علوم رایانش و فناوری اطلاعات
نشریه علمی انجمن کامپیوتر ایران
مجلد ۱۳، شماره۱، ۱۳۹۴
صفحات ۵۴-۶۴
مقاله عادی

نشریه علوم رایانش و فناوری اطلاعات

به سوی ارزیابی سیستم‌های خودتطبیق

اسلام ناظمی[۱] حسین صدر[۲] مژده نظری سلیماندارابی[۳]

[۱]دانشکده مهندسی و علوم کامپیوتر، دانشگاه شهید بهشتی، تهران، ایران
[۲]دانشکده مهندسی برق، رایانه و فناوری اطلاعات، دانشگاه آزاد اسلامی واحد قزوین، قزوین، ایران
[۳]دانشکده فنی و مهندسی، دانشگاه آزاد اسلامی واحد رشت، رشت، ایران

چکیده

در طول چند دهه گذشته استفاده از سیستم‌های خودتطبیق در حوزه مهندسی سیستم‌های نرم‌افزاری به صورت قابل توجهی افزایش یافته است. در همین راستا و برای پیشرفت در حوزه سیستم‌های خودتطبیقی و خودمدیریتی و استفاده از تمامی مزایا و قابلیت‌های آن‌ها به ابزارها و روش‌هایی نیاز است که بتواند سیستم‌های خودتطبیق را نه تنها در زمان طراحی، بلکه در زمان اجرا نیز مورد ارزیابی قرار دهد، زیرا ارزیابی مهمترین عامل برای درک موفقیت هر سیستم نرم‌افزاری است. علی‌رغم استفاده رو به رشد از سیستم‌های خودتطبیقی، ارزیابی این نوع سیستم‌ها کمتر مورد توجه قرار گرفته است و اکثر پژوهش‌های انجام شده در این حوزه به صورت موردی بوده و خاصیت تغییرپذیری این نوع سیستم‌ها را در زمان اجرا مد نظر قرار نمی‌دهد. از طرف دیگر، ویژگی‌های کیفی و متریک‌های مرتبط با آن‌ها که برای ارزیابی سیستم‌های خودتطبیق معرفی شده‌اند، محدود بوده و کلیه خصوصیات انطباقی این نوع سیستم‌ها را در نظر نمی‌گیرند. برای حل این مشکلات، در این مقاله چارچوبی برای ارزیابی سیستم‌های خودتطبیق معرفی شده است که براساس تجزیه و تحلیل دقیق طیفی از کارهای انجام شده در این حوزه بدست آمده است. به همین منظور، در ابتدا خصوصیات انطباق که هدف اصلی ایجاد سیستم‌های خودتطبیقی‌اند شناسایی شده و به ویژگی‌های کیفی متناظرشان که مهمترین معیار برای ارزیابی سیستم‌های نرم‌افزاری می‌باشند، نگاشت شده‌اند. در ادامه نیز متریک‌های کیفیت نرم‌افزار مرتبط با این ویژگی‌های کیفی معرفی شده است. باید در نظر داشت که این ویژگی‌های کیفی و متریک‌های مرتبط با آن‌ها مختص سیستم‌های خودتطبیق و براساس ساختار و نحوه عملکرد آن‌ها می‌باشند و چارچوب پیشنهادی نیز می‌تواند به عنوان یک زیربنای جامع برای ارزیابی سیستم‌های خودتطبیق در نظر گرفته شود.

کلمات کلیدی: سیستم‌های خودتطبیق، ارزیابی، صحت‌سنجی و اعتبارسنجی، کارایی، قابلیت اطمینان، امنیت.

۱- مقدمه

در طول دهه گذشته خودتطبیقی به یک دغدغه اساسی در حوزه مهندسی سیستم‌های نرم‌افزاری به منظور کاهش هزینه بالای نگهداری و تکامل نرم‌افزار و تامین رضایت کاربران از نیازمندی‌های تابعی سیستم‌هایی که در هنگام اجرا با محیط‌های متغیر مواجه هستند، تبدیل شده است. اگرچه مکانیزم‌های تطبیق به طور گسترده‌ای در حوزه مهندسی سیستم‌های نرم‌افزاری پویا مورد بررسی قرار گرفته‌اند، کاربرد آنها در محیط‌های واقعی به علت عدم وجود روشی جامع برای ارزیابی عملکرد آنها هنوز با محدودیت‌هایی مواجه است [۱].

برای سیستم‌های ذاتا غیرتطبیقی که شرایط و محیط اجرای سیستم کاملا شناخته‌شده و پایدار است، تکنیک‌ها و ابزارهای مختلف ارزیابی توسعه یافته‌اند تا در زمان طراحی اعمال شوند. در مقابل ارزیابی کیفیت سیستم‌های خود تطبیقی با توجه به اینکه اهداف و نیازمندی‌های آن‌ها در زمان اجرا و با توجه به محیطی که در آن فعالیت می‌کنند مرتبا در حال تغییر است و این تغییرات منجر به ایجاد تغییر در ساختار سیستم می‌شود، کاری بسیار چالش پذیر است. یکی از مهمترین چالش‌های پیش‌روی روش‌های ارزیابی خودتطبیق انفجار فضای حالات[۱] در این سیستم‌ها است [۲]. تعداد حالات ورودی در این سیستم‌ها ممکن است آنقدر بزرگ باشد که امکان ارزیابی مستقیم این نوع سیستم‌ها وجود نداشته باشد.

مشکل دیگر نیازمندی‌های متغیر با زمان اجرا در این نوع سیستم‌ها است. در سیستم‌های سنتی نیازمندی‌های سیستم در زمان طراحی مشخص شده و براساس آن‌ها پیکربندی و طراحی سیستم صورت می‌گیرد. اما در این نوع سیستم‌ها نیازمندی‌های سیستم در زمان اجرا و با توجه به تغییرات زمینه تغییر می‌یابند. تغییرات در نیازمندی‌ها باعث تغییر در ویژگی‌های کیفی یک سیستم که فاکتور اصلی ارزیابی آن به شمار می‌آیند، می‌شود. از طرف دیگر، باید مد نظر داشت که خصوصیات و ویژگی‌های سیستم‌های خودتطبیق با سیستم‌های سنتی متفاوت است و نمی‌توان از ویژگی‌های کیفی سیستم‌های سنتی برای ارزیابی آن‌ها بهره برد [۲، ۳].

در طول دهه گذشته رویکردها و سیستم‌های خودتطبیق مختلفی توسط محققان این حوزه معرفی شده است که هرکدام از آن‌ها جنبه مشخصی از اهداف سیستم‌های خودتطبیق را پوشش داده و می‌توانند در محیط‌هایی که در آن‌ها نیازمندی‌ها، مدل و محیط اجرایی در زمان اجرا تغییر می‌کنند، به فعالیت بپردازند [۴، ۵]. پس از تجزیه و تحلیل دقیق رویکردهای ارائه شده در این حوزه، این نتیجه حاصل شده است که خصوصیات انطباق و متریک‌های متناظر با آن‌ها به ندرت در مقالات مرتبط با مفهوم سیستم‌های خودتطبیق شناسایی و آدرس‌دهی شده‌اند.

در واقع متریک‌های کیفی معرفی شده متناسب با زمان ارزیابی و مستقل از ارزیاب نبوده و وابسته به تخصص و تجربه ارزیاب در زمینه سیستم مورد نظر می‌باشند. این مساله منجر به ایجاد عدم وجود جامعیت و وجود خروجی‌های مختلف برای طرح‌های یکسان از سیستم‌های نرم‌افزاری می‌شود. با توجه به این محدودیت‌ها نیاز به روش‌هایی برای صحت‌سنجی و اعتبار سنجی سیستم‌های خودتطبیقی در زمان اجرا و شناسایی خصوصیات کیفی مرتبط با این نوع سیستم‌ها بیش از پیش احساس می‌شود، زیرا در صورت عدم شناخت از خصوصیات انطباقی و ویژگی‌های کیفی متناظر با آن‌ها ارزیابی سیستم‌های خودتطبیقی غیر ممکن خواهد بود [۲، ۶].

برای حل این مشکلات در این مقاله با بررسی و تجزیه و تحلیل دقیق کارهای انجام شده در این حوزه، خصوصیات انطباقی سیستم‌های خودتطبیق شناسایی شده و ویژگی‌های کیفی متناظر با آن‌ها معرفی شده است. به بیان دیگر، در این مقاله ابتدا خصوصیات انطباق در سیستم‌های خودتطبیقی شناسایی شده و به ویژگی‌های کیفی متناظر با آن‌ها که مهم‌ترین فاکتور در ارزیابی یک سیستم نرم‌افزاری می‌باشند، نگاشت شده‌اند و در ادامه نیز متریک‌های مناسب برای ارزیابی این ویژگی‌های کیفی معرفی شده است.

برای تعیین خصوصیات انطباقی رویکردهای خودتطبیقی موجود در این حوزه و خصوصیات مورد استفاده در تئوری کنترل مورد بررسی قرار گرفته‌اند. این خصوصیات و متریک‌ها از تئوری کنترل استخراج شده براساس سیستم‌های خودتطبیق دوباره تفسیر شده‌اند. باید در نظر داشت که تفسیر خصوصیات موجود براساس سیستم‌های خودتطبیقی کار ساده‌ای نیست زیرا معنای مفاهیم موجود در سیستم‌های خودتطبیقی با سیستم‌های سنتی متفاوت بوده و سیستم‌های خودتطبیقی جزء سیستم‌های غیر خطی به حساب می‌آیند که محیط اطراف آن‌ها همواره در حال تغییر است [۷].

ادامه مقاله به صورت زیر سازماندهی شده است. در بخش دوم فاکتورهای موثر در ارزیابی سیستم‌های خودتطبیق مورد بررسی قرار گرفته‌اند. خلاصه‌ای از کارهای انجام شده در حوزه ارزیابی سیستم‌های خودتطبیق در بخش سوم آمده است. چارچوب پیشنهادی این مقاله برای ارزیابی سیستم‌های خودتطبیق با استفاده از خصوصیات انطباق، ویژگی‌های کیفی و متریک‌های متناظر با آن در بخش چهارم معرفی شده است. بحث‌های مرتبط با یافته‌های تجزیه و تحلیل‌های مقاله، چالش‌های پیش‌رو و نتیجه‌گیری مقاله در بخش چهارم آمده است.

