

زمان‌بندی قواعد در پایگاه داده پویا به کمک اتوماتای یادگیر

عباس رسول‌زادگان

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

دانشگاه صنعتی امیرکبیر

rasoolzadegan@aut.ac.ir

محمدرضا میبیدی

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

دانشگاه صنعتی امیرکبیر

mmeybodi@aut.ac.ir

چکیده: سیستم مدیریت پایگاه داده پویا^۱ یکی از موضوعات تحقیقاتی مهمی است که در سال‌های اخیر از جهات مختلف مورد بررسی و مطالعه قرار گرفته است. یکی از مؤلفه‌های مهم سیستم پایگاه داده پویا که تأثیر بسزایی در کارایی آن دارد، زمان‌بند قواعد است که موظف به زمان‌بندی قواعد فعال و آماده‌ی اجرا در چرخه‌ی پردازش قواعد پویا می‌باشد. پیش از این روش‌هایی را که تاکنون برای زمان‌بندی قواعد ارائه شده است، در یک محیط آزمایشگاهی مبتنی بر معماری سه بخشی^۲ براساس پنج پارامتر ارزیابی با یکدیگر مقایسه نموده و پس از تحلیل نتایج آزمایشات و بررسی نقاط قوت و ضعف هریک، روش کاراتر را معرفی کرده‌ایم. نحوه‌ی ارزیابی روش‌های زمان‌بندی قواعد براساس پارامترهای پیش گفته به صورت فرمال تعریف شده است در این مقاله، با هدف بهبود زمان‌بندی قواعد بر مبنای بهبود فرایند تخمین احتمال اجرای آنها به کمک اتوماتای یادگیر^۳ روش زمان‌بندی جدیدی ارائه می‌نماییم. سپس آنرا با کاراترین روش ارائه شده‌ی موجود در چارچوب مذکور مقایسه و ارزیابی می‌نماییم. همچنین به منظور پوشش کلیه‌ی مفاهیم سیستم مدیریت پایگاه داده پویا و شبیه‌سازی کامل تر و دقیق تر رفتار آن، ماژول‌هایی را نظیر کامپایلر ارزیابی شروط و اجرای دستورات عمل‌های قواعد پویا به محیط آزمایشگاهی پیش گفته افزوده‌ایم. نتایج آزمایشات حاکی از بهبود فرایند زمان‌بندی قواعد توسط روش جدید می‌باشد.

واژه‌های کلیدی: تخمین احتمال اجرای قواعد، سیستم مدیریت پایگاه داده پویا، زمان‌بندی قواعد، یادگیری، اتوماتای یادگیر.

مقدمه

پایگاه‌های داده متداول دارای ماهیت ایستا هستند، یعنی اعمال پرس و جو، بهنگام سازی، درج، حذف، گزارش‌گیری و غیره فقط زمانی که توسط کاربر درخواست شوند، انجام می‌پذیرند و سیستم مدیریت پایگاه داده ابتکار عملی در هنگام رخ دادن شرایط خاص در سیستم ندارد. بسیاری از برنامه‌های کاربردی مانند برنامه‌های انبارداری، اتوماسیون کارخانه‌ها و سیستم‌های دارای محاسبات پیچیده مالی (مانند بازار سهام) نیاز به نظارت خودکار دارند تا در صورت وقوع رویداد خاصی، واکنش مقتضی انجام شود. برای این منظور باید سیستم پایگاه داده‌ای

طراحی شود که در آن امکان تعریف رویدادهای مورد نظر و واکنش‌های متناظر آنها گنجانده شود، به چنین سیستمی، سیستم پایگاه داده پویا گویند [۱]، [۳]، [۴]، [۵].

رفتار واکنشی سیستم پایگاه داده پویا در قالب مجموعه‌ای از قواعد پویا سازماندهی می‌شود. در حالت کلی، این قواعد شامل سه بخش به نام- های رویداد، شرط و عمل هستند. به همین دلیل به آنها قواعد رویداد- شرط-عمل یا به اختصار ECA^۴ گفته می‌شود [۱]، [۲]. در سیستم پایگاه داده پویا، در ابتدای امر، برنامه کاربردی در حال اجراست. تا زمانی که هیچ رویداد از پیش تعریف شده‌ای در سیستم رخ نداده است اجرای برنامه کاربردی ادامه پیدا می‌کند. به محض اینکه رویدادی در سیستم رخ دهد، مجموعه‌ای از قواعد که با این رویداد، مرتبط هستند فعال شده و وارد لیست قواعد فعال می‌شوند. در ادامه با توجه به روش زمان‌بندی مورد استفاده، یکی از قواعد فعال انتخاب و بخش شرط آن ارزیابی می‌شود و در صورت درست بودن، به لیست قواعد آماده‌ی اجرا اضافه می‌شود. از بین مجموعه قواعد آماده اجرا نیز با توجه به روش زمان‌بندی مورد استفاده، یکی انتخاب و دستورات بخش عمل آن اجرا می‌شود. اگر اجرای این دستورات موجب رخ دادن رویدادهای دیگری در سیستم شود، قواعد مرتبط (قواعد فرزند) نیز به لیست قواعد فعال اضافه می‌شوند. ارزیابی و اجرای قواعد تا زمانی که لیست قواعد فعال و آماده- ی اجرا خالی شوند، ادامه پیدا می‌کند. پس از آن کنترل اجرا به برنامه کاربردی باز می‌گردد. به مجموعه عملیات فوق، چرخه‌ی پردازش قواعد گویند [۱]، [۲]، [۴]، [۵].

دو فاکتور بسیار مؤثر در نحوه‌ی اجرای چرخه‌ی پردازش قواعد، پیوستگی رویداد-شرط و پیوستگی شرط-عمل قواعد است. این دو پیوستگی دارای سه حالت مختلف‌اند: (۱) فوری، (۲) تعویقی و (۳) مستقل. اگر پیوستگی رویداد-شرط یک قاعده از نوع فوری باشد، شرط آن قاعده بلافاصله پس از رخ دادن رویداد متناظرش و فعال شدن قاعده ارزیابی می‌شود. به همین ترتیب اگر پیوستگی شرط-عمل یک قاعده از نوع فوری باشد، پس از فعال شدن آن قاعده و ارزیابی شرط مربوطه، در صورت مثبت بودن نتیجه‌ی ارزیابی، دستورات بخش عمل آن قاعده فوراً اجرا می‌شوند. اگر پیوستگی رویداد-شرط (شرط-عمل) یک قاعده از نوع تعویقی باشد، ارزیابی شرط (اجرای عمل) آن قاعده تا زمان اتمام

روش اتفاقی

در این روش هرگاه که در چرخه‌ی پردازش قواعد نیاز به انتخاب قاعده جدیدی برای اجرا باشد، از بین قواعد آماده‌ی اجرا به صورت اتفاقی یکی برای اجرا انتخاب می‌شود [۱]، [۲]، [۴].

روش اولویت ایستا^۶

در این روش به هر یک از قواعد یک عدد صحیح به عنوان اولویت نسبت داده می‌شود. سپس در هنگام انتخاب قواعد برای اجرا، قاعده‌ای که در بین قواعد آماده‌ی اجرا کوچکترین اولویت را دارد، انتخاب می‌شود [۴]، [۶].

روش مبتنی بر برچسب زمانی

در این روش به هر قاعده که در حالت فعال قرار بگیرد یک برچسب زمانی که مشخص کننده زمان فعال شدن آن قاعده است، تخصیص می‌یابد. در هنگام انتخاب، قاعده‌ای که دارای کوچکترین برچسب زمانی باشد، برای اجرا انتخاب خواهد شد [۱]، [۲]، [۴].

روش مبتنی بر الگوریتم نزدیکترین ضرب‌العجل^۷

این روش یکی از کامل‌ترین روش‌هایی است که تاکنون برای زمان‌بندی قواعد در سیستم‌های پایگاه داده پویای بلادرنگ ارایه شده است و به اختصار EDF نامیده می‌شود. در این روش، هنگام انتخاب، قاعده‌ای که دارای نزدیکترین ضرب‌العجل باشد، برای اجرا انتخاب می‌شود. براساس نحوه‌ی محاسبه اولویت قواعد که مبتنی بر ضرب‌العجل قواعد است، سه نسخه‌ی مختلف از روش EDF طراحی شده است که عبارتند از: EDF_{PD} ، EDF_{SL} و EDF_{DIV} . اولی در حقیقت همان الگوریتم EDF صرف است. در دومی یک تغییر کوچک در الگوریتم EDF داده شده است به این ترتیب که اولویت قاعده‌ی پدر وقتی یک قاعده‌ی فرزند تولید می‌شود، تغییر می‌کند. در سومی اولویت قاعده پدر در هر لحظه براساس سه پارامتر تخمین زمان اجرای قاعده، تعداد قواعد فرزند فعلی و تخمین تعداد قواعد فرزند که در آینده تولید خواهد شد، محاسبه می‌شود [۱]، [۲].

