگان وب‌سرویس‌ها افزایش پیدا کردند و به همین دلیل وب‌سرویس‌هایی با عملکرد یکسان و ویژگی‌های کیفی متفاوت گسترش یافتند، بنابراین مسئله انتخاب وب‌سرویس با بهترین ویژگی کیفی برای سازمان‌ها اهمیت پیدا کرد. از طرفی سازمان‌ها تنها با یک وب‌سرویس نمی‌توانند نیازمندی‌های پیچیده مشتریان را پاسخ دهند، به همین دلیل نیازمند ترکیب چندین وب‌سرویس با هم هستند. از طرفی دیگر با افزایش وب‌سرویس‌ها با عملکردهای متفاوت، در ترکیب آنها، همبستگی، وابستگی و ناسازگاری بین وب‌سرویس‌ها نیز گسترش می‌یابد ولی تاکنون روشی ارائه نشده که وب‌سرویس‌های برتر را بر اساس ویژگی‌های کیفی انتخاب کند و ترکیب آنها با هم، وابستگی، ناسازگاری و همبستگی بین وب‌سرویس‌ها را نقض نکند. در این مقاله سعی می‌کنیم از روش‌های قبلی که به وابستگی یا ناسازگاری یا همبستگی در حالت‌های ساده ترکیب وب‌سرویس‌ها پرداخته‌اند، استفاده کنیم و یک روش جامع پیشنهاد دهیم تا این که حالت‌های پیچیده‌ای که از ترکیب وب‌سرویس‌ها ممکن است رخ دهد را نیز پشتیبانی کنیم و وب‌سرویس مرکب مناسب را از نظر ویژگی‌های کیفی با در نظر گرفتن وابستگی، ناسازگاری و همبستگی بیابیم.

کلیدواژه: الگوریتم ژنتیک، انتخاب وب‌سرویس، ترکیب وب‌سرویس، ناسازگاری، همبستگی، وابستگی، وب‌سرویس، ویژگی کیفی.

## ۱- مقدمه

سازمان‌ها تمام فرایندهای جریان کاری<sup>۱</sup> خود را توسط سرویس‌های درون‌سازمانی برآورده نمی‌سازند، زیرا مطابقت با درخواست‌ها و نیازمندی‌های پیچیده مشتریان و تغییرات زیاد محیط اطراف بسیار سخت و هزینه‌بر است. به همین دلیل سازمان‌ها قسمتی از جریان کاری درون‌سازمانی خود را توسط سرویس‌های برون‌سازمانی برآورده می‌کنند.

این مقاله در تاریخ ۲۴ بهمن ماه ۱۳۹۶ دریافت و در تاریخ ۲۱ مهر ماه ۱۳۹۷ بازنگری شد.

مهدی فرزندی، دانشکده مهندسی و علوم کامپیوتر دانشگاه شهید بهشتی، تهران، ایران، (email: mahdifarzandway@gmail.com).

فریدون شمس علیئی (نویسنده مسئول)، دانشکده علوم و مهندسی کامپیوتر، دانشگاه شهید بهشتی، تهران، ایران، (email: f\_shams@sbu.ac.ir).

(ویژگی کیفی) از مسایل غیر چندجمله‌ای - سخت است [۳] و [۴].

فرض کنید مشتری علاقه‌مند به استفاده از وب‌سرویس‌های جستجو، یافتن و خرید بلیط هواپیمای مناسب از مبدأ به مقصد، جستجو، یافتن و رزرو هتل مناسب در مقصد و پرداخت هزینه به صورت ترکیبی است. برای این منظور سازمان بایستی ابتدا تعدادی وب‌سرویس مناسب با عملکردهای مورد نظر مشتری کشف کند، سپس وب‌سرویس مناسبی از نظر ویژگی کیفی برای هر نیازمندی مجزای مشتری انتخاب کند، در انتها این وب‌سرویس‌ها را به صورت وب‌سرویس مرکب مناسبی که هر سه نیاز مشتری را به صورت یک جا برآورده سازد، با هم ترکیب کند و در اختیار مشتری قرار دهد. در شکل ۱ مثالی از کشف، انتخاب و ترکیب وب‌سرویس‌ها نمایش داده شده است.

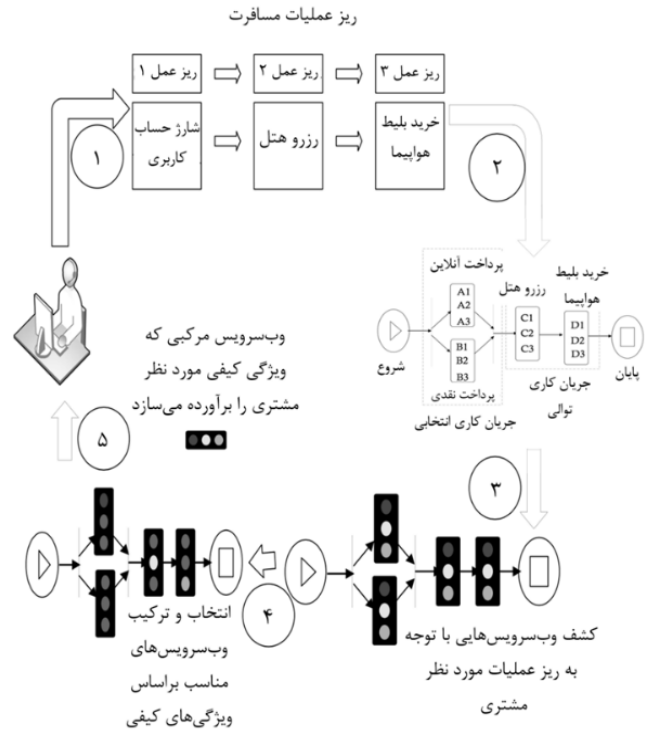
هرچه تعداد فرایندهای موجود در جریان کار و وب‌سرویس‌های کاندید برای تک‌تک فرایندها بیشتر باشند، تعداد حالات ممکن ترکیب وب‌سرویس‌ها برای انجام آن جریان کاری بیشتر خواهد شد. اگر تعداد فرایندهای موجود در جریان کاری برابر  $m$  و تعداد وب‌سرویس‌های کاندید برای هر فرایند برابر  $n$  باشد، آن گاه تعداد حالات ممکن ترکیب وب‌سرویس‌ها برای انجام آن جریان کاری برابر با  $n^m$  خواهد شد. به همین دلیل انتخاب وب‌سرویس‌های مناسبی که عملکردهای مورد نظر مشتری را فراهم کنند و یافتن ترکیب مناسبی از وب‌سرویس‌ها که علاوه بر برآورده‌سازی تمامی عملکردهای مورد نظر مشتری از مجموع ویژگی کیفی مناسبی برخوردار باشد، بدون این که تمامی حالات ممکن ترکیب را بررسی کنیم، تبدیل به یکی از مهم‌ترین مباحث در ترکیب وب‌سرویس‌ها شده است [۶].

## ۲-۳ ویژگی‌های کیفی وب‌سرویس

گام شناسایی وب‌سرویس‌ها یکی از مهم‌ترین و ضروری‌ترین گام‌ها پیش از انتخاب و ترکیب وب‌سرویس‌ها به شمار می‌آید. باید به نحوی از کیفیت خروجی‌های این گام که همان وب‌سرویس‌های کشف‌شده در این گام هستند، شناخت پیدا کرد و برای این منظور از ویژگی‌های کیفی برای سنجش کیفیت وب‌سرویس‌های شناسایی و کشف‌شده، استفاده می‌کنیم. ویژگی‌های کیفی وب‌سرویس‌ها به دو دسته ویژگی‌های کیفی طراحی و ویژگی‌های کیفی پیاده‌سازی و اجرا تقسیم‌بندی می‌شوند. لازم به ذکر است در مراحل کشف، انتخاب و ترکیب وب‌سرویس‌ها، مبنای برتری وب‌سرویس نسبت به وب‌سرویس‌های دیگر ویژگی‌های کیفی سطح پیاده‌سازی و اجرای وب‌سرویس‌ها هستند. از جمله این ویژگی‌های کیفی می‌توان به قابلیت اطمینان، دسترس‌پذیری، هزینه اجرایی، تعامل‌پذیری، امنیت، زمان پاسخ و قابلیت استفاده اشاره کرد.

## ۲-۴ ویژگی‌های کیفی مثبت و منفی و مصالحه آنها

ویژگی‌های کیفی وب‌سرویس‌ها به دو دسته کلی ویژگی‌های کیفی مثبت و منفی تقسیم می‌شوند، ویژگی‌های کیفی منفی به ویژگی‌هایی مانند زمان پاسخ و هزینه گفته می‌شود که هرچه میزان آنها بیشتر باشد، وب‌سرویس ارزش کمتری برای انتخاب شدن دارد. در صورتی که ویژگی‌هایی مانند قابلیت استفاده، تعامل‌پذیری، دسترس‌پذیری، امنیت و قابلیت اطمینان را ویژگی‌های کیفی مثبت می‌نامند، زیرا میزان این ویژگی‌ها هرچه بیشتر باشد وب‌سرویس، کیفیت و ارزش بیشتری برای انتخاب شدن دارد. برای استاندارد کردن ویژگی‌های کیفی منفی ( $QoS_{neg}$ )



شکل ۱: فرایند کشف، انتخاب و ترکیب وب‌سرویس‌ها بر اساس ویژگی‌های کیفی [۵].

## ۲- کارهای پیشین

برای یافتن ترکیب مناسبی از وب‌سرویس‌ها نیازمند گذر از دو مرحله هستیم. این دو مرحله عبارتند از کشف و انتخاب وب‌سرویس‌های مناسب و یافتن ترکیب مناسبی از وب‌سرویس‌ها بر اساس ویژگی‌های کیفی آنها.

### ۱-۲ کشف و انتخاب وب‌سرویس

در کشف و انتخاب وب‌سرویس‌ها، ابتدا بایستی وب‌سرویس‌های مورد نظر مشتری را از نظر ویژگی کارکردی کشف کنیم. برای این منظور از مخزنی<sup>۱</sup> (UDDI) متشکل از تمام وب‌سرویس‌هایی که در اینترنت قرار دارد استفاده می‌کنیم که این مخزن امکان جستجوی وب‌سرویس را برای ما فراهم می‌آورد [۱].

از میان وب‌سرویس‌های کشف‌شده، وب‌سرویس‌های دارای ویژگی کیفی مناسب‌تر را انتخاب کرده و در مجموعه وب‌سرویس‌های ممکن برای بررسی در یافتن ترکیب مناسبی از وب‌سرویس‌ها قرار می‌دهیم [۲].

### ۲-۲ ترکیب وب‌سرویس‌ها

وب‌سرویس‌ها هر کدام کارکرد خاصی را پوشش می‌دهند، گاهی نیاز است برای پوشش کارکرد پیچیده‌ای با ویژگی‌های غیر کارکردی (کیفی) مناسب، دو یا چند وب‌سرویس با هم ترکیب شوند. در ترکیب وب‌سرویس‌های آگاه از کیفیت خدمات (ویژگی کیفی)، از مجموعه وب‌سرویس‌های ممکن، وب‌سرویس‌های مناسبی را انتخاب و با هم ترکیب می‌کنیم تا این که بتوانیم وب‌سرویس مرکب مناسبی را ارائه دهیم. این وب‌سرویس مرکب باید کارکرد و ویژگی‌های غیر کارکردی (کیفی) مورد نظر مشتری را برآورده سازد و لازم به ذکر است که بهینه‌سازی در انتخاب و ترکیب وب‌سرویس‌های آگاه از کیفیت خدمات

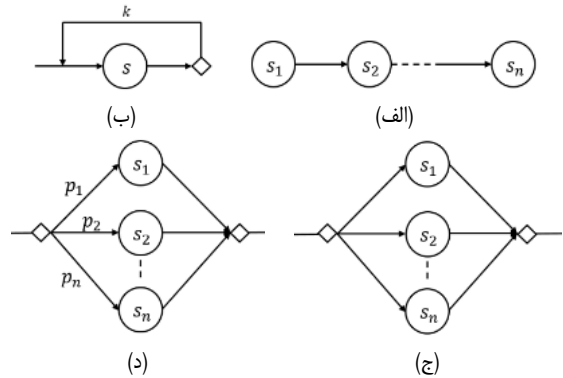
توابع محاسبه ویژگی کیفی و سرویس مرکب نهایی مطابق با شکل‌های مختلف ترکیبی در جدول ۱ آورده شده است.

## ۷-۲ الگوریتم‌های فراابتکاری

الگوریتم‌های فراابتکاری روش‌هایی هستند که بر پایه الهام‌گرفتن از پدیده‌های طبیعی، زیستی یا اجتماعی کار خود را انجام می‌دهند. مزایایی چون مقیاس‌پذیری، تطابق‌پذیری، قابلیت اطمینان و زمان اجرای قابل قبول باعث شده تا انواع مختلفی از این الگوریتم‌ها از جمله الگوریتم ژنتیک [۹]، کلونی زنبور عسل [۱۰]، تجمع ذرات [۱۱]، رقابت استعماری [۱۲] و ... را تبدیل به روش‌هایی نیرومند در حل مسایل بهینه‌سازی و بالطبع مسئله بهینه‌سازی انتخاب و ترکیب و سرویس‌ها کند.