۲- فاکتورهای کلیدی برای ارزیابی سیستم‌های خودتطبیق

هدف از ارزیابی کیفیت نرم‌افزار مشخص کردن توانایی نرم‌افزار در برآورده کردن نیازها و اهداف از پیش تعیین شده است که به عنوان نقطه شروع برای ارزیابی سیستم‌ها به شمار می‌آیند. باید مد نظر داشت که در سیستم‌های خودتطبیقی با توجه به اینکه محیط اطرافشان همواره در حال تغییر است، شناسایی نیازمندی‌ها و اهداف به سادگی سیستم‌های سنتی نیست. به بیان دیگر، محیط اجرایی سیستم‌های خودتطبیق در زمان اجرا تغییر می‌یابد و تغییرات در محیط منجر به ایجاد تغییرات در خود سیستم نیز می‌گردد. در نتیجه ارزیابی سیستم در زمان طراحی نمی‌تواند پاسخگوی تمامی نیازهای سیستم و مشخص کننده تمامی ویژگی‌های کیفی مناسب برای ارزیابی آن باشد، زیرا تغییرات در زمان اجرا منجر ممکن است منجر به تغییر معماری و ساختار سیستم شود [۶]. درنتیجه، نیاز به فاکتورهایی است که مشخص کنند که چه چیزی، در چه جایی و در چه زمانی از چرخه حیات سیستم‌های خودتطبیق باید مورد ارزیابی قرار گیرد. این فاکتورها با مقایسه سیستم‌های خودتطبیق و سیستم‌هایی که در طول اجرا تغییر ناپذیرند و بررسی دغدغه‌هایی که هنگام مواجه با تغییرات محیطی افزایش می‌یابند، مشخص می‌شوند. به طور کلی هدف اصلی این بخش پاسخ دادن به این سه سوال اساسی است که چه چیزی در سیستم‌های خودتطبیق باید مورد ارزیابی قرار گیرد؟ این ارزیابی باید در چه زمانی صورت گیرد؟ در چه جایی در چرخه انطباق این نوع سیستم‌های باید ارزیابی انجام شود؟ این سوالات موارد چالش‌برانگیزی هستند که به وابستگی این نوع سیستم‌ها به زمان اجرا اشاره دارند و پاسخ این سوالات می‌تواند به شناسایی ویژگی‌های کیفی مناسب برای ارزیابی سیستم‌های خودتطبیق و متریک‌های متناظر با آن‌ها کمک کند [۸]. پاسخ این سوالات در ادامه آمده است.

۲-۱- چه چیزی: نیازمندی‌ها و خصوصیات انطباق

برای ارزیابی سیستم‌های خودتطبیق ابتدا باید مشخص شود که چه چیزهایی در این نوع سیستم‌ها باید مورد ارزیابی قرار گیرند. پاسخ به این سوال منوط به شناسایی اهداف انطباق (نیازمندی‌های غیرتابعی سیستم هدف) و خصوصیات انطباق (ویژگی‌های مورد انتظار مکانیزم‌های انطباق) است. شناسایی اهداف و خصوصیات انطباق به منظور ایجاد مدل مناسب برای ارزیابی سیستم‌های خودتطبیق وشناسایی متریک‌های مرتبط با این نوع سیستم‌ها ضروری است [۴].

دغدغه‌های تکنیک‌های صحت‌سنجی و اعتبارسنجی برای ارزیابی سیستم‌های خودتطبیقی می‌توانند براساس دو قسمت اصلی یک سیستم خودتطبیقی دسته‌بندی شوند. دسته اول به اعتبارسنجی سیستم هدف و دسته دوم به اعتبارسنجی مکانیزم تطبیق باز می‌گردد. به بیان دیگر، برای ارزیابی یک سیستم خودتطبیقی، ویژگی‌های مورد انتظار سیستم هدف (اهداف تطبیق) و مکانیزم‌های تطبیق (خصوصیات انطباق) صراحتا شناسایی شده و به عنوان ویژگی‌های کیفی شناخته می‌شوند. لازم به ذکر است که بیشتر خصوصیات شناخته شده برای سیستم‌های خودتطبیقی از تئوری کنترل[۲] گرفته شده و براساس نرم‌افزارهای خودتطبیق تفسیر شده‌اند [۴، ۶].

از طرف دیگر، خصوصیات انطباق را می‌توان براساس اینکه چگونه و کجا مشاهده می‌شوند دسته‌بندی کرد. با در نظر گرفتن اینکه ویژگی‌ها در کدام قسمت از سیستم خودتطبیق مشاهده می‌شوند، برخی از ویژگی‌ها می‌توانند با استفاده از تکنیک‌های صحت‌سنجی ایستا ارزیابی شوند، درحالیکه بقیه نیازمند صحت‌سنجی پویا و نظارت در زمان اجرا هستند. با توجه به مکان مشاهده، ویژگی‌ها نیز

می‌توانند روی سیستم هدف و یا مکانیزم تطبیق ارزیابی شوند. اگرچه، برخی از ویژگی‌ها تنها روی سیستم هدف قابل مشاهده‌اند اما از آن‌ها برای ارزیابی مکانیزم‌های انطباق نیز استفاده می‌شود [۴].

باید در نظر داشت که در صورت عدم وجود متریک‌های صریح و مناسب که بتوان از آن‌ها برای ارزیابی خصوصیات انطباق و ویژگی‌های متناظر با آن‌ها استفاده کرد، نمی‌توان قابلیت‌های سیستم‌های خودتطبیقی و اهمیت ارزیابی در زمان اجرا را نشان داد. از طرف دیگر، اگرچه اهمیت داشتن چنین متریک‌های صریحی واضح است، یک مانع مهم برای ارزیابی پویا این نوع سیستم‌ها کمبود متریک‌های متناسب با ساختار انطباقی است. برای حل این مشکل، این مقاله به دنبال ارائه چارچوبی برای مشخص کردن خصوصیات انطباقی، ویژگی‌های کیفی و متریک‌های متناظر با آن‌ها است.

۲-۲- کجا: جداسازی دغدغه‌ها

همان‌طور که پیش از این اشاره شد، در سیستم‌های خودتطبیقی دو سطح شناسایی شده است. یکی از این سطوح سیستم هدف است که به صورت خودکار براساس تغییرات زمینه تطبیق می‌یابد و سطح دیگر مکانیزم تطبیق است. برای تکنیک‌های صحت‌سنجی و اعتبارسنجی زمان اجرا، مشخص کردن این سطوح و جداسازی مرز و محدوده آن‌ها ضروری است. جداسازی این دغدغه‌ها به ما اجازه توصیف، بررسی و تجزیه و تحلیل مشکلات تحقیقاتی ارزیابی سیستم‌های خودتطبیقی را به طور موثر با تمرکز روی دغدغه‌های هر سطح به طور مشخص فراهم می‌سازد.

باید در نظر داشت که کنترل در یک سیستم خودتطبیق می‌تواند هم به صورت حلقه بازخورد[۳] و هم جلورو[۴] باشد [۴، ۹]. اما از آنجاییکه روش‌های ارزیابی برخط در سیستم‌های خودتطبیق روی سیستم هدف و مکانیزم‌های تطبیق تکیه دارند، همواره روی حلقه بازخورد در این سیستم‌ها برای ارزیابی تاکید شده است. در نتیجه، خروجی‌های اندازه‌گیری شده برای ایجاد یک سیستم با قابلیت تطبیق در زمان اجرا و با کیفیت بالا از اهمیت بالایی برخوردار هستند. به بیان دیگر، سیستم هدف خودش از موقعیت زمینه و عملکردش با درنظرگرفتن رضایت از نیازمندی‌های کارکردی و غیرکارکردیش آگاهی ندارد. در نتیجه، با توجه به اینکه هدف ارزیابی ضمانت کیفیت یک سیستم است و کیفیت به عنوان فاکتور تکمیل نیازمندی‌های یک سیستم به حساب می‌آید، در سیستم‌های خودتطبیقی وظایف صحت‌سنجی و اعتبارسنجی باید به عنوان بخشی از حلقه انتخاب در نظر گرفته شوند. این مساله به این معنی است که علاوه بر اجزای نظارت معمول دامنه که در حلقه تطبیق خودکار در نظر گرفته می‌شوند، اجزای اضافی مرتبط با صحت‌سنجی و تست سیستم هدف نیز مورد نیاز است [۶]. در نتیجه نیاز به متریک‌هایی است که بتوان آن‌ها را روی حلقه انتخاب اندازه‌گیری کرد.

۲-۳- چه زمانی: صحت‌سنجی و اعتبارسنجی در فرآیند تطبیق

استراتژی‌های سنتی ارزیابی، بررسی و تست را قبل از توسعه سیستم و در شرایط مناسب از پیش تعیین شده از عملیات سیستم انجام می‌دهند. این فرآیند بررسی معمولا به صورت خودکار با استفاده از بررسی مدل، اثبات تئوری و ابزار تست صورت می‌گیرد. برای نیازمندی‌های مستقل از دامنه روش‌های سنتی ارزیابی و تکنیک‌های اعتبارسنجی مختلفی معرفی شده‌اند که قبل از اینکه سیستم وارد محیط شود، اعمال می‌شوند. اما اعمال این روش‌ها در زمان اجرا و هنگامی که سیستم براساس محیط‌های پیچیده تطبیق می‌یابد، بسیار هزینه‌بر است [۱].

از طرف دیگر، متغیرهای وابسته به دامنه در زمان طراحی نامحدود و در زمان اجرا محدود می‌باشند. در نتیجه اعمال روش‌های ارزیابی روی این متغیرها در زمان اجرا برای کاهش موثر فضای صحت‌سنجی بسیار با ارزش است. براساس این دیدگاه، مشخص کردن زمانی از فرآیند تطبیق که در آن باید عملیات صحت‌سنجی و اعتبارسنجی اعمال شوند، ویژگی‌های مورد انتظار از سیستم ضمانت شده و از عملیات نامطمئن جلوگیری شود، از اهمیت بالایی برخوردار است [۶]. همان‌طور که پیش از این گفته شد، کمبود روش‌های موثر ارزیابی در زمان اجرا یکی از بزرگترین موانع و چالش‌های اساس برای طیف وسیع سیستم‌های خودتطبیقی به شمار می‌آید، زیرا متریک‌هایی که برای ارزیابی این نوع سیستم‌ها معرفی شده‌اند قابلیت ارزیابی عملکرد سیستم در زمان اجرا را ندارند.

۳- کارهای پیشین

با اینکه مفهوم سیستم‌های خودتطبیقی در چند دهه گذشته باعث ایجاد انقلابی در توسعه سیستم‌های نرم‌افزاری گشته است، اما هنوز توسعه‌دهندگان نرم‌افزار به این سیستم‌ها اعتماد نداشته و آن‌ها جایگزین مناسبی برای سیستم‌های نرم‌افزاری سنتی نمی‌دانند. علت این امر، عدم اطمینان، نبود مکانیزم و روشی جامع برای ارزیابی سیستم‌های خودتطبیقی است.

