



پژوهش‌های نوین در تصمیم‌گیری

دوره ۷، شماره ۱، بهار ۱۴۰۱، صص ۶۱-۸۶

نوع مقاله: پژوهشی

ارزیابی ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل بر اساس استاندارد ایزو ۲۵۰۱۰

سمانه حسین‌دوست^۱، بهمن زمانی^{۲*}، افسانه فاطمی^۳

۱- دانشجوی دکتری، گروه پژوهشی مهندسی نرم‌افزار مدل‌رانده، دانشکده مهندسی کامپیوتر، دانشگاه اصفهان، اصفهان، ایران

۲- دانشیار، گروه پژوهشی مهندسی نرم‌افزار مدل‌رانده، دانشکده مهندسی کامپیوتر، دانشگاه اصفهان، اصفهان، ایران

۳- استادیار، گروه پژوهشی مهندسی نرم‌افزار مدل‌رانده، دانشکده مهندسی کامپیوتر، دانشگاه اصفهان، اصفهان، ایران

تاریخ پذیرش: ۱۴۰۰/۰۷/۱۹

تاریخ دریافت: ۱۴۰۰/۰۵/۲۱

چکیده

با توجه به نقش کلیدی و اهمیت استفاده از ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل در کاربردهای مختلف، مانند مدیریت و برنامه‌ریزی شهری، شبکه‌های اجتماعی، بازارهای مالی، جریان‌های ترافیکی و مدیریت بحران، ارزیابی و مقایسه کمی این ابزارها بر اساس یک چارچوب استاندارد خوش‌تعریف، ارزشمند است. تاکنون پژوهش‌های متعددی به بررسی و مقایسه ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل از جنبه‌های مختلف پرداخته‌اند. با این وجود، هیچ‌یک از این بررسی‌ها بر اساس یک چارچوب استاندارد خوش‌تعریف و به صورت کمی (عددی) انجام نشده است. به‌منظور پر کردن این شکاف تحقیقاتی، در این مقاله پنج مورد از پرکاربردترین ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل شامل AnyLogic، NetLogo، Repast، GAMA و MASON، بر اساس استاندارد شناخته‌شده ایزو ۲۵۰۱۰ در خصوص سنجش کیفیت نرم‌افزار در عمل و به روش هدف-پرسش-معیار با یکدیگر مقایسه و مورد ارزیابی قرار گرفته‌اند. نتایج ارزیابی نشان می‌دهد که ابزار AnyLogic در مقایسه با دیگران، از کیفیت بهتری برخوردار است و این موضوع، بیش‌تر به جذابیت ابزار از دیدگاه کاربر و تلاش کمتر برای توسعه یک برنامه مبتنی بر عامل در مقایسه با سایر ابزارها، باز می‌گردد.

کلیدواژه‌ها: ابزارهای شبیه‌سازی مبتنی بر عامل، استاندارد ایزو ۲۵۰۱۰، مدل‌سازی عامل‌گرا، کیفیت نرم‌افزار در عمل



۱- مقدمه

مدل‌سازی، روشی برای حل مسائل در دنیای واقعی است. از آنجایی که فرآیند ساخت، تخریب یا ایجاد تغییر در دنیای واقعی غالباً پرهزینه، خطرآفرین یا غیر قابل اجراست، در بسیاری از مواقع نمی‌توان برای یافتن راه‌حل‌های مناسب، بر روی اشیای واقعی آزمایش نمود. در این خصوص، مدل‌سازی یک امکان کم‌هزینه، بی‌خطر و قابل پیاده‌سازی است که با اجرای آن می‌توان وضعیت گذشته، حال و آینده یک سیستم را شناسایی نمود [۱]. مدل‌سازی در علوم مختلف و به شکل‌های مختلف انجام می‌شود. یکی از روش‌های مدل‌سازی مطرح، مدل‌سازی ریاضی است که در بسیاری از شاخه‌های علوم و مهندسی، مانند فیزیک، مکانیک، و هوافضا، از قدیم تا به امروز استفاده می‌شود، چراکه زبان ریاضیات، موجز و دقیق است و لذا هنوز هم تحلیل‌های ریاضی در علوم مختلف از اهمیت ویژه‌ای برخوردارند [۲]. برای مدل‌سازی، به‌خصوص وقتی که مدل پیچیده و بزرگ باشد و نیاز به محاسبات و شبیه‌سازی داشته باشد، استفاده از ابزارهای محاسباتی اجتناب‌ناپذیر است. حتی در موارد عمومی‌تر نیز، مانند پدیده‌های پیچیده اجتماعی که از اهمیت بالایی برخوردارند و در زمره فرآیندهای پویا و پیچیده به حساب می‌آیند، نیاز به مدل‌سازی و شبیه‌سازی این پدیده‌ها جهت تدارک پاسخ و رفتار مناسب در برابر آن‌ها، وجود دارد.

شبیه‌سازی مبتنی بر عامل، امکان مناسبی را برای مطالعه سیستم‌های پیچیده فراهم می‌کند. سیستم‌های پیچیده، سیستم‌هایی با تعداد زیادی عامل^۱ هستند که بین آن‌ها تعاملات غیرخطی^۲ برقرار است. یکی از رویکردهای مدل‌سازی سیستم‌های پیچیده، مدل‌سازی مبتنی بر عامل^۳ است. عامل، یک سیستم خودمختار^۴، واکنشی^۵ و کنشگر^۶ است که می‌تواند با عامل‌های دیگر از طریق یک زبان ارتباطی مخصوص^۷، تعامل برقرار کند [۳]. بسته به مسئله مورد نظر، عامل‌ها می‌توانند افراد، گروه‌ها، شرکت‌ها یا کشورها و تعاملات میان آن‌ها را نشان دهند. هر عامل دارای رفتارهای فردی است که از طریق مجموعه‌ای از قوانین نشان داده می‌شود. رفتار منتج شده از یک سیستم چندعاملی^۸، نتیجه تصمیمات و تعاملات عامل‌های موجود در آن است [۴]. [۵]. در رویکرد مدل‌سازی مبتنی بر عامل، می‌توان با استفاده از طیف وسیعی از ابزارهای مدل‌سازی، اقدام به مدل‌سازی منطق تصمیم و تعاملات عامل‌ها کرد که از جمله آن‌ها می‌توان به استفاده از عملگرهای منطقی یا جبری و قوانین «اگر-آن‌گاه» اشاره کرد. از جمله زمینه‌هایی که برای حل مسائل موجود در آن‌ها از مدل‌سازی مبتنی بر عامل استفاده شده است، می‌توان



به شبکه‌های اجتماعی، هوش جمعی، بازاریابی، بازارهای مالی، مهاجرت، جریان‌های ترافیکی و توسعه شهری اشاره نمود [۶]-[۹]. مدل‌سازی مبتنی بر عامل را می‌توان با استفاده از نرم‌افزارها یا زبان‌های برنامه‌نویسی همه منظوره، و یا نرم‌افزارها و ابزارهای طراحی شده خاص منظوره، انجام داد. ابزارها و نرم‌افزارهای خاص منظوره، شامل نیازمندی‌های خاص برای مدل‌سازی عامل‌ها هستند [۱۰]. هر مدل مبتنی بر عامل، شامل چهار عنصر اصلی است: عامل‌ها، محیط استقرار عامل‌ها، قوانینی که چگونگی تعامل عامل‌ها با یکدیگر و با محیط را تعریف می‌کنند، و جهان شبیه‌سازی [۲]، [۱۱]. برای ایجاد یک مدل مبتنی بر عامل، مدل‌ساز باید این عناصر را شناسایی، مدل‌سازی و برنامه‌نویسی کند. پس از مدل‌سازی، برای شبیه‌سازی رفتار عامل‌ها، به یک موتور محاسباتی نیاز است تا مدل را اجرا کند. بدین منظور می‌توان از ابزارهای مدل‌سازی مبتنی بر عامل بهره برد یا مستقیماً با استفاده از زبان‌های برنامه‌نویسی پیاده‌سازی‌ها را انجام داد [۱۰].

در سال‌های اخیر، ابزارها و محیط‌های توسعه مبتنی بر عامل زیادی ارائه شده‌اند. با توجه به اهمیت استفاده از این ابزارها در کاربردهای مختلف، تاکنون پژوهش‌های متعددی به بررسی و مقایسه ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل از جنبه‌های مختلف پرداخته‌اند [۱۲]-[۱۶]. اما هیچ‌یک از این بررسی‌ها بر اساس یک چارچوب استاندارد خوش‌تعریف^۱ و به صورت کمی (عددی) انجام نشده است. به منظور پر کردن این شکاف تحقیقاتی، در این مقاله قصد بر آن است تا پنج مورد از پرکاربردترین این ابزارها براساس استاندارد ISO/IEC ۲۰۱۱:۲۵۰۱۰ [۱۷] از جنبه کیفیت نرم‌افزار در عمل^۲ و به روش هدف-پرسش-معیار^۳ [۱۸] با یکدیگر مقایسه شوند. ابزارهایی که در این مقاله مورد بررسی مقایسه قرار می‌گیرند از جمله پرکاربردترین ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل هستند. این ابزارها عبارتند از: AnyLogic^{۱۲}، NetLogo^{۱۴}، Repast^{۱۴}، GAMA^{۱۵} و MASON^{۱۶} که امکان مدل‌سازی و شبیه‌سازی گرافیکی محیط مورد مطالعه را نیز فراهم می‌آورند. ارزیابی و مقایسه انجام شده در این پژوهش بدین صورت است که ابتدا یکی از معروف‌ترین مسائل مبتنی بر عامل به نام تفکیک شلینگ^{۱۷} با استفاده از این پنج ابزار، مدل‌سازی و شبیه‌سازی می‌شود تا هر یک از معیارهای تعریف شده برای ارزیابی، از جمله میزان تلاش مورد نیاز برای مدل‌سازی و شبیه‌سازی برنامه مورد مطالعه، اندازه‌گیری گردد. سپس با محاسبه مقدار متوسط معیارهای مربوطه، هر یک از اهداف ارزیابی محاسبه و در نهایت میانگین اندازه گرفته می‌شود تا مقدار