## ۸-۲ الگوریتم ژنتیک

الگوریتم ژنتیک (شکل ۳) یکی از انواع الگوریتم‌های فراابتکاری است که تصادفی و هوشمند بوده و بر مبنای یک نظریه زیست‌شناسی به نام داروین بنیان‌گذاری شده است. این الگوریتم از مفاهیم وراثت، جهش، انتخاب ناگهانی و طبیعی و ترکیب ژن‌ها<sup>۱</sup> در علم زیست‌شناسی الهام گرفته است. الگوریتم با ایجاد مجموعه‌ای کاملاً تصادفی از موجودیت‌ها شروع به کار می‌کند و نسل‌های بعدی نیز با انجام اعمالی تصادفی روی نسل‌های قبلی ایجاد می‌شوند. از هر نسل، بهترین موجودیت انتخاب می‌شود، اگر شرایط مورد نظر مشتری را داشته باشد، آن موجودیت به عنوان جواب مسئله معرفی می‌شود و الگوریتم پایان می‌یابد، در غیر این صورت این موجودیت به نسل بعد منتقل می‌شود و علاوه بر این موجودیت برتر از نسل قبلی، مجموعه‌ای کاملاً تصادفی از موجودیت‌هایی جدید همانند نسل قبلی ایجاد و روند قبلی تا زمان یافتن موجودیتی که شرایط مشتری را داشته باشد ادامه می‌یابد. اگر موجودیتی مناسب پیدا نشود، الگوریتم زمانی که به تعداد تکرارهای مشخص شده برای آن برسد، پایان می‌یابد و بهترین موجودیتی که در این تکرارها یافته است را به عنوان جواب مسئله معرفی می‌کند. برای نمایش هر موجودیت در هر نسل از پارامتری به نام کروموزوم<sup>۲</sup> یا ژنوم<sup>۳</sup> استفاده می‌کنیم که هر کروموزوم از تعدادی ژن تشکیل شده است. برای ایجاد جمعیت نسل‌های دوم به بعد، از عملگرهای ژنتیکی روی نسل‌های قبلی استفاده می‌کنیم. انتخاب تصادفی ژن‌ها و ترکیب آنها با هم و ایجاد کروموزوم‌هایی جدید به عنوان فردی در جمعیت نسل جدید یکی از عملگرها است که به آن تقاطع<sup>۴</sup> یا بازترکیبی<sup>۵</sup> می‌گویند و انتخاب تصادفی ژن‌هایی از کروموزوم و جهش آنها، عملگر ژنتیکی دیگری است که به آن جهش<sup>۶</sup> می‌گویند. این عملگرهای ژنتیکی باعث ایجاد نسل جدیدی از کروموزوم‌ها می‌شوند که با نسل قبلی تفاوت دارند. الگوریتم ژنتیک در مسئله ترکیب و سرویس‌ها برای یافتن و سرویس‌های مناسب برای ترکیب با یکدیگر استفاده می‌شود، کروموزوم‌هایی که الگوریتم ژنتیک می‌یابد نشان‌دهنده جریان کاری و سرویس مرکب است و ژن‌های کروموزوم نشان‌دهنده و سرویس‌های کاندیدی هستند که انتخاب شده‌اند و بایستی با هم ترکیب شوند [۱۳]. فرض کنید جریان کاری حاوی  $m$  فرایند و برای هر



شکل ۲: شکل‌های مختلف ترکیب و سرویس‌ها، (الف) ترتیبی، (ب) حلقوی، (ج) موازی و (د) شرطی [۷].

از (۱) و برای استاندارد کردن ویژگی‌های کیفی مثبت  $(QoS_{pos})$  از (۲) استفاده می‌کنیم. در اصل ویژگی کیفی  $(QoS)$  که از (۱) و (۲) به دست می‌آید، مصالحه‌ای بین ویژگی‌های کیفی مثبت و منفی و مثبت برقرار کرده و در ادامه مقاله در محاسبه مجموع ویژگی‌های کیفی ترکیب و سرویس‌ها از مصالحه ایجاد شده بین ویژگی‌های کیفی مثبت و منفی (ویژگی کیفی استاندارد شده  $(QoS)$ ) استفاده می‌کنیم و دیگر دغدغه منفی یا مثبت بودن ویژگی‌های کیفی را نداریم [۶].

در (۱) و (۲)، بیشینه و کمینه هر نوع ویژگی کیفی از اعداد مربوط به آن نوع ویژگی کیفی انتخاب می‌گردند و مثبت یا منفی بودن ویژگی کیفی در انتخاب اعداد در نظر گرفته نمی‌شوند و کمترین یا بیشترین مقدار عددی از بین اعداد ویژگی کیفی، انتخاب و در روابط قرار داده می‌شوند

$$QoS = \frac{QoS_{max} - QoS_{neg}}{QoS_{max} - QoS_{min}} \quad (1)$$

$$QoS = \frac{QoS_{pos} - QoS_{min}}{QoS_{max} - QoS_{min}} \quad (2)$$

## ۵-۲ شکل‌های مختلف ترکیب و سرویس‌ها

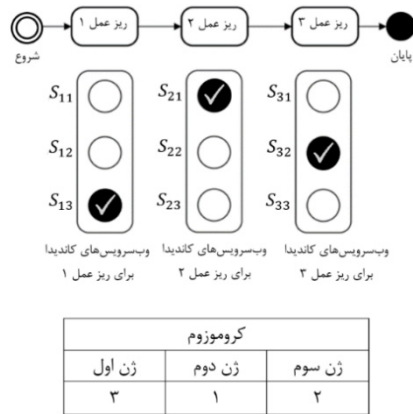
و سرویس‌ها با شکل‌های مختلفی، منطبق بر جریان کاری که نیازمندی کارکردی مشتری را برآورده می‌سازد با یکدیگر ترکیب می‌شوند. شکل‌های مختلف ترکیب و سرویس‌ها شامل ترتیبی، موازی، حلقوی، شرطی و ترکیبی از این موارد در حالات پیچیده است.

- در شکل ۲، شکل‌های مختلف ترکیب و سرویس‌ها آورده شده است.
- در شکل ترتیبی، ابتدا و سرویس  $S_1$ ، سپس و سرویس  $S_2$  و همین طور روند ادامه پیدا می‌کند تا این که در انتها و سرویس  $S_n$  اجرا شده و عملیات پایان یابد.
- در شکل حلقوی، و سرویس  $S_1$  به اندازه  $K$  بار تکرار می‌شود تا این که عملیات مورد نظر به انجام برسد.
- در شکل موازی، و سرویس‌های  $S_1, S_2, \dots, S_n$  به طور هم‌زمان اجرا می‌شوند تا این که عملیات مورد نظر به انجام برسد.
- در شکل شرطی، برای ادامه روند، یکی از و سرویس‌ها بایستی اجرا شود که احتمال ادامه روند با و سرویس  $S_1$  برابر با  $P_1$  و احتمال ادامه روند با  $S_2$  برابر با  $P_2$  و در انتها احتمال ادامه روند با  $S_n$  برابر با  $P_n$  است.

## ۶-۲ محاسبه ویژگی کیفی و سرویس مرکب

حال بایستی روشی برای محاسبه مجموع ویژگی‌های کیفی ترکیب و سرویس‌ها، مطابق با هر کدام از شکل‌های ترکیبی ارائه نماییم.

1. Gene
2. Chromosome
3. Genome
4. Crossover
5. Recombination
6. Mutation



شکل ۴: ساختار کروموزوم [۱۳].

**Algorithm 1** Genetic Algorithm for QoS-aware Web Service Selection and Composition  
**Input:** set of web services , set of quality of them and relation (sequence, parallel, loop,...) between them, best quality of service (bQoS), crossover\_rate, mutation\_rate, population\_size, number\_evolution, maximum\_population\_generation  
**Output:** set of best web services with the best quality of services (QoS)  
 1: population = new population(population\_size)  
 2: generation\_count = 1  
 3: repeat  
 4: population = new population(population\_size - 1) ∪ population.getFittest()  
 5: generation\_count = generation\_count + 1  
 7: crossover(population, crossover\_rate, number\_evolution)  
 9: mutation(population, mutation\_rate, number\_evolution)  
 10: until population.getFittest().getFitness() < bQoS or generation\_count < maximum\_population\_generation  
 11: return population.getFittest()

شکل ۳: الگوریتم ژنتیک.

جدول ۱: توابع محاسبه میزان ویژگی کیفی وبسرویس مرکب نهایی [۸].

ویژگی کیفی	ترتیبی	شرطی	موازی	حلقوی
زمان	$\sum_{i=1}^n T(S_i)$	$\sum_{i=1}^n p_i \times T(S_i)$	$\max\{T(S_i)\}_{i \in \{1, \dots, n\}}$	$K \times T(S)$
هزینه	$\sum_{i=1}^n C(S_i)$	$\sum_{i=1}^n p_i \times C(S_i)$	$\sum_{i=1}^n C(S_i)$	$K \times C(S)$
دسترسی پذیری	$\prod_{i=1}^n A(S_i)$	$\prod_{i=1}^n p_i \times A(S_i)$	$\prod_{i=1}^n A(S_i)$	$A(S)^K$
قابلیت اطمینان	$\prod_{i=1}^n R(S_i)$	$\prod_{i=1}^n p_i \times R(S_i)$	$\prod_{i=1}^n R(S_i)$	$R(S)^K$

\*  $T(S_i)$ : اندازه ویژگی کیفی زمان برای وبسرویس  $S_i$

- انتخاب تصادفی<sup>۱</sup>: به صورت تصادفی تعدادی از کروموزوم‌های جمعیت را به عنوان والدین برای ایجاد جمعیت جدید انتخاب می‌کنیم.  
 - انتخاب نخبگان<sup>۲</sup>: برترین کروموزوم‌های جمعیت را به عنوان والدین برای ایجاد جمعیت جدید انتخاب می‌کنیم. در این مسئله منظور از کروموزوم‌ها، وبسرویس‌های مرکب است و کروموزوم‌های برتر، وبسرویس‌های مرکبی هستند که دارای مجموع ویژگی‌های کیفی بهتری هستند.

### ۲-۹-۲ عملگر تقاطع

در این عملگر دو کروموزوم به عنوان والد انتخاب می‌گردند. سپس به کمک یکی از روش‌هایی که در ادامه توضیح داده می‌شود، ژن‌هایی از دو والد انتخاب می‌گردند که بایستی با یکدیگر جابه‌جا شوند تا این که دو کروموزوم متفاوت نسبت به دو کروموزوم والد ایجاد کنیم.  
 - تقاطع تک‌نقطه‌ای<sup>۳</sup>: در این روش یک نقطه تصادفی انتخاب شده و ژن‌ها از این نقطه به بعد بین دو والد با یکدیگر جابه‌جا می‌شوند (شکل ۶).  
 - تقاطع دونقطه‌ای<sup>۴</sup>: در این روش دو نقطه به صورت تصادفی انتخاب می‌گردند و ژن‌های بین این دو نقطه بین دو والد با یکدیگر جابه‌جا می‌شوند (شکل ۷).  
 - تقاطع چندنقطه‌ای<sup>۵</sup>: در این روش چندین نقطه به صورت تصادفی انتخاب می‌گردند و ژن‌های بین این نقاط بین دو والد با یکدیگر جابه‌جا می‌شوند. در ادامه تصویری از تقاطع چهارنقطه‌ای در شکل ۸ آورده شده است.

فرایند،  $n$  وبسرویس کاندید داریم. بنابراین کروموزوم ما حاوی  $m$  ژن است و هر ژن مقداری بین ۱ تا  $n$  را می‌تواند بپذیرد. به عنوان مثال شکل ۴ یک جریان کاری را نشان می‌دهد که دارای سه فرایند است، بنابراین کروموزوم‌های این مسئله حاوی سه ژن هستند و برای هر فرایند، سه وبسرویس کاندید وجود دارد که تنها یک وبسرویس برای ترکیب انتخاب می‌شود. پس هر ژن این کروموزوم عددی بین یک تا سه را می‌تواند بپذیرد. با توجه به ساختار کروموزوم در شکل ۳ می‌توان نتیجه گرفت که وبسرویس سوم از میان وبسرویس‌های کاندید برای فرایند اول، وبسرویس اول از میان وبسرویس‌های کاندید برای فرایند دوم و وبسرویس دوم از میان وبسرویس‌های کاندید برای فرایند سوم انتخاب شده‌اند (شکل ۵).

### ۲-۹-۲ عملگرهای الگوریتم ژنتیک

در صورتی که از میان کروموزوم‌های موجود، کروموزوم مورد نظر مشتری را نیابیم، اقدام به تولید جمعیت جدیدی از کروموزوم‌ها می‌کنیم. برای این منظور از عملگرهای الگوریتم ژنتیک استفاده می‌نماییم تا جمعیتی برتر تولید کنیم و کروموزوم مورد نظر مشتری را بیابیم.  
 عملگرهای ژنتیک عبارتند از:

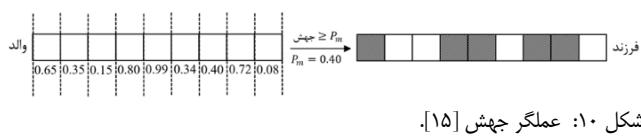
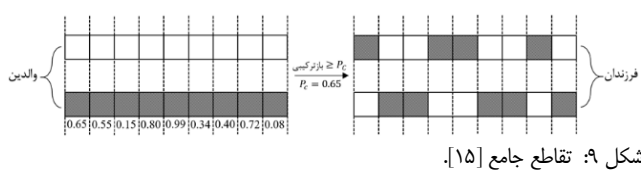
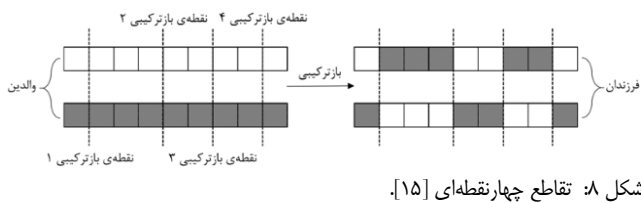
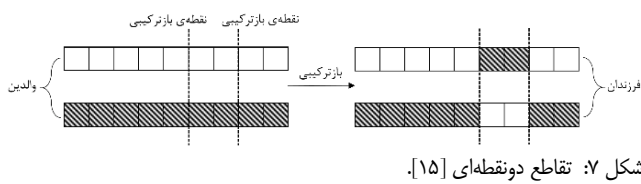
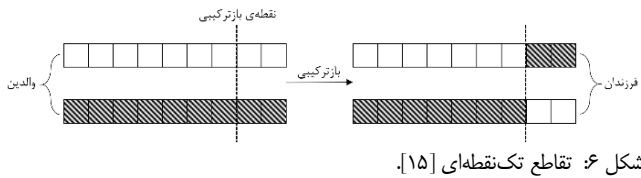
- انتخاب
- تقاطع یا باز ترکیبی
- جهش

و در ادامه به معرفی این عملگرها می‌پردازیم.