روش مبتنی بر الگوریتم ابتدا کوچکترین کار

این روش که مبتنی بر الگوریتم ابتدا کوچکترین کار (SJF) است، به اختصار $Ex-SJF$ نامیده می‌شود. با توجه به اینکه روش جدیدی که در بخش سوم ارائه خواهد شد، بهبودی بر روش $Ex-SJF$ می‌باشد، در این قسمت نحوه‌ی عملکرد روش $Ex-SJF$ را نسبت به سایر روش‌های موجود با تفصیل بیشتری شرح می‌دهیم. در روش $Ex-SJF$ هرگاه که نیاز به انتخاب یکی از تراکنش‌های فعال در سیستم باشد، تراکنشی که دارای کمترین زمان اجراست، انتخاب می‌شود. تفاوت دو روش SJF و $Ex-SJF$ در نحوه‌ی محاسبه زمان اجرای تراکنش‌ها است. در روش $Ex-SJF$ به دلیل تولید پویای تراکنش‌ها، نمی‌توان فاکتور انتخاب آنها را فقط زمان اجرای خود تراکنش‌ها قرار داد. زیرا هر تراکنشی (پدر) که در

تراکنش جاری به تأخیر می‌افتد. در برخی موارد ارزیابی شرط (اجرای عمل) تا رخ دادن رویداد خاصی به تأخیر خواهد افتاد. در حالت مستقل، ارزیابی شرط (اجرای عمل) قاعده در قالب یک تراکنش مستقل و پس از اتمام تراکنش جاری انجام می‌شود [۱].

روش مورد استفاده برای زمان‌بندی قواعد به طور مستقیم در فاکتورهایی مانند زمان پاسخگویی سیستم به تراکنش‌ها، زمان بازگشت تراکنش‌ها، توان عملیاتی سیستم و بطور کلی در کارایی سیستم پایگاه داده پویا، بسیار مؤثر است. بهبود زمان‌بندی قواعد در سیستم مدیریت پایگاه داده پویا یکی از موضوعات تحقیقاتی مهم است [۴]، [۵]. تاکنون چندین روش برای زمان‌بندی قواعد ارائه شده است [۱]، [۳]، [۵]. پیش از این روش‌های زمان‌بندی موجود را در یک محیط آزمایشگاهی مبتنی بر معماری سه بخشی (شبیه‌ساز پایگاه داده پویا^۸) براساس تعدادی پارامتر ارزیابی با یکدیگر مقایسه نموده و روش کارتر ($Ex-SJF_{PRO}$ -V.2.8) را معرفی کرده‌ایم [۱]. در این مقاله با اضافه کردن اتوماتای یادگیر به روش $Ex-SJF_{PRO}$ -V.2.8، درصد بهبود فرایند تخمین احتمال اجرای قواعد و متعاقباً ارائه یک روش زمان‌بندی کارتر از بُعد پارامترهای ارزیابی تعریف شده، برآمده‌ایم. همچنین با افزودن ویژگی‌هایی به شبیه‌ساز پایگاه داده پویا، سعی کرده‌ایم تا حد امکان به شبیه‌سازی و ارزیابی دقیق‌تر رفتار پایگاه داده پویا و عملکرد روش‌های زمان‌بندی بپردازیم.

این مقاله در پنج بخش تنظیم شده است، در بخش دوم به معرفی روش‌های موجود برای زمان‌بندی قواعد در پایگاه داده پویا می‌پردازیم. در بخش سوم با هدف بهبود زمان‌بندی قواعد بر مبنای بهبود فرایند تخمین احتمال اجرای قواعد به کمک اتوماتای یادگیر، روشی جدید معرفی می‌نماییم. در بخش چهارم به بررسی عملکرد روش پیشنهادی جدید در مقایسه با روش‌های زمان‌بندی موجود بر اساس تعدادی پارامتر ارزیابی بر مبنای نتایج حاصل از انجام آزمایشات متعدد و متنوع در یک محیط آزمایشگاهی مبتنی بر معماری سه بخشی می‌پردازیم. سرانجام در بخش پنجم مطالب مطرح شده در مقاله را جمع بندی می‌نماییم.

معرفی روش‌های مختلف زمان‌بندی قواعد

در سیستم مدیریت پایگاه داده پویا به فرایند تخصیص اولویت به قواعد فعال جهت ارزیابی بخش شرط و نیز به قواعد آماده‌ی اجرا جهت اجرای دستورات بخش عمل زمان‌بندی قواعد گویند. همانطور که پیش از این نیز بیان شد، قواعدی آماده‌ی اجرا هستند که اولاً در اثر وقوع رویداد متناظرشان در سیستم، فعال شده باشند و ثانیاً بخش شرطشان در لحظه‌ی ارزیابی درست باشد. در این قسمت روش‌های مختلف زمان‌بندی قواعد را به طور اجمالی معرفی می‌نماییم [۱]، [۲]، [۴]. مطالبی که در ادامه در مورد زمان‌بندی قواعد آماده‌ی اجرا بیان می‌شود، در مورد زمان‌بندی قواعد فعال نیز صادق است و بالعکس.

روش زمان‌بندی $EX-SJF_{PRO}$

در این روش احتمال درستی هر یک از عبارات شرطی موجود در بخش شرط یک قاعده به صورت یکسان $\frac{1}{2}$ در نظر گرفته شده است. با توجه به مطالب فوق، زمان اجرای هر قاعده در این روش نیز از (۱) قابل محاسبه است [۲]، [۳].

روش زمان‌بندی $EX-SJF_{PRO-V.1.8}$

در این روش که پیش از این توسط خودمان ارائه شده است، یک پیمانه تخمین به روش $EX-SJF_{PRO}$ اضافه کرده‌ایم [۱]، بدین ترتیب که در ابتدا زمان اجرای هر قاعده، همانند روش $EX-SJF_{PRO}$ محاسبه می‌شود. سپس با شروع کار سیستم هر بار که بخش شرط یکی از قواعد فعال مورد ارزیابی قرار گیرد، احتمال درستی هر یک از عبارات منطقی (LS_i) موجود در بخش شرط آن قاعده یعنی $P(R, LS_i)$ طبق (۲) محاسبه و ذخیره می‌شود.

$$(2) \quad P(R, LS_i) = \frac{\text{تعداد دفعاتی که نتیجه‌ی ارزیابی } LS_i \text{ درست بوده است}}{\text{تعداد دفعاتی که بخش شرط } R \text{ تاکنون ارزیابی شده است}}$$

این کار برای هر عبارت منطقی در بخش شرط یک قاعده تا زمانی تکرار می‌شود که آهنگ تغییرات احتمال درستی آن عبارت منطقی به حد مطلوب برسد (در حالت کلی ϵ)، در این هنگام مقدار جدید، جایگزین مقدار پیش فرض اولیه ($\frac{1}{2}$) می‌شود. زمانیکه احتمال درستی همه عبارات منطقی یک شرط بهنگام شد، احتمال درستی آن شرط (احتمال اجرای قاعده‌ی صاحب آن شرط) نیز بهنگام می‌شود. و در نهایت زمان اجرای قاعده R زمانی طبق (۱) بهنگام می‌شود که زمان اجرای همه فرزندان بهنگام شده باشند. بدین ترتیب بعد از مدتی که از اجرای سیستم بگذرد (که البته این مدت نسبت به کل زمانی که قرار است سیستم کار کند ناچیز است)، زمان اجرای قواعد با دقت بیشتری نسبت به زمان شروع اجرای سیستم، محاسبه می‌شود [۱].