نهایی به‌عنوان کیفیت نرم‌افزار در عمل برای هر یک از ابزارهای مورد مطالعه به‌دست آید. ادامه مقاله بدین ترتیب سازماندهی شده است. در بخش ۲ پژوهش‌های مرتبط معرفی می‌گردند. بخش ۳، به معرفی ابزارهای مدل‌سازی و شبیه‌سازی مورد مطالعه این پژوهش می‌پردازد. در بخش ۴، ابزارها از جنبه کیفیت نرم‌افزار در عمل و براساس یک مطالعه موردی (مسئله تفکیک شلینگ) مورد ارزیابی قرار می‌گیرند. بخش ۵ به ارائه نتایج ارزیابی و بحث درباره آن‌ها می‌پردازد و در نهایت در بخش ۶ جمع‌بندی، نتیجه‌گیری و کارهای آتی بیان می‌شود.

۲- پژوهش‌های مرتبط

با توجه به اهمیت استفاده از ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل، تاکنون مقاله‌های مروری متعددی به بررسی و مقایسه این ابزارها از جنبه‌های مختلف پرداخته‌اند. در ادامه، به مرور مختصر دستاوردهای هر کدام از این پژوهش‌ها می‌پردازیم و ضعف هر کار یا تفاوت آن با پژوهش حاضر را نیز به اختصار بیان می‌نماییم.

شکشوکی^{۱۸} و جون^{۱۹} [۱۲] در سال ۲۰۰۴، به ارزیابی سه سکوی توسعه مبتنی بر عامل به‌نام‌های Zeus, Jade و Jack پرداخته‌اند. ارزیابی آن‌ها بر چهار ویژگی اصلی پشتیبانی از جاوا، ارزیابی کارایی، پشتیبانی از توسعه، و کارایی سیستم انتقال پیام تمرکز دارد. نتایج ارزیابی این پژوهشگران نشان داد که برای توسعه برنامه‌های مبتنی بر عامل که نیاز به زندگی عامل‌ها در یک ظرف^{۲۰} دارند، بهتر است از سکوی Jade استفاده شود، زیرا کارایی بهتری دارد. در مقابل، برای توسعه برنامه‌های مبتنی بر عامل که در آن‌ها نیاز به زندگی عامل‌ها در ظرف‌های متفاوتی می‌باشد، بهتر است از سکوی Jack استفاده شود. تفاوت این پژوهش با پژوهش حاضر در این است که در ابزارهای مورد نظر این پژوهش، امکان مدل‌سازی بصری توسط کاربر وجود ندارد. همچنین ارزیابی بر اساس یک چارچوب استاندارد خوش‌تعریف انجام نشده است، بلکه هدف اصلی صرفاً ارزیابی کارایی سکوها در شرایط مختلف بوده است. نیکولای^{۲۱} و میدی^{۲۲} [۱۳] در سال ۲۰۰۹، به بررسی ۵۳ سکوی مدل‌سازی و شبیه‌سازی مبتنی بر عامل پرداخته‌اند و هر یک از آن‌ها را بر اساس پنج ویژگی مهم که کاربران هنگام انتخاب یک ابزار در نظر می‌گیرند، توصیف کرده‌اند. این ویژگی‌ها عبارت‌اند از: زبان برنامه‌نویسی مورد نیاز، سیستم عامل مورد نیاز، نوع مجوز^{۲۳}، دامنه و کاربرد اصلی و میزان



پشتیبانی در دسترس برای کاربر. این ویژگی‌ها در قالب طبقه‌بندی‌های کاربرپسند دسته‌بندی شده تا به کاربر برای انتخاب ابزار کمک کند. در این پژوهش، مقایسه بر اساس یک چارچوب استاندارد خوش تعریف و به صورت کمی انجام نشده است.

آلن^{۲۴} [۱۹] در سال ۲۰۱۰، به بررسی ۴۴ ابزار مدل‌سازی و شبیه‌سازی مبتنی بر عامل بر اساس اطلاعات موجود در اینترنت و نیز تجربیات شخصی پرداخته است. در بررسی او به تاریخچه ابزارها، توصیف و کاربرد آن‌ها پرداخته شده است و دسته‌بندی یا معیار مشخصی برای ارزیابی ابزارها وجود ندارد.

سینگ^{۲۵} و همکاران [۱۴] در سال ۲۰۱۱، به بررسی ابزارهای توسعه مبتنی بر عامل و تحلیل نقاط قوت و ضعف آن‌ها پرداخته‌اند. در این پژوهش، پنج سکوی توسعه مبتنی بر عامل بر اساس ویژگی‌های کاربردی آن‌ها با یکدیگر مقایسه شده‌اند. سکوهای مورد مطالعه این پژوهش، Aglet، Voyager، Anchor، JADE و Zeus هستند و ویژگی‌هایی که از آن‌ها برای مقایسه استفاده شده است، عبارت‌اند از: طبیعت محصول^{۲۶} (رایگان یا تجاری بودن آن)، استانداردهای به‌کار گرفته‌شده در آن، تکنیک به‌کار گرفته‌شده برای ارتباطات بین عامل‌ها، کیفیت مکانیزم امنیتی، پشتیبانی از تحرک عامل‌ها^{۲۷} و مکانیزم مهاجرت^{۲۸} عامل‌ها. در نتیجه تحقیق آن‌ها، JADE به‌عنوان جذاب‌ترین سکو نسبت به پنج سکوی دیگر، شناخته‌شده است، زیرا متن‌باز است، کاملاً با جاوا طراحی شده، با رابط برنامه‌نویسی کاربردی^{۲۹} (API) سازگاری دارد، از سطح امنیت خوبی برخوردار است و از تحرک عامل‌ها پشتیبانی می‌کند. تفاوت پژوهش مذکور با پژوهش حاضر در این است که در پژوهش فوق، مقایسه بر اساس یک چارچوب استاندارد خوش تعریف و به صورت کمی انجام نشده است. همچنین امکان مدل‌سازی بصری و تلاش برای توسعه یک برنامه مبتنی بر عامل به‌عنوان یک معیار مقایسه، در نظر گرفته نشده است.

کراواری^{۳۰} و باسیلیادز^{۳۱} [۱۵] در سال ۲۰۱۵، مقایسه‌ای بین ۲۴ سکوی مدل‌سازی و شبیه‌سازی مبتنی بر عامل ارائه داده‌اند. بررسی آن‌ها مبتنی بر مجموعه جامعی از معیارهای مقایسه و ارزیابی [۲۰] است و در آن طبقه‌بندی‌هایی ارائه شده است که به خوانندگان در درک این‌که کدام ابزارها دارای ویژگی‌های مشابه هستند و در هر موقعیت چه ابزاری باید انتخاب شود، کمک می‌کند. آن‌ها برای مقایسه و ارزیابی ابزارها از پنج معیار اصلی استفاده نموده‌اند که عبارت‌اند از: ویژگی‌های سکو، کاربردپذیری، توانایی عملیاتی، عمل‌گرایی^{۳۲} و مدیریت امنیت.



نتایج ارزیابی آن‌ها حاکی از آن است که JADE هنوز هم به‌عنوان محبوب‌ترین سکو، مورد استفاده قرار می‌گیرد. زیرا کاملاً با جاوا طراحی شده است و از انواع مختلف سیستم‌های عامل در وب پشتیبانی می‌کند. همچنین برخی از سکوها، تجاری هستند که بیان‌گر آن است که صنعت هم در حال استفاده از فناوری مبتنی بر عامل است، زیرا فناوری‌های مبتنی بر عامل قابلیت کاربرد در صنعت را نیز دارند. در مقایسه با پژوهش حاضر، در این پژوهش امکان مدل‌سازی و تلاش لازم برای توسعه یک برنامه مبتنی بر عامل با استفاده از ابزارهای مطرح، مورد مقایسه قرار نگرفته است. همچنین مقایسه بر مبنای یک چارچوب کاملاً استاندارد مثل ISO و به‌صورت کمی انجام نشده است.