### ۲-۹-۱ عملگر انتخاب

در این عملگر کروموزوم‌هایی از جمعیت انتخاب می‌کنیم تا عملگرهای تقاطع و جهش را روی آنها انجام دهیم تا جمعیت جدید دیگری تولید کنیم. برای مسئله انتخاب و ترکیب وبسرویس‌ها معمولاً عملگر انتخاب به یکی از دو روش متفاوت زیر پیاده‌سازی می‌شود.

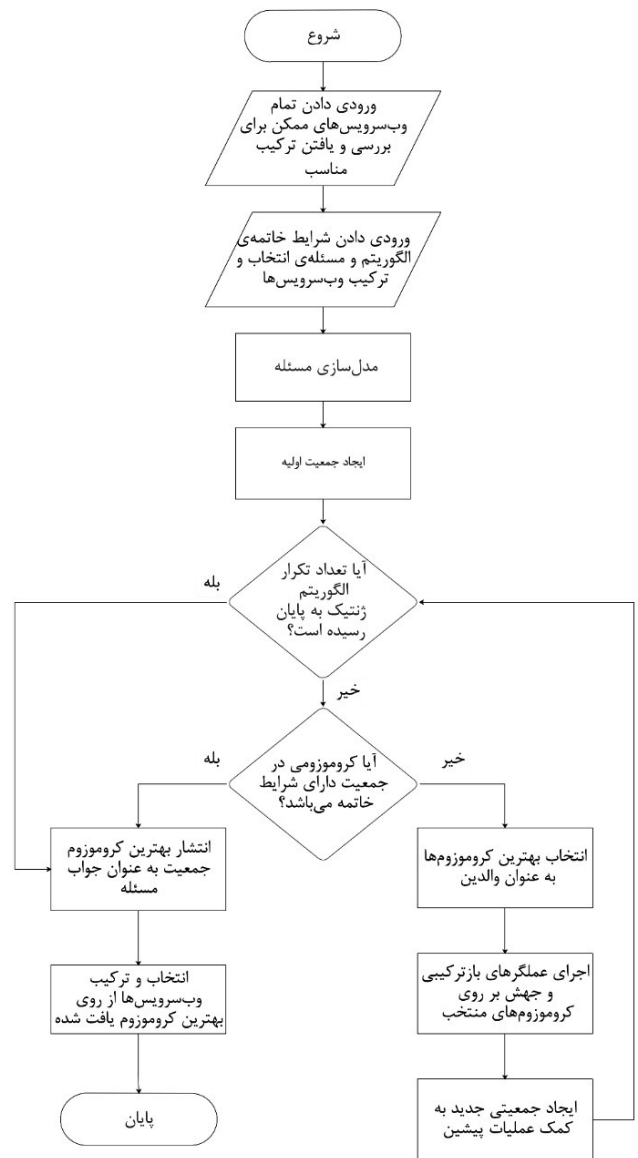
1. Random Selection
2. Elitist Selection
3. Single Point Crossover
4. Two Point Crossover
5. Multipoint Crossover



محلی گرفتار می‌شود که البته همین میزان اندک گرفتار شدن در دام بهینه محلی نیز باعث پایین آمدن کارایی الگوریتم ژنتیک می‌شود. از طرفی خاصیت تصادفی بودن آن منجر شده تا هنگام رسیدن به جواب مناسب، از آن دور شده و در پیدا کردن جواب بهینه دچار مشکلات شود. از این رو برای رفع معایب ذکر شده در طی سال‌های اخیر کارهای مختلفی در زمینه بهبود الگوریتم ژنتیک اولیه صورت گرفته است. از آن موارد می‌توان به ایجاد برخی تغییرات در عملگرها و پارامترهای الگوریتم و یا ترکیب این الگوریتم با سایر روش‌های دیگر برای افزایش کارایی آن اشاره کرد. به عنوان نمونه در [۱۶] این الگوریتم برای حل مسئله ترکیب وب‌سرویس‌ها ارائه شده است. مؤلفان در این مقاله اثر پارامترهای مختلف مثل نرخ جهش و تقاطع را روی توانایی بهینه‌سازی الگوریتم ژنتیک بررسی کرده و نشان داده‌اند که پارامترهای مختلف در کارایی این الگوریتم تأثیرهای متفاوتی را دارند.

در [۱۷] نیز با اشاره به تأثیر انتخاب تابع محاسبه ارزش مجموع ویژگی کیفی و قوانین جهش، خاطرنشان شده که چگونگی انتخاب این پارامترها می‌تواند در بهبود سرعت همگرایی الگوریتم ژنتیک نقش بسزایی داشته باشد. در پژوهشی دیگر از یک بهینه‌ساز محلی برای اصلاح مقادیر ارزش مجموع ویژگی کیفی افراد جمعیت استفاده شده که منجر به اصلاح مقادیر بهینه نهایی می‌شود [۱۸].

کنفرا و همکارانش نیز از این الگوریتم با یک تابع محاسبه ارزش مجموع ویژگی کیفی پویا استفاده کرده‌اند. در این روش ساختار ژنوم آرایه‌ای از اعداد صحیح است که هر مدخل آن به یک سرویس اشاره دارد. عملگر تقاطع در اینجا دونقطه‌ای و عملگر جهش به صورت تصادفی عمل می‌کند [۱۹]. اما در کاری دیگر ساختار ژنوم به صورت آرایه‌ای در نظر



شکل ۵: مراحل اجرایی الگوریتم ژنتیک برای یافتن جواب مسئله انتخاب و ترکیب وب‌سرویس‌ها [۱۴].

تقاطع جامع: اگر تمام ژن‌های کروموزوم را به عنوان نقاط بازترکیبی انتخاب کنیم به آن بازترکیبی جامع گویند. روش کار برای این مورد به این صورت است که با احتمال ثابتی مثل  $P_c$  عمل بازترکیبی را انجام می‌دهیم. برای هر یک از ژن‌های کروموزوم عدد تصادفی بین صفر و یک تولید می‌کنیم. اگر این عدد از مقدار ثابتی مثل  $P_c$  بیشتر بود، آن ژن بین دو والد جابه‌جا می‌شود و در غیر این صورت آن ژن ثابت باقی می‌ماند (شکل ۹).

### ۲-۹-۳ عملگر جهش

در عملگر جهش (شکل ۱۰) برای هر یک از ژن‌های کروموزوم‌ها عددی تصادفی تولید می‌کنیم. اگر این عدد از مقدار ثابت  $P_m$  بیشتر بود در آن ژن، جهش رخ خواهد داد و در غیر این صورت آن ژن بدون تغییر به کروموزوم نسل بعد منتقل می‌شود.

همان طور که اشاره شد الگوریتم ژنتیک به جای جستجو در یک نقطه، مجموعه‌ای از نقاط را جستجو می‌کند و در نتیجه کمتر در دام بهینه

ویژگی‌های کیفی آن کروموزوم از مجموع ویژگی‌های کیفی مورد نظر مشتری بهتر بود آن کروموزوم یا به عبارتی وبسرویس‌های انتخاب شده برای ترکیب را به مشتری اعلام می‌کنیم و الگوریتم پایان می‌یابد. در غیر این صورت الگوریتم با انتخاب کروموزوم‌های برتر و عملگرهای تقاطع و جهش جمعیت جدیدی را تولید می‌کند و تمام عملیات پیشین برای یافتن کروموزوم برتر را در این جمعیت جدید تکرار می‌کنیم و در صورت عدم موفقیت، دوباره جمعیت جدیدی تولید می‌کند تا این که در جمعیتی، کروموزوم برتری نسبت به اطلاعات وارد شده توسط مشتری پیدا کند یا این که تعداد تکرارهای الگوریتم پایان می‌یابد و بهترین کروموزومی را که الگوریتم تاکنون یافته است به عنوان جواب مسئله به مشتری پیشنهاد می‌دهد. در ادامه روابط محاسبه ارزش چندین ویژگی کیفی برای وبسرویس‌ها که به صورت ترتیبی با هم ترکیب شده‌اند آمده است

$$QoS = \begin{cases} \frac{QoS_{max} - QoS_{neg}}{QoS_{max} - QoS_{min}}, & \text{ویژگی کیفی منفی} \\ \frac{QoS_{pos} - QoS_{min}}{QoS_{max} - QoS_{min}}, & \text{ویژگی کیفی مثبت} \end{cases} \quad (3)$$

$$QoS_{\alpha_i} = \sum_{i=1}^m (QoS_{time})_i \quad (4)$$

$$QoS_{\alpha_c} = \sum_{i=1}^m (QoS_{cost})_i \quad (5)$$

$$QoS_{\alpha_r} = \sum_{i=1}^m (QoS_{availability})_i \quad (6)$$

$$QoS_{\alpha_s} = \sum_{i=1}^m (QoS_{reliability})_i \quad (7)$$

$$Fitness = QoS_{final} = F_{obj} = \sum_{i=1}^k W_i \times QoS_{\alpha_i} \quad (8)$$

### ۳- مبانی مسئله

#### ۳-۱ همبستگی ویژگی‌های کیفی در مسئله انتخاب و ترکیب وبسرویس‌ها

همبستگی بین ویژگی‌های کیفی زمانی رخ می‌دهد که میزان ویژگی‌های کیفی وبسرویس‌ها تنها به خود وبسرویس‌های انتخابی همبسته نبوده و به وبسرویس‌های دیگری که در جریان کاری انتخاب و با هم ترکیب می‌شوند نیز همبسته باشد. دو وبسرویس مجاور که به صورت ترتیبی با هم ترکیب شده‌اند اگر از یک شرکت و در یک مکان باشند دیگر نیازی به انتقال اطلاعات بین این دو وبسرویس نداریم و در صورت ترکیب این دو وبسرویس با هم به زمان پاسخ بهتری برای ترکیب آنها دست می‌یابیم، به این حالت، همبستگی در وبسرویس‌های مجاور گویند. برای همبستگی در وبسرویس‌های نامجاور می‌توان به تأمین‌کنندگان وبسرویس‌های متنوعی مثل Microsoft و Amazon اشاره کرد که برای تشویق مشتریان به خرید تعداد وبسرویس‌های متفاوت بیشتری از آنها، قیمت‌های وبسرویس‌ها را کاهش می‌دهند، در صورتی که مشتری دو یا تعداد بیشتری وبسرویس از آنها خریداری کند. مقالاتی که تاکنون در این زمینه ارائه شده‌اند، همگی همبستگی را برای ترکیب وبسرویس‌ها در حالت ترتیبی در نظر گرفته‌اند و راه‌حلی برای آن ارائه کرده‌اند. راه‌حل‌های ارائه شده تاکنون قادر به حل مسئله

گرفته شده که هر مدخل آن به جای اشاره به یک وبسرویس به یک ویژگی کیفی اشاره دارد [۲۰]. عملگر تقاطع در اینجا تک‌نقطه‌ای و عملگر جهش مانند روش قبل عمل می‌کند.

از دیگر تفاوت‌ها میان پیاده‌سازی‌های مختلف الگوریتم ژنتیک، بحث روش کدگذاری این الگوریتم است. برخی از مقاله‌ها روش کدگذاری یک کروموزوم یک‌بعدی را اتخاذ کرده‌اند که البته در صورت افزایش تعداد وبسرویس‌های کاندید، قابلیت اطمینان کروموزوم‌ها در این روش خیلی ضعیف می‌شود [۲۱] و [۲۲]. از طرفی دیگر این روش نمی‌تواند اطلاعات معنایی را نشان دهد و بنابراین برخی از روش‌ها بر پایه الگوریتم کدگذاری ماتریس روابط بنا نهاده شده‌اند، گرچه این روش نیز در موارد زیادی باعث تولید افراد غیر مجاز می‌شود و کارایی الگوریتم را پایین می‌آورد. در [۲۱] از کدگذاری درختی که می‌تواند روابط ترکیب‌های مختلف بین وبسرویس‌ها را مشخص کند صحبت شده است.

با توجه به مطالب گفته شده دیدیم که مدل استاندارد و یا تعریف شده‌ای برای پارامترهای کلیدی الگوریتم ژنتیک در حل مسئله انتخاب و ترکیب وبسرویس‌ها بر اساس ویژگی‌های کیفی وجود ندارد و هر کدام از پژوهش‌ها با اتخاذ سیاست‌ها و پارامترهای مختلف این الگوریتم را مورد استفاده قرار داده و در جهت بهبود کارایی آن تلاش کرده‌اند. این پارامترها شامل الگوهای کدگذاری مثل آرایه صحیح، رشته باینری، یک‌بعدی، ماتریسی و درختی، انواع عملگرهای انتخاب شامل انتخاب تصادفی، نخچیان، انواع عملگرهای تقاطع شامل تک‌نقطه‌ای، دونقطه‌ای، چندنقطه‌ای و جامع و انواع تابع محاسبه ارزش مجموع ویژگی کیفی همراه با فاکتور جریمه یا حذفی و ... است.

