

Generalizing The Concept of Business Processes Structural Soundness from Classic Petri-nets to BPMN2.0 Process Models

Yahya Poursoltani¹, Mohammad Hassan Shirali-Shahreza^{2*}, S.Alireza Hashemi Golpayegani³

¹ Department of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran

² Department of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran

³ APA Research Center-Information Technology Engineering Group, Amirkabir University of Technology, Tehran, Iran

Received: 18 February 2023, Revised: 02 April 2023, Accepted: 11 June 2023

Paper type: Research

Abstract

BPMN2.0 Standard is a modeling language, which can be understood and used by a wide range of users. However, because of its non-formal nature, models (designed using it) can be containing structural errors such as Deadlock (impossibility of executing some of process tasks) and Livelock (infinite repetition of tasks) may be produced by using them. These semantic errors can create anomalies in the workflow of the organization. So far, some researches has been conducted on the validation of these process models and various solutions have been provided to discover some of these structural errors. The question that may be raised about these methods is whether it is possible to definitely guarantee the structural accuracy of a BPMN method model by using any of them? To answer this question, we need a comprehensive definition of a correct BPMN2.0 process model, based on which we can evaluate the comprehensiveness of validation methods and strongly make sure that the considered method can discover all of the structural errors of the process model. In this paper, based on concept of general process models and the concept of soundness (based on process models created using Petri nets) and the generalization of its properties, i.e. Liveness and Boundness to BPMN2.0 process models, a comprehensive definition for a correct (sound) BPMN2 process model provided. Then, the comprehensiveness of the suggested methods of some of the most important researches conducted has been evaluated based on it. This definition can be used as a measure for efficiency of BPMN validation methods.

Keywords: Structural Soundness, BPMN2.0 Processes Models, Process Models, Liveness, Boundness.

* Corresponding Author's email: hshirali@aut.ac.ir

تعمیم مفهوم صحت ساختاری فرآیندهای کسب و کار از شبکه‌های پتری کلاسیک به مدل‌های فرآیندی BPMN

یحیی پورسلطانی^۱، محمد حسن شیرعلی شهرضا^{۲*}، سید علیرضا هاشمی گلیایگانی^۳
^۱ دانش آموخته‌ی کارشناسی ارشد علوم کامپیوتر، دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)، تهران، ایران
^۲ استادیار، دانشکده ریاضی و علوم کامپیوتر، دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)، تهران، ایران
^۳ استادیار، مرکز پژوهشی آپا- گروه مهندسی فناوری اطلاعات، دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)، تهران، ایران

تاریخ دریافت: ۱۴۰۱/۱۱/۲۹ تاریخ بازبینی: ۱۴۰۲/۰۱/۱۳ تاریخ پذیرش: ۱۴۰۲/۰۳/۲۱
نوع مقاله: پژوهشی

چکیده

استاندارد BPMN2.0 یک زبان مدل‌سازی به منظور طراحی فرآیندهای کسب و کار می‌باشد که به علت گویایی بالا، برای طیف وسیعی از کاربران قابل درک و استفاده می‌باشد؛ با این حال، به علت ماهیت غیر صوری آن ممکن است مدل‌هایی حاوی خطاهایی ساختاری هم‌چون بن‌بست (عدم امکان اجرای فرآیند) و بن‌بست حلقوی (تکرار نامتناهی کارها) با استفاده از آن‌ها تولید شود. تاکنون تحقیقات زیادی پیرامون صحت‌سنجی این مدل‌های فرآیندی صورت گرفته است و راهکارهای مختلفی به منظور کشف برخی از این خطاهای ساختاری ارائه شده‌است؛ سوالی که در خصوص این روش‌ها ممکن است مطرح شود این است که آیا می‌توان با استفاده از هرکدام از آن‌ها صحت ساختاری یک مدل فرآیندی BPMN را به صورت قطعی تضمین کرد؟ برای پاسخگویی به این سوال نیازمند یک تعریف جامع از یک مدل فرآیندی BPMN2.0 صحیح هستیم تا بر مبنای آن بتوانیم جامعیت روش‌های صحت‌سنجی را مورد ارزیابی قرار داده و بدین وسیله، مطمئن شویم که روش ارائه شده می‌تواند تمامی خطاهای ساختاری مدل فرآیندی را کشف نماید. در این پژوهش، بر پایه‌ی یک تعریف عام از مدل‌های فرآیندی و مفهوم صحت (بر اساس مدل‌های فرآیندی ایجاد شده با استفاده از شبکه‌های پتری) و تعمیم ویژگی‌های آن، یعنی زنده بودن و کران‌دار بودن به مدل‌های فرآیندی BPMN2.0، یک تعریف جامع برای یک مدل فرآیندی BPMN2.0 صحیح ارائه شده‌است و بر اساس آن، جامعیت روش‌های پیشنهادی برخی از مهم‌ترین پژوهش‌های صورت گرفته مورد ارزیابی قرار داده شده‌است؛ این تعریف می‌تواند به عنوان یک معیار برای سنجش کارایی روش‌های صحت‌سنجی این مدل‌های فرآیندی به کار گرفته‌شود.

کلیدواژه‌گان: صحت ساختاری، زبان مدل‌سازی BPMN 2.0، شبکه‌های پتری، مدل‌های فرآیندی، کراندار بودن، زنده بودن.

۱- مقدمه

قابل استفاده می‌باشد [۳، ۴].

با فراگیر شدن مدل‌های فرآیندی BPMN در بسیاری از سکوه‌های توسعه‌ی کم‌کد و ایجاد نرم‌افزار بدون نیاز به دانش برنامه‌نویسی سنتی (نظیر لزوم تسلط به زبان‌های برنامه‌نویسی متداول)، با کاربرانی مواجه می‌شویم که به این سکوها بیش از حد خوش‌بین هستند و ممکن است احساس کنند که با وجود این سکوها و تولید نرم‌افزار با استفاده از آن‌ها، از بسیاری از خطاهای زمان اجرا جلوگیری می‌شود؛ این درحالی است که بروز خطاهای ساختاری در مدل‌های فرآیندی BPMN و تبدیل شدن آن به یک نرم‌افزار عملیاتی می‌تواند منجر به انتشار این خطاها به سطوح مختلفی از نرم‌افزارهای سازمان شود. در خصوص چالش تضمین کیفیت نرم‌افزارهای ایجاد شده با استفاده از سکوه‌های توسعه‌ی کم‌کد، استفان برامز [۵] عدم وجود کدهای کامپیوتری و در نتیجه، عدم امکان اجرای آزمون‌های متداول نرم‌افزاری نظیر آزمون واحد^۱ را علتی بر بروز خطا در نرم‌افزارهای ایجاد شده با استفاده از سکوه‌های توسعه‌ی کم‌کد می‌داند. با توجه به ماهیت غیر صوری مدل‌های فرآیندی BPMN، طراحان فرآیند معمولاً شاخص واضحی برای صحت‌سنجی مستقیم مدل‌های فرآیندی ندارند و ممکن است متوجه بروز بسیاری از این خطاهای ساختاری نشوند. بنابراین اهمیت صحت‌سنجی ساختاری مدل‌های فرآیندی BPMN پیش از اجرای عملیاتی آن‌ها، بیش از پیش آشکار می‌شود؛ چرا که ممکن است مدل‌های فرآیندی ایجاد شده به وسیله‌ی آن ناصحیح بوده و حاوی خطاهایی ساختاری نظیر بن‌بست^۲ و یا بن‌بست حلقوی^۳ باشند؛ وجود این خطاهای ساختاری در این مدل‌های فرآیندی، می‌تواند ضمن ایجاد اختلال در گردش جریان کار سازمان، منجر به بروز اختلالاتی در سطوح مختلف معماری آن شود و یا ممکن است باعث ایجاد نرم‌افزارهایی با عملکردی ناصحیح شود.

در تحقیقات متعددی (از جمله [۱]، [۶]، [۷]، [۸]، [۹] و [۱۰])، راه‌کارهای مختلفی به منظور صحت‌سنجی مدل‌های فرآیندی BPMN2.0 ارائه شده‌است؛ در هر کدام از آن‌ها، به کشف انواع مختلفی از خطاهای ساختاری نظیر بن‌بست‌ها، بن‌بست‌های حلقوی، پایان‌های چندگانه و غیره پرداخته شده‌است؛ بنابراین در هر کدام از آن‌ها، برداشتهای مختلفی از یک مدل فرآیندی BPMN2.0 ناصحیح ارائه شده‌است و از شاخص واحد و مشخصی برای سنجش صحت یک مدل فرآیندی BPMN 2.0 صحیح استفاده نشده‌است؛ همچنین در بسیاری از این تحقیقات، شاخص‌های تاییدکننده‌ی صحت ساختاری مدل فرآیندی بر مبنای مدل‌های صوری (مدل صوری مقصد که مدل BPMN به آن تبدیل می‌شود) بیان شده‌اند؛

در طی دهه‌های اخیر، با رشد فرآیندهای تجاری و پیچیده‌تر شدن آن‌ها، لزوم ایجاد نرم‌افزارهایی برای مدیریت آن‌ها روز به روز بیشتر حس می‌شد [۱]. پیچیده‌تر شدن روزافزون فرآیندهای کسب و کار منجر به افزایش شکاف بین تحلیل‌گران، توسعه‌دهندگان نرم‌افزارها و کاربران نهایی شد [۱]. به منظور کمک به کاهش این شکاف، استاندارد BPMN به منظور مدل‌سازی فرآیندهای تجاری و توسط گروه مدیریت شی^۱ به وجود آمد و یکی از اهداف مهم آن، کاهش فاصله‌ی بین طراحان یک سیستم و توسعه‌دهندگان آن است؛ چرا که با استفاده از این استاندارد می‌توان فرآیندهای کسب و کار را به صورتی بصری توصیف نمود [۲]؛ در نتیجه توسط طیف وسیعی از کاربران (حتی کاربرانی که با دانش کامپیوتر آشنایی زیادی ندارند) قابل درک خواهد بود.

به گفته هانگ و همکاران [۲] مدل‌های فرآیندی BPMN2.0 دارای دو ویژگی زیر هستند که منجر به ایجاد ماهیتی غیر صوری در آن می‌شود و در نتیجه، صحت‌سنجی و تحلیل آن را با دشواری‌هایی مواجه می‌سازد [۲]:

۱. زبان BPMN به عنوان یک زبان گراف‌گرای ایستا^۲ شناخته شده‌است.
۲. شاخص مناسبی برای ارزیابی و پیش‌بینی رفتارهای پویای این مدل‌های فرآیندی (در زمان اجرا) وجود ندارد؛ این موضوع می‌تواند منجر به بروز خطاهایی در جریان کنترلی^۳ فرآیند و جریان اطلاعات^۴ در آن شود.

در نتیجه، نگاشت این مدل‌های فرآیندی به مدل‌هایی صوری می‌تواند به تحلیل بهتر آن‌ها کمک کند.

سادگی و گویایی این استاندارد، باعث شده‌است تا در بسیاری از سکوه‌های توسعه‌ی کم‌کد مثل Bizagi، Biztop و Ubi Process ایرانی سکوه‌های ایرانی BPMN، بدون نیاز به کدنویسی و صرفاً با ترسیم نمودارهای BPMN، استفاده شود [۳] [۴]. در این خصوص، رابرت واتسکوزکی در قالب یک تحقیق، ضمن معرفی سکوه‌های توسعه‌ی کم‌کد و مبتنی بر زبان BPMN، اساسی‌ترین نیازمندی‌های کاربردی و غیر کاربردی این سکوها را تشریح کرد [۴]. با استفاده از این سکوها، مدل‌های فرآیندی BPMN ترسیم می‌شوند؛ سپس بر مبنای مدل طراحی شده، بدون نیاز به برنامه‌نویسی، واسطه‌های کاربری مورد نیاز برای اجرای فرآیندها ساخته می‌شود و نرم‌افزارهای اجراکننده‌ی آن مدل‌های فرآیندی، ایجاد شده و برای کاربران نهایی

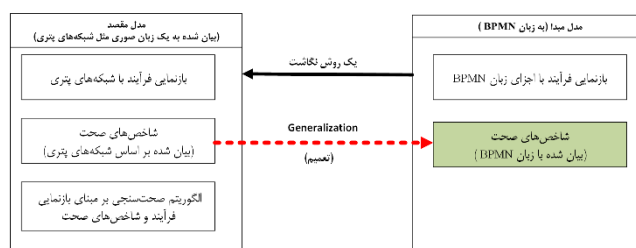
مقایسه کرده و تطبیق دهیم و چنانچه این شاخص‌ها با یکدیگر معادل بودند، از کارآمدی شاخص‌های تعریف شده در مدل فرآیندی مقصد اطمینان حاصل می‌شود.

بنابراین در ابتدا لازم است یک تعریف کاملاً مشخص و واحد، برای یک مدل فرآیندی BPMN2.0 صحیح (با بیان مدل مبدا) ارائه شود تا بتوان در ارزیابی کارهای انجام شده و نیز در تحقیقات بعدی بر اساس این تعریف، جامعیت و کارآمدی روش‌های صحت‌سنجی را ارزیابی نمود؛ علاوه بر آن یکی از موضوعات اساسی در انتقال مدل‌های نرم‌افزاری به یک دیگر، ارزیابی درستی انتقال مدل است که به معنای حصول اطمینان از معادل بودن مدل مقصد با مدل مبدا است؛ این موضوع در حوزه‌ی مهندسی نرم‌افزار مدل رانده^۲، از اهمیت بالایی برخوردار است [۱۱]. بنابراین با دانستن مفهوم صحت ساختاری به بیان مدل‌های BPMN و انتقال این مفاهیم به مدل مقصد، می‌توان با مقایسه‌ی شاخص‌های صحت در مدل مقصد با انتقال یافته‌ی شاخص‌های صحت مدل‌های BPMN، از مناسب بودن شاخص‌های صحت در مدل مقصد اطمینان حاصل نمود.

