



دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دانشکده ریاضی و علوم کامپیوتر

پایان نامه کارشناسی ارشد

علوم کامپیوتر گرایش الگوریتم و نظریه محاسبه

## تحلیل ایستای مدل‌های فرآیندی BPMN2.0

نگارش

یحیی پور سلطانی

استاد راهنما

دکتر محمد حسن شیرعلی شهرضا

استاد مشاور

دکتر سید علیرضا هاشمی گلپایگانی

مهرماه ۱۴۰۰

لِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

## برگ ارزیابی پایان نامه شماره # ۱۵۲۶۲۶

کارشناسی ارشد (تئوری)

یحیی پورسلطانی (۹۸۱۱۲۰۹۷)

دانشکده ریاضی و علوم کامپیوتر

رشته-گرایش علوم کامپیوتر-الگوریتم و نظریه محاسبه

عنوان اولیه پایان نامه تحلیل ایستای مدل های فرآیندی BPMN2

عنوان تغییر یافته تغییر ندارد

تاریخ تصویب ۱۳۹۹/۰۲/۲۴

تعداد واحد پایان نامه ۶

تاریخ دفاع ۱۴۰۰/۰۷/۲۶

نوع برگزاری نشست غیرحضوری

<http://meetings.aut.ac.ir/poursoltani>

### اعضای هیات داوران

#	هیات داوران	نام و نام خانوادگی	کد پورتال	وضعیت حضور
۱	استاد راهنماییک	محمدحسن شیرعلی شهرضا	۱۰۴۲۱	حاضر در نشست مجازی
۲	استاد راهنمایی دو	---	---	---
۳	استاد راهنمایی سه	---	---	---
۴	استاد مشاور یک	سیدعلیرضا هاشمی گلپایگانی	۱۰۳۷۰	حاضر در نشست مجازی
۵	استاد مشاور دو	---	---	---
۶	داور داخلی یک	امین غبی	۱۲۲۲۰	حاضر در نشست مجازی
۷	داور داخلی دو	---	---	---
۸	داور خارجی یک	ناصر محمدزاده	۱۲۷۳۵	حاضر در نشست مجازی
۹	داور خارجی دو	---	---	---
۱۰	داور آخر	---	---	---
۱۱	نماینده تحصیلات تکمیلی	امین غبی	۱۲۲۲۰	حاضر در نشست مجازی

گواهی می گردد ارزیابی دفاع ۱۹/۵۰ (نوزده ممیز پنجه صیدم از بیست،) نمره مقاله یا کار ویژه ۰/۰۰ (صفر) و نمره نهایی ارزیابی دفاع پایان نامه ۱۹/۵۰ (نوزده ممیز پنچ دهم) با درجه عالی ثبت گردیده است.

## مدیریت تحمیلات تکمیلی

ریاضی و علوم کامپیوتر



به نام خدا

تاریخ: مهرماه ۱۴۰۰

## تعهدنامه اصالت اثر

دانشگاه صنعتی امیرکبیر  
(پلی‌تکنیک تهران)

اینجانب یحیی پورسلطانی متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظرارت و راهنمایی استادی دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مأخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است. در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، تسطیح، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مأخذ بلامانع است.

یحیی پورسلطانی

امضا

## تقطیع

همهی بزرگوارانی که بادیدن شرهی این کار خرسند می‌شوند  
و با قلب پاکشان موفقیت من را از خداوند خواسته‌اند و خواهند خواست.  
و به یاد پدر بزرگم، که همواره از دیدن موفقیت‌های تحصیلی ام خوشحال و مسرور می‌شدند.

## منت خدای راعزو جل که طاوش موجب قربت است و به شکر اندرش نمید نعمت.

در ابتدای امر، ایزد منان را سپاس‌گزارم از این که به من توفیق داد تا در دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران) مقطع تحصیلات تکمیلی خود را به انتهای برسانم و این کار پژوهشی را به انتهای رسانده و عرضه نمایم؛ بی تردید تحصیل در این دانشگاه و اتمام این کار پژوهشی، یکی از افتخارات فراموش نشدنی من خواهد بود. در ادامه، از استاد راهنمای خود، جناب آقای دکتر محمدحسن شیرعلی شهرضا نهایت سپاس‌گزاری را دارم که با راهنمایی های مشفقاته خویش زمینه‌های لازم را برای انجام این پژوهش فراهم نمودند و در انجام این کار از هیچ حمایتی دریغ نکردند. همچنین، از استاد مشاور خود، جناب آقای دکتر هاشمی گلپایگانی از گروه مهندسی فناوری اطلاعات دانشگاه صنعتی امیرکبیر کمال تشکر را دارم که علاوه بر جلسات هفتگی آزمایشگاه فناوری های پیشرفته در خدمات و تجارت الکترونیک، در ارتباطی مستمر، منابع و دانش عمیق خویش را به اشتراک گذاشته و بخش مهمی از نتایج این پژوهش را مرهون رهنماوهای روشنگر ایشان هستم. در ادامه، از استاد ارجمند، جناب آقای دکتر رامان رامسین، عضو هیات علمی دانشکده‌ی مهندسی کامپیوتر دانشگاه صنعتی شریف، سپاس‌گزارم که با ارائه‌ی مطالبی مفید در درس الگوها در مهندسی نرمافزار به ایجاد نگرش الگو محور به این جانب کمک کردند؛ در ادامه، از پدر عزیزم سپاس‌گزارم که تجربیات ۳۵ ساله‌ی خود را در اختیارم قرار دادند و این تجربیات، در پیشرفت این پژوهش تاثیر بسزایی داشت؛ در ادامه از مدیریت و کارکنان شرکت دانشبنیان رایانوید سامانه که در پیشنهاد این پژوهش و در راستای نیاز محصولات دانشبنیان یوبی‌پراسس و بیزوتاب به من کمک کردند، سپاس‌گزارم. در نهایت، از اساتید ارجمند، جناب آقایان دکتر امین غیبی و دکتر ناصر محمدزاده، به خاطر داوری این اثر پژوهشی کمال تشکر را دارم و برای این عزیزان، از خداوند منان آرزوی سلامتی دارم و امید است که این کار پژوهشی، در راستای حل بخشی از مشکلات حوزه‌ی فناوری اطلاعات میهن عزیزمان، مفید باشد.

## چکیده

زبان مدل‌سازی BPMN2.0 یک زبان مدل‌سازی استاندارد به منظور مدل‌سازی فرآیندهای کسب و کار می‌باشد. اگرچه فرآیندهای ایجاد شده با استفاده از آن، بسیار گویا هستند، اما به علت ماهیت غیرصوری آن، ممکن است در فرآیندهای ایجاد شده به وسیله‌ی آن، خطاهایی معنایی نظیر بن‌بست و یا اجرای نامتناهی برخی از فعالیت‌ها در یک حلقه (بن‌بست حلقوی) رخ دهد. بسیاری از این مدل‌های فرآیندی توسط سیستم‌های مدیریت فرآیندهای کسب و کار، در سطوح مختلفی از معماری و سیستم‌های اطلاعاتی سازمان‌ها اجرا شده و یا توسط پلتفرم‌های توسعه‌ی کم‌کد و مبتنی بر این زبان، به صورت مستقیم به نرم‌افزارهای قابل اجرا تبدیل می‌شوند؛ بنابراین وقوع خطاهای معنایی در آن‌ها می‌تواند عملکرد نرم‌افزارها، سیستم‌های اطلاعاتی آن سازمان و در نتیجه عملکرد آن را با اختلال مواجه سازد. بنابراین لازم است که پیش از اجرای این مدل‌های فرآیندی، آن‌ها را از نظر نبود خطاهای ساختاری، مورد تحلیل (ایستا) قرار داد؛ بدین منظور، در تحقیقات پیشین، مدل‌های فرآیندی ایجاد شده با این زبان، به مدل‌هایی صوری مثل انواع مختلف شبکه‌های پتری، نگاشته شده و در آن مدل‌ها، مورد تحلیل قرار گرفته‌اند.

در این پژوهش، بر مبنای مفهوم عام مدل‌های فرآیندی و الگوهای جریان کنترلی، نگاشتی بین فرآیندهای ایجاد شده با این زبان و شبکه‌های پتری جریان کار ارائه شده‌است و به وسیله‌ی آن، روشی مبتنی بر استفاده‌ی مجدد از ترکیب الگوها و پادالگوهایی از فرآیندهای صحیح به منظور صحت‌سنجی مدل‌های فرآیندی BPMN2.0 ارائه شده‌است. سپس با معرفی داده‌ساختاری به نام گراف دروازه‌ای، روش پیشنهادی در قالب یک الگوریتم بر روی آن اجرا شده و با این کار، امکان صحت‌سنجی (تحلیل ایستای) مدل‌های فرآیندی BPMN2.0 بر مبنای الگوها و پادالگوهای فرآیندهای صحیح و بدون نیاز به نگاشت به مدل‌های صوری، ممکن شده‌است.