روش زمان‌بندی $EX-SJF_{PRO-V.2.8}$

این روش نیز مانند روش $EX-SJF_{PRO-V.1.8}$ بر مبنای تخمین احتمال اجرای قواعد طراحی و پیاده‌سازی شده است [۱]، [۳]. تفاوت این دو روش در نحوه‌ی تخمین احتمال اجرای قواعد می‌باشد. با توجه به اینکه پیش از زمان اجرا هم بخش شرط کلیه قواعد مشخص است و هم دامنه مقادیر مجاز برای کلیه متغیرهای شرطی موجود در بخش شرط قواعد و با فرض یکنواختی توزیع احتمال متغیرهای شرطی و مستقل بودن عبارات منطقی از هم، می‌توان احتمال درستی هر یک از عبارات منطقی را پیش از زمان اجرا، به صورت ریاضی محاسبه نمود. پس از محاسبه احتمال اجرای بخش عمل قواعد، زمان اجرای قواعد را طبق (۱) محاسبه می‌نماییم. اما به منظور حذف فرض یکنواختی توزیع احتمال متغیرهای شرطی جهت نزدیکی به شرایط سیستم‌های واقعی به روش زیر عمل می‌نماییم.

پس از شروع کار سیستم، در بازه‌های زمانی مشخص (δt_i)، برای هر یک

یک سیستم پایگاه داده پویا اجرا می‌شود به صورت بالقوه مجموعه‌ای از قواعد پویا (فرزندان) را در سیستم فعال می‌کند. ترائکشن‌های حاصل از این قواعد نیز در صورت اجرا، مجموعه‌ای دیگر از قواعد را به صورت بالقوه فعال خواهند کرد و این عمل تا چند سطح ممکن است انجام پذیرد. لذا زمان واقعی اجرای یک ترائکشن، علاوه بر زمان اجرای خودش، شامل زمان اجرای ترائکشن‌های فوری و تعویقی که به صورت بالقوه در اثر اجرای آن ترائکشن تولید و اجرا خواهند شد، نیز می‌باشد. محاسبه زمان واقعی اجرای قواعد در زمان اجرا امکان‌پذیر است. اما انجام آن به دلیل در پی داشتن سربار زمانی زیاد، منجر به یک روش زمان‌بندی ناکارا خواهد شد. لذا تمام نسخه‌های روش $EX-SJF$ زمان اجرای قواعد را پیش از اجرا محاسبه می‌نمایند. اما محاسبه زمان واقعی اجرای یک قاعده پیش از زمان اجرا مستلزم تخمین احتمال اجرای قواعدی است که در دل آن قاعده در زمان اجرا به صورت پویا فعال می‌شوند. هرچه این تخمین دقیق‌تر باشد، مقدار محاسبه شده برای زمان اجرای قاعده به واقعیت نزدیکتر است [۱]، [۲].

همانطور که پیش از این نیز بیان شد، اجرای یک قاعده‌ی فعال منوط به درستی بخش شرط آن در لحظه‌ی ارزیابی است. به عبارت دیگر احتمال اجرای یک قاعده با احتمال درستی بخش شرط آن یکسان است. بخش شرط موجود در بدنه قواعد از عبارات شرطی، عبارات پرس و جوی پایگاه داده، فراخوانی رویه‌ها و توابع و ترکیب منطقی آنها تشکیل شده است. اگر احتمال درستی این عبارات شرطی به صورت قطعی وجود داشته باشد، می‌توان احتمال درست بودن شرط قواعد را به دقت محاسبه نمود. ولی احتمال درستی یک عبارت شرطی غالباً قبل از اجرای آن به صورت قطعی وجود ندارد، هر چهار نسخه‌ی $EX-SJF$ یعنی $EX-SJF_{PRO}$ ، $EX-SJF_{PRO-V.1.8}$ ، $EX-SJF_{PRO-V.2.8}$ و $EX-SJF_{EXA}$ به منظور پیش‌بینی قواعد فرزند یک قاعده به تشکیل درخت اجرای قواعد می‌پردازند [۱]. اما وجه تمایز این نسخه‌ها در نحوه‌ی تخمین احتمال اجرای قواعد فرزند می‌باشد.

روش زمان‌بندی $EX-SJF_{EXA}$

در این روش احتمال درستی بخش شرط قواعد (احتمال اجرای قواعد)، ۱ در نظر گرفته می‌شود [۲]. در نتیجه زمان اجرای قواعد با فرض $P(R)=1$ طبق رابطه‌ای (۱) که در ادامه می‌آید، محاسبه می‌شود:

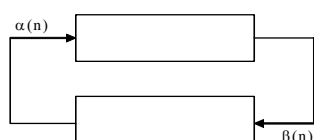
$$\begin{aligned} X^{imm}(R_i) & \quad \text{زمان اجرای } i \text{ امین ترائکشن فوری حاصل از } R \\ X^{def}(R_j) & \quad \text{زمان اجرای } j \text{ امین ترائکشن تعویقی حاصل از } R \\ L(R) & \quad \text{تعداد دستورات بخش عمل } R \\ n^{def}(R) & \quad \text{تعداد قواعد تعویقی تولیدی توسط قاعده‌ی } R \\ n^{imm}(R) & \quad \text{تعداد قواعد فوری تولیدی توسط قاعده‌ی } R \\ X(R) & \quad \text{زمان اجرای بخش عمل قاعده‌ی } R \\ P(R_i) & \quad \text{احتمال درستی شرط } i \text{ امین قاعده فوری حاصل از } R \\ P(R_j) & \quad \text{احتمال درستی شرط } j \text{ امین قاعده تعویقی حاصل از } R \end{aligned}$$

$$X(R) = L(R) + \sum_{i=1}^{n^{imm}(R)} P(R_i) * X^{imm}(R_i) + \sum_{j=1}^{n^{def}(R)} P(R_j) * X^{def}(R_j)$$

تخمین احتمال درستی بخش شرط قواعد، در حین اجرای سیستم می-پردازد. اما در ادامه پیش از تشریح جایگاه و عملکرد اتوماتای یادگیر در زمانبندی قواعد فعال، به تعریف اتوماتای یادگیر و مفاهیم مرتبط می-پردازیم.

اتوماتای یادگیر

اتوماتای یادگیر [۸]، [۹]، ماشینی است که می‌تواند تعدادی متناهی عمل را انجام دهد. هر عمل انتخاب شده توسط یک محیط احتمالی ارزیابی می‌شود و نتیجه ارزیابی در قالب پاسخی مثبت یا منفی به اتوماتا داده می‌شود و اتوماتا از این پاسخ در انتخاب عمل بعدی تاثیر می‌گیرد. هدف نهایی این است که اتوماتا یاد بگیرد تا از بین اعمال خود بهترین عمل را انتخاب کند. بهترین عمل، عملی است که احتمال دریافت پاداش از محیط را به حداکثر برساند. کارکرد اتوماتای یادگیر در تعامل با محیط، در شکل (۱) مشاهده می‌شود.



شکل (۱): ارتباط بین محیط و اتوماتای یادگیر

محیط را می‌توان توسط سه‌تایی $E \equiv \{v, \beta, c\}$ نشان داد که در آن $v \equiv \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه ورودی‌ها، $\beta \equiv \{\beta_1, \beta_2, \dots, \beta_m\}$ مجموعه خروجی‌ها و $c \equiv \{c_1, c_2, \dots, c_r\}$ مجموعه احتمال‌های جریمه می‌باشد. هرگاه β مجموعه دو عضوی باشد، محیط از نوع P می‌باشد. در چنین محیطی $\beta_1 = 1$ به عنوان جریمه و $\beta_2 = 0$ به عنوان پاداش در نظر گرفته می‌شود. در محیط از نوع Q ، $\beta(n)$ می‌تواند به طور گسسته یک مقدار از مقادیر محدود در فاصله $[0, 1]$ و در محیط از نوع S ، $\beta(n)$ متغیر تصادفی در فاصله $[0, 1]$ است. c_i احتمال این است که عمل α_i نتیجه نامطلوب داشته باشد. در محیط ایستا مقادیر c_i بدون تغییر می‌مانند، حال آنکه در محیط غیرایستا این مقادیر در طی زمان تغییر می‌کنند.