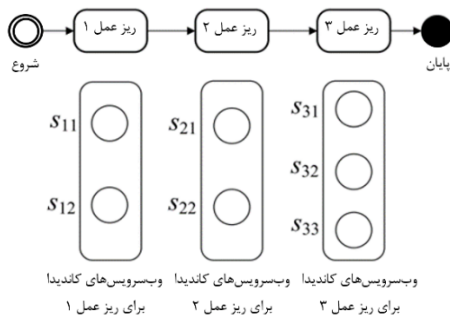
#### ۲-۱۰ تابع محاسبه ارزش ویژگی‌های کیفی (تابع برازندگی)

ویژگی‌های کیفی مثبت و منفی هستند و توسط معیارهای متفاوتی اندازه‌گیری می‌شوند، به عنوان مثال ویژگی کیفی هزینه بر حسب تومان و ویژگی کیفی زمان پاسخ بر حسب میلی‌ثانیه اندازه‌گیری می‌شوند. در انتخاب و ترکیب ترتیبی وبسرویس‌ها بر اساس چندین ویژگی کیفی، ابتدا بایستی تأثیر معیارهای متفاوت اندازه‌گیری مثبت و منفی بودن ویژگی‌های کیفی را از بین ببریم تا بتوانیم ویژگی‌های کیفی متفاوت را با هم جمع کرده و ارزش نهایی ویژگی کیفی وبسرویس مرکب را به دست بیاوریم و با حالات دیگر انتخاب و ترکیب وبسرویس‌ها مقایسه کرده و بهترین حالت ترکیب وبسرویس‌ها را بر اساس تمام ویژگی‌های کیفی مورد نظر مشتری بیابیم.

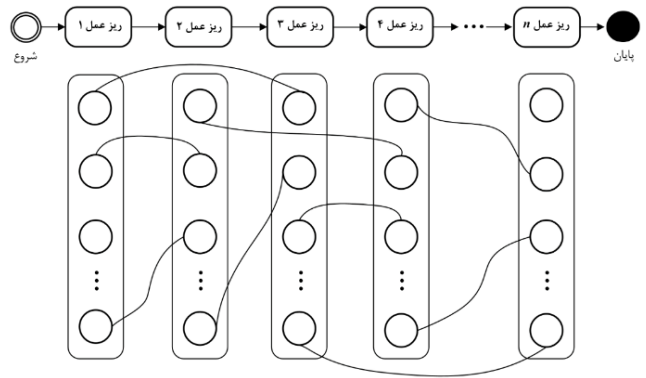
به کمک روابط زیر ارزش نهایی ویژگی کیفی وبسرویس مرکب را به دست می‌آوریم. به کمک (۳) تأثیر معیارهای متفاوت اندازه‌گیری مثبت و منفی بودن ویژگی‌های کیفی را حذف می‌کنیم، سپس با (۴) تا (۷) ارزش مجموع ویژگی‌های کیفی مختلف را در ترکیب ترتیبی وبسرویس‌ها به دست می‌آوریم و نهایتاً با (۸) ارزش نهایی ویژگی‌های کیفی وبسرویس مرکب را با توجه به وزن مورد نظر مشتری برای هر یک از ویژگی‌های کیفی به دست می‌آوریم. اگر ویژگی کیفی مورد نظر مشتری نباشد وزن آن ویژگی کیفی را در (۸) صفر قرار می‌دهیم تا تأثیر آن ویژگی کیفی را از مجموع نهایی حذف کنیم.

به این ترتیب به کمک این روابط ارزش ویژگی‌های کیفی نهایی را برای تمام کروموزوم‌های جمعیت یا به عبارتی برای تمام حالت‌های مختلف انتخاب و ترکیب وبسرویس‌ها محاسبه می‌کنیم و بهترین کروموزوم را از نظر ویژگی‌های کیفی انتخاب می‌کنیم. اگر ارزش نهایی





شکل ۱۲: مثالی از انتخاب و ترکیب ترتیبی وب‌سرویس‌ها [۲۳].



شکل ۱۱: مثالی از شرایط پیچیده در همبستگی بین تعداد زیادی از وب‌سرویس‌ها.

جدول ۲: هزینه اجرایی وب‌سرویس‌ها [۲۳].

وب‌سرویس	سازمان تأمین‌کننده وب‌سرویس	هزینه اجرایی وب‌سرویس (دلار)	دارای همبستگی با وب‌سرویس دیگر	هزینه اجرایی وب‌سرویس در صورت انتخاب وب‌سرویس همبسته با این وب‌سرویس (دلار)
$S_{11}$	Google	۵۰	تعریف نشده	تعریف نشده
$S_{12}$	Microsoft	۶۰	تعریف نشده	تعریف نشده
$S_{21}$	IBM	۳۵	تعریف نشده	تعریف نشده
$S_{22}$	Amazon	۴۰	تعریف نشده	تعریف نشده
$S_{31}$	Microsoft	۹۰	$S_{12}$	۶۰
$S_{32}$	Amazon	۸۵	$S_{21}$	۵۰
$S_{33}$	Oracle	۷۵	تعریف نشده	تعریف نشده

-----: بهینه محلی (انتخاب بهینه)  
 .....: بهینه سراسری (ترکیب بهینه)

### ۳-۳ وابستگی و ناسازگاری بین وب‌سرویس‌ها در مسئله انتخاب و ترکیب وب‌سرویس‌ها

معمولاً در انتخاب وب‌سرویس‌ها باید محدودیت‌های وابستگی و ناسازگاری بین وب‌سرویس‌ها را به منظور حفظ یکپارچگی و تضمین درستی و صحت اجرای وب‌سرویس مرکب در نظر گرفت. در واقع انتخاب وب‌سرویس، یک عملیات مستقل نیست بلکه باید وابستگی و ناسازگاری بین وب‌سرویس‌ها را در نظر بگیرد.

یک وب‌سرویس مرکب‌نشده، نتیجه انتخاب دو وب‌سرویس ناسازگار برای یک وب‌سرویس مرکب است. انتخاب دقیق وب‌سرویس‌ها، لازمه اجتناب از رخ دادن یک وب‌سرویس مرکب‌نشده است که به این موضوع اصطلاحاً صحت و درستی گفته می‌شود.

مسئله‌ای که در این قسمت بررسی می‌شود، یک مسئله بهینه‌سازی محدودیت‌دار است. چگونگی دستیابی به ارزش مجموع ویژگی کیفی مناسب تحت عنوان موضوع بهینه‌سازی تعریف می‌گردد و چگونگی حاصل شدن اطمینان از برآورده شدن محدودیت‌ها یعنی وابستگی‌ها و ناسازگاری‌ها بین وب‌سرویس‌ها تحت عنوان موضوع درستی و صحت معرفی می‌شود که این ۲ موضوع، ۲ چالش عمده در ترکیب وب‌سرویس‌ها هستند و الگوریتم‌های ژنتیکی برای مقابله با این چالش‌ها مورد استفاده قرار می‌گیرند.

با توجه به طبیعت و ماهیت الگوریتم ژنتیک، این الگوریتم به تنهایی می‌تواند روی مباحثی که دارای محدودیت نیستند بهینه‌سازی انجام دهد. خوشبختانه می‌تواند با همکاری و ادغام با برخی از روش‌ها (همچون تابع جریمه، ترمیم و برخی از روش‌های ترکیبی که از محدودیت‌ها (وابستگی‌ها و ناسازگاری‌ها) پشتیبانی می‌نمایند از کارایی خوبی جهت مواجهه با

با وجود همبستگی بین تعداد زیادی از وب‌سرویس‌ها و شرایط پیچیده نیستند (شکل ۱۱).

### ۲-۳ همبستگی ویژگی‌های کیفی در وب‌سرویس‌های نامجاور

در این مثال (شکل ۱۲) اگر به صورت محلی برای هر فرایند بهترین وب‌سرویس را انتخاب کنیم، وب‌سرویس‌های  $S_{11}$ ،  $S_{21}$  و  $S_{33}$  را انتخاب خواهیم کرد و میزان ویژگی کیفی هزینه اجرایی برای وب‌سرویس مرکب برابر  $50 + 35 + 75 = 160$  دلار خواهد شد. حال اگر در این مثال به صورت محلی عمل نکنیم و به صورت سراسری نگاه کنیم (جدول ۲) و همبستگی بین وب‌سرویس‌ها را در نظر بگیریم و وب‌سرویس‌های  $S_{12}$ ،  $S_{21}$  و  $S_{31}$  را انتخاب کنیم، میزان هزینه اجرایی وب‌سرویس مرکب برابر  $60 + 35 + 60 = 155$  دلار می‌شود. وب‌سرویس‌های انتخابی در این حالت همبسته و نامجاور هستند و یا اگر وب‌سرویس‌های  $S_{11}$ ،  $S_{22}$  و  $S_{33}$  را انتخاب کنیم، میزان هزینه اجرایی وب‌سرویس مرکب برابر  $50 + 40 + 50 = 140$  دلار می‌شود. بنابراین در این مثال با اتخاذ رویکرد بهینه سراسری و در نظر گرفتن همبستگی بین وب‌سرویس‌ها ۲۰ دلار یا  $12.5\%$  در هزینه اجرایی وب‌سرویس‌ها بهبود ایجاد کرده‌ایم. دلیل این بهبود، همبستگی وب‌سرویس‌های  $S_{22}$  و  $S_{33}$  است، زیرا این دو وب‌سرویس از یک سازمان هستند و سازمان تأمین‌کننده وب‌سرویس‌ها به منظور تشویق مشتریان به استفاده از وب‌سرویس‌های بیشتر (در صورت استفاده بیش از دو وب‌سرویس)، هزینه اجرایی را از وب‌سرویس دوم به بعد کاهش می‌دهد تا مشتری را برای استفاده بیشتر از وب‌سرویس‌های سازمان خود ترغیب کنند. در این نوع ویژگی کیفی امکان انتخاب وب‌سرویس‌های نامجاور همبسته وجود دارد.

گردند. بنابراین در این الگوریتم باید این موضوع بررسی و در نظر گرفته شود که ممکن است کروموزوم‌های نشدنی، شامل برخی از ژن‌ها که برای ایجاد یک راه‌حل بهینه ضروری هستند باشند. اگر افراد نشدنی مستثنی شوند، ممکن است که الگوریتم ژنتیک به راه‌حل بهینه دست پیدا نکند. در نتیجه استراتژی اتخاذ شده توسط الگوریتم ژنتیک به افراد نشدنی اجازه می‌دهد در جمعیت قرار بگیرند، اما جریمه‌هایی را به تابع برازندگی اعمال می‌نماید. در ادامه به ۲ رهنمود اشاره می‌کنیم که موقع تعریف تابع برازندگی استفاده می‌شوند.

باید تضمین شود که راه‌حل (کروموزوم) نشدنی دارای کمترین ارزش برازندگی نسبت به راه‌حل (کروموزوم) شدنی است. کروموزومی که از محدودیت‌ها تخلف زیادی نموده است، باید به شدت نسبت به کروموزومی که تخلف کمتری داشته است جریمه شود. در ادامه (۹) و (۱۰) به ترتیب تابع برازندگی و جریمه را بیان می‌نمایند

$$Fitness(x) = 0.5 + (0.5 \times F_{obj}(x)) + P(x) \quad (9)$$

$$P(x) = \begin{cases} 0, & \text{if } V(x) = 0 \\ -0.5 - \frac{V(x)}{V_{max}}, & \text{otherwise} \end{cases} \quad (10)$$

در (۹)،  $F_{obj}(x)$  تابع محاسبه ارزش ویژگی‌های کیفی (تابع برازندگی) برای کروموزوم  $x$  است که قبلاً معرفی شد و  $P(x)$  مقدار جریمه‌ای است که برای کروموزوم  $x$  محاسبه و نسبت داده می‌شود. در معادله دوم  $V(x)$  مخفف تعداد کل موارد نقض محدودیت توسط کروموزوم  $x$  و  $V_{max}$  حداکثر تعداد موارد موجود نقض محدودیت در مسئله مطرح شده است. بنابراین هنگامی که  $V(x)$  برابر صفر است، دلالت بر این نکته دارد که کروموزوم  $x$ ، راه‌حلی شدنی است و هیچ موردی از محدودیت‌ها را نقض نکرده است.

با توجه به (۱۰) می‌توان اشاره کرد که اگر کروموزوم  $x$  راه‌حلی شدنی باشد، میزان جریمه آن برابر صفر است. در غیر این صورت مقدار جریمه داده شده به کروموزوم  $x$  به عنوان یک راه‌حل نشدنی از طریق معادله  $-0.5 - V(x)/V_{max}$  محاسبه می‌شود و این معادله تضمین می‌کند که یک راه‌حل نشدنی هرچه محدودیت‌های بیشتری را نقض کند، جریمه بالاتری نیز دریافت خواهد نمود.

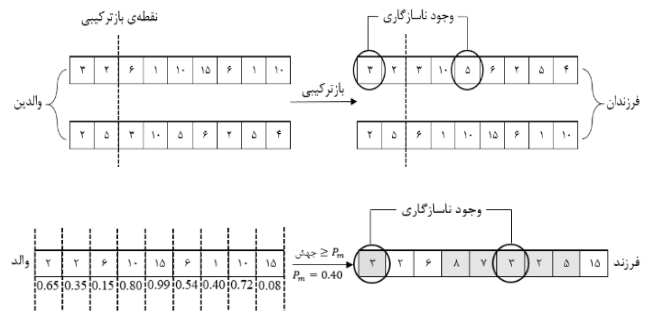
با توجه به (۹) و (۱۰) مقدار تابع برازندگی هر کروموزوم شدنی  $x$  را می‌توان از طریق معادله  $0.5 + (0.5 \times F_{obj}(x))$  محاسبه نمود که مقدار آن همواره در بازه  $[0.5, 1]$  است. مقدار تابع برازندگی هر کروموزوم نشدنی  $x$  را می‌توان از طریق  $(0.5 \times F_{obj}(x)) - V(x)/V_{max}$  محاسبه نمود که مقدار آن همواره در بازه  $[0, 0.5]$  است. بنابراین به کمک این روابط می‌توان تضمین کرد که یک کروموزوم نشدنی نسبت به هر کروموزوم شدنی، ارزش کمتری از تابع برازندگی کسب خواهد کرد [۲۴].