## واژه‌های کلیدی:

زبان مدل‌سازی BPMN2.0 ، صحت ساختاری (Structural Soundness) ، شبکه‌های پتری جریان کار (Verification) ، صحت‌سنجی (Workflow Petri nets) (Model Driven Development)



## فصل اول

### مقدمه

## ۱-۱ زمینه‌ها و ضرورت تحقیق

زبان مدل سازی BPMN 2.0<sup>۱</sup>، یک زبان استاندارد برای مدل سازی فرآیندهای کسب و کار<sup>۲</sup>، خصوصاً در لایه‌ی تحلیل دامنه و طراحی سطح بالای سیستم است. این زبان، ویژگی‌هایی را از مدل‌های پیشین، شامل XPDL و نمودارهای UML، به ارت برده است [۴]. سیستم‌های مدیریت فرآیندهای کسب و کار<sup>۳</sup> که راهکارهایی نرم افزاری برای ایجاد، توسعه و مدیریت چرخه‌ی حیات فرآیندهای کسب و کار هستند، به صورت گسترده‌ای از این زبان استفاده می‌کنند.

### ۱-۱-۱ چالش‌های موجود

زبان مدل سازی BPMN، قابلیت مدل‌سازی فرآیندهای کسب و کار را به خوبی دارد و می‌تواند رخدادهای یک سیستم، همچون موازی سازی، انتخاب و همگام سازی جریان‌های کاری را به خوبی مدل‌سازی کند؛ این زبان مدل‌سازی، ارتباطات بین اجزای شرکت‌کننده در فرآیند را با زبان‌های تعریف فرآیندهای گراف‌گرا، ترکیب کرده و آن را به صورت یکپارچه در آورده است؛ در نتیجه، نمادهای به کار رفته در این مدل فرآیندی، حاوی یک معنی صوری (فرمال) نیستند. به عبارت دیگر، این زبان مدل سازی، یک توصیف نیمه‌صوری<sup>۴</sup> از فرآیند را ارائه می‌دهد. در این صورت، ممکن است که ابزارها و افراد مختلف، یک برداشت واحد و منحصر به فرد را از یک نمودار فرآیندی نداشته باشند. این امر می‌تواند مدل‌های فرآیندی را با مشکلات معنایی<sup>۵</sup>، از جمله بن‌بست<sup>۶</sup>، بن‌بست‌های حلقوی (انتظار مشغول)<sup>۷</sup>، وجود بخش‌های استفاده نشده و... مواجه کند [۴] [۶] [۷]. به عنوان مثال، در شکل ۱-۱، یک بن‌بست رخ داده است؛ در این شکل، فرآیند آغاز شده و به یک دروازه‌ی انشعبان انصصاری می‌رسد؛ سپس، یکی از خروجی‌های دروازه‌ی انشعبان انصصاری فعال می‌شود و نتیجه‌ی آن، به دروازه‌ی اتصال موازی می‌رسد (اگر شاخه‌ی پایینی اجرا شود، بعد از اجرای فعالیت «تولید کالا» به دروازه‌ی اتصال موازی، می‌رسد). اما با توجه به این‌که صرفاً یکی از ورودی‌های دروازه‌ی اتصال موازی، فعال شده است، خروجی دروازه‌ی اتصال موازی فعال نمی‌شود (شرط فعال شدن خروجی دروازه‌ی اتصال موازی، فعال بودن تمام

<sup>1</sup>Business Process Management Notation

<sup>2</sup>Business Process

<sup>3</sup>Business Process Management Systems (BPMS)

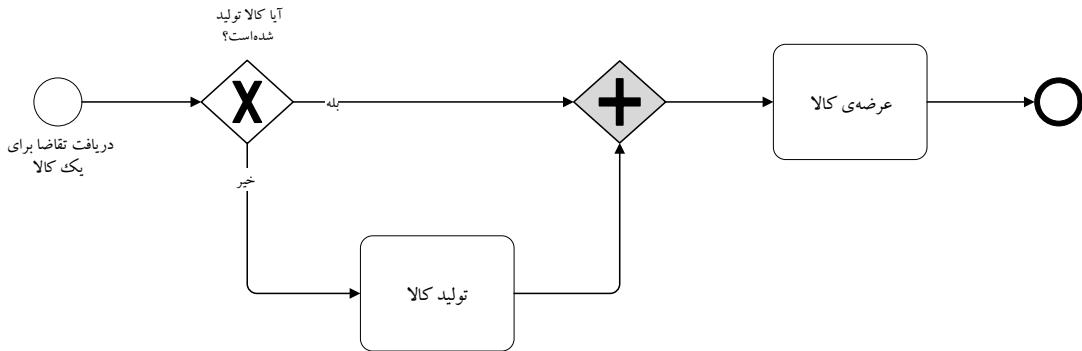
<sup>4</sup>Semi-Formal

<sup>5</sup>Semantic Mistakes

<sup>6</sup>deadlock

<sup>7</sup>livelocks

ورودی‌های آن است). در نتیجه، فرآیند کسب و کار در محل دورازه‌ی اتصال موازی، به بنبست خوده و ادامه پیدا نمی‌کند.



شکل ۱-۱: نمونه‌ای از بروز بنبست در یک فرآیند کسب و کار

از سوی دیگر، مدل‌های فرآیندی BPMN 2.0، می‌توانند به طور مستقیم، توسط پلتفرم‌های توسعه‌ی کم کد و مبتنی بر زبان مدل‌سازی BPMN 2.0 و بدون نیاز به کدنویسی، در دنیای واقعی اجرا شوند. با این کار، توسعه‌ی نرم افزارهای کاربردی، با حداقل کدنویسی (و بعضًا بدون نیاز به کدنویسی) و صرفاً، با ترسیم یک مدل فرآیندی، ممکن خواهد بود. امروزه، با گسترش پلتفرم‌های توسعه‌ی کم کد و استفاده از آن‌ها در اتوماسیون فرآیندهای کسب و کار، امکان تبدیل مستقیم مدل‌های فرآیندی ایجاد شده به وسیله‌ی زبان مدل‌سازی BPMN 2.0 به یک برنامه‌ی قابل اجرا، وجود دارد؛ بنابراین، مدل فرآیندی، عیناً به یک نرم‌افزار قابل اجرا بر روی زیرساخت‌های مختلف نرم افزاری (همانند تلفن‌های همراه، نرم‌افزارهای تحت وب و یا برنامه‌های کاربردی و دیگر بسترها الکترونیکی) تبدیل خواهد شد؛ با توجه به این تبدیل مستقیم، در صورت وقوع هرکدام از خطاهای معنایی در مدل فرآیندی، یک خطای معنایی در عمل کرد نرم‌افزار تولید شده از مدل فرآیندی، به وجود خواهد آمد و عمل کرد نرم‌افزار حاصل را مختل می‌کند؛ این امر باعث تحمیل هزینه‌های بسیار زیاد و بروز اختلال در سیستم‌های اطلاعاتی سازمان و نیز در سطوح مختلفی از لایه‌ی عملیاتی معماری آن سازمان، خواهد شد.

## ۲-۱-۱ اهمیت و ضرورت تحقیق

زبان‌هایی که ساختاری مشابه با زبان مدل‌سازی BPMN 2.0 دارند، به عنوان زبان‌های تعریف فرآیندهای گراف گرا<sup>۸</sup> شناخته شده‌اند. مطالعات نشان داده‌اند که این دسته از زبان‌های مدل‌سازی، انواعی از

<sup>8</sup>Graph oriented process definition

خطاهای معنایی را می‌توانند از خود بروز دهند [۴]. در پژوهش‌های پیشین، روش‌های مختلفی به منظور صحتسنجی مدل‌های فرآیندی BPMN 2.0 ارائه شده‌است؛ با این حال در بسیاری از این روش‌ها، لازم است که مدل فرآیندی به یک زبان مدل‌سازی صوری، مثل شبکه‌های پتری نگاشته شود و عمل صحتسنجی، با استفاده از خواص مدل صوری مقصود انجام بگیرد. نگاشت مدل‌های فرآیندی BPMN به مدل‌های صوری می‌تواند از پیچیدگی بالایی رنج ببرد و عمل صحتسنجی را دچار پیچیدگی کند؛ بنابراین بهتر است رویکردی برای صحتسنجی ارائه شود که طی آن، نیازی به نگاشت یک مدل فرآیندی BPMN 2.0 به یک مدل صوری نباشد؛ با این کار، عمل صحتسنجی مدل‌های فرآیندی BPMN 2.0 بدون نیاز به صوری‌سازی آن، میسر خواهد بود.

از سوی دیگر، با توجه به این که مدل‌های مبتنی بر این زبان مستقیماً بر روی تعداد قابل توجهی از پلتفرم‌های توسعه‌ی کم کد به نرم‌افزارهایی قابل اجرا تبدیل می‌شوند و یا به طور مستقیم در سطوح مختلفی از معماری سازمان به اجرا در می‌آیند، بروز خطاهای معنایی در آن‌ها می‌تواند در عملکرد سیستم‌های اطلاعاتی سازمان و در نتیجه، سطوح مختلفی از معماری آن سازمان، ایجاد مشکل کرده و عملکرد سازمان و یا کسب و کار آن را با اختلال مواجه کند. بنابراین، با توجه به حذف برنامه نویسی سنتی در ایجاد نرم‌افزارهای سازمانی و تجاری و استفاده از پلتفرم‌های توسعه‌ی کم کد و مبتنی بر زبان مدل‌سازی BPMN 2.0، لازم است که رویکرد ارزیابی نرم‌افزارهایی که از این طریق تولید می‌شوند، مورد بازنگری قرار گرفته و توسعه داده شود و ابزاری مناسب برای جلوگیری از بروز این خطاهای معنایی، ایجاد شود.