در چندین دهه گذشته تلاش‌های کمی در راستای ارزیابی سیستم‌های خودتطبیق صورت گرفته است و هرکدام از کارهایی که در این زمینه انجام شده‌اند، از بعدهای مختلفی روی موضوع ارزیابی تمرکز کرده‌اند. به عنوان مثال، منگ[۵] یک نگاشت از مفاهیم پایه‌ای تئوری کنترل به سیستم‌های نرم‌افزاری خودتطبیقی معرفی کرده است [۱۰]. براساس ایده وی، خصوصیات اساسی برای ارزیابی سیستم‌های خودتطبیقی پایداری[۶] و مقاوم بودن[۷] می‌باشند. این دو خصوصیت براساس اینکه به چه چیزی در پارادایم‌های برنامه نویسی، سبک‌های معماری، پارادایم‌های مدل‌سازی و اصول مهندسی نرم‌افزار اشاره می‌کنند، تجزیه و تحلیل و مشخص شده‌اند. لازم به ذکر است که مدل ارائه شده توسط وی تنها توصیفی بود و روی هیچ رویکردی اعمال نشد [۴]. در طبقه‌بندی ارائه شده توسط صالحی و تحویلداری نیز چندین پروژه که مفهوم سیستم‌های خودتطبیقی را براساس دغدغه‌های چه‌گونه، چه چیز، چه زمانی و کجا طبقه‌بندی کردند، مورد ارزیابی قرار گرفتند. آنها همچنین یک نگاه سلسه مراتبی از خصوصیات خود-* را معرفی کرده و روی رابطه آنها با فاکتورهای کیفی سیستم‌های نرم‌افزاری بحث کردند. اگرچه محدوده تحقیق آنها شامل شناسایی متریک‌هایی برای ارزیابی سیستم‌های خودتطبیقی براساس خصوصیات انطباقی و ویژگی‌های کیفی شناخته شده نمی‌شود [۸].

از طرف دیگر، با توجه به اینکه تعریف یکتایی برای سیستم‌های خودتطبیق موجود نیست، ارزیابی و مقایسه آن‌ها کار دشواری است. علاوه بر این، ساختار روبه جلوی این سیستم‌ها نیز پیچیدگی‌هایی را به ارزیابی آن‌ها افزوده است. برای مقابله بر این چالش‌ها، می‌توان ارزیابی سیستم‌های خودتطبیقی را از دیدگاه معماری سیستم خودتطبیقی انجام داد [۱۱]. باید در نظر داشت که معماری یک سیستم نرم‌افزاری خواستگاه خصوصیات کیفی آن است و ملاک برآورده‌سازی یا عدم برآورده‌سازی یک خصوصیت کیفی به نحوه طراحی معماری بستگی دارد. از نظر آقای مک‌کان[۸] و براساس مدل معماری سیستم‌های خودتطبیقی که توسط وی معرفی شده است، یک معماری نرم‌افزار خودتطبیق باید خصوصیات کیفی مانند کیفیت سرویس، هزینه، استحکام، تطبیق‌پذیری، زمان تطبیق و پایداری را برآورده سازد. نکته بسیار مهم این است که در این مقاله تنها خصوصیات کیفی که به طور مستقیم از معماری سیستم قابل اقتباس هستند معرفی شده‌اند و هیچ‌گونه معیار و متریکی برای ارزیابی آن‌ها ارائه نشده است [۱۱].

یک سیستم خودتطبیقی را نیز می‌توان براساس سناریوهای انطباق و به صورت کمی مورد ارزیابی قرار داد [۱۲]. سناریوهای انطباقی توصیفی از رفتار سیستم براساس نیازهای تغییرپذیر است. برخی از سناریوهای انطباقی عبارتند از کاربرد عمومی سیستم، عکس‌العمل سیستم در شرایط غیرمعمول و تغییرپذیر و پیش‌بینی تغییرات آینده در سیستم که همگی مرتبط با تطبیق‌پذیری می‌باشند. برای ارزیابی سیستم‌های خودتطبیقی براساس سناریو دو متریک معرفی شده است که پس از مطرح شدن سناریو تطبیقی، این دو متریک روی معماری‌های مختلف نمونه اعمال می‌شوند و در نهایت هرکدام از معماری‌ها که نمره بهتری بدست آورده باشد، به عنوان معماری منتخب انتخاب خواهد شد. لازم به ذکر است که سناریوهای انطباقی مبتنی بر نیاز کاربران سیستم بوده و مستقل از نیازهای انطباقی هستند که در سیستم‌های خودتطبیقی وجود دارند [۱۲].

روش دیگر برای ارزیابی سیستم‌های خودتطبیقی وابسته به تجربه و تخصص ارزیاب است. در روش‌های ارزیابی تجربی، طراحان براساس تجارب خود محیط مصنوعی را متناسب با نوع کاربرد نرم‌افزار ایجاد می‌کنند و پس از اجرای شبیه‌ساز انتظار دارند که به نتیجه مدنظر برسند. اگر سیستم در محیط شبیه‌سازی شده به درستی عمل کند و به نتیجه مدنظر دست یابد، سیستم به درستی طراحی شده است، در غیر اینصورت طراحی سیستم به درستی صورت نگرفته و باید تغییراتی در آن صورت گیرد. به طور کلی در این نوع ارزیابی، براساس محیطی که سیستم در آن عمل می‌کند می‌توان از تجربیات گوناگونی استفاده کرد. لازم به ذکر است که محیط‌هایی که سیستم‌های تطبیق‌پذیر در آن عمل می‌کنند همواره در حال تغییر است و نمی‌توان یک محیط شبیه‌سازی از پیش‌تعریف شده برای آن‌ها طراحی کرد، در نتیجه استفاده از یک تجربه خاص برای ارزیابی این نوع سیستم‌ها کارآمد نخواهد بود [۱۳].

با توجه به اینکه سیستم‌های خودتطبیق نیاز دارند تا در زمان اجرا و براساس تغییراتی که در اطراف آن‌ها رخ می‌دهد که منجر به ایجاد تغییر در خود این سیستم‌های می‌شود، مورد ارزیابی قرار گیرند، مدلی را برای ارزیابی سیستم خودتطبیق در زمان اجرا معرفی شده است [۶]. براساس این مدل عملیات صحت‌سنجی و اعتبارسنجی یک سیستم خودتطبیق می‌توانند در حلقه انطباق (حلقه MAPE-K) آن صورت می‌گیرد. در این چارچوب همزمان با تطبیق سیستم با تغییرات محیطی، سیستم مورد ارزیابی نیز قرار می‌گیرد. این مدل با اینکه چارچوب مناسبی برای ارزیابی یک سیستم خودتطبیق در زمان اجرا در اختیار محققان قرار می‌دهد، اما هیچ آزمایشی به منظور تعیین درستی آن صورت نگرفته است و این مدل تنها در حد تئوری است. از طرف دیگر، این مدل نیز همانند سایر کارهای معرفی شده تنها نحوه ارزیابی در زمان اجرا را مورد بررسی قرار داده و به ویژگی‌های کیفی و متریک‌های متناظر با آن‌ها اشاره‌ای نکرده است [۶].

با توجه به مطالبی که پیش از این گفته شد، با اینکه تحقیقات مختلفی در زمینه سیستم‌های خودتطبیق و ارزیابی آن‌ها صورت گرفته است، خصوصیات انطباقی و متریک‌های مرتبط با آنها به ندرت در مقالاتی که مرتبط با مهندسی نرم‌افزارهای پویا می‌باشند، شناسایی شده و مورد بررسی قرار گرفته‌اند. ویلگاس[۹] و همکارانش برای اینکه بتوانند خصوصیات یک سیستم خودتطبیقی و ویژگی‌های متناظر با آن‌ها را شناسایی کنند، مدلی معرفی کردند که شامل هشت بعد تجزیه و تحلیلی برای توصیف نرم‌افزار خودتطبیقی است. آن‌ها برای اعتبار بخشیدن به چارچوب پیشنهادی خود طیفی از کارهای انجام شده در حوزه سیستم‌های خودتطبیق رو مورد تجزیه و تحلیل قرار دادند و به کمک آن‌ها خصوصیات انطباق، ویژگی‌های کیفی و متریک‌های متناظر با آن‌ها را شناسایی کردند. با اینکه این مدل می‌تواند به عنوان پایه و اساس ارزیابی سیستم‌های خودتطبیقی به شمار آید، اما خصوصیات و متریک‌های معرفی شده در این مقاله محدود بوده و تمام جنبه‌های سیستم‌های خودتطبیق را در نظر نمی‌گیرد. از طرف دیگر، خصوصیات و ویژگی‌های معرفی شده تنها به صورت تئوریک بوده و هیچ آزمایشی روی آن‌ها به منظور بررسی دقتشان صورت نگرفته است [۴].

همان‌طور که مشخص است، یکی از مهمترین مشکلاتی که روش‌های معرفی شده دارند، نبود خصوصیات کیفی منطبق بر اهداف انطباق و متریک‌های متناظر با آن‌ها برای ارزیابی سیستم‌های خودتطبیقی در زمان اجرای آن‌ها است. اگرچه هرکدام از کارهای انجام شده در این زمینه جنبه خاصی از سیستم‌های خود تطبیق را مورد بررسی قرار داده‌اند، اما بین روش‌های مختلف ارائه شده در این زمینه جامعیت وجود ندارد و ممکن است برای طرح‌های یکسان از سیستم‌های نرم‌افزاری، خروجی‌های مختلفی بدست آید. از طرف دیگر بیشتر متریک‌هایی که برای ارزیابی سیستم‌های خودتطبیق معرفی شده است با مرحله‌ای که ارزیابی در آن در حال اجرا است مطابقت نداشته و نیازمند تجربه و تخصص ارزیاب در زمینه سیستم مورد نظر است و ارزیابی یک معیار از دید دو ارزیاب مختلف ممکن است به دلیل تجارب گوناگون، خروجی‌های مختلفی داشته باشد.

۴- چارچوبی برای ارزیابی سیستم‌های خودتطبیق

قبل از اینکه بخواهیم یک سیستم خودتطبیق را مورد ارزیابی قرار دهیم، ابتدا باید عامل اصلی ایجاد چنین سیستم های را شناسایی کنیم که به آنها اهداف انطباق گفته می‌شود. به طورکلی، انطباق می‌تواند به علت نیاز به وجود رضایت دائمی از نیازهای تابعی و یا تنظیم نیازمندی‌های غیرتابعی در محیط های در حال تغییر رخ دهد [۱۴]. با بررسی کارهای انجام شده در این حوزه می‌توان دریافت که مهم‌ترین دلیل انطباق در سیستم‌ها دستیابی به ویژگی‌های کیفی مورد نظر در این سیستم ها بوده است. از طرف دیگر، خصوصیات سیستم‌های خودتطبیق مانند خودپیکربندی، خودبهینگی نیز می‌توانند به ویژگی‌های کیفی نگاشت شود [۴]. در همین راستا نیز رابطه بین خصوصیات سیستم‌های خودمختار و فاکتورهای کیفی مانند رابطه بین قابلیت اطمینان و خودالتیامی را مورد بررسی قرار داده شده‌اند [۸]. هدف اصلی این مقاله نیز استفاده از ویژگی‌های کیفی متناظر با خصوصیات انطباقی برای ارزیابی سیستم‌های خودتطبیق است.