ابار^{۳۳} و همکاران [۱۶] در سال ۲۰۱۷، مقایسه‌ای جامع در زمینه فناوری محاسبات مبتنی بر عامل و ترکیب آن با حوزه مدل‌سازی و شبیه‌سازی انجام داده‌اند. آن‌ها ۸۵ بسته نرم‌افزاری برای مدل‌سازی و شبیه‌سازی مبتنی بر عامل را با یکدیگر مقایسه کرده‌اند. معیارهای ارزیابی مورد استفاده آن‌ها عبارت است از: مشخصات کد منبع^{۳۴}، انواع پیاده‌سازی عامل‌ها بر اساس مکانیزم‌های تعاملی در طول شبیه‌سازی، نیازمندی‌های زبان برنامه‌نویسی، پشتیبانی از یک واسط گرافیکی برنامه‌نویسی برای کدنویسی و اجرای شبیه‌سازی یا واسط کاربری گرافیکی برای مدل‌سازی، شناسایی کامپایلر، پشتیبانی سیستم‌عامل، و نیازمندی‌ها و محدودیت‌های سخت‌افزاری و نرم‌افزاری برای پیاده‌سازی مدل، میزان تلاش (ورودی) مورد نیاز برای طراحی و توسعه مدل کاربر، قدرت مدل‌سازی محاسباتی یا مقیاس‌پذیری مدل‌های توسعه‌یافته با استفاده از ابزارهای خاص، ارزیابی پوشش دامنه‌های کاربردی مورد حمایت ابزار، و توصیف مختصری از عملکرد مدل‌سازی و شبیه‌سازی فراهم آمده توسط هر یک از ابزارهای مدل‌سازی و شبیه‌سازی مورد مطالعه. هدف اصلی این پژوهش، کمک به دانشمندان و مهندسان در ارزیابی سریع در مورد نحوه انتخاب و بکارگیری صحیح ابزارهای مدل‌سازی و شبیه‌سازی در برنامه‌های تحقیقاتی‌شان بوده است. در این پژوهش، مقایسه بر اساس استاندارد ISO و به‌صورت کمی انجام نشده است.

در پژوهش جاری قصد بر آن است که نقاط ضعف مذکور، برطرف گردند. به‌عبارتی هدف این است که چند مورد از معروفترین ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل که امکان مدل‌سازی یا شبیه‌سازی گرافیکی محیط مورد مطالعه را نیز دارند، براساس یک چارچوب استاندارد خوش تعریف و به‌صورت کمی با یکدیگر مقایسه شوند. در این راستا



مهم‌ترین معیاری که می‌تواند در انتخاب ابزار مناسب برای توسعه‌دهندگان برنامه‌های مبتنی بر عامل، مورد توجه قرار گیرد، میزان رضایت‌بخشی و تلاش مورد نیاز برای توسعه یک برنامه کاربردی است که در این پژوهش به آن پرداخته می‌شود. جدول ۱، مقایسه روش مدنظر این پژوهش را با کارهای مرتبط نشان می‌دهد. شایان ذکر است که در برخی از پژوهش‌های فوق [۱۳]، [۱۵]، [۱۶]، ابزارهای AnyLogic، NetLogo، Repast، GAMA و MASON مورد مطالعه و ارزیابی قرار گرفته‌اند، اما معیارهای مقایسه و روش ارزیابی آن‌ها با پژوهش حاضر، متفاوت است. همچنین، با توجه به این‌که هدف این پژوهش ارزیابی و مقایسه ابزارهایی است که امکان مدل‌سازی یا شبیه‌سازی گرافیکی محیط را فراهم می‌آورند، لذا چارچوب‌ها و سکوهایی مثل JADE که صرفاً امکان برنامه‌نویسی مبتنی بر عامل را فراهم می‌آورند، در این پژوهش مورد بررسی قرار نگرفته‌اند.

جدول ۱. مقایسه پژوهش حاضر با کارهای مرتبط

ردیف	پژوهش	سال ارائه	ابزارهای مورد مطالعه	معیارهای مقایسه	روش ارزیابی
۱	شکشوکی و جون [۱۲]	۲۰۰۴	Zeus Jade Jack	پشتیبانی از جاوا ارزیابی کارایی پشتیبانی از توسعه کارایی سیستم انتقال پیام	پایه‌سازی یک نمونه سیستم انتقال پیام با هر یک از ابزارها و توسعه یک محک برای مقایسه کارایی ابزارها روی سیستم انتقال پیام
۲	نیکولای و میدی [۱۳]	۲۰۰۹	۵۳ ابزار مدل‌سازی و شبیه‌سازی مبتنی بر عامل	زبان برنامه‌نویسی مورد نیاز سیستم‌عامل مورد نیاز نوع مجوز دامنه و کاربرد اصلی میزان پشتیبانی در دسترس برای کاربر	جمع‌آوری اطلاعات از منابع آزاد مثل مقالات، گزارش‌های فنی و وب‌گاه‌های معتبر
۳	آلن [۱۹]	۲۰۱۰	۴۴ ابزار مدل‌سازی و شبیه‌سازی مبتنی بر عامل	بر اساس تاریخچه، توصیف و کاربرد ابزارها	اطلاعات موجود در اینترنت و تجربیات شخصی



ردیف	پژوهش	سال ارائه	ابزارهای مورد مطالعه	معیارهای مقایسه	روش ارزیابی
۴	سینگ و همکاران [۱۴]	۲۰۱۱	Aglet, Voyager, Anchor, JADE, Zeus	نوع مجوز استانداردهای به‌کار گرفته‌شده در پیاده‌سازی محصول تکنیک به‌کار گرفته‌شده برای ارتباطات بین عامل‌ها کیفیت مکانیزم امنیتی پشتیبانی از تحرک عامل‌ها مکانیزم مهاجرت عامل‌ها	بررسی مستندات مربوطه
۵	کراواری و باسیلیادز [۱۵]	۲۰۱۵	۲۴ ابزار مدل‌سازی و شبیه‌سازی مبتنی بر عامل	بر اساس مجموعه جامعی از معیارهای مقایسه و ارزیابی [۲۰]	مطالعه مقالات، آموزش‌ها، وب‌گاه‌ها و فروم‌های مربوطه، به‌علاوه تجربه شخصی نویسندگان در استفاده از ابزار به‌منظور بررسی میزان راحتی یادگیری و استفاده از ابزار
۶	ابار و همکاران [۱۶]	۲۰۱۷	۸۵ ابزار مدل‌سازی و شبیه‌سازی مبتنی بر عامل	ویژگی‌ها و مشخصات فنی ابزارها	برگرفته از وب‌گاه‌های توسعه‌دهندگان که در آن‌ها مستندات، آموزش‌ها و ابزار برای دانلود در دسترس است.
۷	پژوهش حاضر	۲۰۲۰	AnyLogic, NetLogo, Repast, GAMA, MASON	براساس استاندارد ایزو ۲۵۰۱۰ در خصوص کیفیت نرم‌افزار در عمل	به روش «هدف-پرسش-معیار» و به صورت کمی

۳- معرفی ابزارهای شبیه‌سازی مورد مطالعه

همان‌گونه که پیش‌تر اشاره شد، هدف از این مقاله مقایسه و ارزیابی پنج ابزار مطرح در حوزه مدل‌سازی و شبیه‌سازی مبتنی بر عامل است. این ابزارها به این دلیل انتخاب شده‌اند که اولاً



همگی امکان مدل‌سازی و شبیه‌سازی گرافیکی محیط مورد مطالعه را فراهم می‌آورند و ثانیاً در اغلب پژوهش‌های مربوطه، مورد استفاده قرار گرفته‌اند و این نشان از پرکاربرد بودن این ابزارها است [۱۳]، [۱۵]، [۱۶]. ابزارهای انتخاب شده عبارت‌اند از: NetLogo، AnyLogic، Repast، MASON، GAMA. در ادامه هر کدام از این ابزارها در بخشی جداگانه معرفی می‌گردند.

۱-۳- AnyLogic

AnyLogic یک ابزار مدل‌سازی و شبیه‌سازی چند منظوره است که توسط شرکت former XJ Technologies توسعه یافته است. از این نرم‌افزار تاکنون برای شبیه‌سازی بازارهای رقابتی بهداشت و درمان، تولید، زنجیره تأمین تدارکات و خرده‌فروشی، فرآیندهای کسب‌وکار اجتماعی و اکوسیستم دفاعی پویا، مدیریت پروژه و دارایی، عابر پیاده پویا و ترافیک جاده‌ای، هوا فضا و غیره استفاده شده است. از جمله قابلیت‌های AnyLogic می‌توان به قابلیت اجرا بر روی سیستم‌عامل‌های ویندوز، مک و لینوکس، پشتیبانی از رویکرد شیء‌گرا، پشتیبانی از عناصر استاندارد UML، پشتیبانی از زبان جاوا، و پشتیبانی از واسط گرافیکی کاربر به صورت دو بعدی و سه‌بعدی اشاره کرد [۲۱].

۲-۳- NetLogo

NetLogo یک محیط مدل‌سازی چندعاملی است که امکان برنامه‌نویسی نیز دارد. این ابزار دارای کتابخانه‌ای غنی از مدل‌هاست که برای کاربران تازه‌کار که می‌خواهند ساخت مدل‌های اولیه خود را آغاز کنند، مناسب خواهد بود. این ابزار بر مبنای زبان‌های Lisp و Logo نوشته شده و کاربر نیز با همین زبان‌ها می‌تواند برنامه‌نویسی کند. این نرم‌افزار توسط Wilensky Uri در سال ۱۹۹۹ معرفی شد و در مرکز آموزش ارتباطات و مدل‌سازی کامپیوتری توسعه یافت. این نرم‌افزار برای شبیه‌سازی سیستم‌های پیچیده طراحی شده است و کاربردهای آن در علوم ریاضی، فیزیک، شیمی، کامپیوتر، پزشکی، اجتماعی و اقتصادی است. از جمله ویژگی‌های محیط توسعه NetLogo می‌توان به مواردی همچون رایگان و متن‌باز بودن، قابلیت اجرا بر روی سکوها مختلف، سادگی ساختار زبانی، امکان مرتبط نمودن اشیای ایجاد شده و شبکه‌سازی آن‌ها، دارا بودن مجموعه قابل توجهی از اشکال از پیش طراحی‌شده، قابلیت مشاهده مدل به صورت دو بعدی و سه بعدی، اشکال قابل چرخش و مقیاس‌پذیر، خط فرمان



هوشمند، دارا بودن اهرم سرعت که اجازه کنترل سرعت مدل و مشاهده خروجی را با دقت بیشتر می‌دهد و قابلیت رسم نمودار به صورت انعطاف‌پذیر و قدرتمند اشاره کرد [۲۲].

