تاریخ دریافت مقاله: ۹۶/۴/۲۵
تاریخ پذیرش مقاله: ۹۶/۱۰/۲۰

تحلیل عدم قطعیت در ارزیابی معماری نرم‌افزار مبتنی بر نظریه شواهد

علی صداقت‌باف*
دانشجوی دکتری دانشکده مهندسی کامپیوتر ـ دانشگاه علم و صنعت ـ تهران ـ ایران
پست الکترونیکی: ali_sedaghat@comp.iust.ac.ir

محمد عبداللهی ازگمی
دانشیار دانشکده مهندسی کامپیوتر ـ دانشگاه علم و صنعت ـ تهران ـ ایران
پست الکترونیکی: azgomi@iust.ac.ir

چکیده:

پیشرفت‌های اخیر در مهندسی نرم‌افزار منجر به توسعه روش‌ها و ابزارهای قدرتمند به‌منظور ارزیابی ویژگی‌های کیفی بر روی معماری سامانه‌های نرم‌افزاری شده است. هدف از این ارزیابی‌ها کمک به معمار در اتخاذ تصمیمات مناسب در مورد طراحی سامانه بوده و اغلب زمانی انجام می‌گیرند که هنوز اطلاعات کافی در مورد سامانه وجود ندارد. به‌عبارت‌دیگر، این ارزیابی‌ها با نوعی عدم قطعیت همراه هستند. با وجودی که این عدم قطعیت در تصمیمات معماری بسیار تأثیرگذار است، در اغلب روش‌های ارزیابی موجود نادیده گرفته‌شده است. هدف از این مقاله، بررسی کاربردپذیری نظریه شواهد در توصیف صریح این عدم قطعیت و کمک به معمار در تصمیم‌گیری است. ضمن ارائه یک مرور اجمالی بر مبانی نظریه شواهد، یک فرایند شش مرحله‌ای برای ارزیابی معماری مبتنی بر این نظریه معرفی می‌شود. برای نشان دادن کاربردپذیری فرایند پیشنهادی، ارزیابی یک سامانه کنترل ماهواره شرح داده می‌شود. همچنین، نشان داده می‌شود چگونه می‌توان با تحلیل حساسیت، میزان تأثیر عدم قطعیت هر یک از پارامترهای ورودی را بر دقت نتایج ارزیابی سنجید.

واژه‌های کلیدی: معماری نرم‌افزار، ارزیابی کمّی، ویژگی‌های کیفی، عدم قطعیت، نظریه شواهد.

۱- مقدمه

معماری اولین محصول فرایند توسعه سامانه‌های نرم‌افزاری است که تصمیمات کلیدی اتخاذ شده در مورد نیازمندی‌های کارکردی و غیر کارکردی (کیفی) را منعکس می‌کند. بدین ترتیب، هر تصمیمی در مرحله طراحی معماری می‌تواند تأثیر قابل‌توجهی بر کیفیت محصول نهایی داشته باشد. در سال‌های اخیر، روش‌های متعددی به‌منظور ارزیابی زودهنگام معماری ارائه‌شده که هدف از آن‌ها سنجش معیارهای کیفی (مانند کارایی[1] و قابلیت

* نویسنده مسئول

1- performance

اطمینـــان[۲]) بر روی گزینه‌های مختلـــف طراحی معماری و کمک به معمار در انتخاب گزینه‌ای است که بیشترین تطابق را با نیازمندی‌ها و ترجیحات کیفی ســـودبران (ذی‌نفعان) دارد. عـــلاوه بر ایـــن، ارزیابی زودهنـــگام معماری امکان تشخیص و رفع نواقص سامانه را در مراحل اولیه توسعه فراهم می‌کند. این امر می‌تواند در کاهش هزینه‌های توسعه نرم‌افزار بســـیار مؤثر باشد، چراکه اعمال هرگونه تغییری در مراحل بعدی توسعه می‌تواند دشوار و پرهزینه باشد.

یکـــی از چالش‌های مهم ارزیابـــی زودهنگام معماری، عدم قطعیت قابل‌توجهی است که نسبت به ساختار، رفتار و محیط اجرای سامانه وجود دارد [۱]. به عبارت دقیق‌تر، مقدار دقیـــق اغلب پارامترهای مورداســـتفاده در ارزیابی مشـــخص نبـــوده و باید بر اســـاس داده‌هـــای موجود از پروژه‌های مشـــابه و یا نظر متخصصـــان کیفیت، آن‌ها را تخمین زد. تخمین مقدار دقیـــق این پارامترها در عمل کار دشواری است [۲] و این در حالی است که اغلب روش‌هایی که تاکنـــون برای ارزیابی معماری ارائه‌شـــده‌اند، برفرض دانش کافی نســـبت به مقادیر دقیـــق پارامترها در مرحله طراحی معماری اســـتوار هستند. این فرض دور از واقعیت بوده و می‌تواند به تصمیم‌های نادرست در طراحی سامانه منتهی شود.

بر اســـاس [۳]، عدم قطعیـــت در دنیـــای واقعی به دو نوع ذاتی[۳] و معرفتی[٤] قابل‌تقســـیم اســـت. درحالی‌که عدم قطعیت ذاتی به ماهیت پدیده‌ها مرتبط اســـت (مانند پرتاب ســـکه)، عدم قطعیت معرفتی از نبود دانش کافی در مورد پدیده‌ها سرچشمه می‌گیرد. مسلماً، عدم قطعیت در ارزیابی زودهنگام معماری نرم‌افزار از نوع معرفتی است. به‌منظور مدیریت این عدم قطعیت و کنترل تأثیر آن بر نتایج ارزیابی، توصیف صریح آن ضروری است [٤].

هدف از این مقاله، ارائه فرایندی برای ارزیابی معماری نرم‌افزار با در نظر گرفتن عدم قطعیت در پارامترها است. در این فرایند از نظریه شـــواهد به‌منظور توصیف صریح عـــدم قطعیت و تحلیل تأثیر آن بر نتایج ارزیابی اســـتفاده می‌شـــود. اســـتفاده از این روش به معمار امکان سنجش کمّـــی اثر تغییرات در معماری بر روی ویژگی‌های کیفی و همچنین مقایسه کمّی گزینه‌های مختلف طراحی و انتخاب گزینه مناسب را می‌دهد. علاوه بر این، می‌توان پارامترهایی را که عدم قطعیت در آن‌ها بیشترین تأثیر را بر کاهش دقت نتایج ارزیابی دارد، شناسایی نمود.