باید مد نظر قرار داد که ویژگی‌های کیفی بیشتر برای ارزیابی سیستم مدیریت شده مورد استفاده قرار می‌گیرند، به همین دلیل علاوه بر معرفی ویژگی‌های کیفی مناسب برای ارزیابی سیستم های خودتطبیق، در این مقاله نگاشتی بین ویژگی‌های کیفی و خصوصیات انطباق نیز صورت گرفته است که یک روش غیر مستقیم برای ارزیابی خصوصیات انطباقی که بصورت مستقیم روی کنترل کننده قابل مشاهده نیستند، مورد استفاده قرار می‌گیرد [۴]. تعریفی از این خصوصیات کیفی با مراجع مرتبط که آنها را مورد بررسی قرار داده‌اند آمده است.

در ادامه، در ابتدا خصوصیات انطباق که علت اصلی ساخت سیستم های خود تطبیق است به همراه ویژگی‌ها کیفی مؤثر در ارزیابی آن‌ها و مراجع مرتبط معرفی می شوند. در انتها نیز متریک‌های مناسب برای ارزیابی آن‌ها ارایه شده است. لازم به ذکر است که این معیارهای کیفی مرتبط با خصوصیات انطباق با مطالعه مقالاتی که تاثیرگذاری ویژگی‌های کیفی را بر خصوصیات انطباقی مورد بررسی قرار داده اند، استخراج شده‌اند.

۴-۱- خصوصیات انطباق و ویژگی‌های کیفی متناظر با آن

مرحله اول کار، شناسایی خصوصیات انطباق است که در طیف وسیعی از سیستم‌های خودتطبیق از تئوری کنترل گرفته تا مهندسی نرم‌افزار برای ارزیابی فرآیندهای انطباق مورد استفاده قرار گرفته‌اند. تعاریف خصوصیات مختلف به همراه مراجعی که به این خصوصیات اشاره داشته‌اند، در ادامه آمده است و هرکدام از این خصوصیات به ویژگی‌های کیفی که به کمک آن‌ها می‌توان آن‌ها را مورد ارزیابی

قرار داد، نگاشت شده‌اند. جدول ۱ نشان دهنده خصوصیات انطباق و ویژگی‌های کیفی متناظر با آن‌ها می‌باشد که توضیحات آن در ادامه آمده است. از طرف دیگر باید در نظر داشت که برخی از این خصوصیات انطباق روی سیستم مدیریت شده و برخی روی کنترل‌کننده قابل مشاهده هستند و به صورت ایستا و پویا می‌توانند مورد ارزیابی قرار گیرند. نگاشت این خصوصیات به ویژگی‌های کیفی وابستگی این خصوصیات به محلی که می‌توان آن‌ها را مشاهده کرد را از بین می‌برد.

پایداری: این خصوصیت نشان دهنده این است که فرآیند خود تطبیقی تا چه اندازه به سمت اهداف کنترلی گمراه می‌شود. در یک انطباق ناپایدار عملیات کنترلی همواره با این ریسک که در هر بار اجرا ممکن است سیستم مدیریت شده بهبود نیابد و به سطح غیرقابل قبول و خطرناکی برسد، مواجه هستند. به منظور کاهش زمان ناپایداری می‌بایست زمان تطبیق‌پذیری را کاهش داد [۱۰، ۱۵-۱۷]. در نتیجه باید تغییرات و اصطلاحات در سیستم به سرعت صورت گیرد. حساسیت و قابلیت پیشگویی نیز از دیگر فاکتورهای تاثیرگذار روی این خصوصیت انطباقی به شمار می‌آیند [۱۶]. قابلیت پیشگویی به سیستم اجازه می‌دهد که بتواند با استفاده از اطلاعات از پیش جمع‌آوری شده محیط جدید را پیش‌بینی کند و به سیستم در رسیدن به پایداری کمک کند. علاوه بر موارد گفته شده، چهار ویژگی عملکرد، ایمنی، امنیت و یکپارچگی نیز در پایداری یک سیستم تأثیرگذارند [۱۵].

زمان استقرار: مدت زمانی است که طول می‌کشد تا یک سیستم پس از اعمال تغییرات به یک حالت مطلوب دست یابد. فاکتور اصلی تأثیرگذار در زمان استقرار تعداد منابع در دسترس برای انجام عملیات است که با عنوان کم از حد خارج شدن شناخته می‌شود. کم از حد خارج شدن به این معنی است که منابع مورد نیاز سیستم نبایدکمتر از حداقل مدنظر باشند. عامل تأثیرگذار دیگر، حساسیت سیستم نسبت به محیط است. هرچه میزان حساسیت یک سیستم بیشتر باشد، زمان مورد نیاز برای درک و تصمیم‌گیری نسبت به تغییرات کاهش می‌یابد. زمان اتخاذ تصمیم دیگر فاکتور تاثیرگذار روی این خصوصیت است. هرچه یک سیستم بتواند زمان اتحاد تصمیم را در برابر تغییرات کاهش دهد، زمان استقرار آن نیز کاهش می یابد. باید در نظر داشت که کارایی نیز روی این خصوصیت انطباقی تأثیرگذار است [۱۰، ۱۷-۲۰].

صحت: این خصوصیت تعیین کننده این است که به چه میزان به اهداف انطباق دست یافته‌ایم (اهداف انطباق به معنای دستیابی به یکی از خصوصیات غیرکارکردی یا خصوصیات سیستم‌های خود تطبیقی است). به طورکلی، میزان دقت و صحت در یک سیستم به میزان دستیابی به شش ویژگی اصلی سیستم‌های خودتطبیقی و برآورده‌سازی اهداف هر سیستم مانند کارایی، امنیت، درستی تصمیمات اتحادی و میزان استقلال سیستم نسبت به محیط اطراف وابسته است [۲۱، ۲۲].

کم از حد خارج شدن: این خصوصیت انطباقی بیانگر استفاده از منابع محاسباتی در طول فرآیند تطبیق به منظور دستیابی به اهداف انطباقی است. مدیریت کم از حد خارج شدن منابع به منظور جلوگیری از بی‌ثباتی و ناپایداری یک سیستم ضروری است. در سیستم‌های خودتطبیقی این ویژگی نیازمند نوع و تعداد منابع در دسترس محیط و سیستم است و به طور مستقیم روی ویژگی هزینه با توجه به میزان منابع محاسباتی مورد استفاده تأثیرگذار است. در برخی از موارد برای کاهش زمان انتقال می‌بایست از منابع بیشتری نسبت به حد تعیین شده استفاده نمود (کم از حد خارج شدن) [۱۰، ۱۵، ۱۷، ۲۳].

پایان‌دهی: در روش‌های مهندسی نرم‌افزار، طراحان در چرخه MAPE-K لیستی از فرآیندها را برای تطبیق سیستم مدیریت شده تعیین می‌کنند. خصوصیت پایان‌دهی تضمین می کند که این حالات متناهی بوده و حتی اگر سیستم به حالت مطلوب خود دست نیافت، اجرا خاتمه یابد. باید مد نظر قرار داد که هرچه تعداد فرآیندهای متعلق به یک عمل محدودتر باشد، میزان قابلیت اعتماد سیستم به نتایج حاصل از فرآیندها افزایش می‌یابد. در مقابل فرآیندهای نامحدود باعث ایجاد بن بست و عدم اطمینان در سیستم می‌شود. ویژگی پیچیدگی رفتاری که در ارزیابی تجربی بسیار تأثیرگذار است، تأثیر مستقیمی روی این خصوصیت نیز دارد. کارایی نیز دومین فاکتور تأثیرگذار روی این خصوصیت انطباقی است [۲۴، ۲۵].

سازگاری: هدف این خصوصیت حصول اطمینان از یکپارچگی ساختاری و رفتاری سیستم مدیریت شده پس از اجرای فرآیند تطبیق است. درواقع می توان گفت این ویژگی به این معنی است که فرآیندها به صورت مجزا نتایج خود را ثبت کرده ودر نتایج یکدیگر تأثیرگذار نباشد که قابلیت اطمینان در اینجا نقش بسزایی را ایفا می‌کند. باید در نظر داشت که تفاوت سیستم‌های سنتی و خودتطبیقی در نوع پیچیدگی رفتاری آنها است. هرچه این پیچیدگی بیشتر باشد، احتمال اینکه فرآیندها بخواهند روی نتایج حاصل از هم تاثیرگذارند نیز افزایش می‌یابد. همچنین می‌توان از دو ویژگی یکپارچگی و قابلیت نگهداری به عنوان معیارهایی که تاثیر قابل توجهی روی این خصوصیت انطباقی دارند، نام برد [۱۷، ۲۴، ۲۶].

مقیاس‌پذیری: به قابلیت یک سیستم برای کنترل وضعیت هنگام افزایش تقاضای کار یا حفظ کارایی و استفاده از منابع مازاد محاسباتی اشاره دارد. مهمترین فاکتور در برآورده‌سازی این خصوصیت میزان منابع در دسترس است. در نتیجه معیار کم از حد خارج شدن تأثیر مستقیم و مهمی روی این خصوصیت دارد. از طرف دیگر، هنگام توسعه یک سیستم باید کارایی آن را نیز مد نظر قرار داد زیرا توسعه یک سیستم با کارایی پایین، ریسک بالایی خواهد داشت [۱۵-۱۷].

حساسیت: یکی از مهمترین عوامل در سیستم‌های خود تطبیقی میزان حساسیت یک سیستم به تغییراتی است که در محیط اطراف آن رخ می‌دهد. سیستمی که نتواند تغییرات محیط اطراف خود را تشخیص دهد، نمی‌تواند نیازهای خود تطبیقی را برآورده سازد. تعداد حسگرهایی که یک سیستم برای درک محیط اطراف استفاده می‌کند نقش مهمی در میزان حساسیت یک سیستم دارد. هر چه میزان حسگرهای فعال افزایش یابد، سیستم می‌تواند درک درست‌تری از محیط اطراف خود داشته باشد [۱۰، ۱۵، ۲۵].

دانه‌بندی: در سیستم‌های خودتطبیقی دانه‌بندی از اهمیت بالایی برخوردار است. شاید در ظاهر مولفه‌های دانه‌ریز تاثیری روی عملکرد سیستم نداشته باشند و منجر به ایجاد سربار شوند، اما در واقع باعث افزایش قابلیت انعطاف و کارایی خواهند شد. وجود مولفه‌ها با هماهنگی بالا به سیستم در حذف و اضافه‌کردن مولفه جدید کمک کرده و این مسأله افزایش کارایی را به همراه خواهد داشت [۱۰، ۱۵].