۳-۳- Repast

Repast از سکوه‌های مدل‌سازی و شبیه‌سازی مبتنی بر عامل پیشرفته است که در آزمایشگاه ملی Argonne توسعه یافت و در حال تکمیل و ارتقا می‌باشد. Repast Symphony ۲,۲ که سال ۲۰۱۴ منتشر شده است، یک سیستم مدل‌سازی مبتنی بر جاواست که برای استفاده روی ایستگاه‌های کاری و خوشه‌های محاسباتی کوچک طراحی شده است [۲۳]. از جمله قابلیت‌های محیط توسعه Repast Symphony می‌توان به رایگان و متن‌باز بودن، امکان شبیه‌سازی برنامه‌های مبتنی بر عامل با مقیاس بزرگ، توسعه و اجرای مدل‌های بصری، اتصال خودکار به پایگاه داده، ثبت وقایع خروجی به صورت خودکار و بصری‌سازی نتایج اشاره کرد [۱۹].

۳-۴- GAMA

GAMA یک محیط توسعه برای مدل‌سازی و شبیه‌سازی برنامه‌های مبتنی بر عامل است که طراحی، توسعه و نگهداری آن توسط محققان دانشگاه‌های فرانسه و ویتنام انجام شده است. شبیه‌ساز گاما دارای قابلیت‌ها و مزایای بسیاری بوده که برخی از این قابلیت‌ها عبارتند از (۱) حوزه کاربرد چندگانه: گاما با یک رویکرد عمومی توسعه یافته است و می‌تواند برای بسیاری از کاربردها مورد استفاده قرار گیرد. همچنین پلاگین‌هایی متناسب با نیازهای خاص توسعه داده شده است. زمینه‌هایی که گاما اغلب برای آن‌ها استفاده شده است، موضوعاتی مانند مدیریت شهری و اپیدمیولوژی می‌باشد. (۲) زبان برنامه‌نویسی سطح بالا و شهودی: می‌توان مدل‌ها را به راحتی با استفاده از زبان GAML که یک زبان سطح بالا و مبتنی بر عامل است، پیاده‌سازی نمود. (۳) استفاده برای مدل‌های GIS^{۳۵} و مدل‌هایی با تعداد داده زیاد: می‌توان عامل‌ها را از هر مجموعه داده‌ای با هر اندازه، حتی مجموعه‌های شامل داده‌های GIS ساخت و شبیه‌سازی‌ها را در مقیاس بزرگ حتی تا میلیون‌ها عامل انجام داد. (۴) واسط کاربری بیانی: اجرای شبیه‌سازی‌های دوبعدی و سه‌بعدی از ویژگی‌های دیگر این شبیه‌ساز است. گاما امکان داشتن نمایش‌های چندگانه برای یک مدل را فراهم می‌کند. همچنین این امکان را می‌دهد که به راحتی جداول و انواع مدل‌های نموداری را تعریف نمود [۲۴].



۵-۳- MASON

MASON یک کتابخانه شبیه‌سازی چند عاملی مبتنی بر جاوا است که به‌عنوان پایه‌ای برای شبیه‌سازی با زبان جاوا طراحی شده است و از امکانات کافی برای بسیاری از کاربردهای شبیه‌سازی برخوردار است. MASON شامل یک کتابخانه مدل و مجموعه‌ای اختیاری از ابزارهای تصویرسازی دو بعدی و سه بعدی است. این پروژه محصول همکاری مشترک بین آزمایشگاه محاسبات تکاملی جورج میسون و مرکز GMU است. از جمله ویژگی‌های این ابزار می‌توان به سریع و قابل‌حمل بودن، امکان حذف، اضافه و تغییر مدل‌ها به‌صورت مستقل از نمایش گرافیکی آن‌ها، امکان بازیابی مدل‌های طراحی‌شده و اجرای آن‌ها به‌صورت پویا روی سکوهاى مختلف، عدم تغییر نتایج اجرای مدل‌ها در سکوهاى متفاوت، امکان نمایش دو بعدی و سه‌بعدی، و امکان ایجاد تصویر PNG، فیلم، نمودار و جریان‌های داده خروجی اشاره کرد [۲۵].

۴- مدل نظری ارزیابی برای ابزارهای شبیه‌سازی مبتنی بر عامل

به‌منظور ارزیابی ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل، در این پژوهش از استاندارد ISO/IEC ۲۵۰۱۰ استفاده می‌شود. ISO/IEC ۲۵۰۱۰ بخشی از یک استاندارد بین‌المللی به‌نام Systems and software Quality Requirements and Evaluation است که به اختصار SQuaRE نامیده می‌شود. این استاندارد شامل مدیریت کیفیت (ISO/IEC ۲۵۰۰۰)، مدل کیفیت (ISO/IEC ۲۵۰۱۰)، اندازه‌گیری کیفیت (ISO/IEC ۲۵۰۲۰)، نیازمندی‌های کیفیت (ISO/IEC ۲۵۰۳۰)، و ارزیابی کیفیت (ISO/IEC ۲۵۰۴۰) است [۱۷]. با توجه به آن‌که هدف این مقاله، ارزیابی ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل از دیدگاه کاربر بوده است، لذا در این مقاله صرفاً از بخش مدل کیفیت آن استفاده می‌شود. سایر بخش‌های استاندارد SQuaRE عموماً در طراحی و توسعه نرم‌افزارها، مورد استفاده قرار می‌گیرند. در ادامه، ابتدا به اختصار تاریخچه این استاندارد بیان می‌گردد و سپس به تشریح استاندارد مدل کیفیت (ISO/IEC ۲۵۰۱۰) پرداخته می‌شود.

کمیته مشترکی از ISO و IEC در سال ۱۹۹۱ برای نخستین بار استاندارد ISO/IEC ۹۱۲۶ را برای کیفیت محصولات نرم‌افزاری ارائه کرد. در سال ۲۰۰۱ نسخه جدید این استاندارد با عنوان ISO/IEC ۹۱۲۶-۱ ارائه شد. در سال ۲۰۱۱ آخرین نسخه آن به نام ISO/IEC ۲۵۰۱۰ معرفی شد که هم‌اکنون به عنوان نسخه جاری مورد استفاده قرار می‌گیرد [۱۷]. شایان ذکر است که ISO مجموعه‌ای از استانداردها تحت عنوان n-ISO/IEC ۱۵۰۰۴ و



ISO/IEC ۳۳۰۰n را نیز برای تعیین کیفیت فرایند طراحی و تولید معرفی کرده‌است که خارج از موضوع این تحقیق است.

استاندارد بین‌المللی ISO/IEC ۲۵۰۱۰:۲۰۱۱ دو دسته معیار به شرح زیر تعریف می‌کند [۱۷]:

الف) یک دسته معیار با عنوان کیفیت در عمل که عبارت است از درجه‌ای که یک محصول یا سیستم می‌تواند توسط کاربران خاص، مورد استفاده قرارگیرد و نیازهای آن‌ها را برای دستیابی به اهداف خاص در زمینه‌های خاص مورد استفاده، برآورده سازد [۱۷]. ویژگی‌های کیفیت در عمل به پنج مشخصه اصلی به‌نام‌های اثربخشی^{۳۶}، کارایی^{۳۷}، رضایت‌بخشی^{۳۸}، بی‌خطر بودن^{۳۹} و پوشش زمینه^{۴۰} تقسیم‌بندی می‌شوند که برخی از آن‌ها شامل زیرمشخصه‌هایی هستند. به‌عنوان مثال، معیار رضایت‌بخشی، شامل سودمندی^{۴۱}، اعتماد^{۴۲}، لذت‌بخشی^{۴۳} و راحتی^{۴۴} است. به‌طور کلی این معیارها مرتبط با تعامل کاربران با محصول است.

ب) یک دسته معیار با عنوان مدل کیفیت محصول^{۴۵} که شامل هشت مشخصه به‌نام‌های مناسب بودن عملکردی^{۴۶}، قابلیت اطمینان^{۴۷}، بهره‌وری کارایی^{۴۸}، کاربردپذیری^{۴۹}، قابلیت نگهداری^{۵۰}، امنیت^{۵۱}، تطبیق‌پذیری^{۵۲}، قابلیت جابجایی^{۵۳} است و هر کدام از آن‌ها به زیرمشخصه‌هایی تقسیم می‌شود. به‌عنوان مثال، کاربردپذیری شامل ویژگی‌های تشخیص مناسب بودن^{۵۴}، قابلیت یادگیری^{۵۵}، قابلیت عملکردی^{۵۶}، حفاظت از خطای کاربر^{۵۷}، زیبایی واسط کاربری^{۵۸} و دسترس‌پذیری^{۵۹} است. به‌طور کلی معیارهای مطرح شده در مدل کیفیت محصول، مرتبط با ویژگی‌های ایستای نرم‌افزار و ویژگی‌های پویای سیستم کامپیوتری می‌باشند.