در ادامـــه ایـــن مقاله، در بخش دوم بـــه مرور کارهای مرتبط می‌پردازیم. بخش ســـوم به ارائه دانش پیش‌زمینه در مورد نظریه شواهد اختصاص دارد. فرایند پیشنهادی در بخش چهارم معرفی‌شده و کاربردپذیری آن با بررسی یک مطالعه موردی در بخش پنجم نشـــان داده می‌شـــود. درنهایت، نتایج به‌دســـت‌آمده و کارهـــای آینده در بخش ششم شرح داده می‌شود.

۲-کارهای مرتبط

در دو دهه اخیر، روش‌ها و ابزارهای متعددی به‌منظور ارزیابـــی ویژگی‌های کیفـــی معماری نرم‌افزار ارائه‌شـــده اســـت. به‌عنوان نمونـــه، در [۵-۷] از مدل‌هـــای مارکوف، شبکه‌های بیزی و درخت‌های خطا به‌منظور ارزیابی قابلیت اطمینان مدل‌های معماری استفاده‌شـــده اســـت. علاوه بر قابلیت اطمینان، کارایی ازجمله ویژگی‌های کیفی اســـت که ارزیابی آن در دو دهه اخیر بسیار موردتوجه بوده است. نمونه‌هایی از روش‌های مطرح برای ارزیابی این ویژگی در [۸] موردبررسی قرارگرفته است. فرض مشترک اغلب این روش‌ها دقت مقادیر پارامترهای ورودی اســـت. درنتیجه، این روش‌هـــا هیچ راهکاری برای توصیـــف و تحلیل عدم قطعیـــت موجود در پارامترها تأثیـــر آن بر نتایج ارزیابی ارائه نمی‌دهند.

در سال‌های اخیر تلاش‌هایی در راستای کمّی‌سازی عدم قطعیت در ارزیابی معماری نرم‌افزار صورت گرفته اســـت برای نمونه می‌توان به روش‌های شبیه‌ســـازی ارائه‌شـــده در [٤] و [۲] به ترتیـــب به‌منظور ارزیابی کارایی و قابلیت

2- reliability
3- aleatory
4- epistemic

اطمینان معماری نرم‌افزار اشـــاره کرد. در این روش‌ها هر پارامتر غیرقطعی به‌صورت یک متغیر تصادفی با یک تابع توزیع احتمال در نظر گرفته‌شده است. با نمونه‌برداری از تابع توزیع و تکرار فرایند ارزیابی برای هر نمونه می‌توان اثر عدم قطعیت در پارامترهـــای ورودی را بر روی نتایج ارزیابی سنجید. ازجمله معایب این روش‌ها می‌توان به نیاز آن‌ها به تکرار زیاد ارزیابی برای نمونه‌های مختلف داده‌ها اشـــاره کرد که فرایند ارزیابی را بسیار زمان‌بر می‌سازد. علاوه بر این، داده‌های موجود در مرحله طراحی معماری اغلب کمتر از آن است که بتوان تابع توزیع حقیقی را تخمین زد. حتی اگر داده کافی وجود داشـــته باشـــد، اغلب ازنظر آماری پایدار نبوده و تخصیص تابع توزیع مناسب به آن‌ها کار دشواری است [۹].

در [۱۰] استفاده از نظریه امکان[5] و مجموعه‌های فازی به‌عنوان راه‌حل جایگزین پیشنهادشده است. گرچه، استفاده از ایـــن روش نیاز به تخمین توابـــع توزیع احتمال و انجام مکرر فرایند ارزیابی را برطرف می‌ســـازد، اما استفاده از آن در عمل چالش‌برانگیز است [۹]. برای نمونه، بررسی‌ها [۹-۱۱] نشان می‌دهد که ارزیابی قابلیت اطمینان به‌صورت فازی می‌تواند به نتایجی دور از واقعیت منتهی شود. علاوه بر این، استفاده از نظریه امکان مستلزم تخصیص یک تابع توزیع امکان به هر پارامتر اســـت و این در حالی است که برخلاف نظریه احتمال، هیچ مبنای نظری برای ایجاد ارتباط بین توابع توزیع امکان و داده‌های آماری وجود ندارد.

در ســـال ۲۰۰۷ آقای لیو نظریـــه جدیدی به نام نظریه عـــدم قطعیـــت [۱۲] را باهدف کمّی‌ســـازی عـــدم قطعیت معرفتی پایه‌گذاری کرد. همچنیـــن، در [۱۳]معیار جدیدی مبتنی بر این نظریه برای ارزیابی غیرقطعی قابلیت اطمینان معرفی‌شده اســـت. علاوه بر این، در سال ۲۰۱۳، آقای لیو نظریه شـــانس[6] را به‌عنوان بســـطی از این نظریه به‌منظور توصیـــف یکپارچه هـــر دو نوع عدم قطعیـــت معرفی کرد. درعین‌حال، پیچیدگی‌های محاسباتی کاربردپذیری این دو نظریه را محدود ســـاخته است. تا آنجا که نویسندگان این مقالـــه اطلاع دارند، تاکنون کاربـــردی از این دو نظریه در ارزیابی ســـامانه‌های فنی گزارش نشـــده است. در ضمن، همانند نظریـــه احتمال و نظریه امکان، اســـتفاده از نظریه عدم قطعیت، نیازمند تخصیص نوعی تابع توزیع به مقادیر امکان‌پذیر برای پارامترهای غیرقطعی است.

استفاده از نظریه شـــواهد به‌منظور توصیف و تحلیل عدم قطعیت مزایای زیر را به همراه دارد:

- نیازی به تخمین تابع توزیع احتمال برای پارامترهای غیرقطعی نبوده و تنها کافی است بازه شامل مقادیر ممکن برای هر پارامتر مشخص شود،
- در صورت اســـتفاده از نظریه احتمال، نظریه امکان یا نظریه عدم قطعیت باید بر روی یک تابع توزیع خاص برای هر پارامتر به اجماع رسید و این در حالی است که استفاده از نظریه شـــواهد، امکان تجمیع شواهد (هرچند متناقض) برای تخمین هر پارامتر را فراهم می‌کند و
- نظریه شواهد عمومیت بیشتری نسبت به نظریه احتمال و نظریه امکان دارد؛ می‌توان به‌راحتی متغیرهای تصادفی و فازی را به متغیرهای بازه‌ای تبدیل نمود.