جلوگیری از خطا: این خصوصیت سیستم به قابلیت آن در پیشگویی حوادثی که برای ایمنی سیستم رخ می‌دهد و توانایی آن برای مواجه با این حوادث اشاره دارد. بنابراین قابلیت پیشگویی و ایمنی نقش مهمی روی این خصوصیت انطباقی دارند [۱۰].

قابلیت پیش‌بینی: این خصوصیت به توانایی سیستم در پیشگویی رویدادهای آینده اشاره دارد. برای اینکه یک سیستم بتواند رخدادهایی که در آینده رخ خواهند را پیش‌بینی کند، باید درک درستی از خود و محیط اطراف خود داشته باشد. به طور کلی، با داشتن آگاهی در مورد وضعیتی که سیستم در آن قرار دارد می‌توان تا حدودی عکس‌العمل سیستم را پیش‌بینی کرد. از طرف دیگر، دو ویژگی کیفی ،کارایی و قابلیت نگهداری نیز از عوامل موثر بر این خصوصیت می‌باشند [۸، ۱۵].

استحکام: این خصوصیت به پایداری سیستم مدیریت شده اشاره دارد و هدف تضمین دقت، زمان استقرار کوتاه و کم از خارج شدن سیستم حتی در زمانی است که حالت سیستم مدیریت شده متفاوت با حالت مورد نظرو از پیش تعیین شده است. باید مد نظر قرار داد که هر کدام از ویژگی های مطرح شده به نسبت یکسانی تأثیر مستقیمی روی قدرت سیستم دارند. از دو ویژگی ایمنی و در

دسترس‌پذیری می‌توان به عنوان معیارهای موثر روی این خصوصیت انطباق نام برد [۴، ۱۰، ۲۷، ۲۸].

سرعت پاسخ خود مختاری: این خصوصیت به مدت زمان لازم برای درک تغییرات و اعمال عملیات مورد نظر برای تطبیق اشاره دارد. به طور کلی، هر چه سرعت پاسخ به رویدادها و تغییرات خواسته شده بالاتر باشد، عکس‌العمل سیستم نسبت به تغییرات افزایش می‌یابد. در یک سیستم ابتدا باید تغییرات احساس شوند، براساس تغییرات تصمیم مورد نظر در مورد نحوه عکس‌العمل اتخاذ شود و پس از آن سیستم وارد حالت اجرا شود. این خصوصیت به دو ویژگی زمان پایداری و زمان تنظیم کوتاه وابسته است. زمان پایداری به زمان درک تغییرات ، اتخاذ تصمیم مناسب و انتقال آن به سیستم و زمان استقرار به نحوه عکس‌العمل سیستم و انطباق آن با شرایط جدید اشاره دارد [۴، ۱۰، ۲۹].

درجه خود مختاری: در برخی از موارد لازم است قدرت پیشگویی سیستم را کاهش دهیم و توانایی سیستم را در تحمل و رفع مشکلات مورد بررسی قرار دهیم. این معیار درجه خودمختاری یک سیستم را مشخص می‌کند. کاهش قدرت پیشگویی و افزایش صحت و درستی از ویژگی‌های موثر بر این خصوصیت انطباقی می‌باشد [۲۷].

امنیت: در فرآیند انطباق نه تنها سیستم مقصد بلکه تمامی داده‌ها و اجزایی که با کنترل‌کننده به اشتراک گذاشته شده اند باید از دسترسی‌های غیرمجاز (محرمانگی)، اصلاحات (یکپارچگی) وتخریب (دسترس پذیری) حفاظت شوند. این خصوصیت انطباقی هم راستا با امنیت در سیستم های غیرتطبیقی است [۱، ۱۰].

هزینه: سیستم های خود مختار، منجر به ایجاد سربار زیادی می شوند. باید بررسی شود که آیا نتیجه حاصل از خود مختاری ارزش تحمل چنین سرباری را دارد یا خیر. یکی از مهمترین این سربارها هزینه است که از دو بعد نیروی انسانی و منابع مصرفی مورد بررسی قرار می‌گیرد. هرچه خود مختاری یک سیستم افزایش یابد ، حضور عوامل انسانی کمتر خواهد شد اما میزان مصرف منابع افزایش خواهد یافت. در نتیجه، می‌توان از ویژگی کم از حد خارج شدن برای بررسی منابع مورد استفاده بهره برد. میزان کارایی سیستم در استفاده منابع نیز روی هزینه تأثیرگذار است. فاکتور مؤثر دیگر نیز خودبهینگی است [۲۷].

سطح خود مختاری: این خصوصیت مشخص کننده این است که یک سیستم تا چه اندازه ای می تواند نیازهای اصلی یک سیستم خود مختار را پوشش دهد که منجر به کاهش هزینه‌ها می گردد. علاوه بر شش مولفه اساسی سیستم های خود تطبیق، دو ویژگی قابلیت پیشگویی و بازبودن را نیز از ویژگی‌های مؤثر بر این خصوصیت انطباقی می‌باشند [۸].

خود پیکربندی: این خصوصیت نشان دهنده توانایی سیستم برای باز پیکر بندی خود به صورت خودکار و پویا در مقابل تغییراتی است که در سیستم رخ می‌دهند. به طور کلی، پیکربندی به توانایی سیستم بر حذف ، اضافه، تغییر و جابجایی مولفه‌ها اشاره دارد و ویژگی‌هایی مانند قابلیت نگهداری ، قابلیت استفاده، عاملیت و قابلیت حمل روی آن تأثیر گذارند [۱۸، ۳۰].

خود التیامی: این خصوصیت نشان‌دهنده توانایی سیستم در تشخیص خرابی و حل آن است . در واقع سیستم باید بتواند مولفه‌هایی که منجر به خرابی سیستم شده‌اند را شناسایی کند و مولفه سالمی را جایگزین آن نماید. هدف از خود التیامی برآورده‌سازی سه ویژگی قابلیت نگهداری، قابلیت اطمینان و دسترس‌پذیری است [۲۴، ۳۱].

خود حفاظتی: به توانایی سیستم برای حفظ امنیت سیستم اشاره دارد. این خصوصیت را میتوان از جنبه دفاع از سیستم در برابر حملات مخرب و پیش بینی مشکلات و اقداماتی جهت جلوگیری از آنها بررسی کرد. قابلیت نگهداری، عاملیت وکارایی از ویژگی های مؤثر روی این خصوصیت می‌باشند [۳۲].

خود بهینگی: سیستم باید توانایی مدیریت منابع را داشته باشد بطوریکه میزان مفیدی از منابع را به کاربران براساس نیازشان اختصاص دهد. سه ویژگی کارایی، قابلیت نگهداری و قابلیت عملکرد از عوامل موثر در این خصوصیت می باشند [۳۲].

آگاه از زمینه : سیستم های خود تطبیق برای انجام عملیات انطباقی باید نسبت به محیط اطراف خود آگاهی داشته ، تغییرات را درک کنند و نسبت به آن واکنش نشان دهند. برای درک محیط اطراف سیستم بایدحساسیت بالایی داشته باشد. علاوه بر این، کارایی نیز به عنوان یک ویژگی موثر روی این خصوصیت معرفی شده است [۱۰، ۳۳].

آگاه از خود: یک سیستم علاوه برای اجرای عملیات خود تطبیقی علاوه بر آگاهی نسبت به محیط اطراف خود می‌بایست نسبت به وضعیت مولفه‌ها و حسگرهای خود نیز آگاه باشد. در ضمن باید از حساسیت بالایی برخوردار باشد تا بتواند رفتار سیستم را برای جلوگیری از ایجاد پیچیدگی مدیریت کند که منجر به افزایش کارایی می‌شود [۱۰، ۳۳].

باز بودن: این خصوصیت سیستم نشان دهنده توانایی سیستم برای کار در محیط‌های گوناگون و متغیر است. این خصوصیت را می‌توان معادل با ویژگی کیفی قابلیت حمل دانست [۳۳].

۴-۲- متریک‌های انطباقی ارزیابی

متریک‌های خودتطبیقی یک راهکار برای ارزیابی سیستم‌های خودتطبیق با در نظر گرفتن دغدغه‌ها و چالش‌های پیش‌روی فرآیندهای تطبیقی فراهم می‌سازند. درواقع، این متریک‌ها معیاری را برای ارزیابی خصوصیات مورد انتظار در سیستم‌های خودتطبیق معرفی می‌کنند [۳۴]. به عنوان مثال، متریک‌های انطباقی برای ارزیابی سیستم کنترلی جنبه‌های مرتبط با خصوصیات سیستم‌های خودتطبیق (مانند، پایداری، صحت، زمان پاسخ کوتاه و کم از حد خارج شدن) را اندازه‌گیری می‌کنند. در این مقاله، برای مشخص کردن متریک‌های ارزیابی سیستم‌های خود تطبیق، طیف وسیعی از سیستم‌های نرم‌افزاری خود تطبیق که خصوصیات انطباق را روی سیستم مدیریت شده و کنترل‌کننده اعمال می‌کنند شناسایی شده و ویژگی‌های کیفی متناظر با آن‌ها که برای ارزیابی خصوصیات انطباق می‌توان از آنها بهره برد، معرفی شدند. با توجه به اینکه ارزیابی بیشتر ویژگی‌ها تنها با مشاهده کنترل‌کننده امکان پذیر نمی باشد، می‌توان آنها را بوسیله این ویژگی‌های کیفی روی سیستم مدیریت شده نیز مورد ارزیابی قرار داد. در این بخش نیز متریک‌های مرتبط برای اندازه‌گیری این ویژگی‌های کیفی با بررسی طیف وسیعی از تحقیقات انجام شده در حوزه سیستم‌های خودتطبیق معرفی شده‌اند. برای شناسایی متریک‌های مرتبط باید مجموعه‌ای از فاکتورها مانند میزان حافظه مصرفی، توان عملیاتی، زمان پاسخ، نرخ پردازش، زمان میانگین شکست، و زمان میانگین ترمیم که روی ارزیابی ویژگی‌های کیفی تأثیر می‌گذارند، شناسایی شوند [۴، ۳۵]. شناسایی این فاکتورها برای ارزیابی خصوصیات انطباقی روی سیستم مدیریت شده و کنترل‌کننده ضروری است.

کارایی: کارایی یک سیستم نرم‌افزاری وابسته به استفاده بهینه از زمان و منابع است. کارایی زمانی وابسته به زمان دریافت درخواست، پردازش درخواست و ارسال پاسخ است. کارایی سایر منابع که مهم‌ترین آن حافظه است به میزان حافظه مصرفی برای ذخیره اطلاعات و عدم اختصاص حافظه تکراری در سیستم های نرم افزاری اشاره دارد. با استفاده از متریک‌های زیر می‌توان کارایی یک سیستم خودتطبیق را مورد ارزیابی قرار داد. (RU) زمان میانگین مورد نیاز برای تکمیل یک درخواست، (CU) هزینه میانگین برای انجام درخواست کاربر.