در این پژوهش، با در نظر گرفتن مدل‌های فرآیندی BPMN2.0 به عنوان یک مدل فرآیندی (بر اساس مفهوم عام و کلی یک مدل فرآیندی-پیشنهاد شده توسط ماتیس و سکی و همکارانش [۱۲])، بر مبنای یک نگاشت بین شبکه‌های پتری و مدل‌های فرآیندی BPMN2.0 (پیشنهاد شده توسط داکمن و همکارانش [۱۳])، دو مشخصه‌ی یک مدل فرآیندی صحیح ایجاد شده به وسیله‌ی شبکه‌های پتری، یعنی زنده بودن^۳ و کران‌دار بودن^۴، به مدل‌های فرآیندی BPMN2.0 تعمیم داده شده‌اند و در نهایت، مفهوم صحت از شبکه‌های پتری به صحت مدل‌های فرآیندی BPMN2.0 تعمیم داده شده‌است (مسیر فلش خط‌چین قرمز رنگ در شکل ۱). بنابراین در این تحقیق به این سوالات پاسخ داده شده‌است:

۱. تصدیق صحت یک مدل فرآیندی BPMN (با بیان زبان مدل‌سازی BPMN) بر مبنای چه ویژگی‌هایی باید صورت گیرد؟
۲. ویژگی‌های بدست آمده برای صحت یک مدل فرآیندی BPMN (با بیان زبان مدل‌سازی BPMN) در چه وضعیتی باشند تا بتوان گفت که آن مدل فرآیندی صحیح است؟
۳. این ویژگی‌ها به چه شکلی در مدل فرآیندی مبدا (به زبان BPMN) خود را نشان می‌دهند؟
۴. روش‌های پیشنهاد شده در تحقیقات پیشین به منظور صحت‌سنجی مدل‌های فرآیندی BPMN (در مدل‌های مقصد و بازنمایی شده با مدل‌های صوری) تا چه حدی این ویژگی‌ها

در نتیجه به صورت عینی برای کاربران نهایی و تحلیلگرانی که با آن مدل صوری آشنا نیستند، چندان قابل درک و ملموس نیستند [۱]؛ حال آنکه بهتر است در ابتدا، این شاخص‌ها بر روی مدل مبدا (مدل BPMN) به صورتی ملموس و قابل مشاهده تعریف شده‌باشند و پس از صحت‌سنجی فرآیند در مدل مقصد، از برقرار بودن آن شاخص‌ها در مدل مبدا نیز اطمینان حاصل نمود؛ در صورتی که در خصوص یک نمونه مدل فرآیندی BPMN، تایید شاخص‌های صحت ساختاری در مدل مقصد منجر به تایید شاخص‌های صحت ساختاری مدل مبدا گردد، به معنای مناسب بودن و کارا بودن روش صحت‌سنجی می‌باشد. برای درک بهتر این ضرورت، به شکل ۱ توجه کنید.



شکل ۱. نمایی از جایگاه تعمیم مفهوم صحت ساختاری در مدل‌های فرآیندی BPMN

طبق شکل ۱ مدل فرآیندی مبدا^۵ (یک مدل فرآیندی BPMN) حاوی عناصری می‌باشد که در قالب یک روش نگاشت^۶، می‌تواند به یک مدل فرآیندی مقصد^۷ (مثل شبکه‌های پتری) نگاشته شود؛ در نتیجه در مدل مقصد یک بازنمایی جدید از آن مدل فرآیندی به وجود می‌آید. به منظور صحت‌سنجی ساختاری این مدل فرآیندی (در مدل مقصد) لازم است موارد زیر مشخص باشد:

۱. شاخص‌هایی برای صحت مدل فرآیندی^۸ تعیین شده‌باشد تا بتوان بر مبنای آن‌ها، در خصوص صحیح بودن و یا صحیح نبودن مدل فرآیندی به صورت قطعی تصمیم گرفت.
۲. الگوریتمی به منظور صحت‌سنجی مدل فرآیندی (بازنمایی شده در زبان مقصد) وجود داشته‌باشد که بر مبنای شاخص‌های صحت ساختاری (بند ۱)، صحیح بودن و یا صحیح نبودن مدل فرآیندی را تعیین نماید.

بدیهی است که اگر شاخص‌های صحت در مدل فرآیندی مقصد، شاخص‌های مناسبی نباشند آنگاه الگوریتم صحت‌سنجی ارائه شده در مدل فرآیندی مقصد نیز به درستی عمل نخواهد کرد؛ بنابراین برای آن‌که از کارآمد بودن شاخص‌های صحت در مدل مقصد اطمینان حاصل نماییم، لازم است آن را با شاخص‌هایی کلی‌تر (که در زبان مدل‌سازی مبدا - یعنی زبان BPMN - قابل بیان باشد)

صحت‌سنجی تراکنش‌های مدل‌سازی شده به وسیله‌ی مدل‌های فرآیندی BPMN پرداختند [۱۵]؛ با این کار، به صحت‌سنجی بخش‌هایی از فرآیند که امکان لغو تراکنش در آن وجود داشت، پرداخته شده‌است [۱۵]؛ چرا که در بسیاری از مواقع لغو تراکنش می‌تواند منجر به بروز بن‌بست و یا بن‌بست حلقوی شود.

در سال ۲۰۱۱ تحقیقی توسط واتاهیکی و همکاران انجام شد که در آن، روشی برای صحت‌سنجی مدل‌های BPMN ارائه شد که از آن به منظور جلوگیری از بروز خطاهای معنایی در فرآیندهایی که در منابع و زمان دچار محدودیت هستند، استفاده شد [۱۶]. در این پژوهش، مدل فرآیندی BPMN گسترش داده شده و سپس به یک اوتوماتای زمانی نگاشته شده‌است؛ سپس اوتوماتای زمانی بدست آمده، با ابزاری به نام UPPAL مورد ارزیابی قرار گرفته‌است [۱۶]. هدف از اجرای این پژوهش، کشف نقاط گلوگاهی و بن‌بست در مدل‌های فرآیندی BPMN و دارای منابع محدود بود؛ در همان سال استفاده از سایر روش‌های صحت‌سنجی نیز مورد توجه قرار گرفت و در نتیجه، روشی توسط وانگ و همکارانش پیشنهاد شد [۱۷]. در این روش، از منطق زمانی خطی (یا منطق زمانی موقت^۶) به منظور پیش‌بینی خطاهای بن‌بست و بن‌بست حلقوی استفاده شده‌است.

یکی از موقعیت‌هایی که ممکن است منجر به ایجاد خطاهای ساختاری در مدل‌های فرآیندی شود، جریان یافتن فرآیند در بین چند سازمان است؛ در چنین حالتی، جریان یافتن تعاملات در بین سازمان‌ها ایجاد شده و فرآیندهای هرکدام از آن‌ها، به صورتی غیربدهی می‌تواند با فرآیند دیگری ادغام گردد. برای کشف خطاهای احتمالی و ناشی از تعاملات بین سازمانی، در سال ۲۰۱۲ تحقیقی توسط فالسیونی و همکاران صورت گرفت [۱۸]؛ آن‌ها با به کارگیری یک الگوریتم بخش‌بندی^۷، توانستند راهکاری را به منظور بررسی وجود اثرات مخرب بر روی فرآیندهای تعاملی^{۱۸} بین سازمانی و در نتیجه، کشف بن‌بست و بن‌بست حلقوی ارائه کنند [۱۸].

در بین روش‌های صحت‌سنجی مدل‌های BPMN، روش‌های مبتنی بر واری مدل^{۱۹} نیز در صحت‌سنجی ساختاری مدل‌های فرآیندی BPMN مورد توجه بوده‌است؛ به عنوان نمونه می‌توان به تحقیقات صورت گرفته توسط خربوشه و همکاران وی در سال‌های ۲۰۱۲ و ۲۰۱۳ اشاره نمود [۶، ۷]؛ در این تحقیقات، مدل فرآیندی BPMN به ساختاری به نام ساختار کریپکی^{۲۰} نگاشته شد و سپس، از منطق زمانی خطی (یا منطق زمانی موقت^{۲۱}) به منظور صحت‌سنجی ساختاری مدل فرآیندی (به منظور اطمینان از عدم وجود بن‌بست، بن‌بست حلقوی و پایان‌های چندگانه) استفاده شده‌است [۱۹]. این

و مفهوم تعمیم یافته‌ی صحت را ارضا می‌کنند و با آن مطابقت دارند؟

این مقاله در ۴ بخش اساسی مقدمه، روش پیشنهادی، یافته‌های پژوهش و نتیجه‌گیری سازماندهی شده‌است. در بخش مقدمه، پس از بیان ضرورت و کلیاتی از موضوع پژوهش (بخش کنونی)، به بررسی پیشینه‌ی پژوهش (مرور کارهای قبلی) و اساسی‌ترین مفاهیم مورد نیاز برای پیشبرد آن پرداخته شده‌است. سپس در بخش دوم (روش پیشنهادی)، به بیان شاخص‌های پیشنهادی برای صحت یک مدل فرآیندی BPMN (در مدل مبدا) می‌پردازیم و در قالب سه گام اساسی، درست بودن این شاخص‌ها را بر مبنای شبکه‌های پتری اثبات می‌کنیم. در بخش سوم (یافته‌های پژوهش)، خطاهای ساختاری را که توسط کارهای مطرح شده در بخش مقدمه (و زیربخش پیشینه‌ی پژوهش) پوشش داده شده‌اند، با ویژگی‌ها و تعاریف پیشنهادی خود تطبیق داده و در خصوص کارایی آن‌ها بحث می‌کنیم و با یک نتیجه‌گیری، این پژوهش به اتمام می‌رسد.

۱-۱- پیشینه‌ی پژوهش

در خصوص صحت‌سنجی مدل‌های فرآیندی BPMN2.0، از روش‌های مختلفی استفاده شده‌است که در هرکدام از آن‌ها، به کشف انواع مختلفی از خطاهای ساختاری پرداخته شده‌است. در بسیاری از تحقیقات صورت گرفته، کشف بن‌بست‌ها، جزو اهداف اصلی صحت‌سنجی ساختاری این مدل‌های فرآیندی بوده‌است. در این بخش قصد داریم مروری بر تحقیقات صورت گرفته در خصوص روش‌های صحت‌سنجی ساختاری مدل‌های فرآیندی BPMN و خطاهای ساختاری پوشش داده شده توسط هرکدام از آن‌ها بپردازیم.

اولین تحقیقات حوزه‌ی صحت‌سنجی مدل‌های فرآیندی BPMN توسط داکمن و همکارانش در سال ۲۰۰۸ صورت گرفت [۱۳]؛ در این تحقیق، یک روش برای نگاشت مدل‌های BPMN به شبکه‌های پتری به منظور شناسایی خطای ساختاری بن‌بست ارائه شد [۱۳]. به دنبال انجام این تحقیق و مطرح شدن شبکه‌های پتری کلاسیک به عنوان یکی از کارآمدترین مدل‌های صحت‌سنجی ساختاری مدل‌های فرآیندی BPMN، استفاده از آن در تحقیقات بعدی ادامه یافت؛ از جمله می‌توان به تحقیق او-یانگ و همکارش در همان سال اشاره نمود [۱۴]. در این تحقیق به بررسی دسترسی‌پذیری^{۱۵}، کشف بن‌بست و نیز کشف بن‌بست حلقوی با استفاده از شبکه‌های پتری پرداخته شده‌است؛ در همان سال تا کمورا و همکارانش به ارائه‌ی یک روش مبتنی بر شبکه‌های پتری به منظور

و انتقال مدل‌های BPMN به مدل مقصد بود؛ برای رفع این مشکل، در سال ۲۰۱۶، نوم و همکاران یک روش مبتنی بر تحلیل احتمالاتی اجزای BPMN به منظور صحت‌سنجی ساختاری آن ارائه کردند [۹]؛ در این پژوهش، بر مبنای قاعده‌ی ضرب احتمالات^{۲۷}، احتمال وقوع بن‌بست در مدل فرآیندی قابل محاسبه است.