### ۴- روش پیشنهادی

#### ۴-۱ روند عملیاتی الگوریتم ژنتیک برای یافتن جواب

##### مسئله انتخاب و ترکیب و بسرویس‌ها

الگوریتم ژنتیک برای شروع کار خود و یافتن جواب مسئله انتخاب و ترکیب و بسرویس‌ها، جمعیتی از کروموزوم‌ها را ایجاد می‌کند. شماره هر ژن کروموزوم اشاره به شماره و بسرویس انتزاعی در جریان کاری دارد و مقدار عددی‌ای که در ژن‌های کروموزوم‌ها قرار می‌گیرد، شماره و بسرویس کاندیدی است که برای آن فعالیت انتخاب شده است.



شکل ۱۳: ظهور ناسازگاری در فرزندان بعد از عملگرهای تقاطع و جهش.

محدودیت‌ها برخوردار شود. در ادامه یک روش از بین روش‌های مختلف به کارگیری محدودیت‌ها که برای مواجهه با محدودیت‌ها همراه با وابستگی‌ها و ناسازگاری بین و بسرویس‌ها مورد استفاده قرار گرفته است شرح داده خواهد شد.

الگوریتم ژنتیک مبتنی بر جریمه از یک تابع جریمه برای مواجهه با محدودیت‌ها در وابستگی و ناسازگاری بین و بسرویس‌ها درگیر در مسئله استفاده می‌کند.

### ۳-۴ الگوریتم ژنتیک مبتنی بر جریمه

شاید الگوریتم جریمه محبوب‌ترین الگوریتم برای مواجهه و پاسخگویی به محدودیت‌ها باشد. الگوریتم ژنتیک مبتنی بر جریمه از یک تابع جریمه جهت پشتیبانی از محدودیت‌های وابستگی و ناسازگاری بین و بسرویس‌ها استفاده می‌کند. ایده اصلی این الگوریتم و تابع آن، دادن جریمه به راه‌حل‌های نشدنی است. تابع جریمه در صورت مشاهده انحراف و نقض محدودیت‌های وابستگی و ناسازگاری بین و بسرویس‌ها در راه‌حلی که می‌یابد، ارزش راه‌حل یافته شده را کاهش می‌دهد تا این راه‌حل نشدنی، شانس کمتری برای زنده ماندن نسبت به راه‌حل‌های شدنی دیگری که محدودیت‌ها را نقض نکرده‌اند داشته باشد. در الگوریتم ژنتیک مبتنی بر جریمه برای کروموزوم‌هایی که ژن‌های نشدنی دارند و محدودیت‌های وابستگی و ناسازگاری بین و بسرویس‌ها را نقض کرده‌اند، در تابع محاسبه برازندگی، از استراتژی جریمه استفاده شده است [۲۴].

باید توجه داشت که عملگرهای جهش و تقاطع که روی والد‌ها اجرا می‌شود، ممکن است راه‌حل‌های (کروموزوم‌های) نشدنی تولید نمایند. به عنوان مثال فرض کنید که اگر در خانه اول کروموزوم و بسرویس شماره سه انتخاب شود و قرار گیرد و در خانه پنجم کروموزوم و بسرویس شماره پنج انتخاب شود و قرار گیرد، به دلیل این که این دو و بسرویس با هم ناسازگار هستند، راه‌حلی که از ترکیب این و بسرویس‌ها به دست می‌آید در کل به راه‌حلی نشدنی تبدیل می‌شود. همچنین و بسرویس شماره سه در خانه شماره یک با و بسرویس شماره سه در خانه شماره شش با یکدیگر ناسازگار هستند

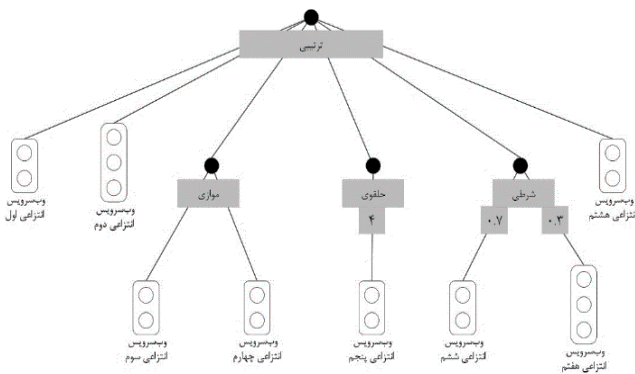
$$S_{13} \text{ has Conflict with } S_{55}$$

$$S_{13} \text{ has Conflict with } S_{67}$$

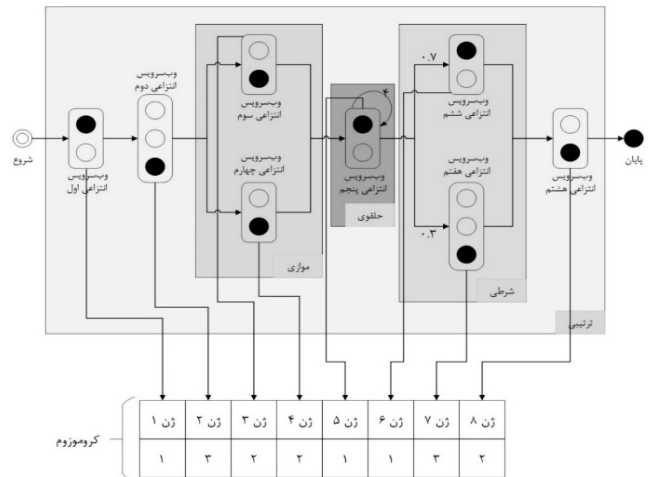
در شکل ۱۳ والد‌ها محدودیت‌های ذکر شده در بالا را نقض نکرده‌اند. این حال پس از عمل تقاطع، فرزند اول محدودیت ناسازگاری را نقض می‌نماید و همچنین عملگر جهش نیز منجر به نقض محدودیت می‌شود. برای پشتیبانی و مواجهه با راه‌حل‌های نشدنی، یک تابع جریمه در تابع برازندگی این الگوریتم تعریف می‌کنیم.

همان گونه که در بالا مطرح گردید، برخی از کروموزوم‌های تولیدی توسط عملگرهای تقاطع و جهش ممکن است به راه‌حلی نشدنی تبدیل





شکل ۱۵: درخت دربردارنده جریان کاری و روابط بین وب‌سرویس‌های مورد نظر مشتری.



شکل ۱۴: ساختار کروموزوم و ژن‌های آن برای جریان کاری مورد نظر مشتری.

جدول ۳: رابطه محاسبه برازندگی وب‌سرویس مرکب برای جریان کاری مورد نظر مشتری.

ویژگی کیفی	رابطه محاسبه برازندگی وب‌سرویس مرکب برای جریان کاری مورد نظر مشتری
زمان ( $QoS_{time}$ )	$QoS_{time} = (QoS_{time}(CWS_{s,x})) + (QoS_{time}(CWS_{t,x})) + (\max\{QoS_{time}(CWS_{r,x}), QoS_{time}(CWS_{v,x})\}) + (4 \times QoS_{time}(CWS_{d,x})) + ((0.7 \times QoS_{time}(CWS_{f,x})) + (0.3 \times QoS_{time}(CWS_{v,x}))) + (QoS_{time}(CWS_{h,x}))$
هزینه ( $QoS_{cost}$ )	$QoS_{cost} = (QoS_{cost}(CWS_{s,x})) + (QoS_{cost}(CWS_{t,x})) + (QoS_{cost}(CWS_{r,x})) + (QoS_{cost}(CWS_{v,x})) + (QoS_{cost}(CWS_{d,x})) + ((0.7 \times QoS_{cost}(CWS_{f,x})) + (0.3 \times QoS_{cost}(CWS_{v,x}))) + (QoS_{cost}(CWS_{h,x}))$
دسترس‌پذیری ( $QoS_{availability}$ )	$QoS_{availability} = (QoS_{availability}(CWS_{s,x})) \times (QoS_{availability}(CWS_{t,x})) \times (QoS_{availability}(CWS_{r,x})) \times (QoS_{availability}(CWS_{v,x})) \times (QoS_{availability}(CWS_{d,x})) \times ((0.7 \times QoS_{availability}(CWS_{f,x})) \times (0.3 \times QoS_{availability}(CWS_{v,x}))) \times (QoS_{availability}(CWS_{h,x}))$
قابلیت اطمینان ( $QoS_{reliability}$ )	$QoS_{reliability} = (QoS_{reliability}(CWS_{s,x})) \times (QoS_{reliability}(CWS_{t,x})) \times (QoS_{reliability}(CWS_{r,x})) \times (QoS_{reliability}(CWS_{v,x})) \times (QoS_{reliability}(CWS_{d,x})) \times ((0.7 \times QoS_{reliability}(CWS_{f,x})) \times (0.3 \times QoS_{reliability}(CWS_{v,x}))) \times (QoS_{reliability}(CWS_{h,x}))$
قابلیت اطمینان (Final)	$Fitness = QoS_{final} = F_{obj} = (W \times (QoS_{time} \times QoS_{cost})) + (W \times (QoS_{cost} \times QoS_{cost})) + (W \times (QoS_{availability} \times QoS_{availability})) + (W \times (QoS_{reliability} \times QoS_{reliability}))$

\*  $W$ : وزن مورد نظر مشتری برای تأثیر ویژگی‌های مختلف در میزان برازندگی نهایی وب‌سرویس مرکب

به کمک الگوریتم جستجوی عمق اول<sup>۱</sup> (DFS) و پیمایش پسوندی<sup>۲</sup> آن، رابطه محاسبه برازندگی را برای وب‌سرویس مرکب می‌یابیم و با توجه به مقادیر ژن‌های کروموزوم‌ها وب‌سرویس‌های انتخابی را یافته و میزان ویژگی کیفی آنها را در رابطه یافت‌شده از پیمایش درخت قرار می‌دهیم و برازندگی وب‌سرویس مرکب را محاسبه و با برازندگی‌ای که مورد نظر مشتری است مقایسه کرده و برای ادامه کار الگوریتم تصمیم می‌گیریم. در ادامه به کمک جدول ۱ و استفاده از الگوریتم جستجوی عمق اول و پیمایش پسوندی که روی درخت شکل ۱۵ اعمال می‌کنیم، روابط محاسبه برازندگی وب‌سرویس مرکب را برای ویژگی‌های کیفی مختلف، برای جریان کاری که در شکل ۱۴ نمایش داده شده است، به دست آورده و در جدول ۳ نمایش می‌دهیم.

به عنوان مثال در ادامه، با توجه به جریان کاری و روابط بین وب‌سرویس‌ها که مشتری مشابه شکل ۱۴ به مسئله ورودی داده بود، در شکل ۱۵ درختی را که دربردارنده این اطلاعات است ترسیم می‌کنیم.

#### ۴-۲ الگوریتم بررسی همبستگی بین وب‌سرویس‌های نامجاور برای ویژگی کیفی هزینه

در بررسی همبستگی وب‌سرویس مورد نظر برای ویژگی کیفی هزینه

به عنوان مثال، ساختار کروموزوم و ژن‌های آن برای جریان کاری مورد نظر مشتری که در بالا ذکر گردید در شکل ۱۴ نمایش داده شده است.

همان طور که قبلاً اشاره شد، الگوریتم ژنتیک مستقل از مسئله کار خود را انجام می‌دهد و بنابراین در این مسئله الگوریتم ژنتیک مستقل از روابط بین وب‌سرویس‌ها و جریان کاری‌ای که طبق آن وب‌سرویس‌ها با هم ترکیب شده‌اند، کار خود را انجام می‌دهد. الگوریتم ژنتیک در ابتدا جمعیتی از کروموزوم‌ها با مقادیری تصادفی برای ژن‌های آنها تولید می‌کند. برای ادامه کار الگوریتم، ابتدا بایستی برازندگی این کروموزوم‌های تولیدی را محاسبه کنیم تا در صورت دارا بودن برازندگی مورد نظر مشتری الگوریتم پایان یابد و مقادیر ژن‌های کروموزوم برتر، وب‌سرویس‌های مناسب برای انتخاب در جریان کاری را برای ما مشخص کند. در غیر این صورت، الگوریتم ژنتیک به کمک عملگرهای جهش و تقاطع و روندی که قبلاً توضیح داده شد، جمعیت جدیدی تولید می‌کند و ادامه می‌یابد تا با یکی از شرایط خاتمه الگوریتم ژنتیک مواجه شود. برای محاسبه برازندگی کروموزوم‌ها بایستی به روابط بین وب‌سرویس‌ها و جریان کاری‌ای که طبق آن وب‌سرویس‌ها با هم ترکیب شده‌اند و کنار هم قرار گرفته‌اند، توجه کرد. برای این منظور درختی را که دربردارنده شکل جریان کاری و روابط بین وب‌سرویس‌ها است تشکیل می‌دهیم و همواره برای محاسبه برازندگی کروموزوم و وب‌سرویس مرکب به این درخت مراجعه می‌کنیم و

1. Depth First Search  
2. Post-Order

**Algorithm 2** Checking Correlation between Target Web Service and Previous Nonadjacent Web Services**Input:** target web service in workflow (TWS), Chromosome**Output:** existence of correlation for target web service

```

1: selected_web_services = Chromosome.selectedWebServicesBeforeTarget(TWS)
2: for  $\forall x \in$  selected_web_services do
3:     if checkCorrelation(x, TWS) then
4:         return true
5:     end if
6: end for
7: return false

```

شکل ۱۶: الگوریتم بررسی وجود همبستگی بین وبسرویس هدف با وبسرویس‌های نامجاور.