جدول ۱- خصوصیات انطباق و ویژگی‌های کیفی موثر بر آنها

خصوصیت انطباق	ویژگی کیفی موثر
پایداری (stability)	تغییر و اصلاح (modification)
	حساسیت (sensitive)
	پیشگویی (anticipate)
	عملکرد (performance)
	یکپارچگی (integrity)
	ایمنی (safety)
	امنیت (security)
صحت (accuracy)	اگاهی از زمینه (self-awareness)
	خود تنظیمی (self-configuration)
	خود التیامی (self-healing)
	آگاهی از خود (self-awareness)
	حفاظت از خود (self-protection)
	خود بهینگی (self-optimization)
	بازبودن (openness)
	پیش‌بینی (anticipate)
	عملکرد (performance)
	امنیت (security)
	صحت تصمیمات اتخاذی (accuracy decision make)
زمان تنظیم کوتاه (short setting time)	زمان اتخاذ تصمیم (decision make time)
	حساسیت (sensitive)
	کم از حد خارج شدن (small overshoot)
	عملکرد (performance)
کم از حد خارج شدن (small overshoot)	عملکرد(performance)
	رفتار پیچیده (complex behavior)
سازگاری (consistency)	رفتار پیچیده(complex behavior)
	یکپارچگی (integrity)
	قابلیت اطمینان (reliability)
	قابلیت نگهداری (maintainability)
مقیاس‌پذیری (scalability)	کم از حد خارج شدن (small overshoot)
	عملکرد (performance)
پایان‌دهی (termination)	عملکرد(performance)
	رفتار پیچیده (complex behavior)
حساسیت (sensitivity)	سنسورهای فعال (active sensor)
جلوگیری از خطا (failure avoidance)	پیش‌بینی (anticipate)
	ایمنی (safety)
دانه‌بندی (granularity)	قابلیت انعطاف (flexibility)
	درشت و ریزدانه بودن (fine/thick grain)
قابلیت پیش‌بینی (anticipate)	کارایی (efficiency)
	قابلیت نگهداری (maintainability)
	اگاهی از زمینه (self-awareness)
	آگاهی از خود(self-awareness)

خصوصیت انطباق	ویژگی کیفی موثر
سرعت پاسخ خودمختاری (speed of automatic response)	پایداری (stability)
	زمان تنظیم کوتاه (short setting time)
استحکام (robustness)	پایداری (stability)
	زمان تنظیم کوتاه (short setting time)
	کم از حد خارج شدن (small overshoot)
	ایمنی (safety)
	دسترس‌پذیری (availability)
سطح خودمختاری (level of autonomy)	اگاهی از زمینه (self-awareness)
	خود تنظیمی (self-configuration)
	خود التیامی (self-healing)
	آگاهی از خود (self-awareness)
	حفاظت از خود (self-protection)
	خود بهینگی (self-optimization)
	بازبودن (openness)
	پیش‌بینی (anticipate)
خودپیکربندی (self-configuration)	قابلیت نگهداری (maintainability)
	قابلیت استفاده (usability)
	عاملیت (functionality)
	قابلیت حمل (portability)
خودالتیامی (self-healing)	قابلیت نگهداری (maintainability)
	قابلیت اطمینان (reliability)
	دسترس‌پذیری (availability)
خودحفاظتی (self-protection)	عاملیت (functionality)
	قابلیت حمل (portability)
خودبهینگی (self-optimization)	قابلیت نگهداری (maintainability)
	عاملیت (functionality)
	کارایی (efficiency)
گاه از زمینه (context awareness)	حساسیت (sensitive)
	عملکرد (performance)
آگاه از خود (slef-awareness)	حساسیت (sensitive)
	عملکرد (performance)
بازبودن (openness)	بازبودن (openness)
	قابلیت حمل(portability)

متریک‌های دیگری نیز برای ارزیابی کارایی وجود دارد. به عنوان مثال، می توان از فرآیندها دارای کار تکراری، فرآیندها با اتلاف وقت و میزان دانه‌بندی معماری سیستم به عنوان متریک‌های موثر بر کارایی نام برد [۱۷، ۲۱]. میانگین زمان پاسخ نیز یک معیار عمومی برای ارزیابی کارایی است .کارایی سیستم را براساس یک تابع توزیع بار نیز می‌توان اندازه‌گیری کرد. این تابع برای اضافه کردن هزینه بار تحمیلی و بار داخلی مولفه‌ها تعریف شده است [۱۵].

قابلیت اطمینان: قابلیت اطمینان در یک سیستم تضمین کننده استمرار فعالیت آن است. برای اندازه‌گیری قابلیت اطمینان متریک (DU) معرفی شده است که نشان دهنده احتمالی است که یک سرویس کار را براساس درخواست کاربر تکمیل کند [۲۱]. از طرف دیگر، این ویژگی را می‌توان به دو زیر ویژگی تحمل خطا و بازگشت از خطا نیز تقسیم کرد که مرتبط با ساز و کارهای موجود در معماری سیستم می‌باشند. برای اندازه‌گیری تحمل خطا باید تعداد سیستم‌هایی که مکانیزم تحمل خطا در آنها وجود دارد شمارش شوند. نسبت تعداد این سیستم‌ها به کل سیستم‌های نرم‌افزاری نشان دهنده مقدار این معیار است. برای محاسبه بازگشت از خطا نیز باید نسبت سیستم‌هایی که مکانیزم بازگشت از خطا در آنها وجود دارد به کل سیستم‌های نرم‌افزاری مشخص شود. از فاکتورهای ارتباطات فعال به ازای هر سرور، زمان پاسخ کلی (متوسط زمانی که طول میکشد هر درخواست پردازش شود)، پهنای باند خروجی (تعداد متوسط بایت‌های خارج شده به ازای هر سرور در هر ثانیه)، زمان پاسخ‌دهی پایگاه داده (میانگین زمانی که طول می‌کشد هر درخواست توسط پایگاه داده پردازش شود)، نرخ عدم پذیرش (درصدی از ارتباطات پذیرفته نشده برای عبور روی دامنه مشتری)، نرخ پذیرش (متمم نرخ عدم پذیرش) و سرورهای فعال (تعداد سرورهایی که به یک حوزه مشتری سرویس می‌دهند) نیز می‌توان برای اندازه‌گیری قابلیت اطمینان در سیستم‌های خودتطبیق بهره برد [۳۶].

یکپارچگی: به منظور استفاده بهینه از سیستم های نرم‌افزاری، این سیستم‌ها باید یکپارچه و پیوسته باشند. یکپارچگی را در سیستم های نرم افزاری می‌توان از سه دیدگاه سکوی کاری، داده و سیستم‌های نرم‌افزاری مورد بررسی قرار داد. یکپاچگی براساس دیدگاه سکوی کاری مرتبط با تعداد سکوهای مختلف به خصوص سیستم عامل‌های متفاوت موجود می‌باشد. یکپارچگی داده، به استانداردهای نگهداری اطلاعات اشاره دارد که براساس آن سیستم‌های مختلف بتوانند با داده‌های سایر سیستم ها ارتباط برقرار کنند. یکپارچگی در سطح سیستم‌های نرم‌افزاری با توجه به تبادل اطلاعات و فرآیندهای کاری بین سیستم‌های نرم‌افزاری سنجیده می‌شود که می‌توان آن‌ها را با شمارش تعداد سکوها، سیستم عامل‌های موجود، محیط‌های مختلف نگهداری داده‌ها نسبت به کل سیستم‌ها سنجید [۳۶، ۳۷].

قابلیت نگهداری: قابلیت نگهداری به توانایی مقابله سیستم نسبت به تغییراتاشاره دارد و به زیرویژگی‌های قابلیت تجزیه و تحلیل و تغییرپذیری تقسیم می‌شود. افزایش بیش از حد سیستم‌های نرم‌افزاری منجر به افزایش پیچیدگی می‌شود که پیرو آن قابلیت تجزیه و تحلیل کاهش می‌یابد. از طرف دیگر استفاده از دانه‌بندی در سطح بالا ، منجربه بهبود تجزیه و تحلیل و افزایش تغییرپذیری می‌گردد. علاوه بر این عدم وجود ساز و کارهای مبتنی بر شرایط ثابت، تعداد مؤلفه های قابل‌تغییر و فناوری‌های روز معیارهای هستند که منجر به افزایش تغییرپذیری خواهند شد. برای اندازه‌گیری قابلیت نگهداری می‌توان از متریک‌های مانند نسبت سیستم‌های رعایت کننده دانه‌بندی به کل سیستم ها، میزان همبستگی داخلی سیستم‌های نرم‌افزاری، میزان اتصال بیرونی سیستم‌های نرم‌افزاری و درصد سیستم‌های استفاده کننده از معماری‌های تغییرپذیر استفاده کرد [۱۰، ۳۴].

امنیت: امنیت را می توان براساس سه دیدگاه تمامیت، محرمانگی و دسترسی مورد بررسی قرار داد. برای اندازه‌گیری این خصوصیت می‌توان از متریک‌هایی مانند نسبت سیستم عامل‌های دارای مکانیزم تصدیق هویت به کل سیستم‌ها ، نسبت سیستم عامل‌ها با مکانیزم تأیید اعتبار به کل سیستم‌ها و وجود یا عدم وجود مکانیزم تصدیق هویت و تأیید اعتبار در شبکه، وجود ابزارهای دیواره آتش، ویروس‌یابی و کشف نفوذ و مهاجم‌یاب مورد ارزیابی قرار داد [۳۶].

قابلیت استفاده: برای اندازه‌گیری این خصوصیت باید متریک‌هایی مانند هزینه، زمان اجرایی، میزان سازگاری، قابلیت‌ها و در دسترس بودن مورد بررسی قرار گیرند [۳۶].

قابلیت حمل: به منظور ارزیابی این خصوصیت باید ویژگی‌های تطابق‌پذیری، قابلیت نصب و قابلیت همزیستی مورد بررسی قرار گیرند. به منظور افزایش تطابق‌پذیری نیاز به مؤلفه هایی در سیستم است که بتواند هسته سیستم را با سایر محیط ها سازگار نماید. برای قابلیت همزیستی نیز سیستم باید مکانیزم‌هایی داشته باشد که بتواند با سیستم‌های موجود در محیط تبادلات لازم را انجام دهد و منابعی را به اشتراک بگذارد . این ویژگی را می‌توان از دید متریک‌هایی مانند وجود یا عدم وجود مولفه برای سازگاری هسته سیستم با محیط، وجود یا عدم وجود مولفه‌هایی برای ارتباط با عوامل خارجی، وجود یا عدم وجود فرآیندهای همروند برای ایجاد اشتراک زمانی برای استفاده از پردازنده و یا وجود مکانیزم بن‌بست مورد اندازه‌گیری قرار داد [۳۶].