۳- نظریه شواهد

بر اساس نظریه شواهد [۱۴]، عدم قطعیت در مقدار یک متغیر را می‌توان در قالب یک بازه توصیف نمود که حدود بالا و پایین آن به ترتیب کمترین و بیشترین مقدار فرضی برای آن متغیر را مشخص می‌کنند. این نظریه به ما اجازه می‌دهد از منابع مختلف شواهد را جمع‌آوری و باهم ترکیب نمـــوده و به درجه‌ای از باور در مورد مقدار حقیقی متغیر برسیم.

در نظریه شـــواهد تمام فرضیات در مورد یک سامانه مجموعـــه Ω را شـــکل داده کـــه در اصطلاح بـــه آن قالب بصیـــرت[7] گویند. اگر P_Ω را به‌عنوان مجموعه توانی Ω در نظـــر بگیریم، به هر p_i به‌عنوان عضـــوی از P_Ω یک مقدار احتمالـــی تخصیص می‌یابد که میزان دانش در مورد p_i را

5- possibility theory
6- chance theory
7- frame of discernment

نشـــان می‌دهد. این دانش به‌صورت $m(p_i)$ مشخص‌شده و فرضیات زیر در مورد آن صادق است:

$$m(p_i): P_\Omega \rightarrow [0,1] \quad (۱)$$

$$m(\emptyset) - 0 \quad (۲)$$

$$\sum_{p_i \subseteq P_\Omega} m(p_i) = 1 \quad (۳)$$

ازآنجاکـــه ممکن اســـت از شـــواهد متعـــدد به‌منظور تخصیص مقدار احتمالی به p_i استفاده‌شده باشد، در [۱۵] قوانین متعددی به‌منظور تجمیع شـــواهد ارائه‌شـــده است. پرکاربردترین این قوانین، قانون دمستر است. با این فرض که تنها دو متخصص در اختیار داشته باشیم، قانون دمستر

$$m_{1-2}(p_i) = \frac{\sum_{p_a \cap p_b = p_i} m_1(p_a).\, m_2(p_b)}{1 - \sum_{p_a \cap p_b = \emptyset} m_1(p_a).\, m_2(p_b)} \quad (۴)$$

ازجمله معایب قانون دمستر این است که در هر شرایطی نمی‌توان از آن اســـتفاده نمود. در حقیقت، این قانون برای مواردی قابل‌اســـتفاده اســـت که بـــه ازای هر متخصص، حداقل یکی از بازه‌های پیشـــنهادی او مقدار حقیقی متغیر را شـــامل شـــود. ازجمله قوانین پرکاربرد دیگر که در هر شرایطی قابل‌استفاده اســـت، قانون ترکیب وزن‌دار[8] است. در این روش به هر یک از شـــواهد یک وزن w_i تخصیص داده می‌شـــود. این وزن میزان اهمیت آن را نشان می‌دهد. درصورتی‌که k تعداد شواهد موجود باشد و تمام شواهد از اهمیت یکســـان برخوردار باشـــند، به ازای $i = 1, \dots, k$ خواهیم داشت $w_i = 1$. در این مقاله از قانون ترکیب وزن‌دار به‌منظور تجمیع شـــواهد استفاده می‌شـــود. این قانون را می‌توان به‌صورت زیر تعریف نمود:

$$m(p_i) = \frac{\sum_{j=1}^{k} w_j m_j(p_i)}{\sum_{j=1}^{k} w_j} \quad (۵)$$

به‌منظور تحلیل تأثیر عدم قطعیت در متغیرهای ورودی یک مســـئله بر متغیرهای خروجی آن، در نظریه شواهد به ازای هر p_i دو معیار باور[9] و امکان‌پذیری[10] تعریف می‌شود که به ترتیب با $Bel(p_i)$ و $Pl(p_i)$ مشخص‌شـــده و حد پایین و بالای باور برای p_i را مشـــخص می‌کنند. این دو معیار با

$$Bel(p_i) = \sum_{p_k \subseteq p_i} m(p_k) \quad (۶)$$

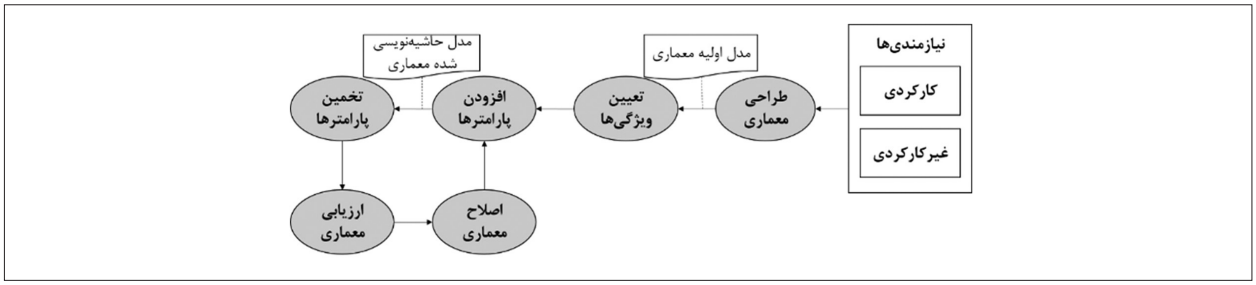
$$Pl(p_i) = \sum_{p_k \cap p_i \neq \emptyset} m(p_k) \quad (۷)$$

۴- فرایند پیشنهادی

در این بخـــش فرایندی مبتنی بر نظریه شـــواهد برای ارزیابی معماری نرم‌افزار معرفی می‌شود. گام‌های فرایند پیشنهادی که در شکل ۱ نیز نشان داده‌شده، عبارت‌اند از:

- **طراحی مدل اولیه معماری**: در این مرحله، مدل معماری ســـامانه بر مبنـــای نیازمندی‌های کارکردی تعریف‌شـــده برای آن، طراحی می‌شود. از هر یک از زبان‌های توصیف معمـــاری نرم‌افـــزار نظیـــر AADL، UML [۱۶] و PCM [۱۷] می‌توان به‌منظور توصیف این طرح اولیه استفاده نمود.
- **تعیین ویژگی‌هـــای کیفی برای ارزیابی**: تاکنون، صدها ویژگی کیفی برای ســـامانه‌های نرم‌افزاری تعریف‌شـــده [۱۸] که تنها تعداد محـــدودی از آن‌ها ازنظر ارزیابی کمّی موردتوجه هستند (مانند کارایی، دسترس‌پذیری[11]، ایمنی[12] و قابلیـــت اطمینـــان). در این مرحلـــه، ویژگی‌هایی که باید به‌صورت کمّی موردسنجش قرار گیرند مشخص می‌شوند.
- **افزودن پارامترهـــای ورودی به مدل معماری**: اطلاعات موجود در مدل اولیه معمـــاری صرفاً جنبه‌های کارکردی سامانه را پوشش می‌دهد و برای ارزیابی ویژگی‌های کیفی کافی نیست. درنتیجه، ضروری است پارامترهای جدیدی به مدل معماری افزوده شـــود. برای نمونه، برای ارزیابی کارایی پارامترهایی نظیر نمایه استفاده و بارکاری سامانه و همچنین زمـــان لازم برای اجرای فعالیت‌های نرم‌افزاری باید به مدل معماری افزوده شود. برای مثال، درصورتی‌که از زبـــان UML برای توصیف مدل معماری استفاده‌شـــده باشد، می‌توان این پارامترها را با استفاده از نمایه MARTE [۱۹] به مدل معماری افزود. زبان PCM نیز به‌صورت توکار حاشیه‌نویسی پارامترهای کارایی را پشتیبانی می‌کند.
- **تخمین مقادیر پارامترهـــای غیرقطعی**: اغلب، اطلاعات کافی برای پیش‌بینی مقدار دقیق بســـیاری از پارامترها در

8- weighted mixture rule
9- belief
10- plausibility
11- availability
12- safety



شکل ۱: فرایند پیشنهادی برای ارزیابی معماری

اختیار معمار نیست. در نبود اطلاعات کافی عوامل مختلفی دخیل است. برای نمونه، میزان و نحوه استفاده کاربران از سامانه به‌راحتی قابل‌سنجش نیست، چراکه هنوز نسخه قابل اجرایی از سامانه وجود ندارد. در این شرایط معمار به جمع‌آوری اطلاعات از منابع مختلف (مانند تجربیات پروژه‌های مشابه یا مشورت با متخصصین کارایی) پرداخته و بازه‌های مقادیر امکان‌پذیر به همراه احتمال صحت هر یک از آن آن‌ها را تخمین زده، سپس، آن‌ها را با استفاده از قانون ترکیب وزن‌دار (فرمول ۵) با یکدیگر ترکیب کرده و به مدل معماری اضافه می‌کند.

• **ارزیابی مدل معماری**: در این مرحله، مدل معماری نسبت به ویژگی(های) کیفی موردنظر مورد ارزیابی قرار می‌گیرد. برای این منظور ممکن است از مدل‌های صوری مانند زنجیره‌های مارکوف[13] یا شبکه‌های پتری[14] استفاده شود. خروجی این مرحله دو معیار باور و امکان‌پذیری است (بخش ۳ را ببینید)، که میزان انطباق نتایج ارزیابی با نیازمندی(های) کیفی مطرح برای سامانه را منعکس می‌کنند.

• **اصلاح مدل معماری**: بر اساس نتایج ارزیابی و در صورت لزوم، مدل معماری مورد بازبینی و اصلاح قرارگرفته و فرایند ارزیابی از گام سوم مجدداً تکرار می‌شود. اصلاح معماری مواردی نظیر تغییر در کیفیت یا کمیت منابع، جابجایی و جایگزینی مؤلفه‌ها را شامل می‌شود و ممکن است به‌صورت دستی یا به‌صورت خودکار و با استفاده از روش‌های قانون-مبنا [۲۰] یا الگوریتم‌های تکاملی [۲۱] انجام شود.

13- Markov chains
14- Petri nets

۵- مطالعه موردی

به‌منظور بررسی کاربردپذیری فرایند پیشنهادی، معماری یک سامانه بیش‌نرم‌افزاری[15] [۲۲] موردمطالعه قرارگرفته است. هدف از توسعه این سامانه کنترل ماهواره‌هایی است که برای شناسایی نقاط با دمای بالا (مانند آتش‌سوزی‌ها) در سطح زمین استفاده می‌شوند. در این مقاله، تنها دو مورداستفاده از این سامانه را بررسی می‌کنیم که عبارت‌اند از: (۱) کنترل موقعیت و چرخش ماهواره و (۲) تشخیص آتش‌سوزی. آنچه مسلم است، قابلیت اطمینان این سامانه بسیار حائز اهمیت است. چراکه زمانی که ماهواره در مدار است، در صورت وقوع هرگونه خرابی قابل تعمیر نیست. به همین دلیل، در معماری این سامانه، افزونگی و سازوکارهای تشخیص خطا پیش‌بینی‌شده است.

۵-۱ طراحی معماری

به‌منظور اثبات کاربردپذیری فرایند ارزیابی پیشنهادی، ابتدا یک منظر ساده از معماری این سامانه (بدون در نظر گرفتن افزونگی) را مورد بررسی قرار می‌دهیم. این منظر که در شکل ۲ نشان داده‌شده معماری سامانه را در قالب مؤلفه‌های تشکیل‌دهنده و ارتباطات بین آن‌ها نشان می‌دهد.

۵-۲ تعیین ویژگی‌های کیفی

ویژگی‌های در نظر گرفته برای سامانه کنترل ماهواره عبارت‌اند از: هزینه، وزن و قابلیت اطمینان. برای ارزیابی این ویژگی‌ها به هر مؤلفه پارامتری با همین نام تخصیص داده‌شده است. همان‌طور که در شکل ۲ نیز مشخص است، مقدار دو پارامتر هزینه و وزن برای هر مؤلفه معلوم است