در سال ۲۰۲۱ تحقیقی توسط فلاویو کوردینی و همکارانش در این خصوص صورت گرفت. تحقیق ایشان عمدتاً دربرگیرنده‌ی مدل‌های فرآیندی تعاملی^{۲۸} است [۱۰]. همان‌طور که می‌دانیم مدل‌های فرآیندی BPMN می‌توانند حاوی بیش از یک Pool باشند که هر کدام از آن‌ها، در برگیرنده‌ی فرآیندهای یک سازمان هستند. این بخش‌ها می‌توانند با یکدیگر تعامل کنند و در نتیجه، تعامل بین سازمان‌ها نیز مدل‌سازی می‌شود. این تعامل (همان‌طور که در تحقیق فالسیونی و همکاران [۱۸] به آن اشاره شده‌است) می‌تواند به صورتی غیر مستقیم منجر به ایجاد خطاهای ساختاری شود. در این تحقیق یک روش صوری به منظور ارزیابی برقراری سه شاخص مرتبط با صحت ساختاری، به منظور صحت‌سنجی مدل‌های فرآیندی BPMN تعاملی ارائه شده‌است [۱۰]. این سه شاخص عبارتند از صحت^{۲۹}، امنیت^{۳۰} و خوش‌ساختاری^{۳۱}؛ شاخص صحت متضمن اجرا شدن تمام فعالیت‌ها و امنیت متضمن عدم تکرار نامتناهی فعالیت‌ها می‌باشد. در کنار این دو ویژگی، خوش‌ساختار بودن نیز به عنوان یک ویژگی معرفی شده‌است که به معنای جمع شدن شاخه‌های انشعابی توسط یک دروازه اتصالی است. در بخشی از این کار تحقیقاتی، ارتباط بین این سه ویژگی به صورت دو به دو بررسی شد و در پایان آن، فرآیندهای کسب و کار از نظر قابلیت ارضای این سه ویژگی دسته‌بندی می‌شوند [۱۰]. در پایان این مقاله، یک ابزار به منظور صحت‌سنجی مدل‌های فرآیندی BPMN و بر اساس روش صوری پیشنهاد شده، ایجاد گشته و بر روی تعدادی از مدل‌های فرآیندی BPMN و تعاملی اجرا شد. اما در پایان و در نتایج این مقاله، مشخص شده‌است که صحیح نبودن یک فرآیند، لزوماً امن بودن آن را نقض نمی‌کند؛ اما در صورتی که یک مدل فرآیندی خوش‌ساختار باشد، الزاماً صحیح و امن است [۱۰]. لازم به ذکر است که شاخص صحیح بودن در مقاله‌ی کوردینی با شاخص صحیح بودن در این پژوهش، تفاوت‌هایی دارد و خاص منظوره تر است.

همان‌طور که دیده شد، در تحقیقات مختلف، روش‌های مختلفی برای کشف این خطای ساختاری صورت گرفته‌است و در بین تمام خطاهای ساختاری، کشف بن‌بست از اهمیت زیادی برخوردار بود و در وهله‌ی بعدی، کشف بن‌بست حلقوی جزو اهداف برخی از این تحقیقات، از جمله [۱]، [۶]، [۷]، [۸]، [۹] و [۱۰] بوده‌است.

روش دارای زمان اجرای مناسب بوده و با استفاده از آن امکان تعقیب مسیر حاوی خطا، وجود دارد. در سال ۲۰۱۴ در طی پژوهش صورت گرفته توسط پرینز و همکاران [۸] یک زبان میانی برای بازنمایی صوری مدل‌های BPMN معرفی شد. این زبان میانی به گراف جریان^{۳۲} کار موسوم شد و به منظور ایجاد یک کامپایلر برای مدل‌های BPMN مورد استفاده قرار گرفت و بر روی نرم‌افزار Eclipse مستقر شد؛ اگرچه مدل میانی پیشنهاد شده، ساده و قابل فهم است، اما این روش دارای محدودیت بوده و از مدل‌های فرآیندی پیچیده با عناصر متعدد پشتیبانی نمی‌کند [۸]. توسعه‌ی روش‌های مبتنی بر وارسی مدل تا سال‌های اخیر (سال ۲۰۲۳) نیز از محبوبیت زیادی برخوردار بوده‌است؛ در این خصوص می‌توان به تحقیق صورت گرفته توسط نیون و همکارانش اشاره نمود که در سال ۲۰۲۲ صورت گرفت [۱]؛ روش مورد استفاده در این کار، بر مبنای وارسی مدل^{۳۳} است و توسط یک وارسی‌گر مدل^{۳۴} مورد بررسی قرار می‌گیرد. مدل فرآیندی داده شده، توسط وارسی‌گر مدل بررسی می‌شود و اگر ویژگی‌های مورد نظر (صحیح بودن) را ارضا نماید، وارسی‌گر یک نتیجه‌ی TRUE تحویل می‌دهد و در غیر این صورت، یک مثال نقض که منجر به نقض ویژگی مورد نظر می‌شود، توسط آن تولید می‌شود. مثال نقض ایجاد شده، بر مبنای یک مدل صوری به نام LTS^{۳۵} بیان شده‌است که درک آن توسط کاربران عادی ممکن است دشوار باشد؛ چرا که این زبان، یک زبان رسمی برای توصیف مدل است که علاوه بر توصیف مدل‌های BPMN، در توصیف سیستم‌های هم روند نیز مورد استفاده قرار می‌گیرد [۲۰]؛ برای درک بهتر مثال‌های نقض توسط کاربران نهایی، در این کار یک روش رنگ آمیزی پیشنهاد شده‌است که مثال نقض ایجاد شده را (که به وسیله‌ی LTS توصیف گشته‌است) به یک مدل BPMN تبدیل کرده و به صورتی واضح (رنگ آمیزی شده) در اختیار کاربر نهایی قرار می‌دهد.

به گفته‌ی صریح نویسندگان مقاله‌ی [۱]، عمده‌ی هدف آن معطوف بر تصدیق ویژگی زنده بودن (قابل اجرا بودن تمامی فعالیت‌های فرآیند) در مدل‌های فرآیندی BPMN است. با این حال، در این مقاله به مفهوم امن بودن مدل فرآیندی نیز توجه شده‌است [۱].

در بین روش‌های صحت‌سنجی، روش‌های مبتنی بر جبر ریاضی نیز مورد استقبال بودند؛ از جمله‌ی روش‌های مدل‌سازی ریاضی می‌توان به بیان صوری مدل‌های BPMN در زبان جبر ماکس پلاس^{۳۶} و سپس صحت‌سنجی آن اشاره نمود که توسط الحکمی در سال ۲۰۱۵ ارائه شد [۲۱]؛ در این کار، از عدم امکان وقوع بن‌بست و وقوع پایان‌های چندگانه اطمینان حاصل می‌گردد.

یکی از دغدغه‌های اصلی کارهای قبلی، پیچیده بودن فرآیند تبدیل

در این پژوهش، مفهوم صحت ساختاری را در مدل‌های فرآیندی BPMN2.0 بر اساس صحت شبکه‌ی پتری معادلشان بازتعریف می‌کنیم. این کار را با تعمیم مفاهیم زنده بودن و کران‌دار بودن از شبکه‌های پتری به مدل‌های فرآیندی BPMN2.0 انجام خواهیم داد.

۱-۲- مفاهیم اساسی مورد نیاز

هر پژوهش بر مبنای بنیان‌های نظری توسعه داده شده توسط پژوهش‌های پیشین رشد می‌کند و این تحقیق نیز از این قاعده مستثنی نیست؛ بنابراین در این بخش، به بیان بنیان‌ها و نتایجی از تحقیقات پیشین می‌پردازیم که در پژوهش جاری مورد استفاده قرار گرفته‌است.

۱-۲-۱- مفهوم مدل‌های فرآیندی (به صورت عام)

مدل‌های فرآیندی BPMN2.0، شبکه‌های پتری و دیگر زبان‌های مدل‌سازی فرآیندها، دارای یک خصیصه‌ی مشترک هستند: «همه‌ی آن‌ها، یک فرآیند کاری را توصیف می‌کنند»؛ بنابراین، همگی آن‌ها مدل‌هایی فرآیندی^{۳۲} هستند. بنابراین نتیجه می‌گیریم که هر مدل فرآیندی، با زبان‌های مختلفی می‌تواند توصیف شود (مشابه با زبان گفتاری که در آن یک مفهوم واحد می‌تواند به زبان‌ها و گویش‌های مختلفی گفته شود) و این مفهوم، مستقل از زبان مدل‌سازی می‌باشد. به همین منظور، ماتیس در کتاب مدیریت فرآیندهای کسب و کار، با در نظر گرفتن مدل فرآیندی به عنوان یک مفهوم عامی که در زبان‌های مختلفی قابل بیان است، یک تعریف مشخص و جامع را ارائه کرده‌است [۱۲]:

تعریف ۱ (مدل‌های فرآیندی) یک مدل فرآیندی، یک گراف جهت‌دار است مثل $G = (N, E)$ به طوری که E مجموعه‌ی یال‌های آن و N مجموعه‌ی گره‌های آن است که به سه زیرمجموعه افراز می‌شوند:

۱. مجموعه‌ی مدل‌های فعالیتی^{۳۳} که هرکدام از اعضای آن نشان‌دهنده‌ی یک فعالیت در مدل فرآیندی هستند. این زیرمجموعه با N_A نشان داده می‌شود.
۲. مجموعه‌ی مدل‌های رخدادی^{۳۴} که هرکدام از اعضای آن نشان‌دهنده‌ی یک رخداد در مدل فرآیندی هستند. این مجموعه با N_E نشان داده می‌شود.
۳. مجموعه‌ی مدل‌های دروازه‌ای^{۳۵} که هرکدام از اعضای آن نشان‌دهنده‌ی یک دروازه (انشعابی و یا اتصالی) در مدل فرآیندی می‌باشند. این مجموعه با N_G نشان داده می‌شود.

همچنین در برخی دیگر از تحقیقات، به بررسی وجود انواع دیگری از خطاها پرداخته شده‌است؛ به عنوان مثال، در پژوهش [۱۶] به کشف نقاط گلوگاهی در مدل‌های فرآیندی و در پژوهش [۱۴] به کشف حلقه‌های نامتناهی پرداخته شده‌است. همچنین در برخی از پژوهش‌ها مثل [۶]، [۷] و [۹]، وجود حالات پایانی متعدد نیز به عنوان خطاهایی ساختاری در نظر گرفته شده‌است.

همان‌طور که مشاهده شد، در تمامی تحقیقات صورت گرفته، صحت‌سنجی بر مبنای فهرست معینی از خطاهای ساختاری صورت گرفته‌است؛ بنابراین بر مبنای یک تعریف جامع از یک مدل فرآیندی صحیح صورت نگرفته‌است؛ در نتیجه در هرکدام از آن‌ها، ممکن است فقط بخشی از جنبه‌های صحت یک مدل فرآیندی بررسی شده‌باشد و بخشی از جنبه‌های صحت آن، بررسی نگردد؛ به عنوان مثال ممکن است در یکی از روش‌های صحت‌سنجی، در یک مدل فرآیندی BPMN ورودی امکان وقوع بن‌بست کشف شود، اما امکان وقوع بن‌بست حلقوی کشف نشود (درحالی که امکان وقوع بن‌بست حلقوی در آن مدل فرآیندی وجود داشته‌باشد). در نتیجه لازم است که یک تعریف جامع برای یک مدل فرآیندی BPMN صحیح ارائه شود تا جامعیت روش‌های صحت‌سنجی بر مبنای آن سنجیده شود؛ علاوه بر این، همانند تحقیق صورت گرفته توسط نیون و همکاران [۱]، این تعاریف به انتقال بهتر ویژگی‌های صحت ساختاری به کاربر نهایی نیز کمک می‌کند و او را از لزوم درک ویژگی‌های صحت ساختاری در قالب مدل‌های صوری معاف می‌کند؛ چرا که تعاریف پیشنهادی در قالب زبان BPMN بیان می‌شود و در نتیجه صحت ساختاری یک مدل فرآیندی BPMN، بر روی خود مدل قابل درک و بررسی خواهد بود؛ از سوی دیگر، در روش‌های مبتنی بر مهندسی نرم‌افزار مدل رانده، لازم است از درستی روش نگاشت بین مدل‌ها اطمینان حاصل نمود؛ در این رویکرد، نگاشتی درست است که در آن، مدل‌های مبدا و مقصد با یک دیگر هم‌ارز باشند. به عنوان مثال در این خصوص، تحقیقی توسط مغزیلی و همکارانش در سال ۲۰۲۰ صورت گرفت که در آن، روشی به منظور اطمینان از درستی نگاشت مدل‌های فرآیندی BPMN به شبکه‌های پتری رنگی ارائه شد [۱۱]؛ با توجه به ضرورت صحت‌سنجی نگاشت‌های پیشنهادی، لازم است برای اطمینان از صحیح بودن شاخص‌های صحت ساختاری در یک مدل صوری (به عنوان مدل مقصد)، شاخص‌های صحت در مدل مبدا، مشخص بوده و پس از نگاشت به مدل مقصد، با شاخص‌های صحت ساختاری مدل مقصد مقایسه شود تا بدین وسیله، از مناسب بودن نگاشت پیشنهادی اطمینان حاصل شود.

۱-۲-۳- مفهوم صحت (بر اساس شبکه‌های پتری)

بر طبق تعریف ارائه شده توسط آلست و همکارانش [۲۳] یک فرآیند ایجاد شده با استفاده از شبکه‌ی پتری صحیح^{۳۸} است، اگر و تنها اگر دو ویژگی زیر در خصوص آن برقرار باشد:

۱. فرآیند، زنده^{۳۹} باشد.

۲. فرآیند، کران‌دار^{۴۰} باشد.

زوراسکی، در مقاله‌ی آموزشی خود در خصوص شبکه‌های پتری، ویژگی زنده بودن را بدین صورت تعریف می‌کند [۲۴]:

تعریف ۲ (زنده بودن یک شبکه‌ی پتری): «یک شبکه‌ی پتری زنده است، اگر و تنها اگر هر نشانه‌گذاری مثل M که به واسطه‌ی شلیک انتقال‌ها از وضعیت اولیه‌ی M_0 قابل تولید باشد، با شلیک دنباله‌ای از انتقال‌ها بدست آید.»