دسترس‌پذیری: این خصوصیت به در دسترس بودن سیستم برای استفاده اشاره دارد. سیستم در صورتی از دسترس خارج می‌شود که در آن شکست رخ دهد. پس برای اندازه‌گیری دسترس‌پذیری یک سیستم می توان معیارهایی مانند وجود یا عدم وجود راهکار تشخیص خطا، وجود راهکار برای ترمیم خطای کشف شده و راهکارهایی برای جلوگیری از بروز نقص مورد اندازه‌گیری قرار داد. . برای اندازه‌گیری دسترس‌پذیری دو متریک تعریف شده است: متریک دسترس‌پذیری $\frac{MTTF}{MTTF+MTTR}$ و متریک مدت از کارافتادگی $\frac{MTTR}{MTTF+MTTR}$ که در آن MTTF به معنی متوسط زمان تا شکست سیستم است. دسترس‌پذیری عددی بین صفر و یک خواهد بود. اگر MTTF نسبت به MTTR[۱۰] خیلی بزرگتر باشد، مدت از کارافتادگی به صورت تقریبی برابر با $\frac{MTTR}{MTTF}$ خواهد بود[۳۸].

عملکرد: این ویژگی نشان‌دهنده توانایی سیستم برای انجام کاری است که برای آن به وجود آمده است. برای انجام یک کاراجزای مختلف سیستم باید در یک مسیر هماهنگ حرکت کنند و اگر به اجزاءِ مسئولیت‌های درست و واقعی واگذار نشود و مکانیزم مناسبی برای ایجاد ارتباط بین مولفه‌ها موجود نباشد، سیستم عملکرد مناسبی نخواهد داشت. این ویژگی کیفی را می‌توان با استفاده از معیارهای امنیت، دقت و صحت و قابلیت انعطاف مورد اندازه‌گیری قرار داد [۱۷].

دقت و صحت: این ویژگی نشان دهنده توانایی یک سیستم یک سیستم در دسترسی صحیح به اهداف از پیش تعیین شده است. بهترین روش برای ارزیابی دقت و صحت براساس تصمیماتی است که در بخش برنامه‌ریزی گرفته می‌شوند. در این بخش یکسری از اهداف برای سیستم مشخص می‌شوند که اگر بتوان میزان رضایت کاربر از درصد تحقق رسیدن به اهداف را در سیستم مشخص کرد ، می‌توان دقت وصحت سیستم را نیز سنجید. یکی از مهم‌ترین معیارها قابلیت ردگیری است . قابلیت ردگیری مشخص کننده این است که مؤلفه‌های درگیر فرآیند انطباقی می‌بایست در یک یا چند کلاس قرار گرفته باشند. هر چه قدر مؤلفه‌های درگیر محدود به تعداد کمتری از کلاس‌ها باشد، ناهمگونی کاهش پیدا کرده و دقت بالاتر می‌رود. معیار دیگر سازگاری است که نشان دهنده این است که اهدافی که فرآیندهای انطباقی دنبال می‌کنند نباید با هم ناسازگار باشند که برای اندازه‌گیری آن باید نسبت فرآیندهایی که شرط سازگاری را رعایت می‌کنند به کل فرآیندها بدست آورد [۱۰، ۳۶].

ایمنی: برای اندازه گیری ایمنی می توان از دو معیار پیچیدگی تعامل[۱۱] و قدرت اتصال[۱۲] بهره برد. پیچیدگی تعامل حدی برای رفتار یک مؤلفه است که می واند

رفتار سایر مؤلفه‌ها را نیز تحت تأثیر خود قرار دهد که می‌توان مقدار آن را از طریق اندازه‌گیری میزان وابستگی مؤلفه‌ها به یکدیگر و میزان استقلال هر مؤلفه از سایر مؤلفه ها بدست آورد. قدرت اتصال نیز از محدوده اتصال سست تا اتصال قوی را شامل می‌شود. سیستم های اتصال قوی وابسته به زمان بوده و نمی توانند منتظر بمانند، در مقابل فرآیندهای اتصال سست میتوانند تأخیر داشته باشند و در حالت انتظار قرار گیرند [۳۴].

بهره‌وری: بهره‌وری را می توان از دو بعد زمان و منابع مورد بررسی قرار داد. برای بهره‌وری در زمان سیاست‌های زمان‌بندی برای اشتراک چندین فرآیند باید مورد بررسی قرار گیرند. برای ایجاد بهره‌وری در زمان فرآیندهای موجود در سیستم باید به صورت چندنخی طراحی گردند تا قابلیت اجرای هم‌زمان را داشته باشند و برای اندازه‌گیری آن می‌توان از نسبت فرآیندها با قابلیت اشتراک زمانی و چندنخی به کل فرآیندهای موجود استفاده کرد. بهره‌وری در منابع نیز به معنای استفاده اشتراکی از منابع است و نسبت مولفه‌های سخت افزاری با قابلیت اجرای همزمان چند فرآیند به تعداد کل مؤلفه‌ها معیاری برای ارزیابی این ویژگی است [۳۶].

قابلیت انعطاف: سادگی، ماژولار بودن و قابلیت توسعه تأثیر مستقیمی روی ویژگی کیفی دارند. سادگی به این معنی است که تعداد ماژول‌ها کم باشد و پیچیدگی گراف تئوریک نیز از حد مشخصی پایین تر باشد .علاوه بر این، برای اندازه‌گیری قابلیت انعطاف می توان عمومیت را نیز اندازه‌گیری کرد. لازم به ذکر است که برای برخی از ویژگی‌های کیفی معرفی شده، متریک‌های ارزیابی تا کنون مورد استفاده قرار نگرفته است و معرفی آن‌ها تنها به صورت تئوریک بوده است [۳۶].

۵- بحث و نتیجه‌گیری

با توجه به اینکه تغییر عنصر غیرقابل انکار دنیای امروز است و این تغییرات منجر به ایجاد تغییر در ساختار سیستم‌های نرم‌افزاری می‌شود، نقش سیستم‌های خودتطبیق که می‌توانند خود را با تغییرات هماهنگ کنند بیش از پیش احساس می‌شود. با اینکه مفهوم خودتطبیقی به عنوان یکی از موضوعات داغ تحقیقاتی در حوزه مهندسی نرم‌افزار به شمار می‌آید و پژوهش‌های مختلفی در این زمینه صورت گرفته است، کمتر به مباحث مرتبط با ارزیابی این سیستم‌ها پرداخته شده است در حالیکه برای درک موفقت یک سیستم ارزیابی آن نقش مهمی ایفا می‌کند.

با توجه به اینکه محیط اطراف سیستم‌های خودتطبیق همواره در حال تغییر است و این تغییرات منجر به تغییر در ساختار و نیازمندی‌های سیستم می‌شود، ارزیابی این نوع سیستم‌ها در مقایسه با سیستم‌های سنتی متفاوت بوده و با چالش‌هایی مواجه است. مهمترین مشکل موجود برای ارزیابی سیستم‌های خودتطبیقی کمبود ویژگی‌های کیفی متناسب با این سیستم‌ها و متریک‌های متناظر با آن‌ها برای ارزیابی است. به همین منظور در این مقاله، طیف وسیعی از کارهای انجام شده در حوزه سیستم‌های خودتطبیق مورد تجزیه و تحلیل قرار گرفته و خصوصیات انطباقی که مهمترین دلیل برای ایجاد سیستم‌های خودتطبیق می‌باشند و در کارهای پیشین مورد بررسی قرار نگرفته بودند، استخراج شده‌اند. با توجه به اینکه برخی از این خصوصیات کیفی روی سیستم مدیریت شده و برخی روی کنترل‌کننده قابل مشاهده‌اند، ارزیابی آن‌ها با مشکلاتی مواجه خواهد بود. به همین خصوصیات انطباقی شناسایی شده به ویژگی‌های کیفی متناظر با آن‌ها که مهمترین معیار برای ارزیابی این سیستم‌های می‌باشند نگاشت شده‌اند. در ادامه نیز متریک‌های ارزیابی برای اندازه‌گیری ویژگی‌های کیفی معرفی شده‌اند. چارچوب ارائه شده برای ارزیابی می‌تواند به عنوان یک زیربنا برای ارزیابی سیستم‌های خودتطبیقی بحساب آید.

اگر چه دقت بسیاری صورت گرفته تا بهترین ویژگی‌های استخراج شوند، اما باز هم می‌توان با مطالعه مقالات و شناخت کافی از خصوصیات انطباقی جدید، ویژگی‌های کیفی مرتبط با این سیستم‌ها را تکمیل‌تر کرد. در واقع می‌توان گفت هرکدام از این ویژگی‌های کیفی تکه‌هایی از یک پازل هستند که می‌توانند هربار تکه‌هایی جدیدی (ویژگی‌های کیفی) به آن افزود و آن را تکمیل‌تر کرد. کار دیگری که می‌توان در این زمینه انجام داد نیز بررسی صحت، درستی و میزان تاثیر ویژگی‌های کیفی انتخاب شده است. به طور کلی، گسترش و توسعه معیارهای ارائه شده برای خصوصیات انطباقی و متریک‌های ارزیابی برای معیارهای کیفی خود می‌تواند زمینه تحقیقی جدید در آینده باشد.

مراجع

[1] M. Ahmad, N. Belloir, and J.-M. Bruel, *Modeling and Verification of Functional and Non-Functional Requirements of Ambient Self-Adaptive Systems*. Journal of Systems and Software, 2015.

[2] R. De Lemos, and et. al., *Software engineering for self-adaptive systems: A second research roadmap, in Software Engineering for Self-Adaptive Systems II*, Springer, pp. 1-32, 2013.

[3] F. D. Macías-Escrivá, and et. al., *Self-adaptive systems: A survey of current approaches, research challenges and applications*. Expert Systems with Applications, 40 (18): pp. 7267-7279, 2013.

[4] N. M. Villegas, and et. al., *A framework for evaluating quality-driven self-adaptive software systems*. in *Proc of the 6th international symposium on Software engineering for adaptive and self-managing systems*, ACM, 2011.

[5] C. Krupitzer, and et. al., *A survey on engineering approaches for self-adaptive systems*. Pervasive and Mobile Computing, 17: pp. 184-206, 2015.

[6] G. Tamura, and et. al., *Towards practical runtime verification and validation of self-adaptive software systems*, in *Software Engineering for Self-Adaptive Systems II*, Springer, pp. 108-132, 2013.

[7] J. L, Hellerstein, S. Singhal, and Q. Wang, *Research challenges in control engineering of computing systems*. Network and Service Management, IEEE Transactions, 6(4): pp. 206-211, 2009.

[8] M. Salehie, and L. Tahvildari, *Self-adaptive software: Landscape and research challenges*. ACM Transactions on Autonomous and Adaptive Systems (TAAS), 4(2): pp. 14, 2009.