مشخصه‌های تعریف‌شده در هر دو دسته، مرتبط با همه محصولات نرم‌افزاری و سیستم‌های کامپیوتری است. شایان ذکر است که ویژگی‌های مربوط به «کیفیت در عمل» با ویژگی‌های «مدل کیفیت محصول» در برخی موارد با یکدیگر همپوشانی دارند. به‌عنوان مثال، کاربردپذیری علاوه‌بر آن‌که در دسته معیارهای مدل کیفیت محصول است، به‌عنوان زیرمجموعه‌ای از کیفیت نرم‌افزار در عمل نیز تعریف شده است که شامل اثربخشی، کارایی و رضایت‌بخشی است [۱۷]. همان‌طور که در بخش ۱ بیان شد، هدف این پژوهش ارزیابی ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل بر اساس استاندارد ایزو ۲۵۰۱۰ و از جنبه کیفیت نرم‌افزار در عمل است، از طرفی با توجه به زمینه مورد استفاده ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل، مشخصه‌های بی‌خطر بودن و پوشش زمینه عموماً برای ارزیابی و



انتخاب این دسته از ابزارها، اثرگذار نیست. لذا در این مقاله، تنها به ارزیابی بر اساس مشخصه‌های اثربخشی، کارایی و رضایت‌بخشی که مجموعاً کاربردپذیری نامیده می‌شود، پرداخته می‌شود. برای این منظور، از رویکرد هدف-پرسش-معیار استفاده می‌شود. این رویکرد اولین بار در سال ۱۹۹۴، توسط باسیلی و همکاران برای توسعه مدل‌های ارزیابی کیفیت محصول، مورد استفاده قرار گرفت [۱۸]. این رویکرد، امکان گسترش و ارتقای مدل ارزیابی کیفی را در هر زمان می‌دهد.

در روش هدف-پرسش-معیار، در گام اول اهداف ارزیابی مشخص می‌شوند. منظور از اهداف، مواردی است که می‌خواهیم برای یک محصول یا ابزار مورد ارزیابی قرار گیرند. سپس هر هدف به مجموعه‌ای از پرسش‌های قابل اندازه‌گیری شکسته می‌شود. یک پرسش می‌تواند برای یک یا چند هدف، مورد استفاده قرار گیرد. در نهایت، مجموعه‌ای از معیارها برای پاسخ‌گویی به هر پرسش به صورت کمی، در نظر گرفته می‌شود. یک معیار می‌تواند برای بیش از یک پرسش، مورد استفاده قرار گیرد. جدول ۲، اهداف، پرسش‌ها و معیارهای تعریف‌شده برای ارزیابی ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل را نشان می‌دهد. این مدل ارزیابی از پژوهش حسین و همکاران [۲۶] برای ارزیابی برنامه‌های کاربردی موبایل اقتباس شده و برای دامنه مورد نظر این پژوهش سازگار شده است.

جدول ۲. رویکرد هدف-پرسش-معیار پیشنهادی برای ارزیابی مشخصه‌های کاربردپذیری در ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل (برگرفته از [۲۶])

معیارها	پرسش‌ها	اهداف	مشخصه‌های کاربردپذیری
(a) رضایت از روند نصب	۱- نصب ابزار چقدر آسان است؟	سادگی ^{۶۱}	اثربخشی
(b) زمان مورد نیاز برای نصب			
(c) تعداد تعاملات مورد نیاز با کاربر هنگام نصب ابزار			
(d) رضایت از بخش راهنمای ابزار	۲- آیا یادگیری ابزار آسان است؟		
(e) زمان صرف‌شده برای یادگیری			
(f) تعداد عناصر اتمی مدل	۳- برای توسعه یک شبیه‌سازی مبتنی بر		



معیارها	پرسشها	اهداف	مشخصه‌های کاربردی پذیری
(g) تعداد خطوط کد نوشته شده به صورت دستی	عامل، چه میزان تلاش مورد نیاز است؟	دقت ۶۲	
(h) رضایت از خروجی	۴- آیا به سادگی می‌توان از خروجی استفاده کرد؟		
(i) رضایت از بهینه‌سازی اندازه صفحه نمایش			
(j) رضایت از خروجی	۵- آیا ابزار، دقیق است؟	دقت ۶۳	
(k) تعداد خطاها			
(l) زمان صرف شده برای شروع به کار ابزار	۶- چقدر زمان توسط ابزار برای پاسخگویی صرف می‌شود؟	زمان‌بری ۶۳	کارایی
(m) زمان صرف شده برای پاسخ‌گویی در حین مدل‌سازی و شبیه‌سازی			
(n) زمان صرف شده برای یادگیری	۷- چقدر زمان توسط کاربر برای یادگیری صرف می‌شود؟	خصیصه‌ها ۶۴	
(o) رضایت از بخش راهنمای ابزار	۸- آیا ابزار راهنمای مناسبی را ارائه می‌دهد؟		
(p) تعداد منابع سیستم که نمایش داده می‌شود	۹- چقدر اطلاعات در مورد منابع سیستم نمایش داده می‌شود؟		
(q) تعداد درخواستها برای به‌روزرسانی ابزار	۱۰- آیا ابزار، به‌روزرسانی خودکار ارائه می‌دهد؟		
(r) درصد باتری مصرف شده در طول نصب ابزار	۱۱- چه مقدار باتری توسط ابزار مصرف می‌شود؟	اعتماد	رضایت‌بخشی
(s) رضایت از خروجی	۱۲- آیا شبیه‌ساز طبق برنامه از پیش تعریف شده (هنگام طراحی) رفتار می‌کند؟		
(t) تعداد خطاها در طول یک شبیه‌سازی	۱۳- آیا ابزار برای استفاده از امنیت کافی برخوردار است؟		
(v) یافتن راهنما به آسانی	۱۴- آیا کاربران از واسط کاربری راضی هستند؟	لذت‌بخشی	
(w) رضایت از ویرایش‌گر مدل‌سازی			
(x) رضایت از ویرایش‌گر کد			
(y) رضایت از خروجی اجرای مدل از لحاظ گرافیک و واسط کاربری			



معیارها	پرسش‌ها	اهداف	مشخصه‌های کاربردی پذیری
(Z) زمان صرف‌شده برای یادگیری	۱۵- آیا واسط کاربری برای کاربران آشناست؟		
(Z ₁) رضایت از واسط کاربری			
(Z ₂) احساس رضایت در حین یادگیری ابزار			
(Z ₃) رضایت از نتایج و پیامدهای استفاده از ابزار	۱۶- آیا هدف مورد نظر کاربر در کار با ابزار محقق شده است؟	سوزمندی	
(Z ₄) دستیابی به اهداف عملی کاربر از شبیه‌سازی			

به‌منظور یکسان‌سازی نتایج، برای هر یک از معیارهای جدول ۲، از مقیاس لیکرت پنج سطحی^{۶۵} استفاده می‌شود. در این مقیاس، هر یک از مقادیر ۱ تا ۵ به ترتیب به‌عنوان ۱:خیلی کم، ۲:کم، ۳:متوسط، ۴:زیاد و ۵:خیلی زیاد تعریف شدند.

۵- ارائه نتایج و بحث

در این بخش، ابزارهای معرفی‌شده در بخش ۳، بر اساس استاندارد ISO/IEC ۲۵۰۱۰:۲۰۱۱، از جنبه کیفیت نرم‌افزار در عمل و به روش هدف-پرسش-معیار ارزیابی می‌شوند. برای این منظور، یک مطالعه موردی واحد به‌نام تفکیک شلینگ [۲۷] در هر پنج ابزار مورد بررسی و آزمایش قرار می‌گیرد. این مدل، به‌عنوان یک مثال اجرایی در نسخه رایگان هر پنج ابزار موجود است. بدین ترتیب می‌توان معیارها را تا حدی مستقل از ویژگی‌های شخصیتی یا خبرگی کاربران اندازه‌گیری نمود و صرفاً بر روی ویژگی‌های ابزارها تمرکز کرد. به‌منظور اندازه‌گیری معیارهای مطرح در این مقاله، ابتدا نسخه‌های رایگان هر پنج ابزار، از وب‌گاه مربوط به آن‌ها دانلود و بر روی سیستم عامل ویندوز نصب گردید. پس از یادگیری ابزارها با استفاده از مستندات مربوطه، مدل تفکیک شلینگ در هر ابزار، توسط نویسندگان مقاله اجرا و مورد بررسی قرار گرفت. سپس بر اساس کار با ابزار و بررسی و اجرای مدل تفکیک شلینگ، به



سؤالات موجود در جدول ۲ بر اساس مقیاس لیکرت پاسخ داده شد. به‌عنوان مثال، برای پرسش شماره ۳، پاسخ‌ها به‌صورت نشان داده شده در جدول ۳ است.

جدول ۳. تلاش مورد نیاز برای توسعه مدل تفکیک شلینگ با استفاده از ابزارهای مورد مطالعه

پرسش	معیارها	AnyLogic	Repast Symphony	NetLogo	MASON	GAMA
برای توسعه یک شبیه‌سازی مبتنی بر عامل، چه میزان تلاش مورد نیاز است؟	(f) تعداد عناصر اتمی مدل	۱۱	۰	۸	۰	۰
	(g) تعداد خطوط کد نوشته‌شده به‌صورت دستی	۲۰	۱۰۸	۸۴	۲۱۸	۹۹

پس از آن، به‌منظور پیش‌گیری از ابهام و دقیق شدن نتایج، برای برخی از سؤالات، جدول ۴ به‌عنوان یک جدول نگاشت تهیه گردید. نحوه بخش‌بندی بازه‌ها و نگاشت آن‌ها به هر یک از اعداد موجود در این جدول، بر اساس پاسخ‌های داده شده به پرسشنامه بوده است. به‌عنوان مثال در ردیف ۴ این جدول، برای تعداد خطوط کد نوشته شده به صورت دستی، همان‌طور که جدول ۳ نشان داده شده است، اعداد بین بازه ۲۰ تا ۲۱۸ می‌باشند. بنابراین بازه‌ها به شکل متوازن و به‌صورت ارائه شده در جدول ۴ در نظر گرفته شده‌اند. پس از نرمال‌سازی نتایج بر اساس مقیاس لیکرت و منطبق با جدول ۴، نتایج ارزیابی در جدول ۵ نشان داده شده است.