روال‌های تضمین کیفیت فرآیندهای کسب و کار [۳] و نیز محصولات نرم‌افزاری [۸]، در دو گروه کلی صحتسنجی<sup>۹</sup> و اعتبارسنجی<sup>۱۰</sup> قرار می‌گیرند که روش‌های آزمون نرم‌افزار نیز در بخشی از این روال‌ها می‌گنجد. تحلیل‌های ایستا، بخشی از روال‌های صحتسنجی محسوب می‌شوند که بدون اجرای نرم‌افزار می‌توانند گروه مهمی از خطاهای معنایی مدل را شناسایی نمایند.

## ۲-۱ بیان کلی مسئله

زبان مدل‌سازی BPMN 2.0 یک زبان مدل‌سازی به منظور مدل‌سازی فرآیندهای کسب و کار است. اگرچه گویایی این زبان باعث درک بهتر فرآیندهای کسب و کار ایجاد شده به وسیله‌ی آن می‌شود، اما به علت

<sup>9</sup>Verification

<sup>10</sup>Validation

ماهیت غیرصوری آن، ممکن است در مدل‌های فرآیندی ایجاد شده به وسیله‌ی آن‌ها، خطاهای ساختاری ایجاد شده و تشخیص این خطاهای دشوار باشد. تاکنون تحقیقات زیادی در خصوص صوری‌سازی مدل‌های فرآیندی BPMN 2.0 صورت گرفته و تحلیل ایستای آن‌ها، در مدل‌های مقصد و بر اساس خواص ساختاری آن‌ها صورت گرفته است؛ با این حال، تحقیقات زیادی در خصوص صحتسنجی مدل‌های فرآیندی BPMN 2.0 بدون نیاز به صوری‌سازی آن‌ها، صورت نگرفته است و لازم است راهکاری ارائه شود که به وسیله‌ی آن بتوان بدون نیاز به نگاشت مدل‌های فرآیندی BPMN 2.0 به مدل‌های صوری، عمل صحتسنجی را انجام داد.

از سوی دیگر، همان‌طور که پیش‌تر گفته شد، امروزه ایجاد بسیاری از نرم‌افزارهای اتوماسیون فرآیندهای کسب و کار، با بهره‌گیری از پلتفرم‌های توسعه‌ی کم کد و مبتنی بر مدل‌های فرآیندی BPMN 2.0، صورت می‌گیرد. در بسیاری از این پلتفرم‌ها، نمودارهای فرآیندهای کسب و کار، توسط کاربر ترسیم شده و در خروجی، برنامه‌هایی کاربردی و قابل اجرا به دست خواهند آمد که می‌توانند فرآیند ترسیم شده را اجرا کنند؛ بنابراین پیاده‌سازی نرم افزار، به وسیله‌ی ترسیم نمودارهای فرآیندی انجام شده است (به جای کدنویسی سنتی). از سوی دیگر، در صورت وجود خطاهای معنایی در مدل فرآیندی ایجاد شده و تبدیل مدل فرآیندی به یک نرم‌افزار قابل اجرا، نرم‌افزار تولید شده نیز حاوی خطاهای معنایی خواهد بود و این خطاهای معنایی، از وجود خطاهایی معنایی در مدل فرآیندی ایجاد شده، نشأت گرفته است. وقوع خطاهای معنایی در عمل کرد نرم‌افزارها در هر کدام از مراحل ایجاد نرم‌افزار، هزینه‌های زیادی را بر تیم توسعه‌ی نرم‌افزار تحمیل می‌کند.

در برنامه‌نویسی سنتی، تحلیل ایستای<sup>۱۱</sup> کد، می‌تواند از وقوع بسیاری از خطاهای معنایی<sup>۱۲</sup> در نرم‌افزارها جلوگیری کند. تحلیل ایستا، به معنای تحلیل کد نوشته شده، بدون نیاز به اجرای آن می‌باشد. با توجه به این‌که در برخی از پلتفرم‌های مدیریت فرآیند و حتی سایر سیستم‌ها به جای استفاده از زبان‌های برنامه‌نویسی سنتی، از نمودارهای فرآیندی BPMN 2.0 به منظور پیاده‌سازی عمل کرد نرم‌افزار استفاده می‌شود و در عمل هیچ کدی نوشته نمی‌شود، لازم است که موارد زیر انجام شود:

۱. معیارهایی برای صحت<sup>۱۳</sup> این مدل‌های فرآیندی و در نتیجه، بازنگاری فرآیندهای صحیح از فرآیندهای غیر صحیح تعریف شود؛ برای این منظور، لازم است که مفهوم صحت برای این مدل‌های فرآیندی، بازتعریف شود.

<sup>11</sup> Static Analisys

<sup>12</sup> Semantic Errors

<sup>13</sup> Soundness

۲. یک روش مدون، به منظور صحتسنجی مدل‌های فرآیندی 2.0 BPMN ارائه شود تا توسط تحلیل‌گران و طراحان فرآیندهای کسب و کار قابل استفاده باشد (از جمله کسانی که با استفاده از پلتفرم‌های توسعه‌ی کم‌کد و مبتنی بر زبان 2.0 BPMN مبادرت به ایجاد نرمافزار می‌کنند).

۳. داده‌ساختار و الگوریتمی ارائه شود تا به وسیله‌ی آن بتوان عمل صحتسنجی را با دقت مطلوبی بر روی مدل‌های فرآیندی 2.0 BPMN انجام داد.

با ارائه‌ی یک روش برای تحلیل ایستای مدل‌های فرآیندی، می‌توان علاوه بر صحتسنجی مستقیم این مدل‌های فرآیندی و بدون استفاده از روش‌های صوری، به معیارهایی به منظور تولید نرمافزارهایی با کیفیت و با استفاده از پلتفرم‌های توسعه‌ی کم‌کد و مبتنی بر این مدل‌های فرآیندی دست یافت و از سروایت خطاهای معنایی به وجود آمده در فرآیندهای ایجاد شده با 2.0 BPMN، به سطوح بالاتر نرم افزارهای تولید شده توسط این پلتفرم‌ها و صرف هزینه‌های فراوان مراقبت و نگهداری، جلوگیری کرد [۴]. با ارائه‌ی این روش و به کارگیری آن در سیستم‌های مدیریت کسب و کار در قالب یک ابزار، از اجرای فرآیندهای حاوی این خطاهای معنایی جلوگیری می‌شود؛ همچنین با به کار گیری این ابزار در پلتفرم‌های توسعه‌ی کم‌کد و مبتنی بر مدل‌های فرآیندی، کیفیت نرم افزارهای ایجاد شده به وسیله‌ی این پلتفرم‌ها، ارتقاء یافته و احتمال وجود اختلال در عمل کرد منطقی و معنایی این نرم افزارها، کاهش می‌یابد.

### ۱-۳ مفروضات و معلومات

در این مسئله، فرض می‌کنیم که یک فرآیند کسب و کار، با زبان مدل‌سازی BPMN2.0 مدل شده است. یکی از مهم‌ترین مفروضات اولیه در این مسئله، آن است که: «در تحلیل ایستای یک فرآیند داده شده، تحلیل ساختاری<sup>۱۴</sup> آن مدنظر است و محتوای اجزای آن (همچون شروط حاکم بر دروازه‌های فرآیند، نوع رخدادها و...) در نظر گرفته نمی‌شوند». همچنین، پیش از اعمال تحلیل ایستا، مفروضات زیر را در نظر می‌گیریم و در صورتی که این مفروضات برقرار نبودند، طی یک فرآیند پیش‌پردازش، این مفروضات را برقرار می‌کنیم. این مفروضات به شرح زیر هستند:

- فرآیند، دارای یک نقطه‌ی شروع و یک نقطه‌ی پایان است؛ در نتیجه، خاصیت یک شبکه‌ی جریان کار را دارد. در صورت عدم برقراری، طی فرآیند پیش‌پردازش، این قاعده اصلاح شود.

<sup>14</sup>Structural Analysis

- فرآیند فاقد رخدادهای الصاق شده به فعالیت‌ها است؛ در صورت وجود، طی فرآیند پیش‌پردازش (که در فصل‌های بعدی توضیح داده خواهد شد) با استفاده از دروازه‌های مناسب، جایگزین خواهد شد.
  - درجه‌ی ورودی و خروجی فعالیت‌ها و رخدادها، برابر با ۱ است. در صورت عدم برقراری، طی فرآیند پیش‌پردازش، با جایگزینی دروازه‌های مناسب، این قاعده برقرار خواهد شد.
  - فرآیند برای یک واحد سازمانی است و در آن، استخراها و خطوط شناوری، موجود نمی‌باشند.
  - درجه‌ی ورودی دروازه‌های انشعابی، برابر با ۱ است.
  - درجه‌ی خروجی دروازه‌های اتصالی، برابر با ۱ است.
- در نهایت، با کاوش الگوهای صحت بر روی مدل فرآیندی، بخش‌های صحیح فرآیند کنار گذاشته شده و در بخش‌های باقی‌مانده، پادالگوهای صحت مورد کاوش قرار می‌گیرند.