شد، کاهش دهیم تا این راه‌حل نشدنی حتی در صورتی که به کمک همبستگی بین وبسرویس‌ها برازندگی مناسب‌تری پیدا کرده باشد، شانس کمتری برای زنده ماندن نسبت به راه‌حل‌های شدنی دیگری که محدودیت‌ها را نقض نکرده باشند، داشته باشد. بنابراین برای هر یک از کروموزوم‌ها بعد از بررسی همبستگی و محاسبه برازندگی، بررسی می‌کنیم که آیا ژن‌های نشدنی دارند و محدودیت‌های وابستگی و ناسازگاری را نقض کرده‌اند که در این صورت برازندگی یافته شده شامل یک استراتژی جریمه خواهد شد (برای محاسبه برازندگی نهایی هر یک از کروموزوم‌ها با در نظر گرفتن جریمه برای راه‌حل‌های نشدنی از (۹) و (۱۰) استفاده می‌کنیم).

در انتها بعد از بررسی همبستگی، وابستگی و ناسازگاری، کروموزومی را که دارای بهترین برازندگی است به عنوان راه‌حل برای جریان کاری مورد نظر مشتری اعلام می‌کنیم.

#### ۴-۴ چگونگی ورودی دادن اطلاعات به وبسرویس‌ها و

#### بررسی همبستگی، وابستگی و ناسازگاری بین آنها به

#### کمک روش پیشنهادی

مشتری برای مسئله انتخاب و ترکیب وبسرویس‌ها، جریان کاری‌ای که نشان‌دهنده روابط بین وبسرویس‌ها است و اطلاعاتی در مورد وبسرویس‌های کشف شده و وابستگی، ناسازگاری و همبستگی بین وبسرویس‌ها را به کمک قواعدی که در ادامه تعریف می‌شود به الگوریتم ژنتیک ورودی می‌دهد:

- جریان کاری مورد نظر مشتری شامل مجموعه‌ای از فعالیت‌ها است. برای هر یک از فعالیت‌ها، وبسرویس انتزاعی<sup>۱</sup> (AWS) را در نظر گرفته و به صورت زیر آن را به الگوریتم ژنتیک ورودی می‌دهیم

$$AWS = \{AWS_1, AWS_2, \dots, AWS_m\} \quad (11)$$

- برای هر یک از وبسرویس‌های انتزاعی که متناظر با فعالیتی است، چندین وبسرویس کاندیدی<sup>۲</sup> (CWS) که قابلیت به انجام رساندن آن فعالیت را دارند و کشف شده‌اند، به صورت زیر به الگوریتم ژنتیک ورودی داده می‌شوند

$$AWS_i = \{CWS_{i,1}, CWS_{i,2}, \dots, CWS_{i,n}\}, 1 \leq i \leq m \quad (12)$$

- اطلاعاتی که مشتری در مورد هر کدام از وبسرویس‌های کاندید، همچون شرکت تأمین‌کننده وبسرویس، نوع و میزان ویژگی کیفی به مسئله ورودی می‌دهد، به صورت زیر برای هر وبسرویس کاندید به الگوریتم ژنتیک ورودی داده می‌شود

$$CWS_{i,j} = \langle \langle Company'sName \rangle, \langle \langle QoS_\alpha Kind \rangle, \langle QoS_\alpha Value \rangle, \dots \rangle \rangle \quad (13)$$

1. Abstract Web Services
2. Candidate Web Services

بایستی تمام وبسرویس‌های قبل از آن را بررسی کنیم و دلیل این که می‌توانیم به صورت نامجاور عمل کنیم این است که در همبستگی بین وبسرویس‌ها برای ویژگی کیفی هزینه، هدف سازمان‌های تأمین‌کننده وبسرویس، تشویق مشتریان به انتخاب وبسرویس‌های بیشتری از سازمان آنها در جریان کاری مورد نظر مشتری است و در این مورد هم نیازی به بررسی همبستگی وبسرویس‌ها به صورت مجاور نیست و چندین وبسرویس از یک سازمان می‌توانند از مکان‌های مختلف در جریان کاری انتخاب شوند و سازمان تأمین‌کننده نیز به هدف خود یعنی فروش بیشتر وبسرویس‌ها رسیده است. بنابراین در این مورد برای بررسی همبستگی هر وبسرویس، تمام وبسرویس‌های انتخابی قبل از آن را بررسی و مشاهده می‌کنیم که آیا از سازمان وبسرویس مورد بررسی، وبسرویس دیگری قبل از آن در جریان کاری از همان سازمان انتخاب شده است که اگر انتخاب شده باشد هزینه اجرایی جدید و بهبود یافته‌ای که خود سازمان اعلام کرده است را برای آن وبسرویس در نظر گرفته و از این به بعد آن را در محاسبات مورد استفاده قرار می‌دهیم.

برای محاسبه برازندگی کروموزوم یا وبسرویس مرکب مورد بررسی، به کمک این الگوریتم برای هر یک از وبسرویس‌های انتخاب شده در ژن‌های کروموزوم امکان وجود همبستگی را بررسی می‌کنیم و در صورت یافتن همبستگی در هر یک از وبسرویس‌ها در محاسبه برازندگی از هزینه اجرایی جدید و بهبود یافته بعد از اعمال همبستگی استفاده می‌کنیم و هزینه اجرای وبسرویس مرکب نیز در مجموع بهبود خواهد یافت.

برای ویژگی‌های کیفی دیگری که برای بررسی همبستگی بایستی وبسرویس‌های نامجاور را بررسی کنیم، مشابه هزینه اجرایی عمل می‌کنیم و از الگوریتمی که در شکل ۱۶ آورده شده است استفاده می‌کنیم. الگوریتم روند بررسی وجود همبستگی در هر یک از وبسرویس‌ها با وبسرویس‌های نامجاور قبل از خود در شکل ۱۶ آورده شده است. به این الگوریتم کروموزومی که نشان‌دهنده وبسرویس‌های انتخابی است ورودی داده می‌شود تا برای بررسی همبستگی وبسرویس هدف بتوانیم با مراجعه به این کروموزوم، تمام وبسرویس‌های دیگری که انتخاب شده‌اند را یافته و مورد بررسی قرار دهیم و ویژگی کیفی نهایی وبسرویس هدف را به راحتی در صورت وجود همبستگی محاسبه کنیم.

#### ۴-۳ الگوریتم بررسی وابستگی و ناسازگاری

#### بین وبسرویس‌ها

همان گونه که در الگوریتم ژنتیک مبتنی بر جریمه اشاره شد، برخی از کروموزوم‌های تولیدی محدودیت‌های وابستگی و ناسازگاری بین وبسرویس‌ها را نقض می‌کنند و به راه‌حل‌های نشدنی تبدیل می‌گردند. بنابراین بعد از بررسی همبستگی بین وبسرویس‌ها، بایستی هر یک از کروموزوم‌ها را از نظر نقض محدودیت‌های وابستگی یا ناسازگاری نیز بررسی کنیم و در صورت مشاهده نقض محدودیت‌ها ارزش راه‌حل یافته شده را به کمک الگوریتم ژنتیک مبتنی بر جریمه که قبلاً معرفی

را مشخص کند. شرکت‌های ارائه‌دهنده وب‌سرویس‌ها، تغییر میزان ویژگی کیفی در صورت انتخاب وب‌سرویس‌های همبسته و شرط همبستگی با وب‌سرویس‌ها را بایستی اعلام کنند، بنابراین اطلاعات در مورد این شرط را در زمان ورودی گرفتن اطلاعات در مورد وب‌سرویس‌های کاندید از مشتری دریافت می‌کنیم.

در این مقاله به نحوه وارد کردن برای همبستگی نامجاور بین وب‌سرویس‌ها تنها برای ویژگی‌های کیفی هزینه اجرایی اشاره شده است، برای ویژگی‌های کیفی دیگر نیز مشابه این ویژگی‌های کیفی عمل خواهیم کرد و به دلیل تشابه از اشاره به این موارد صرف نظر شده است.

## ۵- نتایج پیاده‌سازی و ارزیابی

در این بخش الگوریتم ژنتیک پیشنهادی با الگوریتم‌های ارائه شده در کارهای پیشین و الگوریتم ژنتیک پایه، بر مبنای تأثیر افزایش همبستگی بین وب‌سرویس‌ها بر میزان برازندگی وب‌سرویس مرکب یافت شده و زمان دستیابی به برازندگی مورد نظر مشتری و تأثیر افزایش وابستگی و ناسازگاری بین وب‌سرویس‌ها بر مدت زمان رسیدن به میزان برازندگی مورد نظر برای وب‌سرویس مرکب، بررسی می‌شوند.

الگوریتم‌های ژنتیک پایه و پیشنهادی با زبان PHP پیاده‌سازی شده‌اند و بر روی کامپیوتری شخصی با پردازنده ۵ GHz Intel Core i۵ و ۸ GB Ram اجرا شده‌اند. نتایج مقایسه این دو الگوریتم در این بخش آورده می‌شود.

### ۵-۱ ویژگی‌های الگوریتم ژنتیک پایه

الگوریتم ژنتیک را به عنوان الگوریتم پایه‌ای که الگوریتم پیشنهادی را با آن مقایسه می‌کنیم در نظر گرفته‌ایم. دلیل این انتخاب این است که الگوریتم پیشنهادی بر مبنای الگوریتم ژنتیک و ایجاد تغییراتی بر روی آن ارائه شده است.

الگوریتم ژنتیکی که به عنوان الگوریتم پایه برای مقایسه با الگوریتم ژنتیک پیشنهادی در نظر گرفته شده است، الگوریتمی است که در صورت مشاهده نقض محدودیت‌های وابستگی و ناسازگاری ساده‌ترین روش را اتخاذ کرده و آن کروموزومی که محدودیت‌ها را نقض کرده، حذف می‌کند و برازندگی آن کروموزوم را صفر قرار می‌دهد. همچنین این الگوریتم نمی‌تواند همبستگی بین وب‌سرویس‌ها را تشخیص داده و تنها ویژگی‌های کیفی اصلی وب‌سرویس‌ها را بدون در نظر گرفتن وجود همبستگی در محاسبه‌های مربوط به برازندگی اعمال می‌کند.

اندازه جمعیت کروموزوم‌ها در هر نسل که تولید و بررسی می‌شوند، ۵۰۰ کروموزوم در نظر گرفته می‌شود. احتمال رخ دادن تقاطع ۰/۸، رخ دادن جهش ۰/۰۵ و تعداد دفعات تولید نسل برای یافتن بهترین ترکیب ۱۰۰۰۰ در نظر گرفته می‌شود.

### ۵-۲ ویژگی‌های الگوریتم ژنتیک پیشنهادی

الگوریتم ژنتیک پیشنهادی در صورت مشاهده نقض محدودیت‌های وابستگی و ناسازگاری توسط کروموزوم‌ها همان طور که در بخش‌های قبل اشاره شد، این کروموزوم‌های ناقص را حذف نکرده و تنها آنها را جریمه کرده و اجازه ایجاد نسل‌هایی از آنها را می‌دهد. زیرا ممکن است با تقاطع یا جهش روی همین کروموزوم‌های ناقص، کروموزومی با جواب بهینه بیابد ولی برازندگی این کروموزوم‌های ناقص محدودیت‌ها را به اندازه‌های کاهش می‌دهد تا مطمئن شود که خود این کروموزوم‌های ناقص به عنوان راه‌حل انتخاب نمی‌شوند.

نحوه و شکل ترکیب وب‌سرویس‌ها به صورت ترتیبی، موازی، شرطی و حلقوی که در جریان کاری مورد نظر مشتری مشخص و به مسئله ورودی داده می‌شود، به صورت زیر به الگوریتم ژنتیک ورودی داده می‌شود

$$CompositeWebService = Sequence$$

$$\left( \begin{array}{c} \dots, \\ AWS_i, \\ Parallel \langle AWS_{i+1}, AWS_{i+2} \rangle, \\ AWS_{i+3}, \\ Conditional \langle \langle p_1, AWS_{i+4} \rangle, \langle p_r, AWS_{i+5} \rangle \rangle, \\ Loop \langle k, AWS_{i+6} \rangle, \\ \dots \end{array} \right) \quad (14)$$

وابستگی بین وب‌سرویس‌های کاندید به صورت زیر توسط مشتری به الگوریتم ژنتیک ورودی داده می‌شوند

$$Dependency = \{ \langle CWS_{i_1, j_1}, CWS_{i_2, j_2} \rangle \}$$

اگر برای وب‌سرویس انتزاعی  $i_1$  وب‌سرویس کاندیدای  $j_1$  انتخاب شود، باید برای وب‌سرویس انتزاعی  $i_2$  نیز وب‌سرویس کاندیدای  $j_2$  انتخاب شود و بالعکس