اتوماتای یادگیر با ساختار ثابت توسط پنج‌تایی $\{v, \beta, F, G, \phi\}$ نشان داده می‌شود که در آن $v \equiv \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه عمل‌های اتوماتا، $\beta \equiv \{\beta_1, \beta_2, \dots, \beta_m\}$ مجموعه ورودی‌های اتوماتا، $\phi \equiv \{\phi_1, \phi_2, \dots, \phi_s\}$ مجموعه وضعیت‌های داخلی اتوماتا، $F: \phi \times \beta \rightarrow \phi$ تابع تولید وضعیت جدید اتوماتا و $G: \phi \rightarrow v$ تابع خروجی می‌باشد که وضعیت کنونی اتوماتا را به خروجی بعدی می‌نگارد. اتوماتای یادگیر با ساختار متغیر را می‌توان توسط چهارتایی $\{v, \beta, p, T\}$ نشان داد که $v \equiv \{\alpha_1, \dots, \alpha_r\}$ مجموعه عمل‌های اتوماتا، $\beta \equiv \{\beta_1, \dots, \beta_m\}$ مجموعه ورودی‌های اتوماتا، $p = \{p_1, \dots, p_r\}$ بردار احتمال انتخاب هریک از عمل‌ها و $p(n+1) = T[v(n), \beta(n), p(n)]$ الگوریتم یادگیری می‌باشد. بردار احتمال انتخاب هریک از عمل‌ها در لحظه n بصورت زیر است:

از متغیرهای شرطی بخش شرط قواعد، تمامی مقادیری که اخذ شده، به همراه مدت زمان اخذ آنها ثبت می‌گردد. در پایان هر α بر اساس اطلاعات بدست آمده توزیع احتمال متغیرها اصلاح و در نتیجه احتمال و زمان اجرای قواعد مجدداً محاسبه و بهنگام می‌شوند. بدین ترتیب که مقادیر فعلی احتمال وقوع مقادیر دامنه‌ی متغیرها را براساس مقادیری که در α گذشته توسط آن متغیرها اخذ شده و همچنین با توجه به مدت زمان اخذ هر مقدار توسط هر متغیر، بهنگام و سپس زمان اجرای قواعد را براساس احتمالات بهنگام شده، مجدداً، طبق (۱) محاسبه می-نماییم. محاسبات و بهنگام سازی‌ها را در زمان بیکاری سیستم انجام می‌دهیم. به عبارت دیگر اندازه‌ی α را طوری تعیین می‌کنیم که پایان آن با زمان بیکاری سیستم مصادف شود. بدیهی است هرچه از شروع کار سیستم بیشتر بگذرد، توزیع‌های احتمالی که برای متغیرها بدست می‌آیند به واقعیت نزدیکتر شده و در نتیجه زمان اجرای قواعد دقیق‌تر محاسبه می‌شوند. اصلاح توزیع‌های احتمال متغیرها را تا زمانی تکرار می‌کنیم، که آهنگ تغییرات زمان اجرای قواعد به حد مطلوب برسد (۶). بدیهی است که تعیین اندازه‌ی ϵ به میزان اهمیت دقت زمان اجرای قواعد در سیستم بستگی دارد.

معرفی روش جدید زمانبندی $^{8}Ex-SJF_{ESTLA}$

این روش نیز مانند روش‌های $Ex-SJF_{PRO-V.1.8}$ و $Ex-SJF_{PRO-V.2.8}$ بر مبنای تخمین احتمال اجرای قواعد طراحی شده است. عبارت دیگر در روش جدید با اضافه کردن یک اتوماتای یادگیر به کاراترین روش موجود [۱]، $Ex-SJF_{PRO-V.2.8}$ ، درصدد افزایش کارایی فرایند زمانبندی قواعد از بُعد پارامترهای ارزیابی تعریف شده، برآمده‌ایم. همانطور که پیش از این نیز بیان شد، فرایند پردازش قواعد پویا، مشتمل بر دو زمانبندی است که عبارتند از: ۱- زمانبندی قواعد فعال جهت انتخاب یکی از قواعد فعال به منظور ارزیابی بخش شرط آنها و ۲- زمانبندی قواعد آماده‌ی اجرا جهت انتخاب یکی از قواعد آماده‌ی اجرا به منظور اجرای دستورات بخش عمل آنها. روش‌هایی که تاکنون ارائه شده‌اند، همگی دارای مکانیزم یکسانی برای هر دو زمانبندی مذکور می‌باشند. بعنوان مثال در روش $Ex-SJF_{PRO-V.2.8}$ ، هر دو زمانبندی قواعد فعال و آماده‌ی اجرا بر مبنای زمان اجرای قواعد عمل می‌کنند. بدین ترتیب که ملاک انتخاب یک قاعده فعال یا آماده‌ی اجرا در هر لحظه، زمان اجرای آن قاعده است و لذا قاعده‌ای انتخاب می‌شود که کمترین زمان اجرا را داشته باشد.

در روش پیشنهادی جدید مکانیزم زمانبندی قواعد فعال و قواعد آماده-ی اجرا با هم متفاوتند. بدین ترتیب که برای زمانبندی قواعد آماده‌ی اجرا همانند روش $Ex-SJF_{PRO-V.2.8}$ عمل می‌نماید. اما برای زمانبندی قواعد فعال، دارای مکانیزم زمانبندی متفاوتی بر مبنای اتوماتای یادگیر است. عبارت دیگر در روش جدید، ملاک زمانبندی قواعد فعال، بر خلاف روش $Ex-SJF_{PRO-V.2.8}$ ، زمان اجرای قواعد نیست. بلکه احتمال درستی بخش شرط آنهاست و اتوماتای یادگیر نیز در این راستا به

بطوریکه

$$P(n) \equiv \{p_1(n), p_2(n), \dots, p_r(n)\}$$

$$\sum_{i=1}^r p_i(n) = 1, \quad \forall n, \quad p_i(n) = \text{Prob}[\alpha(n) = \alpha_i] \quad ()$$

در آغاز فعالیت اتوماتا، احتمال اقدامهای آن با هم برابر و مساوی $\frac{1}{r}$ می‌باشند (که r تعداد اقدامهای اتوماتا می‌باشد). اگر اتوماتای یادگیر در تکرار n ام، یک اقدام خود مانند α_i را انتخاب کند و یک پاسخ مطلوب از محیط دریافت نماید، $p_i(n)$ (احتمال اقدام α_i) افزایش و احتمال سایر اقدامها کاهش می‌یابد. بالعکس در صورت نامطلوب بودن پاسخ دریافتی از محیط، احتمال اقدام α_i کاهش و احتمال سایر اقدامهای اتوماتا افزایش می‌یابد. در هر حال، تغییرات به گونه ای صورت می‌گیرد تا حاصل جمع $p_i(n)$ ها همواره ثابت و مساوی یک باقی بماند. تغییر احتمال اقدامها بصورت زیر می‌باشد. اگر a پارامتر پاداش و b پارامتر جریمه می‌باشند. آنگاه شکل عمومی الگوریتم یادگیری تقویتی خطی (در اتوماتایی با چند اقدام) بصورت زیر است. اگر در گام n ام اقدام α_i انتخاب شده باشد، سپس در گام $n+1$ ام خواهیم داشت:

الف- پاسخ مطلوب از محیط

$$\begin{aligned} p_i(n+1) &= p_i(n) + a[1 - p_i(n)] \\ p_j(n+1) &= (1-a)p_j(n) \quad \forall j, j \neq i \end{aligned} \quad ()$$

ب- پاسخ نامطلوب از محیط

$$\begin{aligned} p_i(n+1) &= (1-b)p_i(n) \\ p_j(n+1) &= \frac{b}{r-1} + (1-b)p_j(n) \quad \forall j, j \neq i \end{aligned} \quad ()$$

با توجه به مقادیر a و b در روابط فوق، سه حالت را می‌توان در نظر گرفت. اگر مقادیر a و b برابر باشند، اتوماتای یادگیر L_{RP} نامیده می‌شود. زمانی که b مساوی با صفر باشد اتوماتای یادگیر L_{RI} نامیده می‌شود. اگر $b < a$ باشد، اتوماتای یادگیر L_{REP} نامیده می‌شود.