## ۴-۱ اهداف تحقیق و مجھولات

### ۴-۱-۱ مجھولات مسئله

موضوع این پژوهش، در خصوص تحلیل ایستای مدل‌های فرآیندی BPMN 2.0 بوده و هدف کلی آن، کشف اجزایی در مدل پیاده‌سازی شده با استفاده از زبان مدل سازی BPMN 2.0 است که ممکن است روند اجرای فرآیند را به علت خطاهای معنایی، مختل کنند. این کار را با کنار گذاشتن بخش‌های صحیح فرآیند انجام خواهیم داد و بخش‌های مشکوک به خطا را از بخش‌های صحیح جدا کرده و بر مبنای نقاط گلوگاهی مورد ارزیابی قرار می‌دهیم.

### ۴-۲-۱ اهداف حل مسئله

#### هدف اصلی

هدف اصلی این پژوهش، ارائه‌ی یک روش و یک ابزار برای تحلیل ایستای مدل‌های فرآیندی BPMN 2.0 می‌باشد.

## اهداف اختصاصی

اهداف اختصاصی این پژوهش، به شرح زیر می‌باشند:

۱. بازتعریف مفهوم صحت برای مدل‌های فرآیندی BPMN 2.0 بر مبنای معناشناسی صوری آن‌ها.
۲. ارائه‌ی روشی الگو-محور به منظور صحتسنجی مدل‌های فرآیندی BPMN 2.0 که با استفاده از رایانه و یا بدون استفاده از آن، قابل استفاده باشد.
۳. ارائه‌ی داده‌ساختار و الگوریتمی که بتواند روش صحتسنجی ارائه شده را با استفاده از رایانه، اجرا کند.
۴. ارائه‌ی یک نگاشت بین شبکه‌های پتری جریان کار و مدل‌های فرآیندی BPMN 2.0

## هدف کاربردی

حاصل این پژوهش، می‌تواند در قالب یک ابزار تحلیل ایستای مدل‌های فرآیندی یکپارچه شده و پیاده‌سازی شود و در سیستم‌های مدیریت فرآیندهای کسب و کار، به منظور کشف خطاهای معنایی و جلوگیری از ورود خطاهای معنایی به اجرای فرآیند کسب و کار، مورد استفاده قرار بگیرد؛ همچنین، این ابزار تحلیل ایستا می‌تواند در پلتفرم‌های توسعه‌ی کم کد، مورد استفاده قرار بگیرد و از انتقال خطاهای معنایی، از مدل‌های فرآیندی به نرم افزارهای تولید شده توسط این پلتفرم‌ها جلوگیری کند.

## هدف متعالی

توسعه‌ی چارچوبی برای مدل‌های فرآیندی و استفاده از آن‌ها در پلتفرم‌های توسعه‌ی کم کد و مبتنی بر فرآیندهای کسب و کار، می‌تواند انگیزه‌ای برای ارائه‌ی راهکارهایی کلی، به منظور ارزیابی و آزمون نرم افزارهای تولید شده توسط پلتفرم‌های توسعه‌ی کم کد باشد. با توجه به گسترش و پیشرفت پلتفرم‌های توسعه‌ی کم کد و استفاده از زبان‌های بصری به جای زبان‌های برنامه نویسی سنتی، ارائه‌ی یک راهکار برای تضمین کیفیت نرم‌افزارهای حاصل از این پلتفرم‌ها، ضروری خواهد بود.

## ۱-۵ محدوده‌ی تحقیق

### ۱-۵-۱ محدوده‌ی تحقیق از دیدگاه مدیریت فرآیندهای کسب و کار

مفهوم صحبت‌سنگی و اعتبارسنجی، علاوه بر حوزه‌ی تضمین کیفیت نرم‌افزار، در خصوص فرآیندهای کسب و کار نیز مطرح می‌شوند. در صحبت‌سنگی فرآیندهای کسب و کار، بررسی می‌کنیم که فرآیند طراحی شده، دارای تناقض و مشکلات ساختاری نباشد. در این تحلیل، به پرسش‌های کیفی پاسخ می‌دهیم. از جمله:

- آیا در فرآیند، بنبست رخ می‌دهد؟
  - آیا موارد کاری ایجاد شده در فرآیند، به انتهای می‌رسند؟
  - آیا شاخه‌هایی که در فرآیند منشعب می‌شوند، بعد از اجرا متصل می‌شوند؟
- در اعتبارسنجی یک فرآیند کسب و کار، بررسی می‌کنیم که فرآیند طراحی شده، با واقعیت خارجی خود منطبق باشد. در برخی منابع، به این تحلیل‌ها، تحلیل‌های زمان طراحی<sup>۱۵</sup> نیز گفته شده است [۹]. برخی از تحلیل‌ها، همچون تحلیل کارایی<sup>۱۶</sup>، در زمان اجرا انجام می‌شوند؛ در تحلیل کارایی، به سوالات کمی پاسخ می‌دهیم؛ از جمله:

- در هر ساعت، چند مورد کاری اجرا می‌شوند؟
- متوسط زمان اجرای فرآیند، برای هر مورد کاری، چقدر است؟
- در طی ۲ روز، چند مورد (کیس) می‌تواند به انتهای برسد؟

در این تحقیق، ما به مقوله‌ی صحبت‌سنگی فرآیندهای کسب و کار و مدل‌سازی شده با زبان BPMN2.0 پرداخته‌ایم و روشی را برای صحبت‌سنگی این مدل‌های فرآیندی، ارائه کرده‌ایم و سپس داده‌ساختاری را تحت عنوان گراف دروازه‌ای پیشنهاد کرده‌ایم تا بتوان این روش را با استفاده از آن و یک الگوریتم پسرو، با استفاده از رایانه اجرا نمود.

<sup>15</sup>design time analysis

<sup>16</sup>Performance Analysis

## ۲-۵ محدوده‌ی تحقیق از دیدگاه مهندسی نرم‌افزار

همان‌طور که می‌دانیم، ارزیابی کیفیت نرم‌افزارها، از اهمیت خاصی برخوردار بوده و بخش زیادی از زمان توسعه‌ی نرم‌افزار را دربر می‌گیرد. با گسترش پلتفرم‌های توسعه‌ی کم کد و کم‌رنگ شدن کد زنی و قابلیت تبدیل مستقیم نمودارهای بصری به نرم‌افزارهای قابل اجرا، لازم است که روش‌های ارزیابی کیفیت نرم‌افزارهای تولید شده (توسط پلتفرم‌های توسعه‌ی کم کد)، متناسب با این شرایط، بازتعریف شوند. در طی فرآیند مهندسی نرم‌افزار، یکی از اصول و مفاهیمی که همواره در ارتباط با نرم‌افزار لازم است رعایت شود، مسأله‌ی صحت سنجی و اعتبارسنجی نرم‌افزار می‌باشد که آزمون نرم‌افزار، بخشی از این فرآیندها را تشکیل می‌دهد. این دو فرآیند و فعالیت‌های ذیل آن‌ها، در شکل ۲-۱ نشان داده شده‌اند. بر اساس این شکل، صحت سنجی<sup>۱۷</sup> به مجموعه‌ای از فعالیت‌ها اشاره دارد که از درستی کارکرد مشخص توسط نرم‌افزار، حصول اطمینان می‌کند؛ در این بخش، بررسی می‌کنیم که «آیا محصول را به درستی ساخته‌ایم؟» [۱۰]؛ برای پاسخ به این سؤال، لازم است که کد نوشته شده را مورد ارزیابی قرار دهیم. فعالیت تحلیل ایستا<sup>۱۸</sup> بخشی از مجموعه‌ی فعالیت‌های صحت‌سنجی است؛ در تحلیل ایستا، متن برنامه نوشته شده مورد تحلیل و بررسی قرار می‌گیرد و در صورت وجود خطاهای نحوی و معنایی در متن آن، به توسعه دهنده‌گان هشدار داده می‌شود. اعتبارسنجی<sup>۱۹</sup> به مجموعه‌ی متفاوتی از فعالیت‌ها اشاره دارد که از انطباق نیازمندی‌های مشتری با عمل کرد نرم‌افزار، حصول اطمینان می‌کند [۱۰]. در این بخش، بررسی می‌کنیم که «آیا محصول نرم افزاری در دنیای واقعی درست کار می‌کند؟». بنابراین، موضوع این تحقیق در حوزه‌ی اعتبارسنجی نرم‌افزارها بوده و در خصوص زیربخش تحلیل ایستا خواهد بود.