ناسازگاری بین وب‌سرویس‌های کاندید به صورت زیر توسط مشتری به الگوریتم ژنتیک ورودی داده می‌شوند

$$Conflict = \{ \langle CWS_{i_1, j_1}, CWS_{i_2, j_2} \rangle \}$$

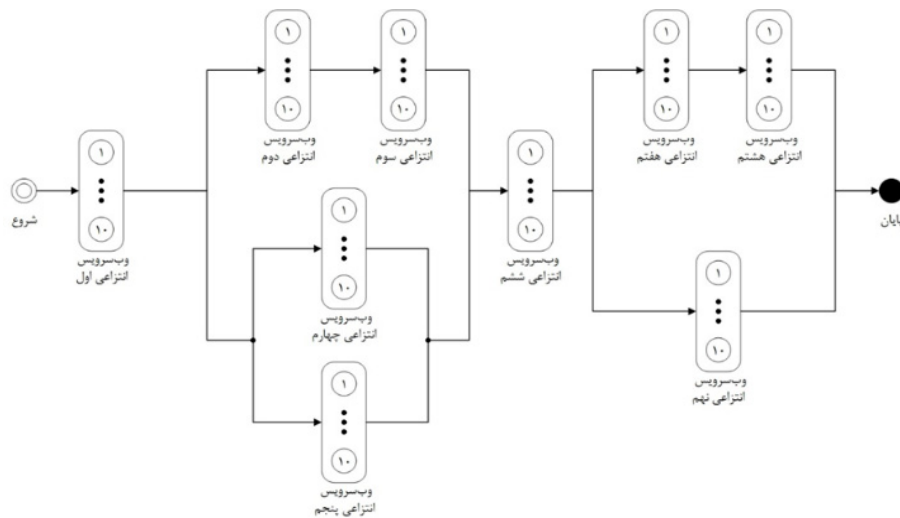
اگر برای وب‌سرویس انتزاعی  $i_1$  وب‌سرویس کاندیدای  $j_1$  انتخاب شود، باید برای وب‌سرویس انتزاعی  $i_2$  وب‌سرویس کاندیدای  $j_2$  انتخاب نشود و بالعکس

مشتری برای همبستگی بین وب‌سرویس‌های کاندید، برای ویژگی کیفی هزینه، هزینه اجرایی اصلی وب‌سرویس و هزینه اجرایی وب‌سرویس در صورت انتخاب وب‌سرویس همبسته دیگری با این وب‌سرویس، برای هر یک از وب‌سرویس‌ها مشخص کرده و طبق قاعده زیر به الگوریتم ژنتیک ورودی می‌دهد تا با توجه به وب‌سرویس‌های انتخاب شده در مجموع برای ترکیب با یکدیگر، یکی از هزینه‌های اجرایی وب‌سرویس انتخاب و در تمام روابط مربوط به محاسبه برازندگی هزینه اجرایی وب‌سرویس مرکب مورد استفاده قرار گیرد

$$Detail\ of\ QoS_{cost}(CWS_{i,j}) = [$$

هزینه اجرایی وب‌سرویس در صورت انتخاب وب‌سرویس همبسته، هزینه اجرایی اصلی

دلیل این که وب‌سرویس‌هایی را که با یکدیگر همبسته هستند همچون وابستگی و ناسازگاری از مشتری دریافت نمی‌کنیم این است که در همبستگی بایستی مشتری تعریف کند که هر وب‌سرویس تحت چه شرایطی با وب‌سرویس‌های دیگر همبسته خواهد شد و این که تعداد حالت‌های همبستگی بین وب‌سرویس‌ها معمولاً بسیار زیاد می‌شود و در جریان‌های کاری پیچیده تشخیص این همبستگی‌ها و وارد کردن آنها به الگوریتم کاری بسیار دشوار و زمان‌گیر است. بنابراین بهتر است که مشتری تنها شرایط همبسته شدن و میزانی که در ویژگی‌های کیفی در صورتی که وب‌سرویس‌های همبسته را انتخاب کنیم، بهبود ایجاد می‌شود



شکل ۱۷: جریان کاری آزمایش تأثیر همبستگی بر میزان برازندگی وبسرویس مرکب یافت‌شده.

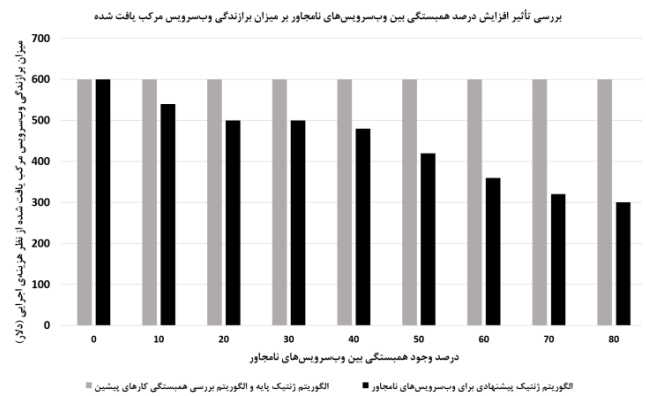
برای این آزمایش، ده عدد تصادفی تولید می‌کنیم و برای هر یک از این وبسرویس‌های کاندید یکی از این اعداد تصادفی تولیدشده را که نشان‌دهنده هزینه اجرایی آن وبسرویس است در نظر می‌گیریم. برای بررسی همبستگی و محاسبه مجموع ویژگی کیفی در صورت وجود همبستگی، در هر یک از این وبسرویس‌ها ۱۰٪ از هزینه اجرایی را کاهش می‌دهیم تا میزان ویژگی کیفی مورد نظر برای اعمال در محاسبات مجموع ویژگی کیفی در صورت وجود همبستگی را داشته باشیم.

در این آزمایش به دلیل این که الگوریتم بررسی همبستگی کارهای پیشین امکان بررسی همبستگی در حالت‌های موازی، شرطی و حلقوی را ندارد، در این جریان کاری مورد آزمایش این الگوریتم دقیقاً مشابه الگوریتم ژنتیک پایه عمل می‌کند و همان نتایج را به دست می‌آورد.

در این آزمایش مدت زمان رسیدن به جواب بهینه مورد نظر مشتری را در صورت بررسی وجود همبستگی بررسی می‌کنیم و نتایج به دست آمده را در ادامه می‌آوریم. مدت زمان رسیدن به جواب بهینه برای الگوریتم ژنتیک پایه در صورتی که همواره مسئله ثابت است، تغییر می‌یابد که این نتیجه به دلیل ماهیت تصادفی بودن الگوریتم ژنتیک رخ می‌دهد.

همان طور که مشاهده می‌کنید الگوریتم بررسی همبستگی کارهای پیشین در زمان بهتری به جواب می‌رسد، به دلیل این که تنها در حالت تریبی ترکیب وبسرویس همبستگی را بررسی می‌کند و در بقیه حالت‌ها همبستگی را بررسی نمی‌کند و از الگوریتم پایه برای یافتن جواب استفاده می‌کند، ولی جواب بهینه را پیدا نمی‌کند زیرا در حالت تریبی از الگوریتم همبستگی و در بقیه حالت‌ها از الگوریتم پایه استفاده می‌کند. جوابی که این الگوریتم در ترکیب پیچیده وبسرویس‌ها که شامل تمامی شکل‌های ترکیب است، پیدا می‌کند دقیقاً شبیه الگوریتم پایه است زیرا در این گونه مسایل دقیقاً شبیه الگوریتم پایه عمل کرده و دنبال جواب می‌گردد. برتری الگوریتم بررسی همبستگی کارهای پیشین نسبت به الگوریتم پایه تنها در مواردی است که تمام وبسرویس‌ها به صورت تریبی با هم ترکیب شده باشند.

همان طور که در این آزمایش مشاهده نمودید به میزانی که درصد همبستگی بین وبسرویس‌ها افزایش پیدا می‌کند، امکان انتخاب وبسرویس‌هایی که با وبسرویس‌های دیگر همبسته بوده و ویژگی کیفی بهتری دارند نیز افزایش پیدا می‌کند. در نتیجه مجموع ویژگی کیفی یا میزان برازندگی وبسرویس مرکب نهایی که از ترکیب این وبسرویس‌های همبسته به دست می‌آید بهبود می‌یابد (شکل ۱۸). البته



شکل ۱۸: آزمایش بررسی تأثیر افزایش درصد همبستگی بین وبسرویس‌های نامجاور بر میزان برازندگی وبسرویس مرکب یافت‌شده.

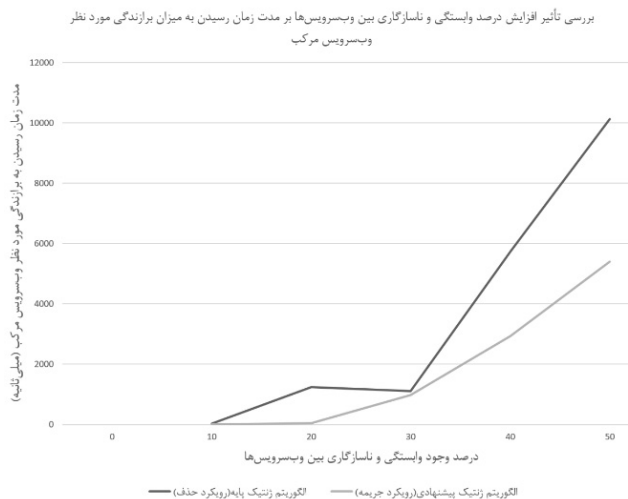
الگوریتم ژنتیک پیشنهادی می‌تواند همبستگی بین وبسرویس‌های نامجاور را تشخیص دهد. بنابراین در محاسبات مربوط به برازندگی وبسرویس مرکب در صورت یافتن همبستگی بین وبسرویس‌های نامجاور از ویژگی کیفی بهبودیافته وبسرویس دارای همبستگی با دیگر وبسرویس‌ها استفاده می‌کند.

الگوریتم ژنتیک پیشنهادی بر پایه الگوریتم ژنتیک ارائه شده و برای این که تنظیمات پارامترها روی نتایج مقایسه تأثیری نگذارد، برای تنظیم پارامترهای الگوریتم ژنتیک پیشنهادی نیز از پارامترهایی که به کمک آن الگوریتم ژنتیک پایه را تنظیم کرده‌ایم استفاده می‌کنیم.

### ۳-۵ آزمایش بررسی تأثیر افزایش همبستگی بین وبسرویس‌ها بر میزان برازندگی وبسرویس مرکب یافت‌شده

در این آزمایش برای جریان کاری نمونه‌ای که در شکل ۱۷ آورده شده است، ارتباط بین درصد وجود همبستگی بین وبسرویس‌های کاندید و میزان برازندگی وبسرویس مرکب یافت‌شده و زمان رسیدن به میزان برازندگی مورد نظر را بررسی می‌کنیم.

همان طور که در بخش قبل اشاره کردیم، با توجه به ویژگی کیفی مورد بررسی، نیازمند بررسی همبستگی بین وبسرویس‌های نامجاور هستیم که برای این حالت الگوریتمی پیشنهاد داده شد. در ادامه آزمایشی برای بررسی الگوریتم همبستگی بین وبسرویس‌های نامجاور انجام داده و نمودار نتایج را ارائه می‌دهیم.



شکل ۲۰: آزمایش بررسی تأثیر افزایش درصد وابستگی و ناسازگاری بین وب‌سرویس‌ها بر مدت زمان رسیدن به میزان برآوردگی مورد نظر وب‌سرویس مرکب.

همان طور که در شکل ۲۰ مشاهده می‌کنید، به میزانی که درصد وابستگی و ناسازگاری بین وب‌سرویس‌ها بیشتر می‌شود، الگوریتم ژنتیک پیشنهادی که از رویکرد دوم که راه‌حل نشدنی را جریمه می‌کند، استفاده می‌کند در مدت زمان بهتری وب‌سرویس مرکب با برآوردگی مورد نظر مشتری را می‌یابد.

در ادامه در جدول ۴ که نشان‌دهنده مقایسه الگوریتم ژنتیک پیشنهادی (که بهبودیافته الگوریتم ژنتیک پایه است) با کارهای پیشین است، آورده شده است.

## ۶- نتیجه‌گیری

الگوریتم ژنتیک پیشنهادی این مقاله برای مسئله انتخاب و ترکیب وب‌سرویس‌ها از همبستگی بین وب‌سرویس‌های نامجاور، وابستگی و ناسازگاری در ترکیب ترتیبی، موازی و حلقوی پشتیبانی کرده و وب‌سرویس مرکب را با مجموع ویژگی‌های کیفی مورد نظر مشتری یا حتی مجموعی بهتر در صورت امکان و انتخاب وب‌سرویس‌های همبسته می‌یابد.

الگوریتم ژنتیک پیشنهادی این مقاله در صورت نقض محدودیت‌های وابستگی و ناسازگاری از رویکرد جریمه استفاده می‌کند و همان طور که در بخش ارزیابی نشان داده شد، استفاده از رویکرد جریمه نسبت به رویکرد حذفی سبب می‌شود که راه‌حل نشدنی را حذف کرده و همین راه‌حل‌های نشدنی به ما در سریع‌تر رسیدن به راه‌حل شدنی کمک کرده و سریع‌تر به وب‌سرویس مرکب مورد نظر مشتری دست می‌یابیم.

الگوریتم ژنتیک پیشنهادی در این مقاله به دلیل ماهیت تصادفی الگوریتم ژنتیک نسبت به الگوریتم‌های دیگری که از فنون مبتنی بر برنامه‌ریزی خطی استفاده می‌کنند و تمام حالات را چک می‌کنند، سریع‌تر بوده و از حافظه کمتری استفاده می‌کند.

## ۷- کارهای آینده

افزایش دادن امکان بررسی وجود همبستگی بین وب‌سرویس‌های مجاور به الگوریتم ژنتیک پیشنهادی به طوری که بتوانیم همبستگی بین وب‌سرویس‌های مجاور را برای ویژگی‌های کیفی دسترس‌پذیری، زمان پاسخ، تعامل‌پذیری و قابلیت اطمینان تشخیص دهیم و بتوانیم بهترین ترکیب وب‌سرویس‌ها را بر اساس ویژگی‌های کیفی در تمام حالات پیچیده ترکیبی (ترتیبی، موازی، شرطی، حلقوی و ترکیبی از این موارد) پیدا کنیم.



شکل ۱۹: آزمایش بررسی تأثیر افزایش درصد همبستگی بین وب‌سرویس‌های نامجاور بر مدت زمان رسیدن به جواب بهینه مورد نظر مشتری.