نحوه‌ی بکارگیری اتوماتای یادگیر در فرایند تخمین

احتمال اجرای قواعد

در روش جدید به کمک یک اتوماتای یادگیر با ساختار متغیر و الگوریتم یادگیری L_{RI} ، به زمانبندی قواعد فعال بر مبنای احتمال درستی شرط آنها می‌پردازیم. وظیفه این اتوماتای یادگیر این است که با توجه به پاسخ‌های دریافتی از محیط، استراتژی زمانبند قواعد را برای انتخاب اقدام مناسب به‌هنگام نماید تا نهایتاً به افزایش کارایی فرایند زمانبندی قواعد پویا از بُعد پارامترهای ارزیابی تعریف شده، منجر گردد. جایگاه اتوماتای یادگیر مذکور در معماری سیستم پایگاه داده پویا در شکل (۳) نشان داده شده است. محیطی که اتوماتای یادگیر مذکور در آن واقع است، از نوع مدل P است، یعنی پاسخی که از طرف محیط به اقدام LA داده می‌شود، دودویی است (در صورتیکه شرط قاعده‌ی فعال انتخاب شده، درست باشد، پاسخ محیط ۱ و در صورتیکه نادرست باشد، پاسخ

محیط ۰ خواهد بود). اقدام اتوماتای مذکور در هر لحظه، انتخاب یکی از قواعد فعال برای ارزیابی بخش شرط آن است. با توجه به متغیر بودن تعداد قواعد فعال در هر لحظه، تعداد اقدامات اتوماتا متغیر خواهد بود [۱۰]. لذا لازم است در هر بار تغییر تعداد قواعد فعال، پردازش‌هایی در رابطه با احتمال انتخاب اقدامهای اتوماتا صورت پذیرد که در ادامه به شرح آن خواهیم پرداخت. اگر تعداد قواعد، n باشد، مجموعه اقدامهای اتوماتا در حالت کلی، $v = \{\alpha_1, \dots, \alpha_n\}$ خواهد بود. یعنی برای هر قاعده، یک اقدام تعریف می‌شود. لذا با توجه به مطالب پیش گفته، بدیهی است که در هر لحظه مجموعه اقدامهای اتوماتا زیر مجموعه‌ای از مجموعه‌ی فوق خواهد بود. V_i نشان‌دهنده i امین زیرمجموعه اقدامهاست بطوریکه $1 \leq i \leq 2^n - 1$. $V(k)$ نشان‌دهنده زیرمجموعه‌ای از V است که در لحظه K فعال است.

شایان ذکر است که احتمال انتخاب هریک از اقدامها در لحظه k متناسب است با احتمال درستی بخش شرط قاعده‌ی متناظر آن اقدام

$$\text{در لحظه } k: p_i(k) = \text{prob}[\alpha(k) = \alpha_i] \approx$$

Correctness probability of rule R_i 's condition at instant k

همانطور که پیش از این بیان شد، در این روش احتمال درستی شرط قواعد در آغاز، بر مبنای تکنیک ارائه شده در روش $EX-SJF_{PRO-V.2.8}$ محاسبه می‌شود، لذا با فرض اینکه $K(0)$ نشان‌دهنده مجموع احتمالات درستی شرط قواعد در آغاز باشد. با توجه به رابطه‌ی (۳) احتمال انتخاب هر اقدام در آغاز در مقیاس ۱ طبق رابطه‌ی زیر محاسبه می‌گردد:

$$p_i(0) = (\text{Correctness probability of rule } R_i \text{'s condition at instant } 0) / k(0)$$

در لحظه k با فرض اینکه اقدامات متعلق به $V(k)$ ، فعال باشند، احتمال انتخاب اقدامها به روش زیر به‌هنگام می‌شوند:

$$K(k) = \sum_{\alpha_q \in v(k)} p_q(k) \quad (۶)$$

الف- پاسخ مطلوب از محیط در جواب اعمال اقدام $\alpha_q \in v(k)$ بر محیط

$$\begin{aligned} p_q(k+1) &= p_q(k) - ap_q(k) + ak(k) & \alpha_q \in v(k) \\ p_j(k+1) &= p_j(k) - ap_j(k) & \forall j, j \neq q, \alpha_j \in v(k) \end{aligned} \quad (۷)$$

ب- پاسخ نامطلوب از محیط در جواب اعمال اقدام $\alpha_q \in v(k)$ بر محیط

$$\begin{aligned} p_q(k+1) &= p_q(k) \\ p_j(k+1) &= p_j(k) & \forall j, j \neq q \end{aligned} \quad (۸)$$

ج- برای هر $\alpha_i \notin v(k)$

$$p_i(k+1) = p_i(k)$$

پس از شروع کار سیستم، در بازه‌های زمانی مشخص (σ_i) احتمال درستی بخش شرط (احتمال اجرای قواعد، براساس اطلاعات بدست آمده اخیر به همان روش بیان شده در بخش ۲-۴-۵، به‌هنگام می‌شوند. به‌هنگام سازی احتمال درستی شرط هر قاعده تا زمانی تکرار می‌شود که

آهنگ تغییرات آن کمتر از ε گردد. متعاقباً زمان اجرای یک قاعده زمانی بهنگام می‌شود که احتمال درستی شرط آن، نهایی و نیز زمان اجرای کلیه‌ی فرزندان آن بهنگام شده باشند. در آغاز هر بازه‌ی زمانی (σ_{i+1}) از اجرای سیستم، احتمال انتخاب اقدامات اتوماتای یادگیر متناسب با احتمال بهنگام شده‌ی درستی شرط قواعد و احتمال انتخاب اقدامات اتوماتای یادگیر در انتهای بازه‌ی زمانی قبلی (σ_i)، طبق (۹) بهنگام می‌شوند.

احتمال انتخاب اقدام نام (قاعده R_i) محاسبه شده به روش $Ex-SJF_{PRO-V.2.8}$ در پایان σ_k در مقیاس ۱

$$PR_{i,1}(\sigma_k)$$

احتمال انتخاب اقدام نام (قاعده R_i) محاسبه شده توسط اتوماتای یادگیر در پایان σ_k

$$PR_{i,2}(\sigma_k)$$

$$\phi(t) = \text{Min}(k/k', 1) \quad \ni \quad k' \geq 25$$

احتمال انتخاب اقدام نام (قاعده R_i) در پایان σ_k توسط اتوماتای یادگیر و در آغاز σ_{k+1}

$$PR_i(\sigma_{k+1}) = \frac{\phi(t) * PR_{i,2}(\sigma_k) + PR_{i,1}(\sigma_k)}{\phi(t) + 1} \quad (9)$$

مقایسه و ارزیابی روش‌های زمان‌بندی قواعد

در مرجع [۲] چارچوبی به منظور مقایسه روش‌های زمان‌بندی قواعد ارائه شده است. این چارچوب شامل پنج پارامتر ارزیابی شامل میانگین زمان پاسخگویی، انحراف معیار زمان پاسخگویی، توان عملیاتی، میزان زمان سریار محاسباتی به ازای هر تراکنش و بهره پردازشگر می‌باشد. اولویت این پارامترها (W_i در رابطه (۱۰)) به ترتیب معرفی آنها در عبارت پیشین می‌باشد.

در این چارچوب به منظور شبیه‌سازی رفتار سیستم پایگاه داده پویا یک محیط آزمایشگاهی به نام شبیه‌ساز سیستم پایگاه داده پویا (ADSS) طراحی و پیاده‌سازی شده است که با پیاده‌سازی روش‌های زمان‌بندی مختلف در آن می‌توان عملکرد آنها را مورد ارزیابی قرار داد. معماری، نحوه‌ی طراحی و پیاده‌سازی ADSS و روش‌های مختلف زمان‌بندی قواعد به طور مفصل در مرجع [۱] شرح داده شده است. ADSS دارای معماری سه بخشی است: (۱) واحد مدیریت اشیای داده، (۲) واحد مدیریت قواعد و (۳) واحد مدیریت تراکنش‌ها. واحد مدیریت اشیای داده که در حقیقت بخش پایگاه داده یک سیستم پایگاه داده پویا را شبیه‌سازی می‌کند، مدیریت کلیه‌ی امور مربوط به اقلام داده را برعهده دارد. واحد مدیریت قواعد که خود شامل چند بخش کوچکتر دیگر است کلیه امور مربوط به حفظ، نگهداری، بهنگام‌سازی، فعال سازی قواعد و تولید تراکنش براساس بخش عمل آنها را بر عهده دارد. واحد مدیریت

تراکنش‌ها مسؤول نگهداری، مدیریت، زمان‌بندی و اجرای تراکنش‌های تولید شده توسط واحد مدیریت قواعد و کاربر می‌باشد.