## ۱-۶ ساختار پایان‌نامه

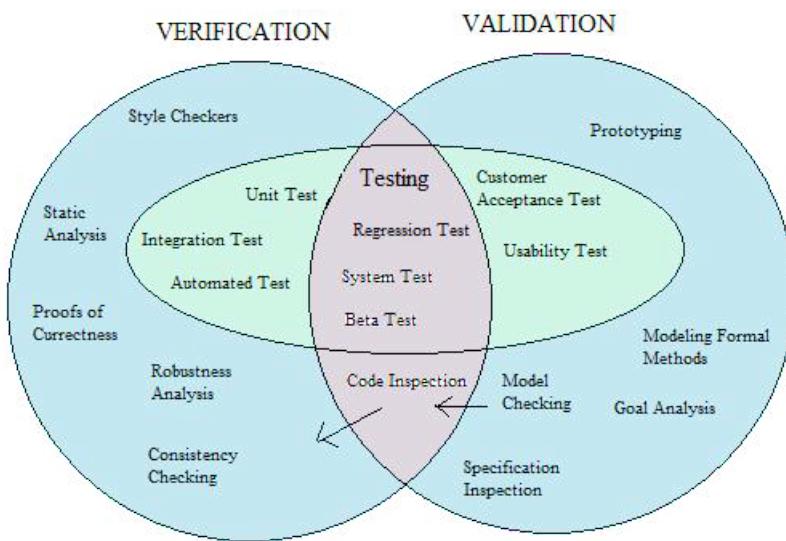
در این فصل، به صورت اجمالی به تشریح صورت مسئله‌ی موجود و چالش‌های پیشرو پرداخته شد و ضرورت و کاربردهای پاسخ آن، تشریح شد. مطالب مورد بررسی در فصول آتی، به شرح زیر می‌باشند:

فصل دوم با توجه به استفاده از مفاهیم مختلف و در حوزه‌های مختلف در پیشبرد این پژوهش، در این فصل به مروری اجمالی و در عین حال مفید بر مفاهیم پایه‌ای مورد نیاز پرداخته شده است؛ از جمله مفاهیمی در حوزه‌ی فرآیندهای کسب و کار، زبان مدل‌سازی BPMN 2.0، شبکه‌های

<sup>17</sup>Verification

<sup>18</sup>Static Analisys

<sup>19</sup>Validation



[۱] شکل ۲-۱: گستره‌ی اعتبارسنجی و صحت‌سنجی نرم‌افزار

پتری، شبکه‌های جریان کار، طبقه‌بندی انواع خطاهای معنایی، مفهوم صحت، بیان مفهوم عام مدل‌های فرآیندی و الگوهای جریان کنترلی در این فصل مطرح شده است؛ در نهایت، فصل با مروری سریع پیرامون مفاهیم اساسی تضمین کیفیت نرم‌افزارها و نیز پلتفرم‌های توسعه‌ی کم کد، به پایان رسیده است.

فصل سوم در این فصل، به مرور و بررسی پژوهش‌های مرتبط با حوزه‌ی پژوهش جاری پرداخته شده است. پژوهش‌های پیشین، در دو بخش کلی دسته‌بندی شده‌اند و درنهایت، دستاوردهای پژوهش جاری با پژوهش‌های پیشین، در آخرین بخش از این فصل، از نظر میزان تشابه و ارزش افزوده، مقایسه شده است.

فصل چهارم در این فصل، راه حل پیشنهادی به تفصیل مورد بررسی قرار گرفته و تشریح شده است. در بخش اول این فصل، یک دورنما از راه حل پیشنهادی ارائه شده است که به خواننده توصیه می‌شود که مطالعه‌ی این فصل را با بررسی آن دورنما آغاز نماید تا دچار سردرگمی نشود. سپس، بخش‌های راه حل پیشنهادی، با زبان ریاضی مدل‌سازی می‌شوند و راه حلی به منظور صحت‌سنجی مدل‌های فرآیندی ارائه شده است. در نهایت، با معرفی داده‌ساختار گراف دروازه‌ای و بیان روش پیشنهادی در قالب یک الگوریتمی که بر روی گراف دروازه‌ای اجرا می‌شود این فصل به اتمام می‌رسد.

فصل پنجم در این فصل، به ملزمات پیاده‌سازی روش پیشنهادی، پرداخته شده است؛ سپس، نتایج

حاصل از اجرای برنامه‌ی پیاده‌سازی شده بر روی تعدادی از گراف‌های دروازه‌ای، از سه منظر مورد تحلیل و بررسی قرار گرفته است.

فصل ششم در این فصل به مروری اجمالی بر دستاوردهای اصلی و نوآوری‌های پژوهش و نیز دستاوردهای ذیل پژوهش پرداخته شده است و با ارائه‌ی پیشنهاداتی در راستای بهبود روش پیشنهادی و به کارگیری عملی روش پیشنهادی در صنعت مهندسی نرم‌افزار، پژوهش به پایان رسیده است.

## فصل دوم

### مفاهیم و تعاریف اولیه

## ۱-۲ مفاهیم و تعاریف اساسی در فرآیندهای کسب و کار

### ۱-۱-۲ فرآیندهای کسب و کار

در حوزه‌ی فناوری اطلاعات،<sup>۱</sup> کسب و کار مقوله‌ای مهم است. با پیشرفت فناوری اطلاعات، بخش مهمی از کسب و کارها، به صورت الکترونیکی انجام می‌شود. برای خدمترسانی در یک کسب و کار سازمانی، مجموعه‌ای از افراد و عوامل در کنار هم قرار می‌گیرند تا بتوانند، کالا و یا خدمتی را به مشتری نهایی عرضه کنند. در خصوص این کسب و کارها، پرسشی قابل طرح است: «چگونه با به کارگیری عوامل سازمان (از جمله نیروهای انسانی و منابع سازمان) می‌توان خدماتی را ارائه کرد و یا کالایی را فروخت؟» از این پرسش، می‌توان متوجه شد که فروش یک محصول و یا ارائه یک خدمت، در گرو اجرای فعالیت‌هایی مرتبط است که باید در کنار یکدیگر، انجام شود. این زنجیره‌ی فعالیت‌ها، با تعدادی ورودی آغاز شده و درنهایت، منجر به ارائه یک کالا و یا خدمت، به عنوان محصول خواهد شد [۱۱].

**تعريف ۱-۱-۲.** به مجموعه‌ای از فعالیت‌های مرتبط که توسط افراد و یا سیستم‌های یک سازمان انجام شده و برای محصول، مشتری داخلی و مشتری خارجی، تولید ارزش می‌کند، یک فرآیند کسب و کار<sup>۲</sup> گفته می‌شود [۱۱].

### ۲-۱-۲ اجزای فرآیندهای کسب و کار

در تعريف ۱-۱-۲ یک فرآیند کسب و کار را تعريف کردیم. یک فرآیند کسب و کار، شامل یک یا چند ورودی است.

**تعريف ۲-۱-۲.** به هرکدام از ورودی‌های یک فرآیند که در طول فرآیند مورد پردازش قرار می‌گیرد، یک مورد<sup>۳</sup> گفته می‌شود [۱۱].

معمولاً، موردها موضوع اصلی هر جریان کاری هستند. مثال‌هایی از موردها در فرآیندهای کسب و کار به شرح زیر است [۱۱] :

۱. هر درخواست پرداخت از یک اداره‌ی بیمه.

۲. هر درخواست برای دریافت وام مسکن.

<sup>1</sup>Information Technology (IT)

<sup>2</sup>Business Process (BP)

<sup>3</sup>Case

۳. هر سفارش از یک فروشگاه اینترنتی.

۴. هر بیمار در یک مطب.

هر مورد، در طول یک فرآیند کسب و کار حرکت کرده و طی فعالیت‌های مختلف، مورد پردازش قرار می‌گیرد و در نهایت، باید از فرآیند خارج گردد.

## ۲-۲ زبان مدل‌سازی BPMN2.0

در این بخش، به معرفی مختصر زبان مدل‌سازی BPMN2.0 می‌پردازیم. این زبان، محصولی از گروه مدیریت شی<sup>۴</sup> است. هدف اصلی BPMN 2.0 ارائه‌ی یک نمادگذاری است که به راحتی توسط همه‌ی کاربران قابل فهم باشد؛ از تحلیل‌گران کسب‌وکار که پیش‌نویس‌های اولیه‌ی فرآیندها را ایجاد می‌نویستند، تا توسعه دهنده‌گان فنی و مسئولین اجرایی که آن فرآیندها را اجرا خواهند کرد و در نهایت، برای افراد کسب‌وکار که آن فرآیندها را مدیریت و نظارت خواهند کرد [۲]. با استفاده از زبان BPMN2.0 می‌توان یک فرآیند کسب و کار را مدل‌سازی کرد و علاوه بر آن، می‌توان تعامل‌های نرم‌افزارهای مورد استفاده در فرآیندهای کسب و کار را نیز با محیط بیرون، مدل‌سازی نمود؛ با این کار امکان مدل‌سازی و مدیریت خطاهای، جلوگیری از دوباره‌کاری‌ها و مدیریت بازگشت کارها، با حداقل پیچیدگی امکان‌پذیر خواهد بود. [۱۲]

در حالت کلی، می‌توان اجزای این زبان مدل‌سازی را به بخش‌های زیر تقسیم‌بندی کرد:

اشیای جریانی<sup>۵</sup> این اجزا، بلوک‌های سازنده‌ی فرآیندهای کسب و کار هستند و شامل موارد زیر می‌باشند [۱۲] :

۱. فعالیت‌ها<sup>۶</sup>

۲. رخدادها<sup>۷</sup>

۳. دروازه‌ها<sup>۸</sup>

<sup>4</sup>Object Management Group (OMG)

<sup>6</sup>activities

<sup>7</sup>events

<sup>8</sup>gateways

رخدادها (یا رویدادها) وقایعی هستند که طی فرآیندهای کسب و کار، در دنیای واقعی رخ می‌دهند. فعالیت‌ها بیان‌گر کار انجام شده در طول فرآیندهای کسب و کار هستند. از دروازه‌ها، برای نشان دادن رفتارهای انشعاب و اتصال جریان‌های کنترلی در بین فعالیت‌ها و رخدادها، استفاده می‌شود [۲].