[9] J. L. Hellerstein, and et. al., *Feedback control of computing systems*, John Wiley & Sons, 2004.

[10] A. C. Meng, *On evaluating self-adaptive software*, in *Self-Adaptive Software*., Springer, pp. 65-74, 2001.

[11] J. A. Mc Cann, and M. C. Huebscher. *Evaluation issues in autonomic computing*. in *Grid and Cooperative Computing-GCC 2004 Workshops*. Springer, 2004.

in *Formal Aspects of Component Software*, Springer, pp. 34-52, 2012.

[26] M. Léger, T. Ledoux, and T. Coupaye, *Reliable dynamic reconfigurations in a reflective component model*, in *Component-Based Software Engineering*, Springer, pp. 74-92, 2010.

[27] J. Dowling, and V. Cahill. *Self-managed decentralised systems using K-components and collaborative reinforcement learning*. in *Proc of the 1st ACM SIGSOFT workshop on Self-managed systems*, ACM, 2004.

[28] J. Cámara, and et. al., *Testing the robustness of controllers for self-adaptive systems.* Journal of the Brazilian Computer Society, 20 (1): pp. 1-14, 2014.

[29] A. Ganek, and et. al., *Autonomic Computing: Concepts, Infrastructure, and Applications*, in *Overview of Autonomic Computing: Origins, Evolution, Direction*, CRC Press, pp. 3-18, 2006.

[30] L. K. Chan, S. W. Cheng, and F. A. Spiring, *A new measure of process capability: Cpm.* Journal of Quality Technology, 20(3): pp. 162-175, 1988.

[31] A. G. Ganek, and T. A. Corbi, *The dawning of the autonomic computing era.* IBM systems Journal, 42 (1): pp. 5-18, 2003.

[32] J. O. Kephart, and D. M. Chess, *The vision of autonomic computing.* Computer, 36 (1): pp. 41-50, 2003.

[33] M. Parashar, and S. Hariri, *Autonomic computing: An overview*, in *Unconventional Programming Paradigms*, Springer, pp. 257-269, 2005.

[34] P. Reinecke, K. Wolter, and A. Van Moorsel, *Evaluating the adaptivity of computing systems*. Performance Evaluation, 67(8): pp. 676-693, 2010.

[35] M. Barbacc, and et. al., *Quality Attributes*. 1995, DTIC Document.

[36] R. Khan, K. Mustafa, and S. Ahson, *Software Quality. Concepts and Practices. Alpha Science*, Oxford, 2006.

[37] R. Calinescu, and et. al., *Self-adaptive software needs quantitative verification at runtime.* Communications of the ACM, 55(9): pp. 69-77, 2012.

[38] G. Candea, J. Cutler, and A. Fox, *Improving availability with recursive microreboots: a soft-state system case study.* Performance Evaluation, 56(1): pp. 213-248, 2004.

[12] X. Liu, and Q. Wang. *Study on application of a quantitative evaluation approach for software architecture adaptability*. in *Quality Software,. Fifth International Conference on*. IEEE, 2005.

[13] D. N. Chin, *Empirical evaluation of user models and user-adapted systems.* User modeling and user-adapted interaction, 11(1-2): pp. 181-194, 2001.

[14] S. Taranu, and J. Tiemann. *On assessing self-adaptive systems*. in *Pervasive Computing and Communications Workshops (PERCOM Workshops)*, IEEE International Conference, 2010.

[15] K. Appleby, and et. al., *Oceano-SLA based management of a computing utility*. in *Integrated Network Management Proc, IEEE/IFIP International Symposium*, IEEE, 2001.

[16] J. Floch, and et. al., *Using architecture models for runtime adaptability.* Software, IEEE, 23 (2): pp. 62-70, 2006.

[17] C. Lu, and et. al., *Performance specifications and metrics for adaptive real-time systems*. in *Real-Time Systems Symposium, Proc. The 21st IEEE*. IEEE,2000.

[18] D. Garlan, and et. al., *Rainbow: Architecture-based self-adaptation with reusable infrastructure.* Computer, 37 (10): p. 46-54, 2004.

[19] V. Kumar, and et. al., *Middleware for enterprise scale data stream management using utility-driven self-adaptive information flows.* Cluster Computing, 2007. 10(4): pp. 443-455.

[20] J. White, D. C. Schmidt, and A. Gokhale, *Simplifying autonomic enterprise java bean applications via model-driven development: a case study*, in *Model Driven Engineering Languages and Systems*, Springer, pp. 601-615, 2005.

[21] V. Cardellini, and et. al., *Qos-driven runtime adaptation of service oriented architectures*. in *Proc of the the 7th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering*, ACM, 2009.

[22] A. Solomon, and et. al., *Business process adaptation on a tracked simulation model*. in *Proc of the 2010 Conference of the Center for Advanced Studies on Collaborative Research*, 2010. IBM Corp.

[23] S. Parekh, and et. al., *Using control theory to achieve service level objectives in performance management.* Real-Time Systems, 23(1-2): pp. 127-141, 2002.

[24] H. Ehrig, and et. al., *Formal analysis and verification of self-healing systems*, in Fundamental Approaches to Software Engineering, Springer, pp. 139-153, 2010.

[25] G. Tamura, and et. al., *QoS contract-aware reconfiguration of component architectures using e-graphs*,

اسلام ناظمی در سال ۱۳۳۳ در شهر سراب دیده به جهان گشود. در سال ۱۳۵۶ مدرک کارشناسی خود را در رشته ریاضیات کاربردی و تحقیق در عملیات از مدرسه عالی برنامه‌ریزی و کاربرد کامپیوتر و پس از آن در سال های ۱۳۶۸ و ۱۳۷۳ مدارک کارشناسی ارشد خود را به ترتیب در رشته مهندسی سیستم‌ها گرایش‌های تحقیق در عملیات و اقتصاد اخذ کرد. در ادامه در سال ۱۳۸۳ موفق به اخذ مدرک دکتری تخصصی در رشته مهندسی صنایع گرایش فناوری اطلاعات شد. دکتر ناظمی از سال ۱۳۵۷ عضو هیئت علمی مدرسهٔ عالی برنامه‌ریزی و کاربرد کامپیوتر و از سال ۱۳۶۵ تاکنون عضو هیئت علمی دانشگاه شهید بهشتی می‌باشند. ایشان به مدت ۶ سال معاون آموزشی و تحصیلات تکمیلی دانشکده مهندسی برق و کامپیوتر، به مدت ۳ سال مدیر توسعه فناوری اطلاعات آموزش دانشگاه و در حال حاضر نیز معاون آموزشی و تحصیلات تکمیلی دانشکده مهندسی و علوم کامپیوتر هستند. وی دانشیار گروه مهندسی کامپیوتر، فناوری اطلاعات دانشگاه شهید بهشتی می‌باشند. علایق پژوهشی نامبرده عبارتند از: مهندسی سیستم‌های نرم‌افزاری خودتطبیق، توسعه نرم‌افزار در مقیاس بزرگ، سیستم‌های اطلاعات مدیریت و وب‌کاوی. تاکنون بیش از نود مقاله از دکتر ناظمی در مجلات و کنفرانس‌های معتبر داخلی و خارجی به چاپ رسیده است. همچنین توسط ایشان تاکنون ده کتاب در حوزه ریاضیات، مدیریت پروژه، مهندسی نرم‌افزار، کیفیت نرم‌افزار و تئوری بازی‌ها انتشار یافته است. وی عضو انجمن کامپیوتر ایران، انجمن مهندسی برق، انجمن فرماندهی و کنترل و انجمن انفورماتیک ایران و عضو هیات مدیره انجمن انفورماتیک ایران هستند.

آدرس پست‌الکترونیکی ایشان عبارت است از:
nazemi@sbu.ac.ir

حسین صدر در سال ۱۳۶۵ در شهر رودسر (استان گیلان) متولد شد. در سال ۱۳۸۹ مدرک کارشناسی و سال ۱۳۹۲ مدرک کارشناسی ارشد خود را با احراز رتبه اول از دانشگاه آزاد اسلامی واحد علوم و تحقیقات اخذ نمود. در حال حاضر عضو هیئت علمی دانشگاه آزاد اسلامی و دانشجوی دکتری مهندسی کامپیوتر گرایش سیستم‌های نرم‌افزاری در دانشگاه آزاد اسلامی واحد قزوین می‌باشد. زمینه‌های پژوهشی نامبرده عبارت است از: پردازش زبان‌های طبیعی، وب‌کاوی، متن کاوی و بازیابی اطلاعات، مهندسی سیستم‌های نرم‌افزاری خودتطبیق و محاسبات نرم می‌باشد. تاکنون از ایشان هجده مقاله در مجلات و کنفرانس‌های معتبر به چاپ رسیده است. عضو IEEE، انجمن رمز ایران و باشگاه پژوهشگران جوان و نخبگان می‌باشند.

آدرس پست‌الکترونیکی ایشان عبارت است از:
sadr@qiau.ac.ir

مژده نظری سلیماندرابی در سال ۱۳۶۸ در شهر رشت متولد شد. در سال ۱۳۹۱ مدرک کارشناسی از دانشگاه پیام نور و سال ۱۳۹۳ مدرک کارشناسی ارشد خود را با کسب رتبه ممتاز از دانشگاه آزاد اسلامی واحد علوم و تحقیقات اخذ نمود. نامبرده پس از اتمام دوره کارشناسی ارشد به عنوان مدرس دانشگاه آزاد اسلامی واحد لاهیجان مشغول به کار و پژوهش است و تاکنون شش مقاله ایشان در مجلات و کنفرانس‌ها به چاپ رسیده است. موضوعات تحقیقاتی ایشان عبارتند از: داده کاوی، وب کاوی و متن کاوی، مهندسی سیستم‌های خودتطبیق و محاسبات نرم و کاربردهای آن. عضو باشگاه پژوهشگران جوان و نخبگان می‌باشند.

آدرس پست‌الکترونیکی ایشان عبارت است از:
m.nazari@iaurasht.ac.ir

اطلاعات بررسی مقاله:

تاریخ ارسال: ۱۳۹۴/۰۴/۲۵
تاریخ اصلاح: ۱۳۹۴/۰۶/۰۴
تاریخ قبول شدن: ۱۳۹۴/۰۶/۱۴
نویسنده مرتبط: حسین صدر، دانشکده مهندسی برق، رایانه و فناوری اطلاعات، دانشگاه آزاد اسلامی واحد قزوین، قزوین، ایران.

---

[1] State Explosion
[2] Control Theory
[3] Feedback
[4] Feed Forward
[5] Meng
[6] Stability
[7] Robustness
[8] McCann
[9] Villegas
[10] Mean Time to Recover (MTTR)
[11] Interaction Complexity
[12] Coupling Strength