در بخش طراحی و پیاده‌سازی شبیه‌ساز سیستم پایگاه داده پویا، سعی شد با افزودن ویژگی‌هایی به ADSS موجود، رفتار سیستم مدیریت پایگاه داده پویا را به طور کاملتری شبیه‌سازی و متعاقباً عملکرد روش-های زمان‌بندی موجود را با دقت بیشتری مقایسه و ارزیابی نماییم. از جمله‌ی این ویژگی‌ها عبارتند از افزودن کامپایلرهای شرط و دستور به سیستم موجود، بقسمی که به کمک آنها می‌توانیم هنگام تولید قواعد پویا از عبارات شرطی و دستورات واقعی در بخش شرط و عمل قواعد استفاده کنیم و در زمان اجرا نیز با کامپایل عبارات شرطی بخش شرط قواعد فعال و دستورات بخش عمل قواعد در حال اجرا، محیط شبیه-سازی را تا حد امکان به محیط‌های واقعی نزدیک‌تر نماییم. در تولید تصادفی قواعد پویا، پارامترهای مورد نیاز جهت پوشش کلیه‌ی حالات و ویژگی‌های قواعد پویا در سیستم‌های واقعی در نظر گرفته شده‌اند

آزمایشات در سه حالت مختلف انجام شده است: حالت فوری، حالت تعویقی و حالت ترکیبی. در حالت فوری از پایگاه قاعده‌ای استفاده شده است که در آن پیوستگی رویداد-شرط و شرط-عمل قواعد فقط از نوع فوری است. در حالت تعویقی نیز پیوستگی رویداد-شرط و شرط-عمل قواعد فقط از نوع تعویقی می‌باشد و سرانجام در حالت ترکیبی از پایگاه قاعده‌ای استفاده شده است که در آن پیوستگی رویداد-شرط و شرط-عمل قواعد از هر سه نوع فوری، تعویقی و مستقل می‌تواند باشد [۲]، [۴]. نتایج حاصل از مقایسه‌ی روش‌های زمان‌بندی مختلف در حالت‌های تعویقی، فوری و ترکیبی به ترتیب در جداول (۱)، (۲) و (۳) نشان داده شده است. عددی که مقابل هر روش زمان‌بندی و پارامتر ارزیابی در هر خانه نوشته شده است، نشان دهنده رتبه آن روش زمان‌بندی در بین کلیه روش‌های مورد ارزیابی از نقطه نظر آن پارامتر ارزیابی است. رتبه-بندی نهایی روش‌ها بر طبق (۱۰) محاسبه می‌شود. شایان ذکر است که رتبه نهایی روشی بهتر خواهد بود، که مقدار $S(k)$ متناظرش کمتر باشد.

جدول (۱): نتایج ارزیابی روش‌های زمان‌بندی در حالت تعویقی

روش‌ها		پارامترهای ارزیابی	میانگین زمان پاسخگویی	انحراف معیار زمان پاسخگویی	توان عملیاتی	سر بار محاسباتی به ازای هر تراکنش	بهره‌ی پردازشگر
	اتفاقی	۵	۶	۳	۱	۳	
	اولویت ایستا	۵	۶	۳	۱	۳	
	برچسب زمانی	۵	۵	۳	۱	۳	
	EDF _{PD}	۵	۵	۳	۱	۳	
	EDF _{DIV}	۵	۸	۴	۱	۳	
	EDF _{SL}	۵	۷	۵	۱	۱	
	E _x -SJF _{EXA}	۴	۷	۴	۱	۲	
	E _x -SJF _{PRO}	۴	۴	۴	۱	۲	
	E _x -SJF _{PRO-V.1.8}	۳	۳	۳	۱	۲	
	E _x -SJF _{PRO-V.2.8}	۲	۲	۲	۱	۲	
	E _x -SJF _{ExTLA}	۱	۱	۱	۱	۲	

جدول (۲): نتایج ارزیابی روش‌های زمان‌بندی در حالت فوری

روش‌ها	پارامترهای ارزیابی	میانگین زمان پاسخگویی	انحراف معیار زمان پاسخگویی	توان عملیاتی	سر بار محاسباتی به ازای هر تراکنش	بهره‌ی پردازشگر
اتفاقی	۵	۵	۲	۳	۳	۳
اولویت ایستا	۵	۵	۲	۳	۳	۳
برچسب زمانی	۵	۵	۲	۱	۱	۱
EDF _{PD}	۴	۴	۱	۱	۲	۲
EDF _{DIV}	۴	۴	۱	۱	۲	۲
EDF _{SL}	۳	۳	۴	۲	۲	۲
E _x -SJF _{EXA}	۴	۴	۵	۳	۲	۲
E _x -SJF _{PRO}	۳	۳	۴	۲	۲	۲
E _x -SJF _{PRO} -V.1.8	۲	۲	۳	۳	۲	۲
E _x -SJF _{PRO} -V.2.8	۲	۲	۱	۳	۲	۲
E _x -SJF _{EsTLA}	۱	۱	۲	۲	۲	۲

جدول (۳): نتایج ارزیابی روش‌های زمان‌بندی در حالت ترکیبی

روش‌ها	پارامترهای ارزیابی	میانگین زمان پاسخگویی	انحراف معیار زمان پاسخگویی	توان عملیاتی	سر بار محاسباتی به ازای هر تراکنش	بهره‌ی پردازشگر
اتفاقی	۷	۷	۵	۳	۱	۱
اولویت ایستا	۶	۶	۴	۴	۲	۱
برچسب زمانی	۶	۶	۴	۴	۲	۱
EDF _{PD}	۵	۵	۳	۵	۲	۱
EDF _{DIV}	۵	۵	۳	۵	۲	۱
EDF _{SL}	۵	۵	۶	۶	۲	۱
E _x -SJF _{EXA}	۴	۴	۳	۳	۲	۱
E _x -SJF _{PRO}	۴	۴	۳	۳	۲	۱
E _x -SJF _{PRO} -V.1.8	۳	۳	۲	۲	۲	۱
E _x -SJF _{PRO} -V.2.8	۲	۲	۱	۱	۲	۱
E _x -SJF _{EsTLA}	۱	۱	۱	۱	۲	۱

$k \in \text{Rule Scheduling Approaches Set}$

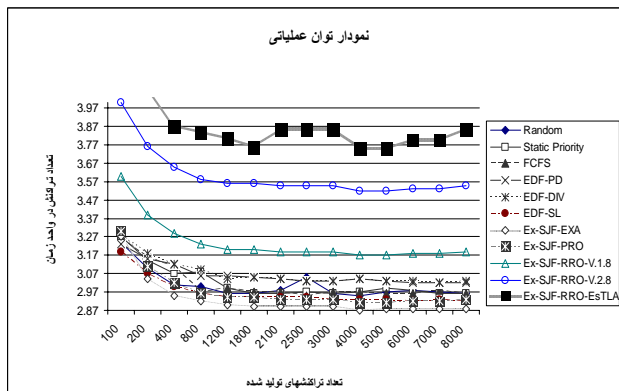
W_i is the weight of i th evaluation metric

$$S(k) = \sum_{j=1}^3 \sum_{i=1}^5 W_i * Rnk_{i,j}(k) \quad (10)$$

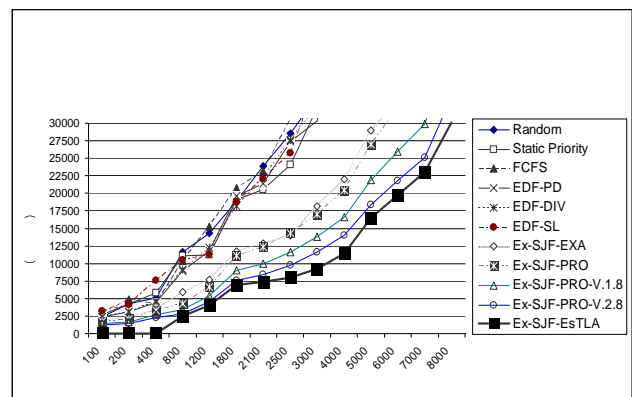
$j=1,2 \text{ and } 3$ indicates deferred, immediate and composite state, respectively

$Rnk_{i,j}(k)$ is the rank of approach k in j th state based on i th evaluation metric

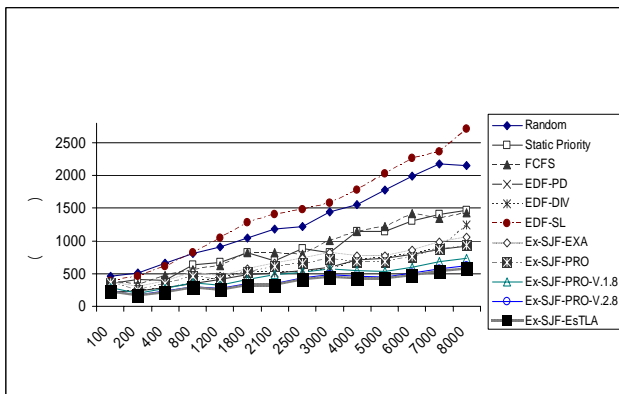
$S(k)$ is the total score of approach k among all approaches