**خطوط شناوری و استخرها** جنبه‌های سازمانی فرآیندهای کسب و کار، با استفاده از خطوط شناوری نشان داده می‌شوند. خطوط شناوری را می‌توان با یک سلسله مراتب دوسری‌جی نشان داد:

#### ۱. استخرها<sup>۹</sup>

۲. خطوط شنا<sup>۱۰</sup> (در برخی از متون از آن‌ها به عنوان خطوط شناوری <sup>۱۱</sup> نیز یاد می‌شود.)

استخرها، نمایانگر سازمان‌هایی هستند که در یک فرآیند کسب و کار تعاملی <sup>۱۲</sup> با یک دیگر مشارکت می‌کنند و می‌توانند دربرگیرنده‌ی فرآیندهای کسب و کار باشند؛ فرآیند کسب و کار هر سازمان، از سازمان دیگر جدا بوده و در استخر مربوط به همان سازمان، ترسیم می‌شود. بخش‌هایی از فرآیند هر سازمان می‌تواند با بخش‌هایی از فرآیند سازمان دیگر تعامل کند.

هر استخر می‌تواند حاوی یک یا چند خط شنا باشد. هر خط شنا در استخر، نشان‌دهنده‌ی یک بخش از سازمان بازنمایی شده در آن استخر است.

مصنوعات<sup>۱۳</sup> در این زبان مدل‌سازی، مصنوعات اجزایی هستند که به منظور توضیح بیشتر و وضوح بیشتر فرآیند به کار می‌روند و ماهیت اجرایی ندارند؛ بنابراین، این اجزا روند اجرای فرآیند کسب و کار را تحت تاثیر قرار نمی‌دهند و صرفاً برای اهداف اطلاعاتی مورد بهره‌برداری قرار می‌گیرند.

اشیای متصل‌کننده<sup>۱۴</sup> اجزای یک مدل فرآیندی، به وسیله‌ی این اشیا به یک دیگر متصل می‌شوند.

در ادامه، به توضیح این اجزا می‌پردازیم.

<sup>9</sup>Pools

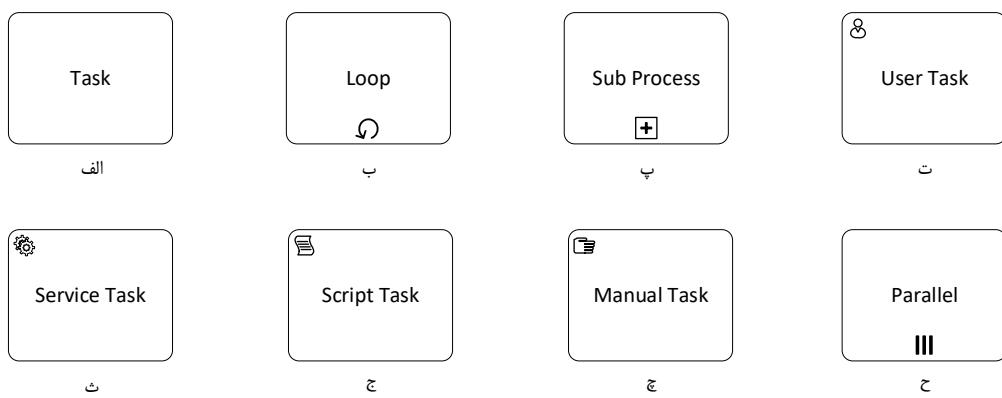
<sup>10</sup>Lanes

<sup>11</sup>Swim Lanes

<sup>12</sup>Collaborative BP

## ۱-۲-۲ فعالیت‌ها

فعالیت‌ها<sup>۱۵</sup> واحدهای کاری یک فرآیند را تشکیل می‌دهند. در زبان مدل‌سازی BPMN2.0 هر فعالیتی که غیر قابل تجزیه باشد، به صورت یک وظیفه در نظر گرفته می‌شود. وظایف، فعالیت‌هایی هستند که غیرقابل تجزیه هستند. فعالیت‌ها، می‌توانند یک فرآیند دیگر را فعال و اجرا کنند. در یک فرآیند، انواع مختلفی از فعالیت‌ها می‌توانند به کار بروند. در ادامه، به آن‌ها اشاره‌ای خواهیم داشت [۲] [۱۲]. در



شکل ۱: مجموعه فعالیت مورد استفاده در زبان BPMN2.0

یک مدل فرآیندی BPMN2.0 انواع مختلفی از فعالیت‌ها می‌تواند به کار برود. در ادامه، توضیحاتی را در خصوص هر کدام از این انواع ارائه می‌کنیم.

**وظایف**<sup>۱۶</sup> ساده‌ترین نوع فعالیت هستند که غیرقابل تجزیه هستند؛ این بدان معنا است که نمی‌توان چنین فعالیت‌هایی را به فعالیت‌های ریزتری شکست. این فعالیت‌ها را با یک مستطیل نشان می‌دهیم (شکل ۱-۲ - الف) [۲].

**حلقه‌ها** یک حلقه، می‌تواند متشكل از چند وظیفه و یا چند فرآیند باشد. وظایفی که در یک حلقه قرار گیرند، می‌توانند در بازه‌های زمانی (مثلاً هر چند دقیقه یک بار) انجام شوند (شکل ۱-۲ - ب) [۱۲].

**زیرفرآیندها**<sup>۱۷</sup> زیرفرآیندها، نوعی از فعالیت‌ها هستند که می‌توانند در بطن خود حاوی یک فرآیند با مجموعه‌ای از وظایف باشند. این زیرفرآیندها، می‌توانند عمق دلخواهی داشته باشند؛ این بدان

<sup>15</sup>Activities

<sup>16</sup>Task

<sup>17</sup>Subprocess

معناست که هر زیرفرآیند می‌تواند در یک زیر فرآیند دیگر نیز ظاهر شود. زیرفرآیندها را با یک مستطیل حاوی علامت بعلاوه، نشان می‌دهند (شکل ۱-۲ - پ) [۲].

**وظایف کاربر وظایف کاربر<sup>۱۸</sup>**، فعالیتهای سنتی جریان کار را نشان می‌دهند. زمانی که اجرای فرآیند به یک وظیفه‌ی کاربری برسد، برای ادامه‌ی فرآیند لازم خواهد بود که این فعالیت، توسط کاربر انجام شود. پر کردن یک فرم کامپیوترا و ارسال اطلاعات آن توسط یک اپراتور انسانی، می‌تواند مثالی از چنین فعالیتی باشد. این فعالیت را با نماد یک مستطیل و شکل انسان در سمت بالا- راست آن، نشان می‌دهند (شکل ۱-۲ - ت) [۲].

**وظایف سرویسی وظیفه‌ی سرویسی<sup>۱۹</sup>**، نوعی وظیفه است که توسط یک قطعه‌ی نرمافزاری اجرا می‌شود. این قطعه‌ی نرمافزاری می‌تواند با استفاده از یک رابط سرویس تحت وب و یا واسط یک نرمافزار کاربردی<sup>۲۰</sup>، در فرآیند به ارائه‌ی خدمت، بپردازد. چنین فعالیتهایی را با استفاده از یک مستطیل و با یک نماد چرخدنده در گوشه‌ی آن، نشان می‌دهیم (شکل ۱-۲ - ث) [۲].

**وظایف اسکریپتی وظایف اسکریپتی<sup>۲۱</sup>**، فعالیتهایی هستند که با اجرای مجموعه‌ای از دستورات اجرا می‌شوند. زبان مورد استفاده در این وظایف، می‌تواند بسته به سیستم مدیریت فرآیند کسب و کار متفاوت باشد؛ به عنوان مثال، در برخی از این سیستم‌ها، از زبان برنامه‌نویسی جاوا اسکریپت استفاده می‌شود. هنگامی که اجرای دستورات تکمیل شود، اجرای فعالیت نیز تکمیل می‌شود [۲]. این فعالیتها را با نماد مستطیل و با نماد یک نوشه در گوشه‌ی آن، نشان می‌دهند (شکل ۱-۲ - ج).