جدول ۴. نگاشت معیارهای ارزیابی پیشنهادی به مقیاس لیکرت

ردیف	معیار	مقیاس لیکرت متناظر				
		۵	۴	۳	۲	۱
۱	زمان مورد نیاز برای نصب (بر حسب دقیقه) * صفر به معنای عدم نیاز به نصب نرم‌افزار است.	بیش‌تر از ۴	[۳, ۴]	[۲, ۳]	[۱, ۲]	(۱, ۰]
۲	تعداد تعاملات مورد نیاز با کاربر هنگام نصب ابزار	بیش‌تر از ۸	۸ و ۷	۶ و ۵	۴ و ۳	۲ و ۱
۳	تعداد عناصر اتمی مدل	بیش‌تر از ۲۰	۱۶-۲۰	۱۱-۱۵	۶-۱۰	۰-۵



ردیف	معیار	مقیاس لیکرت متناظر				
		۵	۴	۳	۲	۱
۴	تعداد خطوط کد نوشته شده به صورت دستی	بیش‌تر از ۲۰۰	۱۵۱-۲۰۰	۱۰۱-۱۵۰	۵۱-۱۰۰	۵۰-۱
۵	تعداد خطاها	بیش‌تر از ۳	۳	۲	۱	۰
۶	زمان صرف شده برای شروع به کار ابزار (برحسب ثانیه)	بیش‌تر از ۸۰	۶۱-۸۰	۴۱-۶۰	۲۱-۴۰	۲۰-۱
۷	تعداد منابع سیستم که نمایش داده می‌شود	بیش‌تر از ۳	۳	۲	۱	۰
۸	تعداد درخواست‌ها برای به‌روزرسانی ابزار	در هر بار اجرای نرم‌افزار	۳	۲	۱	۰
۹	میزان مصرف باتری در حین نصب ابزار (بر حسب درصد)	بیش‌تر از ۴	[۳, ۴]	[۲, ۳]	[۱, ۲]	[۰, ۱]

جدول ۵. مقادیر اندازه‌گیری شده برای هر یک از معیارهای ارزیابی مورد مطالعه

GAMA	MASON	NetLogo	Repast Symphony	Anyllogic	معیارها
۵	۵	۵	۵	۴	(a) رضایت از روند نصب
۱	۱	۳	۳	۵	(b) زمان مورد نیاز برای نصب
۱	۱	۲	۳	۴	(c) تعداد تعاملات مورد نیاز با کاربر هنگام نصب ابزار
۳	۳	۴	۴	۵	(d) رضایت از بخش راهنمای ابزار
۲	۳	۲	۲	۲	(e) زمان صرف شده برای یادگیری
۱	۱	۲	۱	۳	(f) تعداد عناصر اتمی مدل
۲	۵	۲	۳	۱	(g) تعداد خطوط کد نوشته شده به صورت دستی
۴	۳	۴	۵	۴	(h) رضایت از خروجی
۵	۵	۵	۴	۴	(i) رضایت از بهینه‌سازی اندازه صفحه نمایش
۴	۳	۴	۵	۴	(j) رضایت از خروجی
۱	۱	۱	۱	۱	(k) تعداد خطاها
۱	۱	۱	۴	۵	(l) زمان صرف شده برای شروع به کار ابزار



معيارها	AnyLogic	Repast Symphony	NetLogo	MASON	GAMA
(m) زمان صرف‌شده برای پاسخ‌گویی در حین مدل‌سازی و شبیه‌سازی	۵	۵	۴	۴	۴
(n) زمان صرف‌شده برای یادگیری	۲	۲	۲	۳	۲
(o) رضایت از بخش راهنمای ابزار	۵	۴	۴	۳	۳
(p) تعداد منابع سیستم که نمایش داده می‌شود	۲	۱	۱	۱	۲
(q) تعداد درخواست‌ها برای به‌روزرسانی ابزار	۵	۱	۱	۱	۱
(r) درصد باتری مصرف شده در طول نصب ابزار	۵	۳	۳	۱	۱
(s) رضایت از خروجی	۴	۵	۴	۳	۴
(t) تعداد خطاها در طول یک شبیه‌سازی	۱	۱	۱	۱	۱
(u) امنیت هنگام کار با ابزار	۵	۳	۳	۳	۳
(v) یافتن راهنما به آسانی	۵	۴	۵	۳	۳
(w) رضایت از ویرایش‌گر مدل‌سازی	۴	۲	۳	۱	۱
(x) رضایت از ویرایش‌گر کد	۵	۵	۳	۵	۴
(y) رضایت از واسط کاربری	۵	۳	۴	۲	۲
(z) زمان صرف‌شده برای یادگیری	۲	۲	۲	۳	۲
(z ^۱) رضایت از واسط کاربری	۵	۳	۳	۲	۲
(z ^۲) احساس رضایت در حین یادگیری ابزار	۵	۳	۴	۳	۳
(z ^۳) رضایت از نتایج و پیامدهای استفاده از ابزار	۵	۵	۵	۵	۵
(z ^۴) دستیابی به اهداف عملی کاربر از شبیه‌سازی	۵	۵	۵	۵	۵

حال برای محاسبه کمی هر یک از اهداف ارزیابی، میانگین معیارهای مربوط به آن‌ها، مطابق با رابطه‌های ۱ تا ۷ محاسبه می‌گردد. به‌منظور سادگی انجام محاسبات و جلوگیری از ابهام و پیچیدگی، همه معیارها با وزن یکسان در نظر گرفته شده‌اند و هر یک از معیارها، بر اساس اینکه آیا رابطه مستقیم یا معکوس با هدف مورد نظر دارند، به‌صورت مستقیم یا معکوس ($\frac{1}{\text{معیار}}$) در رابطه لحاظ شده‌اند. به‌عنوان مثال، معیارهای b و c رابطه معکوس با سادگی دارند. لذا در رابطه ۱، به‌صورت $\frac{1}{b}$ و $\frac{1}{c}$ در نظر گرفته شده‌اند. نتایج اندازه‌گیری برای هر یک از ابزارهای مورد مطالعه، در نمودار ۱ نشان داده شده است.



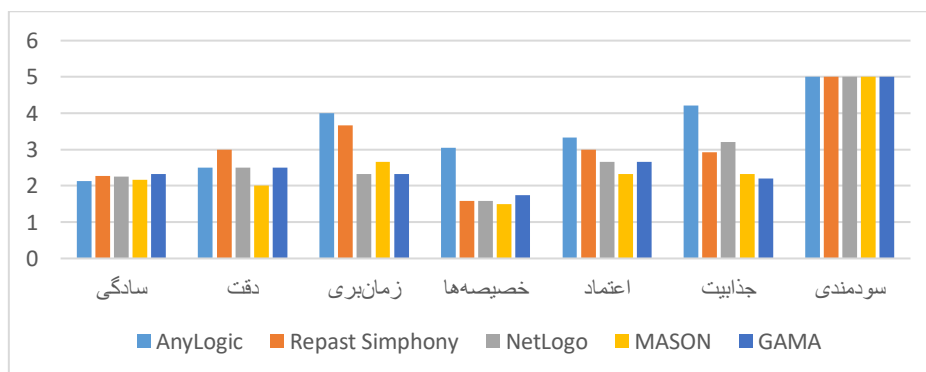
$$\text{Simplicity} = \frac{a + \frac{1}{b} + \frac{1}{c} + d + \frac{1}{e} + \frac{1}{f} + \frac{1}{g} + h + i}{9} \quad (1), \quad \text{Accuracy} = \frac{j + \frac{1}{k}}{2} \quad (2)$$

$$\text{Time Taken} = \frac{l+m+n}{3} \quad (3), \quad \text{Features} = \frac{o+p+q + \frac{1}{r}}{4} \quad (4)$$

$$\text{Trust} = \frac{s + \frac{1}{t} + u}{3} \quad (5), \quad \text{Pleasure} = \frac{v+w+x+y + \frac{1}{z} + z_1 + z_2}{5} \quad (6)$$

$$\text{Usefulness} = \frac{z_2 + z_3}{2} \quad (7)$$

در ادامه، مشخصه‌های کاربردپذیری نیز بر اساس میانگین‌گیری از معیارهای مربوطه، با فرض برابر بودن ارزش همه پارامترها و با در نظر گرفتن رابطه مستقیم یا معکوس هر یک از پارامترها با معیار هدف، مطابق با رابطه‌های ۸ تا ۱۰ اندازه‌گیری می‌شوند. مقادیر به‌دست آمده برای هر یک از ابزارهای مورد مطالعه، در نمودار ۲ نشان داده شده است. در نهایت، برای محاسبه معیار کاربردپذیری به‌عنوان زیرمجموعه‌ای از معیار کیفیت نرم‌افزار در عمل، از رابطه ۱۱ استفاده می‌شود. نتایج نهایی در نمودار ۳ نشان داده شده است.