به بیان ساده‌تر، در [۲۳] زنده بودن بدین صورت تعریف شده است:

تعریف ۳ «یک شبکه‌ی پتری زنده است، اگر و تنها اگر تمامی انتقال‌های آن، در موقعیتی بتوانند فعال شده و شلیک شوند.»

ویژگی زنده بودن برای یک فرآیند، تضمین می‌کند که تمامی فعالیت‌های آن، اجرا می‌شوند و در نتیجه وجود خطاهایی ساختاری مثل بن‌بست^{۴۱}، کارهای مرده^{۴۲} و فعالیت‌های بدون ورودی و یا خروجی، منتفی است.

یکی دیگر از ویژگی‌هایی که برای صحیح بودن یک فرآیند لازم است، کران‌دار بودن است؛ زوراسکی در مقاله‌ی آموزشی خود در خصوص شبکه‌های پتری [۲۴] این ویژگی را به صورت زیر تعریف می‌کند:

تعریف ۴ (کران‌دار بودن یک شبکه‌ی پتری): «یک شبکه‌ی پتری را $k - bound$ (کران‌دار) می‌نامیم اگر و تنها اگر تعداد توکن‌های موجود در آن بیشتر از k نشود.»

این ویژگی بیان می‌کند که برای یک شبکه‌ی پتری مثل $PN = (P, T, F)$ تعداد توکن‌هایی که در یک مکان مثل $p \in P$ تولید می‌شوند، محدود است و به صورت بی‌کران رشد نمی‌کند. با توجه به این که پیش از هر مکان (بجز مکان شروع)، یک انتقال قرار دارد و توکن‌های داخل آن مکان، به واسطه‌ی شلیک انتقال واقع در قبل از آن به وجود می‌آید، کران‌دار بودن بدان معنا خواهد بود که در چنین شبکه‌ای، انتقالی وجود نخواهد داشت که به تعداد دفعات نامتناهی شلیک شود. کران‌دار بودن یک فرآیند، تضمین می‌کند که در آن، فعالیتی وجود ندارد که با تعداد دفعات نامتناهی اجرا شود.

بنابراین، هر فرآیند مدل‌سازی شده با BPMN را می‌توان یک مدل فرآیندی دانست که در آن، مجموعه‌ی مدل‌های فعالیتی، رخدادی و دروازه‌ای متناظر است با فعالیت‌ها، رخدادهای و دروازه‌ها در مدل فرآیندی BPMN2.0. بنابراین، یک مدل فرآیندی BPMN2.0 مثل b می‌تواند به صورت $b = (N, E)$ تعریف شود به طوری که مجموعه‌ی N به سه زیرمجموعه‌ی فعالیت‌ها (N_A) ، رخدادهای (N_E) و دروازه‌ها (N_G) افراز شود. بنابراین خواهیم داشت:

$$N = N_A \cup N_E \cup N_G$$

به گونه‌ای که:

$$N_A \cap N_E \cap N_G = \emptyset$$

۱-۲-۲- شبکه‌های جریان کار

برای تعمیم مفهوم صحت به مدل‌های فرآیندی BPMN2.0، لازم است این مدل فرآیندی یک شبکه‌ی جریان کاری^{۴۳} باشد [۱۷]. یک شبکه‌ی جریان کار، فرآیندی است که دارای دقیقاً یک نقطه‌ی شروع و دقیقاً یک نقطه‌ی پایانی باشد [۱۷]. باید توجه کرد که در صورتی که یک فرآیند دارای ویژگی یک شبکه‌ی جریان کار نباشد، می‌تواند با اعمال تغییراتی به یک شبکه‌ی جریان کار تبدیل شود:

۱. تمامی شاخه‌های منتهی به رخدادهای پایانی، به یک دروازه‌ی انحصاری متصل شود (از نوع اتصالی) و خروجی دروازه، به یک رخداد پایانی متصل شود.
۲. تمامی شاخه‌های آغاز شده از رخدادهای شروع، به واسطه‌ی یک دروازه‌ی انحصاری (انشعابی) به یک رخداد شروع یکتا متصل شود.

با این کارها، فرآیند داده شده به یک شبکه‌ی جریان کار تبدیل خواهد شد. چرا که فرآیند، دقیقاً با یکی از رخدادهای آغازین، شروع شده و دقیقاً با یکی از رخدادهای پایانی، پایان می‌یابد. بنابراین، مفهوم تعمیم یافته‌ی صحت، می‌تواند برای تمامی انواع مختلفی از مدل‌های فرآیندی BPMN2.0 به کار رود.

اگرچه در برخی از کارهای پژوهشی مثل [۶]، [۷] و [۹] وجود پایان‌های چندگانه به عنوان خطاهای ساختاری در نظر گرفته شده است، اما در استاندارد BPMN وجود شروع‌های چندگانه (مشروط بر آن که بجز یک رخداد شروع، سایر رخدادهای آغازین به صورت شروع با تحریک^{۴۴} باشند) و پایان‌های چندگانه مجاز است [۲۲] و چنین ویژگی منجر به نقض صحت ساختاری مدل‌های فرآیندی (توصیف شده با استاندارد BPMN) نمی‌شود.

۴-۲-۱- نگاهت بین مدل‌های فرآیندی BPMN2.0 و

شبکه‌های پتری

در این پژوهش، نگاهت ارائه شده توسط داکمن و همکارانش [۱۳] مبنایی برای صوری‌سازی مدل‌های فرآیندی BPMN2.0 و تعمیم مفهوم صحت از شبکه‌های پتری به مدل‌های BPMN قرار گرفته‌است. در این نگاهت (در جدول ۱ نشان داده شده‌است) هرکدام از ساختارهای مدل‌های فرآیندی BPMN2.0 به یک ساختاری از شبکه‌های پتری نگاهت شده‌است.

این ویژگی، عدم وجود حلقه‌های اجرایی نامتناهی و یا بن‌بست‌های حلقوی را تضمین می‌کند.

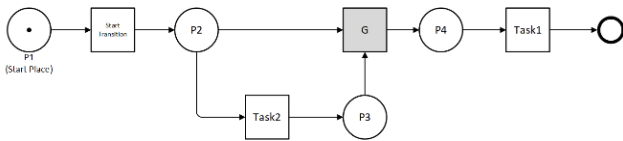
تعریف ۵ (امن بودن یک شبکه‌ی پتری): یک شبکه‌ی پتری امن است، اگر و تنها اگر تعداد توکن‌ها در هر یک از مکان‌های آن بیشتر از ۱ نشود [۲۴].

تعریف ۶ (امن بودن یک مدل فرآیندی): یک مدل فرآیندی امن است، اگر و تنها اگر تعداد موارد کاری فعال برای هرکدام از فعالیت‌ها، بیشتر از ۱ مورد نباشد [۱۰].

جدول ۱. نگاهت ارائه شده توسط داکمن در [۱۳]

معادل عنصر در شبکه‌های پتری	عنصر در زبان BPMN	نوع گره (یا ساختار) از منظر مدل‌های فرآیندی عام
		فعالیتی
		رخدادی
		دروازه‌ای

بن‌بست رخ خواهد داد. حال، این مدل فرآیندی را بر طبق جدول ۱ به یک شبکه‌ی پتری نگاشت می‌دهیم. در نتیجه شکل ۳ بدست می‌آید.

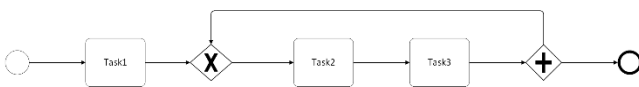


شکل ۳. تبدیل یافته‌ی مدل فرآیندی ترسیم شده در شکل ۲ به شبکه‌ی پتری معادل بر اساس نگاشت داکمن

در شکل ۳ در صورتی که یک توکن از مکان شروع (مکان P_1)، توسط اولین انتقال (انتقال Start Transition) به دومین مکان (مکان P_2) شلیک شود، انتقال Task2 فعال خواهد شد، اما با توجه به این که هنوز توکنی در مکان خروجی Task2 واقع نیست، انتقال G فعال نمی‌شود. سپس با شلیک Task2، توکن از مکان دوم حذف خواهد شد و در مکان خروجی Task2 یک توکن قرار خواهد گرفت. با توجه به این که فقط یکی از ورودی‌های انتقال G فعال خواهد بود، این انتقال هیچگاه فعال نشده و شلیک نخواهد شد و در نتیجه یک بن‌بست در این انتقال رخ می‌دهد. نظر به این که وضعیتی رخ داده‌است که در آن G فعال نشده‌است، در این شبکه‌ی پتری، شرط زنده بودن نقض شده و در نتیجه، این مدل فرآیندی صحیح نمی‌باشد.

۱-۲-۶- بن‌بست حلقوی

تکرار فعالیت‌های یک فرآیند به صورت نامتناهی، یکی از مشکلاتی است که صحت ساختاری آن را تهدید می‌کند. مدل فرآیندی ترسیم شده در شکل ۴ مثالی از وقوع این مشکل در یک مدل فرآیندی BPMN می‌باشد.



شکل ۴. نمونه‌ای از بروز بن‌بست حلقوی در یک مدل فرآیندی BPMN2.0

در شکل ۴ بعد از اجرای فعالیت Task1، دو فعالیت بعدی اجرا می‌شوند. سپس، جریان اجرای فرآیند به یک دروازه‌ی موازی می‌رسد. با موازی‌سازی جریان فرآیند، فرآیند از هر دو مسیر خروجی آن ادامه می‌یابد و در نتیجه، دو فعالیت Task2 و Task1 اجرا می‌شوند؛ سپس یکی از این شاخه‌ها به نقطه‌ی پایان رسیده و ظاهراً فرآیند به اتمام می‌رسد؛ این درحالی است که چنین اتفاقی رخ نمی‌دهد و در حالی که اجرای یکی از شاخه‌ها به پایان رسیده است، شاخه‌ی دیگر خروجی دروازه‌ی موازی، جریان فرآیند را به

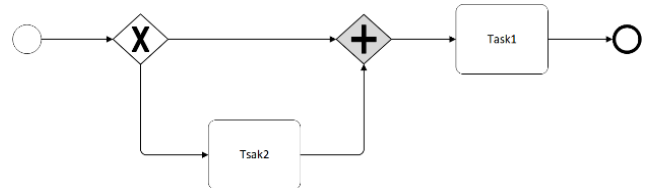
همان‌طور که در جدول ۱ مشاهده می‌شود، انتقال‌هایی که درجه‌ی ورودی و یا خروجی بزرگ‌تر از ۱ دارند، معادل با دروازه‌های موازی هستند و انتقال‌هایی که درجه‌ی ورودی و خروجی برابر با ۱ است، معادل با رخدادها و یا فعالیت‌ها هستند. با تکیه بر این واقعیت که تعریف ۱ در خصوص مدل‌های فرآیندی ایجاد شده با استفاده از استاندارد BPMN و شبکه‌های پتری صادق است (چون هر دو نشان‌دهنده‌ی یک مدل فرآیندی هستند) می‌توان اجزای BPMN و اجزای معادل آن‌ها را به سه مجموعه‌ی مدل‌های فعالیت‌ی، رخدادی و دروازه‌ای تقسیم نمود. البته در خصوص شبکه‌های پتری لازم است کارکرد انتقال از قبل مشخص باشد. این تقسیم بندی در ستون نوع گره (یا ساختار) بر مبنای مدل‌های فرآیندی عام (تعریف ۱) گفته شده‌است.

۱-۲-۵- نقض زنده بودن فرآیند به علت وقوع بن‌بست

آلست ون هی و همکارانش [۲۳] در کتاب مدیریت جریان‌های کاری، برای یک شبکه‌ی پتری مثل $PN = (P, T, F)$ بن‌بست را این‌گونه تعریف کردند:

تعریف ۷ (بن‌بست): «بن‌بست به وضعیتی اطلاق می‌شود که در طی آن، توکن متناظر با یکی از موارد کاری در یکی از مکان‌ها باقی بماند و هیچ کدام از انتقال‌های پس از آن شلیک نشوند؛ در نتیجه امکان خروج توکن از آن مکان به هیچ وجه میسر نخواهد بود.»

به عنوان نمونه، در شکل ۲ یک نمونه از بروز بن‌بست در یک مدل فرآیندی BPMN2.0 قابل مشاهده است.