**وظایف دستی وظایف دستی<sup>۲۲</sup>**، فعالیتهایی هستند که بدون پشتیبانی سیستم‌های نرمافزاری اجرا می‌شوند. فعالیتهایی از قبیل ارسال نامه، جابجایی یک کالا در یک محیط لجستیک (مثل انبار) از این قبیل فعالیتها هستند؛ بنابراین، اجرای واقعی این فعالیتها در محیط خارج از سیستم مدیریت فرآیندهای کسب و کار انجام می‌شود؛ با این حال، ممکن است اطلاعات تکمیلی مثل وضعیت نهایی فعالیت، تکمیل و یا عدم تکمیل آن و یا علت عدم اجرای آن، در سیستم ثبت

<sup>18</sup>User Task

<sup>19</sup>Service Task

<sup>20</sup>Application Interface (API)

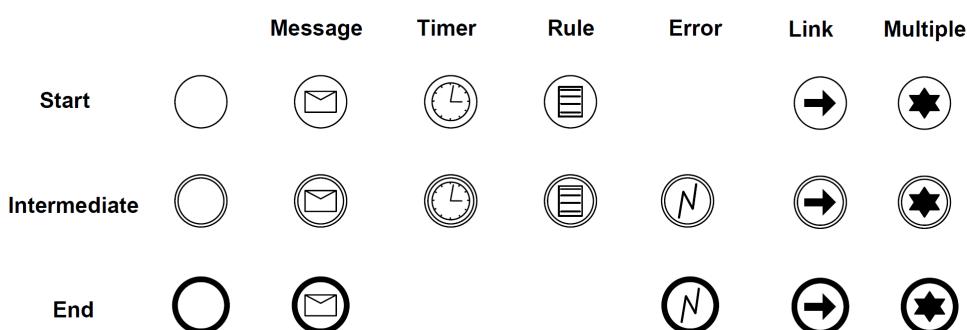
<sup>21</sup>Script Task

<sup>22</sup>Manual Tasks

شود و از این اطلاعات، به منظور اتخاذ تصمیم‌های بعدی استفاده نمود [۲]. این وظایف، با یک مستطیل به همراه نماد یک دست در گوشی آن نشان داده می‌شوند (شکل ۱-۲ - چ).

**فعالیت‌های چند نمونه‌ای موازی** یک فعالیت چند نمونه‌ای موازی، می‌تواند متشکل از چند وظیفه و یا چند فرآیند باشد. وظایفی که در یک چند نمونه‌ای موازی قرار گیرند، می‌توانند به صورت موازی و همزمان با یک‌دیگر، اجرا شوند (شکل ۱-۲ - ح) [۱۲].

## ۲-۲-۲ رخدادها



شکل ۲-۲: مجموعه رخدادهای مورد استفاده در زبان BPMN2.0 [۲]

رخدادها از اجزای جریانی یک مدل فرآیندی BPMN2.0 هستند. یک رخداد، اتفاقی است که در جریان یک فرآیند کسب و کار اتفاق پیش می‌آید و معمولاً، دارای یک علت<sup>۲۳</sup> بوده و درنهایت، نتیجه‌ای را بر اجرای آن فرآیند خواهد داشت. رخدادها، با دایره نشان داده می‌شوند. رخدادها می‌توانند در بخش‌های مختلفی از فرآیند، مورد استفاده قرار گیرند. بر اساس محل قرارگیری، رخدادها به سه دسته‌ی رخدادهای شروع، رخدادهای میانی و رخدادهای پایانی تقسیم می‌شوند. در هر موقعیت، انواعی از رخدادها می‌تواند مورد استفاده قرار گیرد که در ادامه، انواع آن‌ها را معرفی می‌کنیم.

**رخدادهای معمولی** برای بروز این دسته از رخدادها، محرك خاصی وجود ندارد.

**رخداد پیام** <sup>۲۴</sup> با بروز این رخداد، یک پیام ارسال می‌شود. در برخی از سیستم‌های مدیریت فرآیندهای کسب و کار، مثل بیزارجی<sup>۲۵</sup> می‌توان این رخداد را به گونه‌ای تنظیم کرد که با رخداد آن، منتظر

<sup>23</sup>Trigger

<sup>24</sup>Message Event

<sup>25</sup>Bizagi

دریافت پیام از بخش‌های دیگری از فرآیند بود. این رخداد می‌تواند در شروع فرآیند، میانه‌ی فرآیند و انتهای فرآیند، مورد استفاده قرار گیرد.

**رخداد زمانی<sup>۲۶</sup>** این رخداد به گونه‌ای تنظیم می‌شود که در زمان معینی رخ دهد. به عنوان مثال، می‌توان در میانه‌ی فرآیند این رخداد را قرار داد که با رسیدن جریان فرآیند به آن، برای مدت زمان معینی فرآیند متوقف شده و صبر کند. همچنین می‌توان به منظور انتظار در هنگام تکرار بخشی از فرآیند از این رخداد استفاده نمود. این رخداد می‌تواند در شروع فرآیند و میانه‌ی فرآیند مورد استفاده قرار گیرد.

**رخداد شرطی<sup>۲۷</sup>** این رخداد زمانی بروز پیدا می‌کند که یک قاعده‌ی خاصی برقرار باشد. این رخداد می‌تواند در شروع و یا میانه‌ی یک فرآیند به کار گرفته شود.

**رخداد پیوندی<sup>۲۸</sup>** یک پیوند، مکانیزمی است برای اتصال انتهای یک فرآیند به ابتدای فرآیند دیگر. معمولاً به منظور جلوگیری از شلوغ شدن فرآیند، مورد استفاده قرار می‌گیرد.

**رخداد چندگانه<sup>۲۹</sup>** استفاده از آن، به این معناست که برای فعال‌سازی گام بعدی، بیش از یک راه وجود دارد که وقوع یکی از آن‌ها، برای ادامه‌ی فرآیند کافی است.

**رخداد خطأ<sup>۳۰</sup>** با استفاده از این رویداد در میانه و یا انتهای فرآیند، می‌توان وقوع یک خطأ را اعلام نمود. با اعلام خطأ، می‌توان رویه‌هایی را برای مدیریت آن اجرا نمود.

**رخدادهای کرانی** در برخی مواقع، رخدادها می‌توانند بر روی مرز فعالیت‌ها قرار گیرند. در چنین موقعی، در زمانی که اجرای فرآیند به این فعالیت برسد، فرآیند متوقف شده و منتظر می‌ماند تا رخداد مربوطه رخ دهد. در صورتی که رخداد مربوطه رخ داد، اجرای فرآیند از کمان متصل به رخداد ادامه خواهد یافت و در غیر این صورت، ادامه‌ی اجرای فرآیند از فعالیت، ادامه خواهد داشت.

**نمونه ۱-۲-۲.** به عنوان نمونه، در شکل ۳-۲ یک فرآیند برگزاری یک جلسه‌ی مباحثه مشاهده می‌شود. در مرز فعالیت برگزاری جلسه، یک رخداد زمانی قرار دارد که تا مدتی معین صبر می‌کند. با رسیدن اجرای فرآیند به این فعالیت، به همان مدت زمان معین، فرآیند متوقف می‌شود و جلسه‌ی

<sup>26</sup>Timer Event

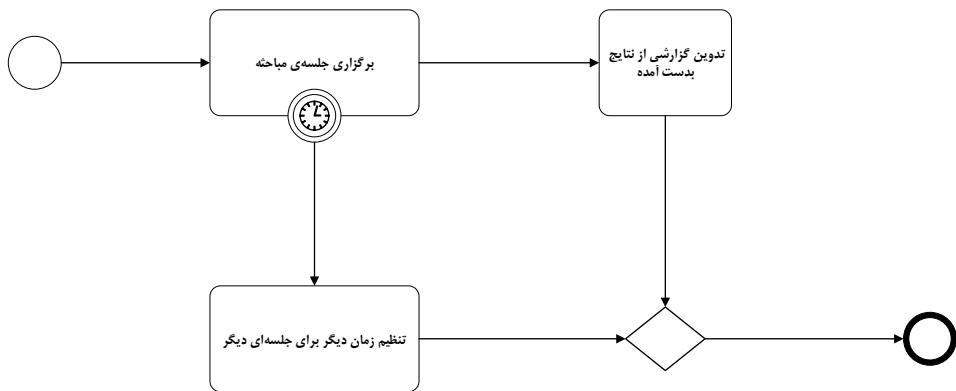
<sup>27</sup>Rule Event

<sup>28</sup>Link Event

<sup>29</sup>Multiple Event

<sup>30</sup>Multiple Event

مباحثه تشکیل می‌شود. اگر قبل از اتمام این مدت زمان، جلسه به نتیجه برسد، گزارش آن تدوین شده و فرآیند تمام می‌شود؛ اما اگر جلسه تمام شود و به نتیجه نرسیده باشد، زمان دیگری برای جلسه‌ی بعدی تعیین خواهد شد و فرآیند به اتمام می‌رسد.