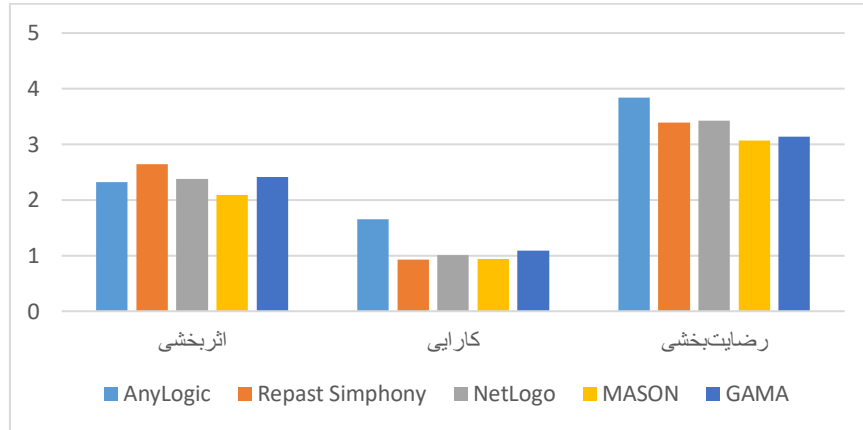
نمودار ۱. نتایج به‌دست آمده برای هر یک از اهداف ارزیابی ابزارهای مورد مطالعه

$$\text{Effectiveness} = \frac{\text{Simplicity} + \text{Accuracy}}{2} \quad (8)$$

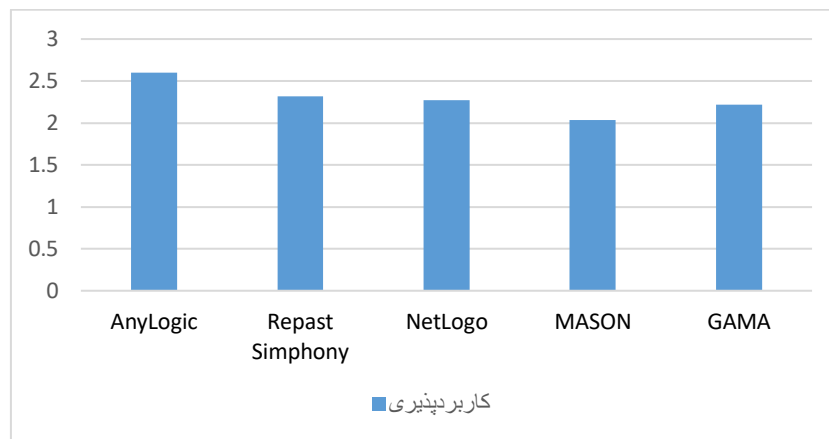
$$\text{Efficiency} = \frac{\frac{1}{\text{Time Taken}} + \text{Features}}{2} \quad (9)$$

$$\text{Satisfaction} = \frac{\text{Trust} + \text{Pleasure} + \text{Usefulness}}{3} \quad (10)$$

$$\text{Usability} = \frac{\text{Effectiveness} + \text{Efficiency} + \text{Satisfaction}}{3} \quad (11)$$



نمودار ۲. نتایج ارزیابی مشخصه های کاربرد پذیری برای هر یک از ابزارهای مورد مطالعه



نمودار ۳. مقادیر به دست آمده برای کاربرد پذیری هر یک از ابزارهای مورد مطالعه

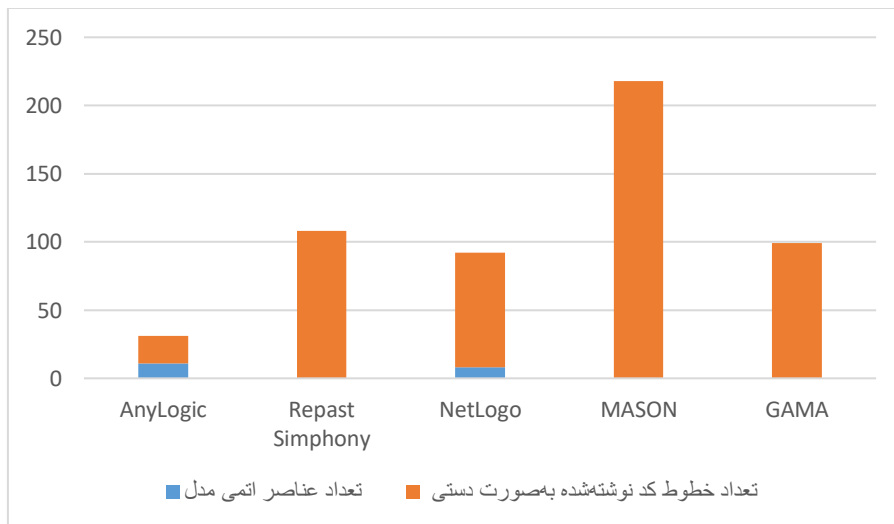
بر اساس بررسی ها و نتایج ارائه شده در جدول ۵ و نمودارهای ۱ تا ۳، یافته های این پژوهش به شرح زیر است.

۱- ابزار AnyLogic، دارای بالاترین میزان کاربرد پذیری و ابزار MASON کمترین مقدار کاربرد پذیری را داراست. با کمی کاوش در جداول ۴ و ۵ مشخص می شود که دلیل این امر به میزان رضایت بخشی بالا از دیدگاه کاربر، در ابزار AnyLogic باز می گردد.



۲- ابزار AnyLogic با فراهم آوردن یک واسط کاربری مناسب شامل ویرایش‌گر مدل‌سازی و همچنین ویرایش کد کاربرپسند، توانسته است رضایت کاربران را جلب کند.

۳- همچنین با بررسی میزان تلاش مورد نیاز برای مدل‌سازی و شبیه‌سازی تفکیک شلینگ (معیارهای f و g در جدول ۲ که نتایج مربوط به آن پیش از نگاشت به مقیاس لیکرت در جدول ۳ و نمودار ۴ نشان داده شده است)، مشخص می‌شود که مدل‌سازی و شبیه‌سازی تفکیک شلینگ با ابزار AnyLogic، به کمترین میزان تلاش برای توسعه نیاز داشته است. زیرا این ابزار امکان مدل‌سازی و تولید خودکار کد را بیش از سایر ابزارها در اختیار کاربر قرار می‌دهد.



نمودار ۴. تلاش مورد نیاز برای توسعه مدل تفکیک شلینگ با استفاده از ابزارهای مورد مطالعه

شایان ذکر است که نتایج این پژوهش، صرفاً بر اساس معیارهای کاربرپذیری و پرسشنامه مطرح شده در جدول ۲ به دست آمده است. اگر هرکدام از این ابزارها را بر اساس معیارهای دیگری، همچون رایگان بودن، متن باز بودن، سبک بودن ابزار و مانند آن مورد ارزیابی قرار دهیم، ممکن است نتایج متفاوتی حاصل شود. به عنوان مثال، ابزار NetLogo یکی از ابزارهای پرکاربرد در جامعه دانشگاهی است. با وجود آن‌که در NetLogo میزان کدنویسی عموماً بیش‌تر از ابزار AnyLogic است، اما به دلیل انعطاف‌پذیری بالایی که دارد، دانشجویان



با یک بار صرف وقت برای یادگیری کدنویسی در این ابزار، می‌توانند به راحتی انواع مدل‌های مبتنی بر عامل را تولید و اجرا کنند. همچنین ابزار AnyLogic با وجود آن‌که المان‌های گرافیکی زیادی در اختیار کاربران قرار می‌دهد، اما نسخه حرفه‌ای این ابزار، رایگان نیست و به همین دلیل کم‌تر مورد استفاده دانشگاهیان قرار می‌گیرد.

۶- جمع‌بندی و نتیجه‌گیری

مدل‌سازی مبتنی بر عامل، یک روش مدل‌سازی و محاسباتی برای شبیه‌سازی فرآیندهای پویا است که در آن‌ها عامل‌های خودمختار نقش اساسی ایفاء می‌کنند. این رویکرد در طول سال‌های گذشته، بیش از پیش مورد توجه قرار گرفته و اغلب برای مدل‌سازی و شبیه‌سازی تصمیم‌گیری‌های فردی و رفتارهای اجتماعی و سازمانی، از ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل استفاده شده است [۱۰].

در این پژوهش، پنج مورد از پرکاربردترین ابزارهای مدل‌سازی و شبیه‌سازی مبتنی بر عامل معرفی شدند و بر اساس استاندارد ISO/IEC ۲۵۰۱۰:۲۰۱۱، از جنبه کیفیت نرم‌افزار در عمل مورد ارزیابی قرار گرفتند. ارزیابی به‌روش هدف-پرسش-معیار و بر مبنای انجام یک مطالعه موردی واحد انجام شد، تا بتوان میزان تلاش مورد نیاز برای توسعه یک برنامه مبتنی بر عامل را به‌صورت کمی اندازه‌گیری نمود. نتایج ارزیابی نشان داد که اگرچه معیار کاربردپذیری برای هر یک از این ابزارها، مقادیر نزدیک به هم دارد، اما برای AnyLogic بالاترین میزان رضایت‌بخشی از دیدگاه کاربر، بیش‌ترین مقدار کاربردپذیری به دست آمده است. دلیل این امر به احتمال زیاد به میزان تلاش کم برای توسعه مدل‌های مبتنی بر عامل بستگی داشته است. زیرا نرم‌افزار AnyLogic امکان مدل‌سازی و تولید خودکار کد را بیش از سایر ابزارها فراهم می‌آورد. در حالی که در سایر ابزارها، تلاش بیش‌تری برای کدنویسی به‌صورت دستی مورد نیاز است. بنابراین برای توسعه برنامه‌هایی که نیاز به اجرای سناریوهای متعدد با کمترین میزان دخالت برنامه‌نویس دارند و نیز برنامه‌هایی که جنبه‌های گرافیکی و بصری در آن‌ها از اهمیت ویژه‌ای برخوردار است، ابزار AnyLogic مناسب به نظر می‌رسد. همچنین در مواردی که تیم توسعه از متخصصین دامنه‌های کسب‌وکار مثل مهندسين صنايع يا اقتصاد تشكيل شده باشد که از دانش برنامه‌نویسی عمومی مثل جاوا برخوردار نباشند، استفاده از ابزار NetLogo مفید است. زیرا این ابزار برای دامنه وسیعی از کاربردهای مختلف، کتابخانه‌های تخصصی ارائه داده است که کار برنامه‌نویسی را متخصصین دامنه



آسان می‌کند.