شکل ۲. نمونه‌ای از بروز خطای ساختاری بن‌بست در یک مدل فرآیندی BPMN

در این شکل با شروع فرآیند و رسیدن جریان فرآیند به دروازه‌ی انحصاری، دقیقاً یکی از شاخه‌های دروازه‌ی انحصاری را فعال می‌کند؛ در نتیجه، جریان فرآیند از یکی از خروجی‌های دروازه‌ی انحصاری ادامه یافته و فقط یکی از ورودی‌های دروازه‌ی موازی (هم‌گام سازی) فعال خواهد شد؛ این درحالی است که ادامه‌ی اجرای فرآیند منوط به فعال شدن تمامی ورودی‌های دروازه‌ی هم‌گام سازی (دروازه‌ی موازی اتصالی) است؛ در نتیجه، در محل این دروازه، یک

۲- روش پیشنهادی

۲-۱- چارچوب کلی راه حل پیشنهادی

رویکرد اصلی در این تحقیق برای ارائه‌ی تعاریف پیشنهادی و نشان دادن درستی آن‌ها، استفاده از اثبات ریاضی است. به منظور تعمیم مفهوم صحت به مدل‌های فرآیندی BPMN2.0، لازم است در ابتدا، ویژگی‌های اساسی یک فرآیندی صحیح، یعنی زنده بودن و کران‌دار بودن را از شبکه‌های پتری به آن تعمیم دهیم؛ بدین وسیله خواهیم دانست که اجزای مدل فرآیندی BPMN به چه صورتی باید در کنار یک دیگر قرار بگیرند تا مدل فرآیندی حاصل صحیح باشد؟ در ادامه، بعد از تعمیم این دو مفهوم به مدل‌های فرآیندی BPMN2.0، یک تعریف جامع برای یک مدل فرآیندی BPMN2.0 صحیح، ارائه خواهد شد. تعمیم این دو مفهوم بر مبنای دو اصل صورت می‌گیرد:

۱. بر اساس تعریف ۱ مدل‌های فرآیندی BPMN و فرآیندهای ایجاد شده به وسیله‌ی شبکه‌های پتری، همگی مدل‌هایی فرآیندی هستند.
۲. در بین مدل‌های فرآیندی BPMN و شبکه‌های پتری، ساختارهایی که دارای الگوهای جریان کنترلی یکسان هستند، با رعایت نگاشت پیشنهاد شده توسط داکمن [۱۳] می‌توانند به یک دیگر نگاشته شوند.

در بخش‌های بعدی، در ابتدا مفهوم کران‌دار بودن و سپس مفهوم زنده بودن را از شبکه‌های پتری به مدل‌های فرآیندی BPMN تعمیم می‌دهیم و با تعمیم این دو ویژگی مفهوم صحت ساختاری را بر اساس این دو مفهوم تعمیم یافته، به مدل‌های فرآیندی BPMN2.0 تعمیم می‌دهیم.

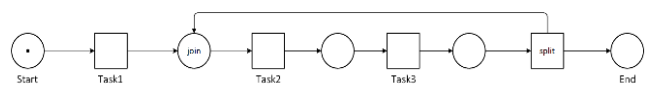
رویکرد پیشنهادی در شکل ۶ به تصویر کشیده شده‌است؛ در این شکل، مستطیل‌های خاکستری نشان‌دهنده‌ی تعاریف مختص شبکه‌های پتری است که تحت نگاشت ارائه شده توسط داکمن به تعاریف پیشنهادی این تحقیق تبدیل شده‌است؛ همچنین مستطیل‌های سفید، تعاریف پیشنهادی ارائه شده توسط این تحقیق را نشان می‌دهد که بر پایه‌ی تعاریف حاکم بر شبکه‌های پتری و نگاشت ارائه شده توسط داکمن ارائه شده‌است.

ورودی دروازه‌ی انحصاری هدایت می‌کند و مجدداً هر دو فعالیت‌های Task2 و Task3 اجرا می‌شوند و این روند ادامه می‌یابد؛ در نتیجه، دو فعالیت Task2 و Task3 به تعداد دفعات نامتناهی اجرا می‌شوند و یک حلقه‌ی اجرایی نامتناهی در فرآیند اجرا می‌شود. در چنین شرایطی در محل دروازه‌ی انشعاب موازی، یک بن‌بست حلقوی رخ داده‌است.

آلست و همکاران [۲۳]، بن‌بست حلقوی را بدین صورت تعریف کرده‌اند:

تعریف ۸ (بن‌بست حلقوی): «بن‌بست حلقوی به معنای به دام افتادن یک مورد کاری (توکن) در یک چرخه‌ی بی‌پایان است.»

در شکل ۵ تبدیل یافته‌ی مدل فرآیندی ترسیم شده در شکل ۴ بر اساس نگاشت ارائه شده در جدول ۱ قابل مشاهده است. در این شکل، یک توکن در مکان شروع قرار دارد. با شلیک این توکن به وسیله‌ی انتقال Task1، فعالیت Task1 اجرا می‌شود. سپس، توکن به ترتیب توسط انتقال‌های Task2 و Task3 شلیک می‌شود تا در نهایت، در مکان خروجی انتقال Task3 (و ورودی انتقال split) واقع می‌شود. با فعال شدن انتقال split و شلیک شدن آن، یک توکن در مکان پایانی و یک توکن دیگر در مکان join قرار می‌گیرد. سپس برای توکن قرارگرفته در مکان join، مجدداً به ترتیب توسط دو انتقال Task2 و Task3 و Split شلیک شده و مجدداً در مکان Join قرار می‌گیرد؛ این چرخه مجدداً و به تعداد دفعات نامتناهی تکرار می‌شود؛ در نتیجه، انتقال split به تعداد دفعات نامتناهی شلیک می‌شود و در نتیجه، تعداد توکن‌های موجود در مکان پایانی، به اندازه‌ی نامتناهی رشد خواهد کرد؛ ضمن این که انتقال‌های Task2 و Task3 نیز به تعداد دفعات نامتناهی شلیک می‌شوند و در نتیجه، این دو فعالیت به تعداد دفعات نامتناهی اجرا خواهند شد.



شکل ۵. تبدیل یافته‌ی مدل فرآیندی ترسیم شده در شکل ۴ بر

اساس نگاشت ارائه شده در جدول ۱

با توجه به این که تعداد توکن‌های موجود در مکان پایانی به تعداد دفعات نامتناهی رشد می‌کند، شرط کران‌دار بودن در این شبکه‌ی پتری نقض می‌شود؛ در نتیجه این شبکه‌ی پتری، یک شبکه‌ی پتری صحیح نخواهد بود.

تعریف ۴) قابل پذیرش است.

حال، اگر یک مدل فرآیندی به وسیله‌ی شبکه‌های پتری مدل‌سازی شود، می‌دانیم که فعالیت‌های آن، بخشی از انتقال‌های آن را تشکیل خواهند داد (طبق جدول ۱)؛ در نتیجه از منظر مدل‌های فرآیندی عام (بحث شده در بخش ۱-۲-۱) و مشخص شده در ستون ابتدایی جدول ۱) بخشی از انتقال‌های یک شبکه‌ی پتری، مدل‌های فعالیت‌ی یک مدل فرآیندی را تشکیل می‌دهند و شلیک آن‌ها به معنای اجرای آن فعالیت می‌باشد؛ از سوی دیگر در زبان مدل‌سازی BPMN این مدل‌های فعالیت‌ی (از دیدگاه عام) در قالب فعالیت‌ها (مستطیل‌ها) نشان داده می‌شوند و همان‌طور که در جدول ۱ نشان داده شده‌است، مدل‌های فعالیت‌ی در مدل‌های فرآیندی BPMN و در شبکه‌های پتری می‌توانند به یکدیگر نگاشته شوند. بنابراین می‌توان مفهوم کران‌دار بودن را در مدل‌های فرآیندی BPMN به صورت زیر تعریف نمود:

تعریف ۹ (کران‌دار بودن در مدل‌های فرآیندی BPMN)
 یک مدل فرآیندی BPMN2.0 را کران‌دار می‌نامیم، اگر و تنها اگر تمامی فعالیت‌های آن (و یا رخداد‌های آن) به تعداد دفعات متناهی اجرا شوند و در نتیجه، فاقد بن‌بست حلقوی و یا خط‌های ناشی از آن، باشند.

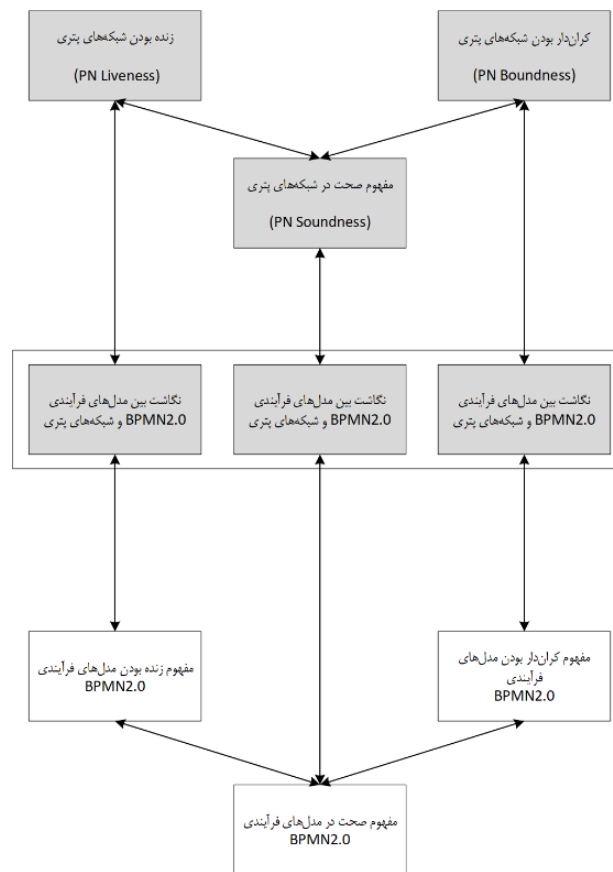
بر اساس این تعریف، نمونه‌هایی از مصادیق بن‌بست حلقوی می‌تواند شامل موارد زیر باشد:

۱. وجود حلقه‌های اجرایی نامتناهی.
۲. وجود بخش‌هایی از یک مورد کاری در سطح فرآیند، پس از اتمام آن.
۳. وجود ارجاع‌هایی به یک مورد کاری پایان یافته در سطح فرآیند.

در ادامه نشان می‌دهیم که صادق بودن تعریف ۹ در خصوص یک مدل فرآیندی BPMN2.0، معادل با کران‌دار بودن شبکه‌ی پتری معادلش است (طبق نگاشت ارائه شده در جدول ۱).

قضیه ۲ یک مدل فرآیندی BPMN2.0، کران‌دار است، اگر و تنها اگر شبکه‌ی پتری معادل با آن، کران‌دار باشد.

برهان: فرض کنید $b = (N_b, E)$ یک مدل فرآیندی BPMN2.0 باشد و $PN = (P, T, F)$ ، شبکه‌ی پتری معادل با آن باشد که در اثر نگاشت ارائه شده در جدول ۱ بدست آمده‌باشد. با توجه به این که هر دو مدل، مدل‌هایی فرآیندی هستند (طبق تعریف ۱) می‌توان گره‌های آن‌ها را به سه زیرمجموعه‌ی مدل‌های فعالیت‌ی (با زیروند



شکل ۶. شمایی از به کارگیری نگاشت ارائه شده توسط [۱۳] به منظور تعمیم مفاهیم کران‌دار بودن، زنده بودن و صحت

۲-۲- تعمیم مفهوم کران‌دار بودن به مدل‌های فرآیندی BPMN

در بخش ۱-۲-۳، یک مدل فرآیندی ایجاد شده به وسیله‌ی یک شبکه‌ی پتری را صحیح دانستیم، اگر و تنها اگر زنده و کران‌دار باشد و در همان بخش گفته شده که یک شبکه‌ی پتری کران‌دار است، اگر و تنها اگر رشد تعداد توکن‌های موجود در تمامی مکان‌های آن محدود باشد؛ این به این معناست که تعداد توکن‌های موجود در مکان‌های آن، به صورت نامتناهی رشد نمی‌کند.

با توجه به این که پیش از هر کدام از مکان‌های شبکه‌های پتری (بجز مکان آغازین و پس از مکان پایانی)، یک انتقال وجود دارد و توکن‌های ایجاد شده در مکان‌های آن، در اثر شلیک انتقال واقع شده در قبل از آن به وجود می‌آیند، می‌توان قضیه‌ی زیر را در این خصوص بیان نمود:

قضیه ۱ یک شبکه‌ی پتری کران‌دار است، اگر و تنها اگر تمامی انتقال‌های آن به تعداد دفعات متناهی شلیک شوند.

درستی این قضیه، بر اساس تعریف کران‌دار بودن [۲۴] (بر اساس

حلقوی و یا تبعات آن باشد. اگر PN کران‌دار نباشد، یک مکان مثل $p \in P$ در آن وجود خواهد داشت که تعداد نامتناهی توکن دریافت می‌کند و این توکن‌ها، در اثر شلیک یک انتقالی مثل $t \in T$ به وجود می‌آیند؛ بنابراین، بر اساس انتقال t موارد زیر ممکن هستند:

- اگر t تبدیل یافته‌ی یک فعالیت مثل $a \in N_A$ و یا رخداد مثل $e \in N_E$ باشد، بدان معنا خواهد بود که فعالیت a و یا رخداد e ، با تعداد دفعات نامتناهی اجرا می‌شوند که با کران‌دار بودن b (طبق جدول ۱) در تناقض است.
- اگر t تبدیل یافته‌ی یک دروازه‌ی موازی موازی مثل $g \in N_G$ باشد، بدان معنا خواهد بود که عمل موازی‌سازی و هم‌گام سازی در b بی نهایت مرتبه اجرا می‌شود و این امر نشان‌دهنده‌ی یک حلقه‌ی نامتناهی اجرایی در b است که با کران‌دار بودن آن، در تناقض است.

بنابراین، فرض خلف باطل و حکم برقرار است.