15- software intensive

جدول ۱: تخمین پارامترهای ارزیابی برای سامانه کنترل ماهواره

پارامتر	منبع	بازه اولیه	احتمال اولیه	بازه پس از تجمیع	احتمال پس از تجمیع
قابلیت اطمینان منبع تغذیه	متخصص ۱	[۰/۹۸, ۱]	۱	[۰/۹۸, ۱]	۱
قابلیت اطمینان ردیاب خورشید	متخصص ۱	[۰/۹۸, ۱]	۰/۶	[۰/۹۸, ۱]	۰/۳
		[۰/۹۶, ۰/۹۸]	۰/۴	[۰/۹۶, ۰/۹۸]	۰/۲
	متخصص ۲	[۰/۹۵, ۰/۹۹]	۱	[۰/۹۵, ۰/۹۹]	۰/۵
قابلیت اطمینان کامپیوتر مدیریت	متخصص ۱	[۰/۹۸, ۱]	۱	[۰/۹۸, ۱]	۰/۵
	متخصص ۲	[۰/۹۷, ۰/۹۹]	۱	[۰/۹۷, ۰/۹۹]	۰/۵
قابلیت اطمینان ژیروسکوپ	متخصص ۱	[۰/۹۶, ۰/۹۹]	۱	[۰/۹۶, ۰/۹۹]	۱
	متخصص ۲	[۰/۹۶, ۰/۹۹]	۱		
قابلیت اطمینان چرخ‌های واکنش	متخصص ۱	[۰/۹۷, ۰/۹۹]	۱	[۰/۹۷, ۰/۹۹]	۱
قابلیت اطمینان حسگر ستاره	متخصص ۱	[۰/۹۸, ۰/۹۹]	۱	[۰/۹۸, ۰/۹۹]	۱
	متخصص ۲	[۰/۹۸, ۰/۹۹]	۱		
قابلیت اطمینان تبادل‌گر اطلاعات	متخصص ۱	[۰/۹۵, ۰/۹۸]	۰/۸	[۰/۹۵, ۰/۹۸]	۰/۸
		[۰/۹۶, ۰/۹۷]	۰/۲	[۰/۹۶, ۰/۹۷]	۰/۲
قابلیت اطمینان دوربین مادون‌قرمز	متخصص ۱	[۰/۹۵, ۰/۹۸]	۱	[۰/۹۵, ۰/۹۸]	۱
	متخصص ۲	[۰/۹۵, ۰/۹۸]	۱		

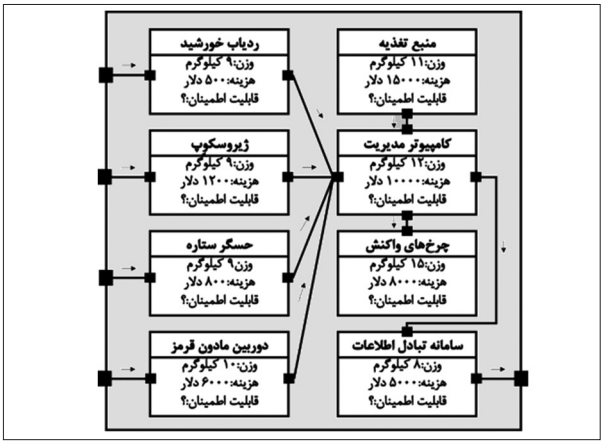
ولـــی اطلاعات کافی در مورد قابلیـــت اطمینان مؤلفه‌ها در اختیار نیست.

۵-۳-تخمین پارامترهای غیرقطعی

بازه‌های تخمین زده‌شـــده بـــرای پارامترهای نامعلوم به همراه نتایج تجمیع آن‌ها در جدول ۱ آمده اســـت. برای مثال، برای تخمین قابلیت اطمینان منبع تغذیه تنها ازنظر یک متخصص استفاده‌شده است. این متخصص نیز با اطمینان کامل بازه [۰/۹۸, ۱] را پیشـــنهاد داده اســـت. این در حالی اســـت که برای تخمین قابلیت اطمینان ردیاب خورشید دو متخصص نظر داده‌اند. متخصص اول دو بازه را پیشنهاد داده اســـت که احتمالات تخصیص داده‌شده به این بازه‌ها نشان می‌دهد اطمینان او از صحت بازه اول بیشتر است.

۵-٤ - ارزیابی معماری

پس از تجمیع شواهد و افزودن آن‌ها به مدل معماری، می‌تـــوان ویژگی‌های کیفـــی موردنظر را مـــورد ارزیابی قـــرارداد. برای ســـامانه کنتـــرل ماهواره، هزینـــه و وزن ســـامانه به‌صورت حاصل جمع هزینـــه و وزن هر یک از



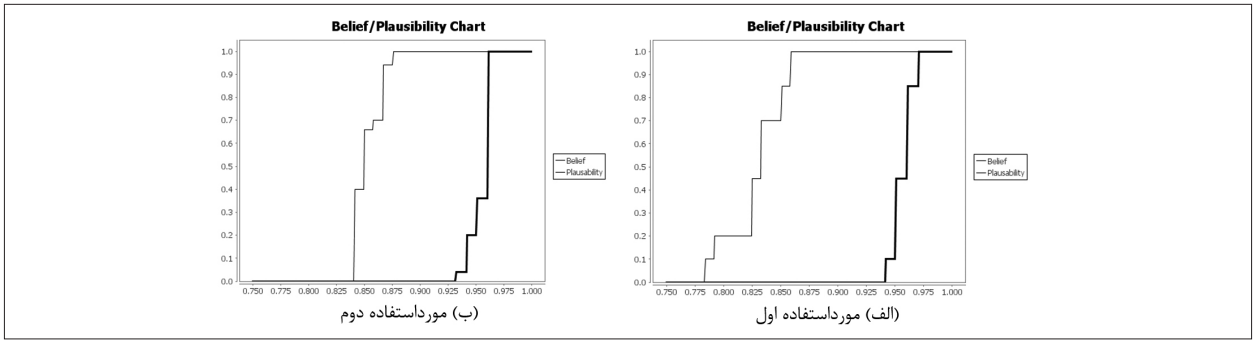
شکل ۲: معماری اولیه سامانه کنترل ماهواره

مؤلفه‌ها محاسبه می‌شود. برای ارزیابی قابلیت اطمینان نیز از نمودار بلوکی قابلیت اطمینان[16] اســـتفاده می‌کنیم. بدین ترتیب که برای ارزیابی قابلیت اطمینان مورداســـتفاده اول (یعنی کنترل موقعیت و چرخش ماهواره)، ترکیب ســـری مؤلفه‌هـــای کامپیوتر مدیریت، ردیاب خورشـــید، حســـگر ســـتاره، منبع تغذیه، ژیروســـکوپ و چرخ واکنش و برای مورداستفاده دوم، ترکیب سری کامپیوتر مدیریت، ردیاب خورشید، منبع تغذیه، دوربین مادون‌قرمز و سامانه تبادل