در آینده می‌توان ابزارهای مدل‌سازی و شبیه‌سازی پیش‌تری را با این شیوه یعنی تهیه پرسشنامه و در جامعه آماری بزرگ‌تری، به‌صورت کمی مورد ارزیابی قرار داد تا برای متخصصین کسب‌وکار و افرادی که نیاز به توسعه برنامه‌های کاربردی مبتنی بر عامل دارند، در انتخاب ابزار مناسب راه‌گشا باشد.

۷- پی‌نوشت‌ها

۱. Agent
۲. Non-linear interactions
۳. Agent-Based Modeling
۴. Autonomous
۵. Reactivity
۶. Pro-activeness
۷. Agent-communication language
۸. Multi-agent system
۹. Well-defined
۱۰. Quality in use
۱۱. Goal-Question-Metric (GQM)
۱۲. <https://anylogic.com/>
۱۳. <https://ccl.northwestern.edu/netlogo/>
۱۴. <https://repast.github.io/>
۱۵. <https://gama-platform.github.io/>
۱۶. <https://cs.gmu.edu/~eclab/projects/mason/>
۱۷. Schelling Segregation
۱۸. Shakshuki
۱۹. Jun
۲۰. Container
۲۱. Nikolai
۲۲. Madey
۲۳. License
۲۴. Allan
۲۵. Singh
۲۶. Nature of product
۲۷. Agent mobility
۲۸. Migration mechanism
۲۹. Application Programming Interface
۳۰. Kravari
۳۱. Bassiliades
۳۲. Pragmatics
۳۳. Abar
۳۴. Source code
۳۵. Geometric information system
۳۶. Effectiveness
۳۷. Efficiency
۳۸. Satisfaction
۳۹. Freedom from risk
۴۰. Context coverage
۴۱. Usefulness
۴۲. Trust
۴۳. Pleasure
۴۴. Comfort
۴۵. Product quality model
۴۶. Functional suitability
۴۷. Reliability
۴۸. Performance efficiency
۴۹. Usability
۵۰. Maintainability
۵۱. Security
۵۲. Compatibility
۵۳. Portability
۵۴. Appropriateness recognizability
۵۵. Learnability
۵۶. Operability
۵۷. User error protection
۵۸. User interface aesthetics



- | | |
|-------------------|---------------------------|
| ۵۹. Accessibility | ۶۳. Time Taken |
| ۶۰. Basili | ۶۴. Features |
| ۶۱. Simplicity | ۶۵. ۵-point Likert scales |
| ۶۲. Accuracy | |

۸- منابع

- [۱] A. Borshchev, *The big book of simulation modeling: multimethod modeling with AnyLogic 6*. AnyLogic North America, ۲۰۱۳.
- [۲] D. J. Barnes and D. Chu, "Agent Based Modeling," in *Guide to Simulation and Modeling for Biosciences*, Springer London, ۲۰۱۵, pp. ۱۵-۷۸.
- [۳] M. Wooldridgey and P. Ciancarini, "Agent-oriented software engineering: The state of the art," in *International workshop on agent-oriented software engineering*, ۲۰۰۰, pp. ۱-۲۸.
- [۴] E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems," *Proceedings of the national academy of sciences*, vol. ۹۹, no. suppl ۳, pp. ۷۲۸۰-۷۲۸۷, ۲۰۰۲.
- [۵] M. Niazi and A. Hussain, "Agent-based computing from multi-agent systems to agent-based models: a visual survey," *Scientometrics*, vol. ۸۹, no. ۲, pp. ۴۷۹-۴۹۹, ۲۰۱۱.
- [۶] E. Abolfathi, A. Toloie Eshlaghy, and M. reza Hamidizadeh, "An Operating Anatomy for Agent-Based Modeling Stand on the Categorization of Research Done In the Humanities: The Diffusion of Innovation in Iran," *Modern Research in Decision Making*, vol. ۳, no. ۲, pp. ۱-۲۵, ۲۰۱۸.
- [۷] M. Khani, abbas saghaei, and K. Heidarzadeh Hanzae, "Modeling New Product Launch Strategies within agent-based simulation," *Modern Research in Decision Making*, vol. ۴, no. ۴, pp. ۱۷۶-۱۹۶, ۲۰۱۹.
- [۸] F. jahanseir khararoudi, A. Azar, and T. Karimi, "Decision- making on dimensions of ambidexterity using an agent-based modeling approach," *Modern Research in Decision Making*, vol. ۵, no. ۴, pp. ۱-۱۸, ۲۰۲۰.
- [۹] D. Helbing, "Agent Based Modeling," in *Social Self-Organization*, D. Helbing,



Ed. Springer Berlin Heidelberg, ۲۰۱۲, pp. ۲۵-۷۰.

- [۱۰] C. Macal and M. North, "Introductory Tutorial: Agent-Based Modeling And Simulation," in *Proceedings of the 2014 Winter Simulation Conference*, ۲۰۱۴, pp. ۳۶۵۰-۳۶۶۱.
- [۱۱] C. W. Loh, S. Nissom, W. S. Cheah, and N. Jali, "Agent oriented methodology for malaria transmission modelling and simulation," in *Proceedings of the 6th International Conference of Computing & Informatics*, ۲۰۱۷, pp. ۱۶۰-۱۶۶.
- [۱۲] E. Shakshuki and Y. Jun, "Multi-agent development toolkits: An evaluation," in *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, ۲۰۰۴, pp. ۲۰۹-۲۱۸.
- [۱۳] C. Nikolai and G. Madey, "Tools of the trade: A survey of various agent based modeling platforms," *Journal of Artificial Societies and Social Simulation*, vol. ۱۲, no. ۲, p. ۲, ۲۰۰۹.
- [۱۴] A. Singh, D. Juneja, and A. K. Sharma, "Agent development toolkits," *International Journal of Advancements in Technology*, vol. ۲, no. ۱, pp. ۱۵۸-۱۶۴, ۲۰۱۱.
- [۱۵] K. Kravari and N. Bassiliades, "A Survey of Agent Platforms," *Journal of Artificial Societies and Social Simulation*, vol. ۱۸, no. ۱, p. ۱۱, ۲۰۱۵.
- [۱۶] S. Abar, G. K. Theodoropoulos, P. Lemarinier, and G. M. P. O'Hare, "Agent Based Modelling and Simulation tools: A review of the state-of-art software," *Computer Science Review*, vol. ۲۴, pp. ۱۳-۳۳, ۲۰۱۷.
- [۱۷] I. S. O. Iso, "Iec۲۵۰۱۰: ۲۰۱۱ systems and software engineering--systems and software quality requirements and evaluation (square)--system and software quality models," *International Organization for Standardization*, vol. ۳۴, p. ۲۹۱۰, ۲۰۱۱.
- [۱۸] V. R. B. G. Caldiera and H. D. Rombach, "The goal question metric approach," *Encyclopedia of software engineering*, pp. ۵۲۸-۵۳۲, ۱۹۹۴.
- [۱۹] R. Allan, "Survey of Agent Based Modelling and Simulation Tools," ۲۰۱۰.



- [۲۰] L. Braubach, A. Pokahr, and W. Lamersdorf, “A universal criteria catalog for evaluation of heterogeneous agent development artifacts,” *From Agent Theory to Agent Implementation (AT2AI-6)*, pp. ۱۹–۲۸, ۲۰۰۸.
- [۲۱] Wikipedia, “AnyLogic.” [Online]. Available: <https://en.wikipedia.org/wiki/AnyLogic>. [Accessed: ۰۹-Mar-۲۰۲۰].
- [۲۲] “NetLogo.” [Online]. Available: <http://ccl.northwestern.edu/netlogo/>. [Accessed: ۱۰-Mar-۲۰۲۰].
- [۲۳] “The Repast Suite Documentation.” [Online]. Available: <https://repast.github.io/index.html>. [Accessed: ۱۰-Mar-۲۰۲۰].
- [۲۴] “GAMA-Platform.” [Online]. Available: <https://gama-platform.github.io/wiki/Home>. [Accessed: ۰۹-Mar-۲۰۲۰].
- [۲۵] “MASON Multiagent Simulation Toolkit.” [Online]. Available: <https://cs.gmu.edu/~eclab/projects/mason/>. [Accessed: ۱۰-Mar-۲۰۲۰].
- [۲۶] A. Hussain, N. L. Hashim, N. Nordin, and H. M. Tahir, “A metric-based evaluation model for applications on mobile phones,” *Journal of Information and Communication Technology*, vol. ۱۲, pp. ۵۵–۷۱, ۲۰۱۳.
- [۲۷] F. McCown, “Schelling’s Model of Segregation.” [Online]. Available: <http://nifty.stanford.edu/۲۰۱۴/mccown-schelling-model-segregation/>. [Accessed: ۰۹-Mar-۲۰۲۰].