۲-۳- تعمیم مفهوم زنده بودن به مدل‌های فرآیندی

BPMN

در بخش ۱-۲-۳ گفته شد که یک شبکه‌ی پتری زنده است، اگر و تنها اگر تمامی انتقال‌های آن، در وضعیتی شلیک شوند؛ در چنین شبکه‌ای، انتقالی وجود ندارد که در هیچ وضعیتی شلیک نشود؛ در نتیجه بن‌بست و تمامی خطاهای مشابه آن (نظیر وظایف مرده) در آن اتفاق نمی‌افتد.

نتیجه ۱: در صورتی که یک فرآیند سازمانی ترسیم شده با شبکه‌ی پتری زنده باشد، فعالیتی در آن وجود ندارد که اجرا نشود.

حال، اگر یک مدل فرآیندی به وسیله‌ی شبکه‌های پتری مدل‌سازی شود، انتقال‌های آن بیانگر فعالیت‌های آن فرآیند خواهند بود؛ در نتیجه از منظر مدل‌های فرآیندی عام (بحث شده در بخش ۱-۲-۲-۱) انتقال‌های یک شبکه‌ی پتری، مدل‌های فعالیتی آن مدل فرآیندی را تشکیل می‌دهند و شلیک آن‌ها به معنای اجرای آن فعالیت می‌باشد؛ از سوی دیگر در زبان مدل‌سازی BPMN این مدل‌های فعالیتی (از دیدگاه عام) در قالب فعالیت‌ها (مستطیل‌ها) نشان داده می‌شوند و همان‌طور که در جدول ۱ نشان داده شده‌است، مدل‌های فعالیتی در مدل‌های فرآیندی BPMN و شبکه‌های پتری به یک دیگر نگاشته می‌شوند. بنابراین می‌توان مفهوم زنده بودن را در مدل‌های فرآیندی BPMN به صورت زیر تعریف نمود:

A ، رخدادی (با زیروند E) و دروازه‌ای (با زیروند G) تقسیم نمود. بنابراین خواهیم داشت:

$$N_b = N_A \cup N_E \cup N_G$$

به همین ترتیب نیز می‌توان گره‌های شبکه‌های پتری (که مشتمل بر انتقال‌ها و مکان‌ها هستند) را نیز به صورت زیر افراز نمود:

$$P \cup T = B_A \cup B_E \cup B_G$$

با توجه به این که در شبکه‌های پتری، فعالیت‌ها و رخداد‌های میانی با انتقال‌ها نشان داده می‌شوند (فعالیت‌ها و رخداد‌های میانی، بخشی از انتقال‌ها را نشان می‌دهند) برای فعالیت‌ها خواهیم داشت:

$$B_A \subseteq T$$

و برای رخداد‌های میانی و آغازین نیز خواهیم داشت:

$$B_E - \{p_{start}, p_{end}\} \subseteq T$$

p_{start} و p_{end} به ترتیب مکان‌های آغازین و پایانی هستند) در نتیجه، به طور کلی برای مدل‌های رخدادی در شبکه‌ی پتری PN خواهیم داشت:

$$B_E \subseteq T \cup P$$

در خصوص مدل‌های دروازه‌ای در شبکه‌های پتری، طبق جدول ۱ مشاهده کردیم که برخی از دروازه‌ها با انتقال‌ها و برخی دیگر با مکان‌ها مدل‌سازی می‌شوند؛ بنابراین در خصوص مدل‌های دروازه‌ای داریم:

$$B_G \subseteq T \cup P$$

به عنوان فرض خلف، فرض کنید که مدل b (یک مدل فرآیندی BPMN است) بر طبق تعریف ۹ کران‌دار باشد (یعنی بر طبق تعریف ارائه شده، هر کدام از فعالیت‌های آن به تعداد دفعات متناهی اجرا شود و در نتیجه، فاقد بن‌بست حلقوی و خطاهای مشتق شده از آن باشد) اما PN کران‌دار نباشد. در نتیجه، مکانی مانند $p \in P$ در PN وجود خواهد داشت، به طوری که که تعداد توکن‌هایی که به آن وارد می‌شود، تا حد بی‌کران رشد می‌کند. بنابراین، یک انتقال مانند $t \in T$ وجود خواهد داشت که پیش از مکان p قرار دارد و با شلیک شدن به تعداد دفعات نامتناهی، توکن‌هایی را به این مکان وارد می‌کند. اگر این انتقال تبدیل یافته‌ی یک رخداد و یا یک فعالیت باشد (طبق جدول ۱)، در مدل فرآیندی b یک فعالیتی مثل $a \in N_A$ و یا رخدادی مثل $e \in N_E$ وجود خواهد داشت که به تعداد دفعات نامتناهی اجرا شده‌است. در نتیجه، مدل فرآیندی b کران‌دار نخواهد بود. در نتیجه فرض خلف باطل و حکم برقرار می‌شود.

برای اثبات سمت دیگر قضیه، به عنوان فرض خلف، فرض کنید که PN کران‌دار نباشد و در عین حال، b کران‌دار باشد و فاقد بن‌بست

تعریف ۱۰ یک مدل فرآیندی BPMN2.0 را زنده می‌نامیم، اگر و تنها اگر تمامی فعالیت‌های آن اجرا شده و نیز رخدادهای آن قابل اجرا باشد و در نتیجه، فاقد بن‌بست و یا تبعات ناشی از آن باشد. تبعات ناشی از بن‌بست، می‌تواند شامل موارد زیر باشد [۲۳]:

۱. وجود وظایف مرده (وظایفی که هرگز اجرا نمی‌شوند).
۲. وجود فعالیت‌هایی که ورودی و یا خروجی نباشند.

در ادامه، نشان می‌دهیم که تعریف ۱۰، تعریفی درست است؛ برای این منظور نشان می‌دهیم که یک مدل فرآیندی BPMN زنده است، اگر و تنها اگر شبکه‌ی پتری معادل آن (تحت نگاشت ارائه شده در جدول ۱) زنده باشد.

قضیه ۳ یک مدل فرآیندی BPMN2.0 (طبق تعریف ۱۰) زنده است، اگر و تنها اگر شبکه‌ی پتری معادل با آن، زنده باشد.

برهان: فرض کنید $b = (N_b, E)$ یک مدل فرآیندی BPMN2.0 باشد و $PN = (P, T, F)$ شبکه‌ی پتری معادل با آن باشد که تحت نگاشت ارائه شده در جدول ۱ بدست آمده باشد.

با توجه به این که هر دو مدل، مدل‌هایی فرآیندی هستند می‌توان گره‌های آن‌ها را به سه زیرمجموعه‌ی مدل‌های فعالیت‌ی (با زیروند A)، رخدادی (با زیروند E) و دروازه‌ای (با زیروند G) تقسیم نمود. بنابراین خواهیم داشت:

$$N_b = N_A \cup N_E \cup N_G$$

به همین ترتیب نیز می‌توان گره‌های شبکه‌های پتری (که مشتمل بر انتقال‌ها و مکان‌ها هستند) را نیز به صورت زیر افراز نمود:

$$P \cup T = B_A \cup B_E \cup B_G$$

با توجه به این که در شبکه‌های پتری، فعالیت‌ها و رخداد‌های میانی با انتقال‌ها نشان داده می‌شوند برای فعالیت‌ها خواهیم داشت:

$$B_A \subseteq T$$

و برای رخداد‌های میانی و آغازین نیز خواهیم داشت:

$$B_E - \{p_{start}, p_{end}\} \subseteq T$$

p_{end} و p_{start} به ترتیب مکان‌های آغازین و پایانی هستند. در نتیجه، به طور کلی برای مدل‌های رخدادی در شبکه‌ی پتری PN خواهیم داشت:

$$B_E \subseteq T \cup P$$

در خصوص مدل‌های دروازه‌ای در شبکه‌های پتری، طبق جدول ۱ مشاهده کردیم که برخی از دروازه‌ها با انتقال‌ها و برخی دیگر با مکان‌ها مدل‌سازی می‌شوند؛ بنابراین در خصوص مدل‌های دروازه‌ای

داریم:

$$B_G \subseteq T \cup P$$

به عنوان فرض خلف، فرض کنید که مدل b زنده باشد (یعنی بر طبق تعریف ۱۰ فاقد بن‌بست و یا تبعات آن باشد؛ بنابراین تمامی فعالیت‌های آن قابل اجرا بوده و تمامی رخداد‌های آن نیز قابل وقوع خواهد بود) اما PN زنده نباشد؛ بنابراین، انتقالی مثل $t \in T$ در آن وجود خواهد داشت، که شلیک نمی‌شود. در این صورت:

- اگر t تبدیل یافته‌ی یک فعالیت مثل $a \in N_A$ باشد (یعنی $t \in B_A \subseteq T$) و یا یک رخداد مثل $e \in N_E$ در b باشد (در این صورت $t \in B_E \subseteq T \cup P$)، فعالیت a و یا رخداد e در b وجود خواهد داشت، به طوری که اجرا نمی‌شود و این اتفاق، با فرض زنده بودن b در تناقض است.
- اگر t تبدیل یافته‌ی یک دروازه‌ی موازی (انشعابی و یا اتصالی) باشد (در این صورت $t \in B_G \subseteq T \cup P$)، دروازه‌ای موازی مثل $g \in N_G$ در مدل فرآیندی b وجود خواهد داشت، به طوری که جریان اجرای فرآیند، هیچ‌گاه از آن عبور نخواهد کرد. در نتیجه، b حاوی یک بن‌بست است و با فرض زنده بودن آن در تناقض است.

از سوی دیگر، برای اثبات سمت دیگر قضیه، به عنوان فرض خلف، فرض کنید که PN زنده باشد، اما b زنده نباشد. در این صورت بر طبق جدول ۱ داریم:

- فعالیت‌ی مثل $a \in N_A$ در b وجود خواهد داشت، به طوری که اجرا نمی‌شود. در نتیجه، انتقالی مثل $t \in B_A \subseteq T$ (معادل با آن فعالیت) در شبکه‌ی PN وجود خواهد داشت که تبدیل یافته‌ی آن فعالیت است و شلیک نمی‌شود و این اتفاق، با زنده بودن شبکه‌ی PN در تناقض است. در نتیجه فرض خلف باطل و حکم، برقرار است.

- رخدادی مثل $e \in N_E$ در b وجود خواهد داشت، به طوری که اجرا نمی‌شود. در نتیجه، انتقالی مثل $t \in B_E \subseteq T \cup P$ (معادل با آن رخداد) در شبکه‌ی PN وجود خواهد داشت که تبدیل یافته‌ی آن فعالیت است و شلیک نمی‌شود و این اتفاق، با زنده بودن شبکه‌ی PN در تناقض است. در نتیجه فرض خلف باطل و حکم، برقرار است.

- دروازه‌ی اتصال موازی مثل $g \in N_G$ در b وجود خواهد داشت، به طوری که حداقل یکی از ورودی‌های آن فعال نمی‌شود؛ در نتیجه، انتقالی مثل $t \in B_G \subseteq T \cup P$ (معادل با آن رخداد) در شبکه‌ی PN وجود خواهد داشت که تبدیل یافته‌ی آن دروازه

برهان: به عنوان برهان خلف، فرض کنید b صحیح نباشد، اما PN صحیح باشد. با توجه به این که b صحیح نیست، برطبق قضیه ۲ و قضیه ۳ حداقل یکی از شرایط کران‌دار بودن و زنده بودن برای آن برقرار نیست؛ در نتیجه بر طبق قضیه ۲ و قضیه ۳ برای PN نیز حداقل یکی از شرایط کران‌دار و زنده بودن، نباید برقرار باشد و در نتیجه، PN نباید صحیح باشد که با فرض خلف در تناقض است. در نتیجه، فرض خلف باطل و حکم برقرار خواهد بود.

از سوی دیگر، فرض کنید PN یک شبکه‌ی پتری صحیح باشد، اما مدل فرآیندی b صحیح نباشد؛ برطبق قضیه ۲ و قضیه ۳، PN صحیح و کران‌دار خواهد بود، درحالی که b زنده یا کران‌دار نیست؛ درنتیجه، مدل فرآیندی b صحیح نخواهد بود. در نتیجه فرض خلف باطل و حکم برقرار خواهد شد. □

۳- یافته‌های پژوهش

در بخش قبل موفق شدیم در قالب تعریف ۱۱ تعریفی را برای صحت ساختاری مدل‌های فرآیندی BPMN ارائه کنیم. یکی از موارد حائز اهمیت در خصوص این تعاریف، بیان آن‌ها بر اساس استاندارد BPMN است. بنابراین ارائه دهندگان روش‌های صوری پیشنهادی می‌توانند معیارهای صحت ساختاری خود را با این تعاریف تطبیق دهند و از کامل بودن معیارهای خود اطمینان حاصل کنند.

۳-۱- تعریف صحت ساختاری بر مبنای مدل‌های

BPMN

همان‌طور که در بخش قبل تشریح کردیم، صحیح بودن یک مدل فرآیندی BPMN مستلزم برقرار بودن دو شرط زنده بودن و کران‌دار بودن آن است؛ بنابراین، روش‌های پیشنهاد شده توسط تحقیقاتی قابل اتکا هستند که هر دو جنبه‌ی زنده بودن و کران‌دار بودن را به صورتی کلی (و بدون تاکید بر خطاهای خاص) مورد بررسی قرار داده باشند.