16- reliability block diagram

جدول ۲: نتایج ارزیابی ویژگی‌های کیفی برای سامانه کنترل ماهواره

قابلیت اطمینان مورداستفاده اول		قابلیت اطمینان مورداستفاده دوم		هزینه	وزن
بازه	احتمال	بازه	احتمال		
[۰/۸۵۸۹, ۰/۹۷۰۲۹]	۰/۱۵	[۰/۸۶۶۷۶, ۰/۹۶۰۵۸]	۰/۲۴	۳۳۰۰۰ دلار	۸۳ کیلوگرم
[۰/۷۹۱۵۶, ۰/۹۵۰۸۸]	۰/۱	[۰/۸۴۹۰۷, ۰/۹۴۱۳۶]	۰/۱۶		
[۰/۸۳۲۶, ۰/۹۶۰۵۸]	۰/۲۵	[۰/۸۴۰۲۲, ۰/۹۶۰۵۸]	۰/۴		
[۰/۸۵۰۱۳, ۰/۹۶۰۵۸]	۰/۱۵	[۰/۸۷۵۸۸, ۰/۹۵۰۷۷]	۰/۰۶		
[۰/۷۸۳۴۸, ۰/۹۴۱۳۷]	۰/۱	[۰/۸۵۸, ۰/۹۳۱۷۵]	۰/۰۴		
[۰/۸۲۴۱, ۰/۹۵۰۹۷]	۰/۲۵	[۰/۸۴۹۰۶, ۰/۹۵۰۷۷]	۰/۱		



(الف) مورداستفاده اول (ب) مورداستفاده دوم

شکل ۳: روند تغییرات دو معیار باور و امکان‌پذیری برای قابلیت اطمینان معماری اولیه سامانه کنترل ماهواره

اطلاعات لحاظ می‌شود.

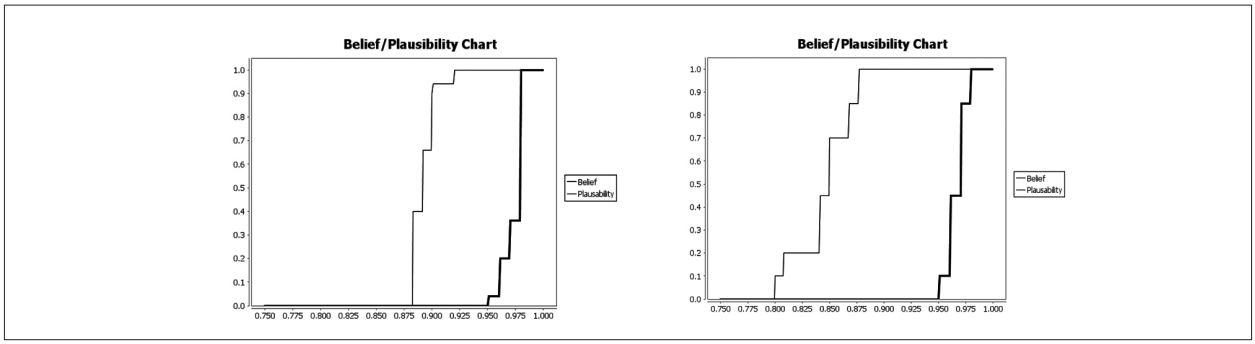
قابلیت اطمینان سامانه برای هر مورداستفاده از ضرب قابلیت اطمینان مؤلفه‌های فعال در آن مورداستفاده به دست می‌آید. درصورتی‌که مقدار پارامترها به‌صورت بازه مشخص‌شده باشد، برای ضرب آن‌ها از قوانین ساده ریاضی استفاده می‌کنیم. برای مثال، حاصل‌ضرب دو بازه [a, b] و [c, d] برابر است با: [ac, bd].

نتایج ارزیابی ویژگی‌های کیفی برای معماری اولیه سامانه کنترل ماهواره در جدول ۲ آمده است. با توجه به قطعیتی که در مورد مقدار وزن و هزینه هر مؤلفه وجود دارد، وزن و هزینه سامانه به‌صورت یک مقدار ثابت محاسبه‌شده است. این وضعیت در مورد قابلیت اطمینان برقرار نیست. با توجه به نتایج تجمیع شواهد در جدول ۱، برای قابلیت اطمینان ردیاب سه بازه، برای کامپیوتر مدیریت و تبادل‌گر اطلاعات دو بازه و برای سایر مؤلفه‌ها یک بازه مشخص‌شده است. بدین ترتیب، با در نظر گرفتن مؤلفه‌های فعال در هر مورداستفاده، شش بازه برای قابلیت اطمینان کل سامانه در هر دو حالت امکان‌پذیر خواهد بود. این شش بازه به همراه احتمال صحت هرکدام در جدول ۲ گزارش‌شده است.

بدین ترتیب، می‌توان معیارهای باور و امکان‌پذیری را به ازای آستانه‌های مختلف برای قابلیت اطمینان محاسبه کرد. در شکل ۳ روند تغییرات این دو معیار برای آستانه‌های بین ۰/۷۵ و ۱ نشان داده‌شده است. بر این اساس، می‌توان با احتمال ۰/٦٤ باور داشت قابلیت اطمینان سامانه برای مورداستفاده دوم بیشتر از ۰/۹٦ است. درعین‌حال، به توجه به عدم قطعیت موجود، وقوع این رخداد با احتمال ۱ امکان‌پذیر است.

نتایج ارائه‌شده در شکل ۳ با استفاده از نرم‌افزار SQMETool [۲۳] محاسبه‌شده است. این ابزار به زبان جاوا توسعه‌یافته و به‌عنوان یک افزونه برای محیط برنامه‌نویسی اکلیپس[17] قابل‌استفاده است. در حال حاضر، این ابزار از ارزیابی قابلیت اطمینان، کارایی و امنیت مدل‌های معماری نرم‌افزار که با استفاده از زبان UML توصیف‌شده‌اند پشتیبانی می‌کند. [۲٤]نمونه‌ای از کاربردهای این ابزار در

17- eclipse



(الف) مورداستفاده اول (ب) مورداستفاده دوم

شکل ۴: روند تغییرات دو معیار باور و امکان‌پذیری برای قابلیت اطمینان معماری اصلاح‌شده سامانه کنترل ماهواره

ارزیابی امنیت مدل‌های UML مبتنی بر فرایند پیشنهادی در این مقاله را ارائه می‌دهد.