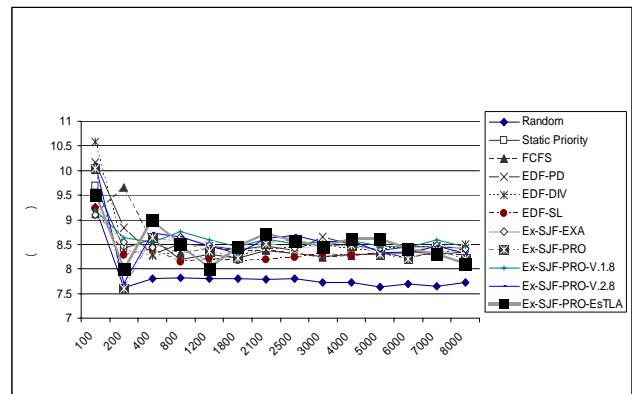
شکل (۳): نمودار توان عملیاتی روش‌های زمان‌بندی در حالت فوری



شکل (۲): نمودار میانگین زمان پاسخگویی روش‌های زمان‌بندی در حالت تعویقی



شکل (۵): نمودار انحراف معیار زمان پاسخگویی روش‌های زمان‌بندی در حالت ترکیبی



شکل (۴): نمودار میانگین زمان سر بار محاسباتی به ازای هر تراکنش روش‌های زمان‌بندی در حالت ترکیبی

شکل‌های (۲) الی (۵) چند نمونه از نمودارهای ارزیابی روش‌های زمانبندی اجرای قواعد در حالات مختلف را نشان می‌دهند. همانگونه که در شکل (۲) ملاحظه می‌شود، دو روش E_X-SJF_{ESTLA} و E_X-SJF_{PRO} دارای کمترین میانگین زمان پاسخگویی بین کلیه روش‌ها در حالت تعویقی هستند. روش‌های E_X-SJF_{PRO} ، $E_X-SJF_{PRO-V.1.8}$ و E_X-SJF_{PRO} از نظر میانگین زمان پاسخگویی در رتبه‌های بعدی قرار دارند. سایر روش‌ها از این حیث دارای رفتار مشابهی هستند. با توجه به نمودار شکل (۲) به این نتیجه می‌رسیم که استفاده از روش E_X-SJF_{ESTLA} موجب کاهش ۹/۸ درصدی میانگین زمان پاسخگویی نسبت به روش $E_X-SJF_{PRO-V.2.8}$ و استفاده از روش $E_X-SJF_{PRO-V.2.8}$ موجب کاهش ۱۵/۸ درصدی میانگین زمان پاسخگویی نسبت به روش E_X-SJF_{PRO} و استفاده از روش $E_X-SJF_{PRO-V.1.8}$ موجب کاهش ۱۸/۸ درصدی میانگین زمان پاسخگویی نسبت به روش‌های E_X-SJF_{PRO} و E_X-SJF_{EXA} می‌شود (عملکرد دو نسخه‌ی روش E_X-SJF در زمینه‌ی میانگین زمان پاسخگویی در حالت تعویقی بسیار نزدیک به هم بوده ولی روش E_X-SJF_{PRO} اندکی بهتر از روش E_X-SJF_{EXA} عمل می‌کند). این درحالیست که دو روش اخیر (E_X-SJF_{PRO} و E_X-SJF_{EXA}) نیز به نوبه‌ی خود موجب کاهش ۴۳ درصدی میانگین زمان پاسخگویی نسبت به سایر روش‌ها در حالت تعویقی خواهند شد. نکته‌ی دیگری که از این شکل مشخص است، شیب نمودار است که برای روش‌های E_X-SJF_{ESTLA} ، $E_X-SJF_{PRO-V.2.8}$ ، $E_X-SJF_{PRO-V.1.8}$ از بقیه روش‌ها کمتر بوده که نشان می‌دهد سرعت رشد میانگین زمان پاسخگویی در این سه روش از بقیه روش‌ها بسیار کمتر است.

شکل (۳) رفتار روش‌های زمانبندی در حالت فوری از نقطه نظر توان عملیاتی را نشان می‌دهد. روش E_X-SJF_{EXA} دارای کمترین توان عملیاتی است. دو روش E_X-SJF_{PRO} و EDF_{SL} دارای توان عملیاتی بیشتری نسبت به E_X-SJF_{EXA} هستند. روش‌های $E_X-SJF_{PRO-V.2.8}$ و $E_X-SJF_{PRO-V.1.8}$ از این حیث در رتبه‌های بهتری نسبت به سه روش پیش قرار دارند به طوریکه روش $E_X-SJF_{PRO-V.1.8}$ موجب افزایش ۹ درصدی توان عملیاتی روش E_X-SJF_{PRO} می‌شود و روش E_X-SJF_{PRO} نیز موجب افزایش ۱۱ درصدی توان عملیاتی روش E_X-SJF_{PRO} می‌شود. اما این روش‌ها نیز خود از نظر این معیار ارزیابی بعد از چهار روش E_X-SJF_{ESTLA} ، «اولویت ایستا»، FCFS و «انتخاب اتفاقی» قرار می‌گیرند. به طوریکه توان عملیاتی روش E_X-SJF_{ESTLA} ۷ درصد بهتر از توان عملیاتی روش $E_X-SJF_{PRO-V.2.8}$ است. همانگونه که در شکل (۳) نشان داده شده است دو روش EDF_{PD} و EDF_{DIV} دارای بیشترین توان عملیاتی در این حالت هستند.

همانگونه که در شکل (۴) ملاحظه می‌شود روش «انتخاب اتفاقی» دارای کمترین «سربار محاسباتی به ازای هر تراکنش» در حالت ترکیبی است در حالیکه رفتار بقیه‌ی روش‌ها در رابطه با این معیار تقریباً یکسان است. شکل (۵) نشان می‌دهد که دو روش E_X-SJF_{ESTLA} و E_X-SJF_{PRO}

$V.2.8$ دارای کمترین انحراف معیار زمان پاسخگویی هستند. روش E_X-SJF_{ESTLA} موجب کاهش ۶/۹ درصدی انحراف معیار زمان پاسخگویی روش $E_X-SJF_{PRO-V.2.8}$ شده است. روش $E_X-SJF_{PRO-V.2.8}$ نیز به نوبه‌ی خود موجب کاهش ۱۶/۴ درصدی انحراف معیار زمان پاسخگویی روش $E_X-SJF_{PRO-V.1.8}$ شده است. بعد از این سه روش، نسخه‌های E_X-SJF_{PRO} و E_X-SJF_{EXA} به همراه دو روش EDF_{PD} و EDF_{DIV} به ترتیب دارای کمترین انحراف معیار زمان پاسخگویی هستند. روش $E_X-SJF_{PRO-V.1.8}$ نیز موجب کاهش ۲۱/۴ درصدی انحراف معیار زمان پاسخگویی روش E_X-SJF_{PRO} شده است و پس از آنها دو روش «اولویت ایستا» و FCFS دارای کمترین انحراف معیار زمان پاسخگویی هستند. نمودار شکل (۵) همچنین نشان می‌دهد دو روش EDF_{SL} و «انتخاب اتفاقی» به ترتیب دارای بیشترین انحراف معیار زمان پاسخگویی هستند. نکته‌ی دیگری که از این شکل مشخص می‌شود شیب تند نمودار برای روش‌های EDF_{SL} و «انتخاب اتفاقی» است که نشان می‌دهد در این روش‌ها در صورت افزایش تعداد تراکنش‌های تولید شده، تجمع زمان‌های پاسخگویی تراکنش‌ها حول نقطه‌ی میانگین به شدت کاهش خواهد یافت. شیب نمودار برای روش‌های E_X-SJF_{ESTLA} ، $E_X-SJF_{PRO-V.2.8}$ و $E_X-SJF_{PRO-V.1.8}$ از همه کمتر بوده و نشان می‌دهد سرعت رشد انحراف معیار زمان پاسخگویی در این چهار روش از بقیه بسیار کمتر است.