همان طور که در شکل ۱۹ مشاهده می‌کنید، مدت زمان دستیابی به جواب بهینه مورد نظر مشتری مقداری افزایش پیدا می‌کند.

در این مقاله به بررسی همبستگی بین وب‌سرویس‌ها برای ویژگی‌های کیفی دسترس‌پذیری، زمان پاسخ، تعامل‌پذیری و قابلیت اطمینان که نیازمند بررسی وب‌سرویس‌های مجاور است نمی‌پردازیم و در کارهای آتی به این موارد خواهیم پرداخت.

## ۵-۴ آزمایش بررسی تأثیر افزایش وابستگی و ناسازگاری بین وب‌سرویس‌ها بر مدت زمان رسیدن به میزان برآوردگی مورد نظر وب‌سرویس مرکب

همان طور که اشاره کردیم در صورتی که کروموزوم، وب‌سرویس‌هایی که محدودیت‌های وابستگی یا ناسازگاری را نقض کرده‌اند برای ترکیب با هم انتخاب کرده باشد، دو رویکرد داریم. رویکرد اول این است که این کروموزوم حاوی راه‌حل نشدنی را کامل از جمعیت حذف کنیم که این رویکرد در الگوریتم ژنتیک پایه پیاده‌سازی شده است. رویکرد دوم این است که کروموزوم حاوی راه‌حل نشدنی را جریمه کرده ولی از جمعیت حذف نکنیم زیرا ممکن است در صورت حذف‌نکردن این کروموزوم با عملگرهای تقاطع و جهش ما سریع‌تر به کروموزوم‌های حاوی راه‌حل شدنی دست یافته و جواب مورد نظر مشتری را پیدا کنیم. بنابراین در الگوریتم ژنتیک پیشنهادی رویکرد جریمه‌ای را پیاده‌سازی کرده‌ایم.

در ادامه این دو رویکرد را با هم مقایسه کرده و نشان می‌دهیم که با استفاده از رویکرد دوم در مدت زمان کمتری به راه‌حلی برای انتخاب و ترکیب وب‌سرویس‌های مناسب دست می‌یابیم که میزان برآوردگی مورد نظر مشتری برای وب‌سرویس مرکب را برآورده می‌سازد.

برای این آزمایش شش وب‌سرویس انتزاعی به صورت ترتیبی با هم ترکیب شده‌اند و برای هر وب‌سرویس انتزاعی شش وب‌سرویس کاندید برای انتخاب شدن وجود دارد و وب‌سرویس‌های وابسته و ناسازگار به صورت تصادفی انتخاب می‌شوند. مدت زمان رسیدن به برآوردگی مورد نظر برای وب‌سرویس مرکب برای دو الگوریتم ژنتیک پایه و پیشنهادی محاسبه و با یکدیگر مقایسه می‌شوند.

در این آزمایش با بررسی تأثیر افزایش درصد وابستگی و ناسازگاری بین وب‌سرویس‌ها بر مدت زمان رسیدن به برآوردگی مورد نظر برای وب‌سرویس مرکب، عملکرد دو الگوریتم ژنتیک پایه و پیشنهادی را با هم مقایسه می‌کنیم.



جدول ۴: مقایسه الگوریتم ژنتیک پیشنهادی با کارهای پیشین.

شاخص‌های مورد بررسی	الگوریتم ژنتیک پیشنهادی (روش پیشنهادی)	الگوریتم بررسی همبستگی (کارهای پیشین)	الگوریتم بررسی وابستگی و ناسازگاری (کارهای پیشین)	الگوریتم ژنتیک پایه (کارهای پیشین)
فن استفاده شده برای ترکیب وب سرویس‌ها	فن مبتنی بر طرح‌ریزی هوش مصنوعی	فن مبتنی بر برنامه‌ریزی خطی	فن مبتنی بر طرح‌ریزی هوش مصنوعی	فن مبتنی بر طرح‌ریزی هوش مصنوعی
نحوه انتخاب وب سرویس‌ها نسبت به وب سرویس‌های دیگر انتخاب شده	بهینه سراسری*	بهینه سراسری	بهینه محلی**	بهینه محلی
یافتن وب سرویس مرکب بهینه با پشتیبانی از چند ویژگی کیفی	✓	×	✓	✓
پشتیبانی از کدام حالت‌های مختلف ترکیبی برای یافتن وب سرویس مرکب بهینه	ترتیبی، موازی، شرطی و حلقوی	ترتیبی	ترتیبی، موازی، شرطی و حلقوی	ترتیبی، موازی، شرطی و حلقوی
امکان بررسی وجود همبستگی بین وب سرویس‌های نامجاور	✓ (ترتیبی، موازی، شرطی و حلقوی)	✓ (ترتیبی)	×	×
امکان بررسی وجود وابستگی و ناسازگاری بین وب سرویس‌ها	✓ (ترتیبی، موازی، شرطی و حلقوی)	×	✓ (ترتیبی، موازی، شرطی و حلقوی)	×

\* در انتخاب وب سرویس، وب سرویس‌هایی که قبل تر و بعدتر انتخاب شده‌اند، در نظر گرفته می‌شوند.  
 \*\* در انتخاب وب سرویس، وب سرویس‌هایی که قبل تر و بعدتر انتخاب شده‌اند، در نظر گرفته نمی‌شوند.

[۱۳] ش. پویا، "ترکیب وب سرویس‌ها بر اساس معیارهای کیفی سرویس با استفاده از الگوریتم ژنتیک"، *اولین همایش ملی رویکردهای نوین در مهندسی کامپیوتر و بازیابی اطلاعات ایران*، ۸ صص، رشت، ایران، ۱۷-۱۷ مهر ۱۳۹۲.

- [14] D. Wang, Y. Yang, and Z. Mi, "A genetic-based approach to web service composition in geo-distributed cloud environment," *Computers & Electrical Engineering*, vol. 43, no. C, pp. 129-141, Apr. 2015.
- [15] F. Gao, E. Curry, M. I. Ali, S. Bhiri, and A. Mileo, "QoS-aware complex event service composition and optimization using genetic algorithms," in *Proc. Int. Conf. on Service-Oriented Computing, ICSSOC'14*, pp. 386-393, Paris, France, 3-6 Nov. 2014.
- [16] M. C. Jaeger and G. Muhl, "QoS-based selection of services: the implementation of a genetic algorithm," in *Proc. Communication in Distributed Systems, 15 ITG-GI Symp.*, 12 pp., Bern, Switzerland, 26 Feb. 2 Mar.. 2007.
- [17] S. Su, C. Zhang, and J. Chen, "An improved genetic algorithm for web services selection," in *Proc. 7th IFIP WG 6.1 Int. Conf. on Distributed Applications and Interoperable Systems*, pp. 284-295 Paphos, Cyprus, 6-8 Jun. 2007.
- [18] M. Tang and L. Ai, "A hybrid genetic algorithm for the optimal constrained web service selection problem in web service composition," in *Proc. IEEE Congress on Evolutionary Computation*, 8 pp., Barcelona, Spain, 18-23 Jul. 2010.
- [19] G. Canfora and M. Di Penta, "A framework for QoS-aware binding and re-binding of composite web services," *J. of Systems and Software*, vol. 81, no. 10, pp. 1754-1769, Oct. 2008.
- [20] F. Mardukhi, N. Nematbakhsh, K. Zamanifar, and A. Barati, "QoS decomposition for service composition using genetic algorithm," *Applied Soft Computing*, vol. 13, no. 7, pp. 3409-3421, Jul. 2013.
- [21] L. J. Zhang and B. Li, "Requirements driven dynamic services composition for web services and grid solution," *J. of Grid Computing*, vol. 2, no. 2, pp. 121-140, Jun. 2004.
- [22] G. Canfora, M. Di Penta, R. Esposito, and M. L. Villani, "An approach for QoS-aware service composition based on genetic algorithms," in *Proc. of the 7th Annual Conf. on Genetic and Evolutionary Computation, ACM*, pp. 1069-1075, Washington DC, USA, 25-29 Jun. 2005.
- [23] S. Deng, H. Wu, D. Hu, and J. Zhao, "Service selection for composition with QoS correlations," *IEEE Trans. on Services Computing*, vol. 9, no. 2, pp. 291-302, Mar.-Apr. 2016.
- [24] L. Ai and M. Tang, "A penalty-based genetic algorithm for QoS-aware web service composition with inter-service dependencies and conflicts," in *Proc. IEEE Int. Conf. on Computational Intelligence for Modelling Control & Automation*, pp. 738-743, Vienna, Austria, 10-12 Dec. 2008.

## مراجع

- [1] S. Y. Lina, C. H. Laia, C. H. Wub, and C. C. Lob, "A trustworthy QoS-based collaborative filtering approach for web service discovery," *J. of Systems and Software*, vol. 93, pp. 217-228, Jul. 2014.
- [2] P. Wang, K. M. Chao, and C. C. Lo, "On optimal decision for QoS-aware composite service selection," *Expert Systems with Applications*, vol. 37, no. 1, pp. 440-449, Jan. 2010.
- [3] A. E. Yilmaz and K. Pinar, "Improved genetic algorithm based approach for QoS aware web service composition," in *Proc. IEEE Int. Conf. on Web Services, ICWS'14*, pp. 463-470, Anchorage, AK, USA, 27 Jun.-2 Jul. 2014.
- [4] D. Ardagna and B. Pernici, "Adaptive service composition in flexible processes," *IEEE Trans. on Software Engineering*, vol. 33, no. 6, pp. 369-384, Jun. 2007.
- [5] U. Shehu, G. Epiphaniou, and G. A. Safdar, "A survey of QoS-aware web service composition techniques," *International J. of Computer Applications*, vol. 89, no. 12, pp. 10-17, Mar. 2014.
- [6] M. A. Amiri and H. Serajzadeh, "QoS aware web service composition based on genetic algorithm," in *Proc. 5th Int. Symp. on Telecommunications, IST'10*, pp. 502-507, Tehran, Iran, 4-6 Dec. 2010.
- [7] M. Alrifai, T. Risse, and W. Nejdl, "A hybrid approach for efficient web service composition with end-to-end QoS constraints," *ACM Trans. on the Web*, vol. 6, no. 2, Article 7, May 2012.
- [8] L. Wang, J. Shen, and J. Yong, "A survey on bio-inspired algorithms for web service composition," in *Proc. IEEE 16th Int. Conf. on Computer Supported Cooperative Work in Design, CSCWD12*, pp. 569-574, Wuhan, China, 23- 25 May 2012.
- [9] D. Silva, A. Sawczuk, E. Moshi, H. Ma, and S. Hartmann, "A QoS-aware web service composition approach based on genetic programming and graph databases," in *Proc. Int. Conf. on Database and Expert Systems Applications, DEXA'17*, pp. 37-44, Lyon, France, 28-31 Aug. 2017.
- [10] J. Zhou and X. Yao, "A hybrid artificial bee colony algorithm for optimal selection of QoS-based cloud manufacturing service composition," *The International J. of Advanced Manufacturing Technology*, vol. 88, no. 9-12, pp. 3371-3387, Feb. 2017.
- [11] M. Zavvar, et al., "Measuring service quality in service-oriented architectures using a hybrid particle swarm optimization algorithm and artificial neural network (PSO-ANN)," in *Proc. 3th Int. Conf. on Web Research, ICWR'17*, pp. 78-83, Tehran, Iran, 19-20 Apr. 2017.
- [12] A. Jula, Z. Othman, and E. Sundararajan, "Imperialist competitive algorithm with PROCLUS classifier for service time optimization in cloud computing service composition," *Expert Systems with Applications*, vol. 42, no. 1, pp. 135-145, Jan. 2015.



**فریدون شمس علیئی** در سال ۱۳۶۶ مدرک کارشناسی مهندسی کامپیوتر با گرایش نرم‌افزار خود را از دانشگاه شهید بهشتی و در سال ۱۳۶۹ مدرک کارشناسی ارشد مهندسی کامپیوتر با گرایش نرم‌افزار خود را از دانشگاه صنعتی شریف دریافت نمود. فعالیت آموزشی خود را از سال ۱۳۶۸ برای تدریس درس مهندسی نرم‌افزار در دانشگاه صنعتی شریف آغاز کرد. در سال ۱۳۷۱ برای ادامه تحصیل دکترای علوم کامپیوتر در دانشگاه منچستر انگلستان وارد گردید و در سال ۱۳۷۵ موفق به اخذ درجه دکترا در مهندسی نرم‌افزار از دانشگاه مذکور گردید. دکتر شمس از سال ۱۳۷۶ در دانشکده مهندسی کامپیوتر دانشگاه شهید بهشتی مشغول به فعالیت آموزشی گردید و اینک نیز دانشیار این دانشکده می‌باشد. وی همچنین مبدع رشته تحصیلی جدیدی تحت عنوان گرایش معماری سازمانی در کارشناسی ارشد مهندسی فناوری اطلاعات است. زمینه‌های علمی مورد علاقه نام‌برده شامل مهندسی نرم‌افزار، معماری نرم‌افزار، معماری سازمانی، معماری سرویس‌گرا، سیستم‌های مقیاس وسیع و مهندسی فرایند است.

**مهدی فرزندوی** در سال ۱۳۹۴ مدرک کارشناسی مهندسی کامپیوتر گرایش نرم‌افزار خود را از دانشگاه شهید بهشتی و در سال ۱۳۹۷ مدرک کارشناسی ارشد فناوری اطلاعات گرایش معماری سازمانی خود را از دانشگاه شهید بهشتی دریافت نمود. زمینه‌های علمی مورد علاقه نام‌برده شامل موضوعاتی مانند معماری نرم‌افزار، معماری سازمانی، معماری سرویس‌گرا، یکپارچه سازی سرویس می‌باشد.