۵-۵ اصلاح معماری

در این مرحله تغییراتی در مدل کنونی معماری باهدف بهبود یک یا چند ویژگی کیفی اعمال می‌شود. ویژگی‌های کیفی اغلب مستقل از یکدیگر نیستند. درنتیجه، اصلاح معماری به‌منظور بهبود یک ویژگی کیفی می‌تواند اثری سوء بر سایر ویژگی‌ها داشته باشد. برای مثال، یکی از روش‌های رایج بهبود قابلیت اطمینان استفاده از افزونگی است [۷]. درعین‌حال، باید توجه داشت، استفاده از این روش به‌منظور بهبود قابلیت اطمینان سامانه کنترل ماهواره منجر به افزایش وزن ماهواره خواهد شد. درنتیجه، در تعداد مؤلفه‌های افزونه‌ای که می‌توان به سامانه اضافه کرد محدودیت وجود دارد.

شکل ٤ تأثیر استفاده از یک دوربین و یک حسگر افزونه را بر قابلیت اطمینان سامانه کنترل ماهواره نشان می‌دهد. در طراحی جدید، می‌توان با احتمال ۰/۹٦ باور داشت که قابلیت اطمینان سامانه برای مورداستفاده دوم بیشتر از ۰/۹٦ است. این امر نشان‌دهنده اثر مثبت افزونگی بر قابلیت اطمینان است. البته، باید در نظر داشت همراه با بهبود در قابلیت اطمینان، وزن و هزینه ساخت سامانه به ترتیب به میزان ۱۹ کیلوگرم و ٦۸۰۰ دلار افزایش می‌یابد.

۵-۶ تحلیل حساسیت

فاصله زیاد بین نمودار باور و نمودار امکان‌پذیری قابلیت اطمینان سامانه کنترل ماهواره بیانگر عدم قطعیت بالا در پارامترهای ورودی و درنتیجه نیاز به جمع‌آوری شواهد بیشتر در مورد آن‌هاست. با تحلیل حساسیت[18] می‌توان تأثیر عدم قطعیت در هر پارامتر بر نتایج ارزیابی را به‌صورت کمّی سنجید و پارامترها را بر اساس نیاز به کسب اطلاعات بیشتر اولویت‌بندی نمود. در [۲۵] معیارهای متعددی برای این منظور تعریف‌شده است. در این بخش، معیار GH[19] مورداستفاده قرار می‌گیرد. این معیار به‌صورت زیر قابل‌تعریف است:

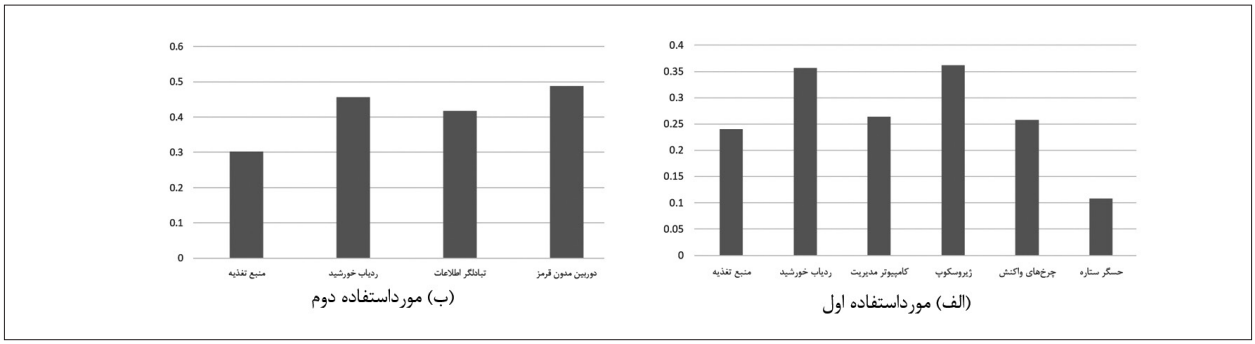
$$GH(m) = \sum_{g \in G} m(g) \log_2 |g| \qquad (۸)$$

که در آن $m(g)$ احتمال تخصیص داده‌شده به بازه g را مشخص می‌کند.

به‌منظور تحلیل حساسیت نتایج ارزیابی قابلیت اطمینان سامانه کنترل ماهواره به هر یک از پارامترهای غیرقطعی، کافی است بازه‌های قابلیت اطمینان این سامانه را هر بار با تخصیص یک مقدار ثابت به یکی از پارامترهای غیرقطعی به دست آورده و نتیجه محاسبه معیار GH برای آن بازه‌ها را با مقدار به‌دست‌آمده برای نتایج آمده در جدول ۲ مقایسه کنیم. واضح است که اختلاف دو مقدار تأثیر کاهش عدم قطعیت در پارامتر موردنظر بر نتایج ارزیابی را نشان می‌دهد.

نتایج مقایسه در شکل ۵ ارائه‌شده است. این نتایج نشان می‌دهد که برای مورداستفاده اول کاهش عدم قطعیت در

18- sensitivity analysis
19- generalized Hartley



(الف) مورداستفاده اول (ب) مورداستفاده دوم

شکل ۵: تأثیر کاهش عدم قطعیت در پارامترهای ورودی بر نتایج ارزیابی قابلیت اطمینان سامانه کنترل ماهواره

قابلیت اطمینان مؤلفه چهارم (ژیروسکوپ) بیشترین تأثیر را در افزایش دقت نتایج ارزیابی دارد. این در حالی است که برای مورداستفاده دوم تأثیر قابلیت اطمینان دوربین مادون‌قرمز بیش از سایرین است.