نتایج آزمایشات و تحلیل‌های مربوطه حاکی از آن است که در مجموع روش E_X-SJF_{ESTLA} از سایر روش‌های مورد ارزیابی کاراتر است. همانطور که ملاحظه می‌شود زمانبندی قواعد فعال بر مبنای احتمال درستی شرط آنها و نه بر مبنای زمان اجرای آنها به کمک اتوماتای یادگیر، منجر به بهبود روش زمانبندی E_X-SJF از نظر میانگین زمان پاسخگویی، انحراف معیار زمان پاسخگویی و توان عملیاتی می‌شود. فرایند تخمین احتمال وقوع قواعد در E_X-SJF_{ESTLA} به گونه‌ای است که سربار محاسباتی بر سیستم تحمیل نمی‌کند. لذا نسخه‌ی E_X-SJF_{ESTLA} با سایر نسخه‌های مبتنی بر الگوریتم SJF از نظر میزان سربار محاسباتی به ازای هر تراکنش و بهره‌پردازشگر برابری می‌کند. جدول (۴) درصد کاراتر بودن روش E_X-SJF_{ESTLA} را نسبت به روش $E_X-SJF_{PRO-V.2.8}$ (کاراترین روش زمانبندی موجود) در سه حالت تعویقی، فوری و ترکیبی براساس سه پارامتر ارزیابی میانگین زمان پاسخگویی، انحراف معیار زمان پاسخگویی و توان عملیاتی نشان می‌دهد.

جدول (۴) درصد کاراتر بودن روش E_X-SJF_{ESTLA} نسبت به روش $E_X-SJF_{PRO-V.2.8}$

حالات مقایسه	پارامترهای ارزیابی	میانگین زمان پاسخگویی	انحراف معیار زمان پاسخگویی	توان عملیاتی
فوری	۱/۸۶٪	۶/۴٪	۷٪	
تعویقی	۹/۸٪	۱۱/۹٪	۶/۷٪	
ترکیبی	۷/۱۸٪	۶/۹٪	۱۲/۳٪	

- Rasoolzadegan, A., "A New Rule Scheduling Approach based on Estimation of Rule Execution Probability in Active Database System", MSc Thesis, Amirkabir University of Technology (Tehran PolyTechnic), 2007.
- Alesheykh, R., "An Effective Rule Selection Approach in Active Database Systems", MSc Thesis, Amirkabir University of Technology (Tehran PolyTechnic), 2005.
- Rasoolzadegan, A.; Alesheykh, R.; Abdollahzadeh, A.; "A New Approach for Event Triggering Probability Estimation in Active Database Systems to Rule Scheduling Improvement", 2nd IEEE International Conference on Information & Communication Technologies: From Theory To Applications, Damascus, Syria, April 24 - 28, 2006.
- Rasoolzadegan, A.; Alesheykh, R.; Abdollahzadeh, A.; "Measuring Evaluation Parameters in Benchmarking Rule Scheduling Methods in Active Database Systems", The IEEE International Conference on Computer and Communication Engineering, Kuala Lumpur, Malaysia, 2006.
- Vadua, A.; *Rule Development for active database*, PhD Thesis, CS Department, University of Zurich, 1999.
- Sivasankaran, R. M.; Stankovic, J. A.; Towsley, D.; Purimetla, B.; Ramamritham, K.; "Priority Assignment in Real-Time Active Databases", The International Journal on Very Large Data Bases, Vol. 5, No. 1, January 1996.
- Ceri, S.; Gennaro, C.; Paraboschi, S.; Serazzi, G.; "Effective Scheduling of Detached Rules in Active Database", IEEE Transaction on Knowledge and Data Engineering, Vol. 15, No.1, 2003.
- Narendra, K. S.; Thathachar, L.; "Learning Automata: An Introduction", Prentice Hall, 1989.
- Thathachar, M. A. L.; Sastry, P. S.; "Varieties of Learning Automata: An Overview", IEEE Transaction on Systems, Man and Cybernetics-Part B: Cybernetics, Vol. 32, No. 6, pp. 711-722, 2002.
- Thathachar, M. A. L.; Harita, B. R.; "Learning Automata with Changing Number of Actions", IEEE Transaction on Systems, Man and Cybernetics, Vol. SMC-17, NO. 6, pp. 1095-1100, 1987.

آخر نویس ها

- ¹ Active Database Management System
- ² Three Tier Architecture
- ³ Learning Automaton
- ⁴ Event - Condition - Action
- ⁵ Active Database System Simulator
- ⁶ Total Order (Static Priority)
- ⁷ Earliest Deadline First
- ⁸ Extended-Shortest Job First-(based on)-Estimation-(using)-Learning Automata

همانطور که مشاهده می‌شود با بهبود فرایند زمانبندی قواعد فعال در نسخه جدید ($EX-SJF_{ESTLA}$) نسبت به نسخه‌های ارائه شده پیشین، کارایی فرایند زمانبندی قواعد از نظر میانگین زمان پاسخگویی، انحراف معیار زمان پاسخگویی و توان عملیاتی افزایش یافته است. از نظر منطقی نیز انتظار می‌رود که چنین بهبودی حاصل شود. همانطور که پیش از این نیز بیان شد، از بین قواعد فعال فقط آنهایی اجرا می‌شوند که شرطشان در لحظه‌ی ارزیابی درست باشد و سایر قواعد فعال از گردونه‌ی پردازش خارج می‌شوند. اما زمانی که صرف انتخاب این دسته از قواعد فعال و نیز ارزیابی آنها می‌شود از جمله زمان‌های غیر مفید کارکرد سیستم محسوب می‌شود که مستقیماً در توان عملیاتی (تعداد قواعد اجرا شده در واحد زمان)، میانگین زمان پاسخگویی (میانگین زمان انتظار قواعد برای اجرا) و به طور غیر مستقیم در انحراف معیار زمان پاسخگویی (میانگین فاصله‌ی زمان پاسخگویی هر قاعده از میانگین زمان پاسخگویی) تأثیرگذار است. بدیهی است هرچه اولویت پردازش قواعد فعالی که بدون اجرا شدن از گردونه پردازش، خارج می‌شوند کمتر باشد، کارایی سیستم از بُعد پارامترهای ارزیابی پیش‌گفته بهتر می‌شود. لذا اگر فرایند زمانبندی قواعد فعال، مبتنی بر احتمال درستی شرط قواعد باشد، دستیابی به این هدف را ممکن می‌سازد. علاوه بر آن اگر مکانیزمی وجود داشته باشد که به تخمین دقیقتر احتمال درستی شرط قواعد در حین اجرای سیستم بپردازد، امکان نیل به این هدف را بیشتر می‌سازد. کاری که در روش جدید به کمک اتوماتای یادگیر امکانپذیر شده است. سربار محاسباتی فرایند زمانبندی قواعد در روش اخیر نسبت به روش $EX-SJF_{PRO-V.2.8}$ فقط در حد بهنگام سازی احتمال درستی شرط برخی از قواعد فعال، طبق (۷) است که بسیار ناچیز و قابل اغماض می‌باشد.

نتیجه

در این مقاله ابتدا پایگاه داده پویا و چرخه‌ی پردازش قواعد را تعریف کردیم. سپس جایگاه و اهمیت فرایند زمانبندی قواعد را در چرخه‌ی پردازش قواعد بیان نمودیم. در ادامه به معرفی روش‌های مختلف زمانبندی قواعد پرداختیم. سپس به منظور بهبود کارترین روش زمانبندی موجود ($EX-SJF_{PRO-V.2.8}$) درصدد بهبود فرایند زمانبندی قواعد فعال به کمک اتوماتای یادگیر برآمدیم. بر همین اساس روش جدیدی ارائه کردیم و آن را $EX-SJF_{ESTLA}$ نامیدیم. در نهایت به بررسی عملکرد روش پیشنهادی جدید در مقایسه با روش‌های زمانبندی موجود بر اساس تعدادی پارامتر ارزیابی بر مبنای نتایج حاصل از انجام آزمایشات متعدد و متنوع در یک محیط آزمایشگاهی مبتنی بر معماری سه بخشی (ADSS) پرداختیم. نتایج آزمایشات بهبود فرایند زمانبندی قواعد را در نسخه‌ی $EX-SJF_{ESTLA}$ از نظر میانگین زمان پاسخگویی، انحراف معیار زمان پاسخگویی و توان عملیاتی نشان می‌دهد.