۳-۱-۱- بیان مفهوم زنده بودن یک مدل فرآیندی به زبان

BPMN

طبق تعریف ۱۰، زنده بودن یک مدل فرآیندی BPMN به معنای آن است که تمامی رخداد‌های میانی و فعالیت‌های آن، قابلیت اجرا شدن داشته‌باشند؛ درستی این تعریف، بر مبنای تعریف زنده بودن در شبکه‌های پتری (تعریف ۲) در قالب قضیه ۳ اثبات گشت و در نتیجه‌ی آن، مشاهده شد که زنده بودن یک مدل فرآیندی BPMN (بر اساس تعریف پیشنهادی) معادل است با زنده بودن شبکه‌ی پتری معادل آن. در مقایسه با تحقیق صورت گرفته توسط

است و شلیک نمی‌شود و این اتفاق، با زنده بودن شبکه‌ی PN در تناقض است. در نتیجه فرض خلف باطل و حکم، برقرار است.

۲-۴- تعمیم مفهوم صحت به مدل‌های فرآیندی

BPMN

در بخش‌های قبل، دو مفهوم زنده بودن و کران‌دار بودن را برای مدل‌های فرآیندی BPMN2.0، تعمیم دادیم و نشان دادیم که این دو مفهوم، معادل با مفاهیم کران‌دار بودن و زنده بودن شبکه‌های پتری معادل با آن مدل فرآیندی است؛ وجود هر یک از این خواص در مدل‌های فرآیندی BPMN2.0، متضمن عدم وجود بخشی از خطاهای ساختاری در آن خواهد بود. برای تعمیم مفاهیم زنده بودن و کران‌دار بودن، از مفهوم علم مدل‌های فرآیندی و نیز، از نگاشت ارائه شده توسط داکمن [۱۳] استفاده کردیم.

همان‌طور که در بخش ۱-۲-۳- مشاهده شد، صحت یک مدل فرآیندی ترسیم شده با استفاده از شبکه‌های پتری، مستلزم برقراری دو شرط زنده بودن و کران‌دار بودن است. در بخش‌های ۲-۲- و ۲-۳- به ترتیب مفاهیم کران‌دار بودن و زنده بودن را با استفاده از نگاشت ارائه شده توسط داکمن و نیز مفهوم عام مدل‌های فرآیندی، به مدل‌های BPMN تعمیم دادیم و ضمن ارائه‌ی تعاریف مجزا برای کران‌دار بودن و زنده بودن مدل‌های فرآیندی BPMN، در قضیه ۲ و قضیه ۳ نشان دادیم کران‌دار بودن و زنده بودن یک مدل فرآیندی، با کران‌دار بودن و زنده بودن شبکه‌ی پتری معادل با آن، معادل است. حال، لازم است مفهوم نهایی صحت را نیز بر مبنای تعاریف زنده بودن و کران‌دار بودن، به مدل‌های فرآیندی BPMN تعمیم دهیم. بنابراین تعریف زیر را برای یک مدل فرآیندی BPMN صحیح پیشنهاد می‌کنیم:

تعریف ۱۱ یک مدل فرآیندی BPMN2.0 صحیح است، اگر و تنها اگر کران‌دار (طبق تعریف ۹) و زنده (طبق تعریف ۱۰) باشد.

تعریف ۱۱ بیان می‌دارد که یک مدل فرآیندی BPMN2.0 سالم است، اگر و تنها اگر فاقد بن‌بست و خطاهای ناشی از آن و نیز، فاقد بن‌بست حلقوی و خطاهای ناشی از آن باشد. به منظور اثبات این ادعا، قضیه‌ی زیر را مطرح می‌کنیم:

قضیه ۴ فرض کنید که مدل فرآیندی b یک مدل فرآیندی BPMN2.0 و $PN = (P, T, F)$ ، شبکه‌ی پتری معادل آن، بر طبق نگاشت ارائه شده در جدول ۱ باشد. در این صورت b صحیح است، اگر و تنها اگر PN صحیح باشد.

تمامی فعالیت‌های (و رخدادهای) موجود در آن مدل است؛ از سوی دیگر، ویژگی‌هایی مثل کشف حلقه‌ی بی‌نهایت، کشف بن‌بست حلقوی و امن بودن، ناظر بر برقراری ویژگی کران‌دار بودن مدل فرآیندی است؛ چرا که برقراری آن‌ها مستلزم اجرای هرکدام از فعالیت‌ها (و رخدادها) به تعداد دفعات متناهی است. در جدول ۲ خطاهای ساختاری و یا شاخص‌های پیشنهادی صحت ساختاری توسط تحقیقات پیشین، به همراه ارجاع به آن تحقیقات فهرست شده‌اند و مشخص شده‌است که هرکدام از آن‌ها محقق‌کننده‌ی کدام یک از تعاریف زنده بودن و یا کران‌دار بودن (تعریف ۱۰ و تعریف ۹) هستند.

جدول ۲. طبقه‌بندی قابلیت‌های الگوریتم‌های صحت‌سنجی بر مبنای تعاریف پیشنهادی زنده بودن و کران‌دار بودن

ردیف	شاخص / خطای ساختاری مورد بحث	شاخص مرتبط بر اساس تعاریف پیشنهادی	تحقیقات مرتبط
۱	امن بودن	کران‌دار بودن	[۱۰]، [۱]
۲	کشف بن‌بست حلقوی	کران‌دار بودن	[۱۴]، [۶]، [۱۸]، [۶]، [۷]
۳	کشف حلقه‌های بی‌نهایت	کران‌دار بودن	[۱۴]
۴	کشف نقاط گلوگاهی	زنده بودن	[۱۶]
۵	بررسی دسترسی‌پذیری	زنده بودن	[۱۳]، [۱۵]، [۱۴]، [۱۷]
۶	بررسی احتمال وقوع بن‌بست	زنده بودن	[۹]
۷	کشف بن‌بست	زنده بودن	[۱۰]، [۱]، [۱۳]، [۱۴]، [۱۷]، [۱۶]، [۶]، [۱۸]، [۶]، [۷]، [۸]، [۲۵]

۳-۳-۳- ارزیابی روش‌های پیشنهادی صحت‌سنجی ساختاری بر اساس تعاریف پیشنهادی

در جدول ۳ فهرستی از تحقیقات صورت گرفته در حوزه‌ی صحت‌سنجی مدل‌های فرآیندی BPMN وجود دارد؛ در این جدول مشخص شده‌است که هرکدام از کارهای قبلی با در نظر گرفتن کدام یک از خطاهای ساختاری و یا کدام یک از شاخص‌های صحت ساختاری، اقدام به صحت‌سنجی مدل‌های فرآیندی کرده‌اند؛ چنان‌چه حداقل یکی از ویژگی‌های کران‌دار بودن و حداقل یکی از ویژگی‌های زنده بودن در خصوص یک روش صحت‌سنجی قابل بررسی باشد، بر طبق تعریف ۱۱ و قضیه ۴، تمامی جنبه‌های صحت (یعنی زنده بودن و کران‌دار بودن مدل فرآیندی) توسط آن روش پیشنهادی، پوشش داده شده‌است (و این موضوع در آخرین ستون

فلاویو کوردینی [۱۰] نیز می‌توان مشاهده کرد که او، این شاخص را (با نام شاخص صحت) جزو نتایج خوش‌ساخت بودن مدل فرآیندی BPMN در نظر گرفته‌است.

۳-۱-۲- بیان مفهوم کران‌دار بودن یک مدل فرآیندی به زبان BPMN

طبق تعریف ۹ کران‌دار بودن یک مدل فرآیندی BPMN به معنای اجرای متناهی هرکدام از رخدادهای میانی و فعالیت‌های آن است؛ درستی این تعریف بر مبنای تعریف امن بودن و کران‌دار بودن شبکه‌های پتری (تعریف ۴ و تعریف ۵) در قالب قضیه ۲ مورد اثبات قرار گرفت و در نتیجه‌ی آن، مشاهده شد که کران‌دار بودن یک مدل فرآیندی BPMN (بر اساس تعریف پیشنهادی) معادل است با کران‌دار بودن تبدیل یافته‌ی آن به شبکه‌ی پتری معادل. در مقایسه با تحقیق صورت گرفته توسط فلاویو کوردینی [۱۰] نیز می‌توان مشاهده کرد که او، این شاخص (شاخص امن بودن که حالت خاصی از شاخص کران‌دار بودن است) را جزو نتایج خوش‌ساخت بودن مدل فرآیندی BPMN در نظر گرفته‌است.

۳-۱-۳- صحیح بودن یک مدل فرآیندی BPMN

طبق تعریف ۱۱ یک مدل فرآیندی BPMN صحیح است، اگر و تنها اگر کران‌دار و زنده باشد؛ همچنین درستی این تعریف در قالب قضیه ۴ اثبات گشت؛ بر اساس اثبات صورت گرفته در این قضیه، صحیح بودن یک مدل فرآیندی BPMN با صحیح بودن شبکه‌ی پتری معادل با آن هم‌ارز است. در این صورت بر طبق تعریف ۹ (کران‌دار بودن مدل‌های فرآیندی BPMN)، مدل فرآیندی فاقد فعالیت و یا رخدادهایی است که به تعداد دفعات نامتناهی اجرا شود؛ همچنین بر طبق تعریف ۱۰ (کران‌دار بودن مدل‌های فرآیندی BPMN) تمامی فعالیت‌ها و رخدادهای میانی این مدل‌های فرآیندی، در موقعیتی قابل اجرا خواهند بود.

۳-۲- طبقه‌بندی خطاهای ساختاری و شاخص‌های صحت ارائه شده توسط تحقیقات پیشین

با بررسی خطاهای ساختاری مورد بررسی در تحقیقات صورت گرفته در حوزه‌ی صحت‌سنجی مدل‌های فرآیندی BPMN2.0، می‌توان مشاهده کرد که بخشی از ویژگی‌های مورد بررسی در صحت‌سنجی ساختاری، مثل قابلیت کشف بن‌بست، بررسی احتمال وقوع بن‌بست، بررسی دسترسی‌پذیری و کشف نقاط گلوگاهی، ناظر بر زنده بودن مدل‌های فرآیندی هستند؛ چرا که برقرار بودن آن‌ها مستلزم اجرای

تحقیقات صورت گرفته توسط فلاویو کوردینی و نیون رعایت شده‌است و زنده بودن در حالت کلی در نظر گرفته شده‌است؛ همچنین، در این دو کار تحقیقاتی شاخص امن بودن نیز در نظر گرفته شده‌است که حالت خاصی از کران‌دار بودن است (طبق تعریف ۵ و تعریف ۶).

جدول ۳ بیان شده‌است؛ اما این موضوع به این معنا نیست که آن روش، صحت ساختاری را به صورت ۱۰۰ درصد می‌تواند تضمین کند؛ چرا که ممکن است ویژگی‌های مورد بررسی، فقط بخشی از خطاهای ساختاری ناقص زنده بودن و یا کران‌دار بودن را پوشش دهد؛ بنابراین روش‌هایی به صورت قطعی قابل اتکا هستند که کران‌دار بودن و زنده بودن را در حالت کلی و بر طبق تعریف ۹ و تعریف ۱۰ تضمین نمایند؛ این موضوع تا حد خیلی خوبی در

جدول ۳. ارزیابی خطاهای مورد بررسی در تحقیقات پیشین بر مبنای تعاریف پیشنهادی

آیا تمامی جنبه‌های صحت پوشش داده شده است؟	بررسی کران‌دار بودن			بررسی زنده بودن			روش پیشنهاد شده
	امن بودن	کشف بن‌بست حلقوی	کشف حلقه‌های بی‌نهایت	کشف نقاط گلوگاهی	بررسی دسترسی پذیری	بررسی احتمال وقوع بن‌بست	
خیر					*		داکمن (۲۰۰۸): استفاده از شبکه‌های پتری کلاسیک [۱۳]
خیر					*		تاکمورا (۲۰۰۸): استفاده از شبکه‌های پتری کلاسیک برای صحت‌سنجی تراکنش‌های ایجاد شده با استاندارد BPMN [۱۵]
بله		*		*	*		او یانگ (۲۰۰۸): استفاده از شبکه‌های پتری کلاسیک [۱۴]
خیر					*		وانگ (۲۰۱۱): استفاده از منطق زمانی خطی (LTL) [۱۷]
خیر				*			واتاهیکی (۲۰۱۱): واری مدل و سپس تصدیق آن با ابزار UPPAL با فرض وجود محدودیت زمانی و منابع در فرآیند [۱۶].
بله		*					خربوشه (۲۰۱۲): استفاده از واری مدل، نگاشت مدل فرآیندی به ساختار کریپیکی و استفاده از منطق زمانی خطی (LTL) [۶].
بله		*					فالسینونی (۲۰۱۲): ارائه‌ی یک الگوریتم Unfolding برای بررسی اثرات مخرب فرآیند [۱۸].
بله		*					خربوشه (۲۰۱۳): بهبود تحقیق پیشین (تحقیق [۶]) [۷].
خیر							پرینز (۲۰۱۴): ارائه‌ی یک مدل به نام گراف جریان کار به منظور ایجاد یک کامپایلر برای مدل‌های فرآیندی BPMN [۸].
خیر							الحکمی و نوم (۲۰۱۵): استفاده از ساختار جبری ماکس پلاس [۲۵].
خیر						*	نوم و الحکمی (۲۰۱۷): یک روش صحت‌سنجی و مبتنی بر ضرب احتمالات [۹]
بله	*			این کار تحقیقاتی بر اساس انواع خاصی از خطاهای ساختاری پیش برده نشده‌است و ابعاد صحت به صورت صوری و با تعاریفی کلی مورد ارزیابی قرار می‌گیرد.			فلاویو کوردینی (۲۰۲۱): در پایان و در نتایج این مقاله، مشخص شده‌است که صحیح نبودن یک فرآیند، لزوماً امن بودن آن را نقض نمی‌کند؛ اما در صورتی که یک مدل فرآیندی خوش‌ساختار باشد، الزاماً صحیح و امن است [۱۰].
بله	*			این کار تحقیقاتی بر اساس انواع خاصی از خطاهای ساختاری پیش برده نشده‌است و ابعاد صحت به صورت صوری و با تعاریفی کلی مورد ارزیابی قرار می‌گیرد.			نیون و همکاران (۲۰۲۲): عمده‌ی هدف این مقاله معطوف بر تصدیق ویژگی زنده بودن در مدل‌های فرآیندی BPMN است. با این حال، در این مقاله به مفهوم امن بودن مدل فرآیندی نیز توجه شده‌است که در این صورت، به تصدیق ویژگی کران‌دار بودن نیز منجر می‌گردد؛ در نتیجه بر اساس شاخص‌های پیشنهادی ما، این کار تحقیقاتی نیز تمامی جنبه‌های صحت ساختاری را پوشش می‌دهد [۱].