نتیجه‌گیری

با توجه به نبود دانش کافی در مرحله طراحی معماری سامانه‌های نرم‌افزاری و لزوم ارزیابی زودهنگام ویژگی‌های کیفی، در این مقاله تأثیر عدم قطعیت در ارزیابی ویژگی‌های کیفی معماری نرم‌افزار موردبررسی قرار گرفت. در روش پیشنهادی، مقادیر پارامترهای غیرقطعی به‌صورت بازه‌ای توصیف‌شده و از مبانی نظریه شواهد به‌منظور تجمیع بازه‌های ورودی و تحلیل نتایج ارزیابی استفاده‌شده است. برای نشان دادن کاربردپذیری روش پیشنهادی، معماری یک سامانه کنترل ماهواره موردمطالعه قرار گرفت. هزینه، وزن و قابلیت اطمینان سه ویژگی کیفی هستند که ارزیابی آن‌ها در این مطالعه موردتوجه قرارگرفته و تأثیر عدم قطعیت در قابلیت اطمینان هر یک از مؤلفه‌ها سامانه بر قابلیت اطمینان دو مورد از موارد استفاده آن موردبررسی قرار گرفت. در ضمن، تأثیر تغییر در معماری، مانند افزودن سازوکارهای افزونگی، بر نتایج ارزیابی مورد تحلیل قرار گرفت.

علی‌رغم این‌که در این مقاله کاربردپذیری نظریه شواهد برای توصیف و تحلیل عدم قطعیت در ارزیابی معماری نشان داده شد، اما روش‌های استفاده‌شده برای ارزیابی ویژگی‌های کیفی بسیار ساده است. بررسی کاربردپذیری روش پیشنهادی برای ارزیابی مدل‌های پیچیده‌تر معماری و پوشش روش‌های کارآمدتر ارزیابی به‌عنوان کار آینده در نظر گرفته‌شده است.

مراجع

1- Autili, M., Cortellessa, V., Di Ruscio, D., Inverardi, P., Pelliccione, P., Tivoli, M., EAGLE: engineering software in the ubiquitous globe by leveraging uncertainty, in Proceedings of the 19th ACM SIGSOFT Symposium and the 13th European Conference on Foundations of Software Engineering (FSE 2011), pp. 488–491, 2011.

2- Meedeniya, I., Moser, I., Aleti, A., Grunske, L., Architecture-based reliability evaluation under uncertainty, in Proceedings of the Joint ACM SIGSOFT Conference--QoSA and ACM SIGSOFT Symposium--ISARCS on Quality of Software Architectures--QoSA and Architecting Critical Systems--ISARCS, pp. 85–94, 2011.

3- Li, Y., Chen, J., Feng, L., "Dealing with uncertainty: a survey of theories and practices", IEEE Transaction on Knowledge and Data Engneering, vol. 25, no. 11, pp. 2463–2482, 2013.

4- Trubiani, C., Meedeniya, I., Cortellessa, V., Aleti, A., Grunske, L., Model-based performance analysis of software architectures under uncertainty, in Proceedings of the 9th International ACM Sigsoft Conference on Quality of Software Architectures, pp. 69-78, 2013.

5- Pham, T.-T., Defago, X., Reliability prediction for component-based software systems with architectural-level fault tolerance mechanisms, in Proceedings of the International Conference on Availability, Reliability and Security, pp. 11–20, 2013.

6- Roshandel, R., Medvidovic, N., Golubchik, L., A Bayesian model for predicting reliability of software systems at the Architectural Level, in Software Achitectures, Components, and Applications, Springer Berlin Heidelberg, pp. 108–126, 2007.

7-Bernardi, S., Merseguer, J., Petriu, D. C., Model-Driven Dependability Assessment of Software Systems. Springer Berlin

nal of Systems and Software, vol. 82, no. 1, pp. 3–22, 2009.
18-Bass, L., Clements, P., Kazman, R., Software Architecture in Practice, 3rd ed. Addison-Wesley Professional, 2012.
19-OMG, UML Profile for MARTE : Modeling and analysis of real-time embedded systems v.1.0, 2009.
20-Xu, J., "Rule-based automatic software performance diagnosis and improvement", Performance Evaluation, vol. 69, pp. 525–550, 2012.
21-Koziolek, A., Koziolek, H., Reussner, R., PerOpteryx: automated application of tactics in multi-objective software architecture optimization, in Proceedings of the Joint ACM SIGSOFT Conference -- QoSA and ACM SIGSOFT Symposium -- ISARCS on Quality of Software Architectures -- QoSA and Architecting Critical Systems -- ISARCS - QoSA-ISARCS '11, pp. 33-42, 2011
22-Grunske, L. Identifying 'good' architectural design alternatives with multi-objective optimization strategies, in Proceedings of the 28th International Conference on Software Engineering - ICSE '06, pp. 849-852, 2006.
23-Sedaghatbaf, A., SQME Tool Web Page (2015), available: http://twcl.iust.ac.ir/projects/sqme.html
24-Sedaghatbaf, A., Azgomi, M. A., "Quantitative evaluation of software security: an approach based on UML/SecAM and evidence theory," The ISC International Journal of Information Security, vol 8, no. 2, pp.137-49, 2016
25-Klir, G. H., Uncertainty and Information: Foundations of Generalized Information Theory. John Wiley & Sons, 2005.
Heidelberg, 2013.
8- Koziolek, H., "Performance evaluation of component-based software systems: a survey," Performance Evaluation, vol. 67, no. 8, pp. 634–658, 2010.
9- Utkin, L. V., Coolen, F. P., Imprecise reliability: an introductory overview, in Computational Intelligence in Reliability Engineering, Springer Berlin Heidelberg, pp. 261–306, 2007.
10- Esfahani, N., Malek, S., Razavi, K., GuideArch: guiding the exploration of architectural solution space under uncertainty, in Proceedings of the International Conference on Software Engineering, pp. 43–52, 2013.
11-Liu, B., "Why is There a Need for Uncertainty Theory?", Journal of Uncertain Systems, vol. 6, no. 1, pp. 3–10, 2012.
12- Liu, B., Uncertainty Theory, in Uncetainty Theory, pp. 205–234, 2007.
13-Zeng, Z., Wen, M., Kang, R., "Belief reliability: a new metrics for products' reliability", Fuzzy Optimization and Decision Making, vol. 12, no. 1, pp. 15–27, 2013.
14-Shafer, G., A Mathematical Theory of Evidence. Princeton University Press, 1976.
15-Sentz, K., Ferson, S., Combination of Evidence in Dempster-Shafer Theory, vol. 835. 2002.
16-Feiler, P. H., Lewis, B. A., Vestal, S., The SAE architecture analysis & design language (AADL) a standard for engineering performance critical systems, in Proceedings of the IEEE Conference on Computer Aided Control Systems Design, pp. 1206–1211, 2007.
17-Becker, S., Koziolek, H., Reussner, R., "The Palladio component model for model-driven performance prediction," Jour-