۴- نتیجه‌گیری

در این مقاله، معنای یک مدل فرآیندی صحیح را در زبان BPMN تعریف کردیم (تعریف ۱۱)؛ صحیح بودن یک مدل فرآیندی BPMN مستلزم زنده بودن و کران‌دار بودن آن است. زنده بودن یک مدل فرآیندی BPMN متضمن اجرای تمامی فعالیت‌ها و رخدادهای آن است و کران‌دار بودن آن متضمن اجرای تمامی فعالیت‌ها و رخدادهای آن به تعداد دفعات متناهی است. درستی این تعاریف در قالب قضیه ۳ و قضیه ۲ بر پایه‌ی شبکه‌های پتری و مفهوم عام مدل‌های فرآیندی اثبات شد و در نهایت در قضیه ۴ اثبات شد که تعریف پیشنهادی در خصوص صحت مدل‌های فرآیندی BPMN با مفهوم صحت در معادل آن مدل فرآیندی در شبکه‌های پتری، هم‌ارز است. در پایان، شاخص‌های مورد بررسی توسط تحقیقات پیشین را بر اساس تعاریف پیشنهادی زنده بودن و کران‌دار بودن در قالب جدول ۲ طبقه‌بندی کرده و بر مبنای آن، در جدول ۳ تحقیقات پیشین بر مبنای پوشش دادن دو ویژگی زنده بودن و کران‌دار بودن، دسته‌بندی شدند. در نتیجه‌ی آن می‌توان مشاهده کرد که بسیاری از تحقیقات پیشین منحصرأ یکی از جنبه‌های یک مدل فرآیندی صحیح را پوشش داده‌اند و در نتیجه، نمی‌توانند صحت ساختاری یک مدل فرآیندی BPMN را تضمین کنند؛ اما در تحقیقات اخیر همانند تحقیقات فلاویو کوردینی [۱۰] و نیون [۱] ضمن پوشش هر دو جنبه‌ی صحت ساختاری، این دو شاخص در حالت کلی و فارغ از نوع خطای ساختاری مد نظر قرار گرفته‌است؛ در نتیجه بر مبنای تعاریف پیشنهادی در این پژوهش، روش پیشنهادی در این دو تحقیق می‌تواند صحت ساختاری این مدل‌های فرآیندی را تضمین نماید.

سپاسگزاری

از دست اندرکاران برگزاری پنجمین همایش ملی معماری سازمانی (پاییز ۱۴۰۰ در دانشگاه فردوسی مشهد و با نظارت آزمایشگاه معماری سازمانی دانشگاه شهید بهشتی) بابت داوری و تایید نسخه‌ی اولیه‌ی این مقاله [۲۶] سپاس‌گزاریم و در پایان از اساتید ارجمند، جناب آقای دکتر کمندی (استادیار گروه علوم مهندسی دانشگاه تهران) و جناب آقای دکتر اکبر مصطفوی (استادیار گروه مهندسی کامپیوتر دانشگاه یزد) بابت بیان نکات ارزشمندشان در نشست ارائه‌ی مقاله، سپاس‌گزاریم.

مراجع

- Cham, S. L. Tapia Tarifa and J. Proença, Eds., 2022// 2022: Springer International Publishing, pp. 90-109.
- [2] F. Huang, F. Ni, J. Liu, F. Yang, and J. Zhu, "A Colored Petri Net Executable Modeling Approach for a Data Flow Well-Structured BPMN Process Model," *IEEE Access*, vol. 10, pp. 86696-86709, 2022, doi: 10.1109/ACCESS.2022.3198969.
- [3] Y. Poursoltani, M. H. Shirali-Shahreza, and S. A. Hashemi Golpayegani, "Using Low-Code Development Platforms for developing Commercial Software based on Business Processes (wrote In Persian)," presented at the Conference: 26th International Computer Conference, Computer Society of Iran, Iran - Tehran, 1399 (In Persian). [Online]. Available: <https://civilica.com/doc/1203571>.
- [4] R. Waszkowski, "Low-code platform for automating business processes in manufacturing," *IFAC-PapersOnLine*, vol. 52, no. 10, pp. 376-381, 2019/01/01/ 2019, doi: <https://doi.org/10.1016/j.ifacol.2019.10.060>.
- [5] S. Braams, "Developing a Software Quality Framework for Low-Code Model Driven Development Platforms Based on Behaviour Driven Development Methodology.," presented at the 27th Twente Student Conference on IT, Enschede, The Netherlands., July 7th, 2017, 2017.
- [6] O. M. Kherbouche, A. Ahmad, and H. Basson, "Detecting structural errors in BPMN process models," in *2012 15th International Multitopic Conference (INMIC)*, 2012: IEEE, pp. 425-431.
- [7] O. M. Kherbouche, A. Ahmad, and H. Basson, "Using model checking to control the structural errors in BPMN models," *IEEE 7th International Conference on Research Challenges in Information Science (RCIS)*, pp. 1-12, 2013.
- [8] T. M. Prinz, N. Spieß, and W. Amme, "A First Step towards a Compiler for Business Processes," Berlin, Heidelberg, 2014: Springer Berlin Heidelberg, in *Compiler Construction*, pp. 238-243.
- [9] M. Naum, O. E. Hichami, M. A. Achhab, and B. E. E. Mohajir, "A probabilistic method for business process verification: Reachability, Liveness and deadlock detection," *2016 4th IEEE International Colloquium on Information Science and Technology (CiSt)*, pp. 128-132, 2016.
- [10] F. Corradini, A. Morichetta, C. Muzi, B. Re, and F. Tiezzi, "Well-structuredness, safeness and soundness: A formal classification of BPMN collaborations," *Journal of Logical and Algebraic Methods in Programming*, vol. 119, p. 100630, 2021/02/01/ 2021, doi: <https://doi.org/10.1016/j.jlamp.2020.100630>.
- [11] S. Meghzili, A. Chaoui, M. Strecker, and E. Kerkouche, "An Approach for the Transformation and Verification of BPMN Models to Colored Petri Nets Models," *International Journal of Software Innovation (IJSI)*, vol. 8, no. 1, pp. 17-49, 2020, doi: 10.4018/IJSI.2020010102.
- [12] M. Weske, "Business Process Modelling Foundation," in *Business Process Management: Concepts, Languages, Architectures*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 73-124.
- [13] R. M. Dijkman, M. Dumas, and C. Ouyang, "Semantics and analysis of business process models in BPMN," *Information and Software Technology*, vol. 50, no. 12, pp. 1281-1294, 2008/11/01/ 2008, doi: <https://doi.org/10.1016/j.infsof.2008.02.006>.
- [14] C. Ou-Yang and Y. Lin, "BPMN-based business process model feasibility analysis: a petri net approach," *International Journal of Production Research*, vol. 46, no. 14, pp. 3763-3781, 2008.
- [15] T. Takemura, "Formal Semantics and Verification of BPMN Transaction and Compensation," in *2008 IEEE Asia-Pacific Services Computing Conference*, 9-12 Dec. 2008 2008, pp. 284-290, doi: 10.1109/APSCC.2008.208.
- [16] K. Watahiki, F. Ishikawa, and K. Hiraishi, "Formal verification of business processes with temporal and resource constraints," in *2011 IEEE International Conference on Systems, Man, and Cybernetics*,

- [1] Q. Nivon and G. Salaün, "Debugging of BPMN Processes Using Coloring Techniques," in *Formal Aspects of Component Software*,

- from *Engineering, Science & IT (IJES)*, vol. 3, no. 3, pp. pp. 20-26, 10/19 2015, doi: 10.3991/ijes.v3i3.4862.
- [22] G. Decker, R. Dijkman, M. Dumas, and L. García-Bañuelos, "The Business Process Modeling Notation," in *Modern Business Process Automation: YAWL and its Support Environment*, A. H. M. Hofstede, W. M. P. Aalst, M. Adams, and N. Russell Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 347-368.
- [23] W. v. d. A. a. K. v. H. The, *Workflow Management. Models, Methods, and Systems*. Massachusetts London, England: MIT Press Cambridge, 2002.
- [24] R. Zurawski and M. Zhou, "Petri nets and industrial applications: A tutorial," *IEEE Transactions on industrial electronics*, vol. 41, no. 6, pp. 567-583, 1994.
- [25] O. E. Hichami, M. Naoum, M. A. Achhab, I. Berrada, and B. E. E. Mohajir, "An Algebraic Method for Analysing Control Flow of BPMN Models," *Int. J. Recent Contributions Eng. Sci. IT*, vol. 3, no. 3, pp. 20-26, / 2015. [Online]. Available: <https://www.online-journals.org/index.php/i-jes/article/view/4862>.
- [26] Y. Poursoltani, M. H. Shirali-Shahreza, and S. A. Hashemi Golpayegani, "Redefining the Soundness for BPMN2.0 Process Models, based on Petri nets," presented at the 5th National Conference on Advances in Enterprise Architecture (NCAEA), Mashhad- Iran, 2021. [Online]. (In Persian). Available: <https://civilica.com/doc/1386000/>.
- 9-12 Oct. 2011 2011, pp. 1173-1180, doi: 10.1109/ICSMC.2011.6083857.
- [17] J. G. Peter Y.H. Wong, "Property specifications for workflow modelling," *Science of Computer Programming*, vol. 76, no. 10, pp. 942-967, 2011, doi: <https://doi.org/10.1016/j.scico.2010.09.007>.
- [18] D. Falcioni, A. Polini, A. Polzonetti, and B. Re, "Livelock and Deadlock Detection for PA Inter-organizational Business Processes," in *Advancing Democracy, Government and Governance*, Berlin, Heidelberg, A. Kö, C. Leitner, H. Leitold, and A. Prosser, Eds., 2012// 2012: Springer Berlin Heidelberg, pp. 125-139.
- [19] O. M. Kherbouche, A. Ahmad, and H. Basson, "Using model checking to control the structural errors in BPMN models," in *IEEE 7th International Conference on Research Challenges in Information Science (RCIS)*, 29-31 May 2013 2013, pp. 1-12, doi: 10.1109/RCIS.2013.6577723.
- [20] G. Barbon, V. Leroy, and G. Salaün, "Debugging of concurrent systems using counterexample analysis," in *Fundamentals of Software Engineering: 7th International Conference, FSEN 2017, Tehran, Iran, April 26-28, 2017, Revised Selected Papers 7*, 2017: Springer, pp. 20-34.
- [21] O. El Hichami, M. Naoum, M. Al Achhab, I. Berrada, and B. E. El Mohajir, "An Algebraic Method for Analysing Control Flow of BPMN Models," *International Journal of Recent Contributions*

^{xx} Workflow Graph

^{xx} Model Checking

^{xx} Model Checker

^{xx} Labelled Transition Systems (LTS)

^{xx} MAX – Plus

^{xx} Likelihood

^{xx} Collaborative Process Models

^{xx} Soundness

^{xx} Safeness

^{xx} Well Structureness

^{xx} Process Model

^{xx} Activity Models

^{xx} Activity Models

^{xx} Activity Models

^{xx} Workflow net

^{xx} Start By Trigger

^{xx} Sound

^{xx} Live

^{xx} Bound

^{xx} Deadlock

^{xx} Dead Tasks

^{xx} Object Management Group (OMG)

^{xx} Static Graph Oriented Language

^{xx} Control Flow Errors

^{xx} Data Flow Errors

^{xx} Unit Testing

^{xx} Dead Lock

^{xx} Livelock

^{xx} Source Process Models

^{xx} Mapping Method

^{xx} Target process Models

^{xx} Structural Soundness Metrics

^{xx} Model Driven Software Engineering (MDSE)

^{xx} Liveness

^{xx} Boundness

^{xx} Reachability

^{xx} Linear Temporal Time (LTL)

^{xx} Unfolding

^{xx} Collaborative Business Processes

^{xx} Model Checking

^{xx} Kripke

^{xx} Linear Temporal Logic (LTL)