[۲] شکل ۳-۲: مثالی از یک رخداد کرانی در یک مدل فرآیندی BPMN2.0

### ۳-۲-۲ دروازه‌ها

در بخش‌هایی از یک فرآیند، ممکن است لازم باشد که چند شاخه، به صورت موازی و همزمان اجرا شوند. همچنین ممکن است که در بخش‌هایی از یک فرآیند استاندارد، لازم شود که شاخه‌ها به یکدیگر رسیده و هم‌گرا شوند و یک هم‌گام‌سازی در شاخه‌های فرآیند صورت بگیرد؛ همچنین، گاهی لازم می‌شود که در بخش‌هایی از یک فرآیند، یک تصمیم‌گیری صورت بگیرد و بر مبنای نتیجه‌ی تصمیم‌گیری، یک یا چندتا از مسیرهای پیش‌رو انتخاب شوند؛ بنابراین لازم است که در زبان مدل‌سازی BPMN2.0 نمادهایی به منظور همگرایی و یا واگرایی مسیرهای فرآیند در نظر گرفته شود. در این زبان مدل‌سازی، چنین نمادهایی تعریف شده‌اند و به دروازه‌ها<sup>۳۱</sup> موسوم هستند. این اجزا وقتی به کار می‌روند که دنباله‌ی جریان، همگرا و یا واگرا شود. تمامی دروازه‌ها با لوزی نشان داده می‌شوند و بر اساس نوع‌شان، نماد داخل لوزی می‌تواند متفاوت باشد. در این زبان مدل‌سازی، پنج نوع دروازه می‌تواند به کار رود که در ادامه، در خصوص هر کدام توضیحاتی ارائه می‌شود. انواع دروازه‌ها در این زبان عبارتند از:

#### ۱. دروازه‌های انحصاری (دوازه‌های داده‌محور یا XOR)

<sup>31</sup>Gateway

<sup>32</sup>Exclusive Gateways

۳. دروازه‌های رویدادی<sup>۳۴</sup>

۳. دروازه‌های جامع (از آن‌ها به عنوان دروازه‌های فصلی و یا دروازه‌های OR نیز یاد شده است)<sup>۳۵</sup>

۴. دروازه‌های موازی (از آن‌ها به عنوان دروازه‌های عطفی نیز نام برده شده است)<sup>۳۶</sup>

۵. دروازه‌های پیچیده<sup>۳۷</sup>

هر دروازه (بجز دروازه‌ی رویدادی که فقط حالت انشعابی دارد)، به دو صورت می‌تواند به کار روند [۲]:

**حالت انشعابی** در این صورت، استفاده از دروازه منجر به ایجاد انشعاب<sup>۳۸</sup> در سطح فرآیند خواهد شد. با ایجاد انشعاب در فرآیند، یک مسیر به چند مسیر مجزا تقسیم خواهد شد. دروازه‌هایی که عمل انشعاب را انجام می‌دهند، دارای حداقل دو کمان خروجی و دقیقاً یک کمان ورودی هستند [۲].

**حالت اتصالی** در این صورت، استفاده از دروازه منجر به ایجاد اتصال<sup>۳۹</sup> در سطح فرآیند خواهد شد. با ایجاد اتصال در فرآیند، چند مسیر متفاوت، به یک مسیر تبدیل خواهند شد. دروازه‌هایی که عمل اتصال را انجام می‌دهند، دارای حداقل دو کمان ورودی و دقیقاً یک کمان خروجی هستند. [۲]

در ادامه، به بررسی اجمالی هر کدام از این دروازه‌ها می‌پردازیم.

**دروازه‌های انحصاری** در استفاده از این دروازه به صورت انشعابی، برای هر خروجی دروازه یک شرط در نظر گرفته می‌شود. چنان‌چه شرط مربوط به هر شاخه از انشعاب برقرار شود، فرآیند از همان شاخه ادامه پیدا می‌کند و ادامه‌ی فرآیند از سایر شاخه‌های انشعاب، منتفی می‌شود. می‌توان برای حالتی که شرط هیچ کدام از شاخه‌ها صحیح نباشد، شاخه‌ای را به عنوان شاخه‌ی پیش‌فرض در نظر گرفت که ادامه‌ی فرآیند، از آن مسیر ادامه پیدا کند.

**نتیجه ۲-۲-۲.** در استفاده از دروازه‌ی انحصاری در حالت انشعابی، ادامه‌ی اجرای فرآیند، از یک و دقیقاً یکی از شاخه‌های انشعاب ادامه پیدا می‌کند. در حالت اتصالی، برای ادامه‌ی جریان فرآیند لازم است که دقیقاً یکی از شاخه‌های ورودی دروازه فعال باشد. [۲]

<sup>۳۳</sup>Event-Based Gateways

<sup>۳۴</sup>Inclusive Gateways

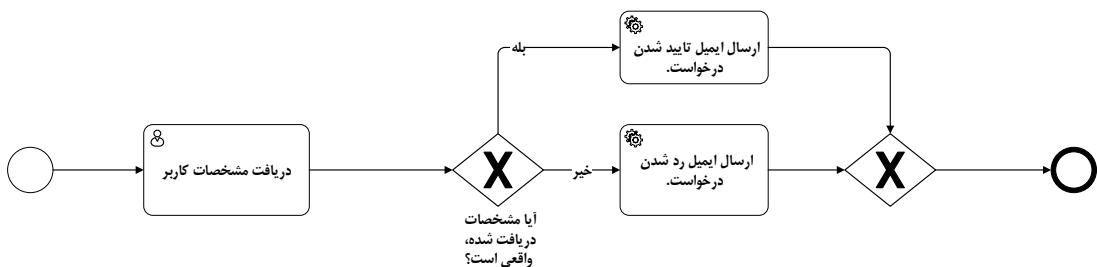
<sup>۳۵</sup>Parallel Gateways

<sup>۳۶</sup>Exclusive Gateways

<sup>۳۷</sup>Split

<sup>۳۸</sup>Join

نمونه ۲-۴-۳. در شکل ۴-۲ مثالی را از به کارگیری این دروازه‌ها مشاهده می‌کنید. همان طور که مشاهده می‌کنید، این دروازه‌ها با نماد لوزی و با یک ضربدر در داخل آن نشان داده می‌شوند. البته می‌توان در داخل لوزی را خالی گذاشت و در این صورت، بازهم دروازه‌ای انحصاری خواهیم داشت.

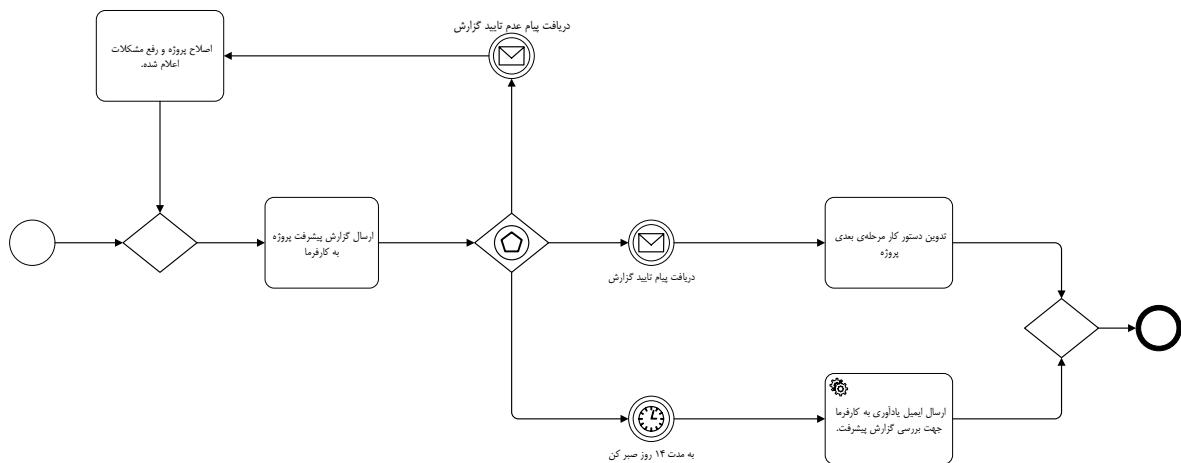


شکل ۲-۴: مثالی از روش به کارگیری دروازه‌ی انحصاری در مدل BPMN2.0

دوازه‌های رویدادی عملکرد این دروازه‌ها، دقیقاً مانند دروازه‌های انحصاری است، با این تفاوت که خروجی‌های شاخه‌های انشعابی این دروازه‌ها، به فعالیت‌ها متصل نمی‌شود و به رویدادها متصل می‌شود [۱۲]؛ در این دروازه‌ها، تمامی رویدادهایی که در شاخه‌های خروجی وجود دارند، فعال می‌شوند و فرآیند از شاخه‌ای ادامه پیدا می‌کند که رویداد مربوط به آن شاخه، زودتر از رویداد سایر شاخه‌ها رخ دهد [۲].

نتیجه ۲-۴-۲. در دروازه‌های انحصاری و مبتنی بر رویداد، مسیر ادامه‌ی اجرای فرآیند بر اساس رویدادهای پیشروی دروازه تعیین می‌شود، این درحالی است که در دروازه‌های انحصاری مبتنی بر داده (بخش قبل) ادامه‌ی مسیر فرآیند بر مبنای شرط تعریف شده بر روی دروازه تعیین می‌شود [۲].

نمونه ۲-۵-۲. در شکل ۵-۲ فرآیندی در خصوص ارسال گزارش پیشرفت پروژه به کارفرما ترسیم شده‌است. بعد از آماده‌سازی گزارش، به مدت ۱۴ روز انتظار در نظر گرفته شده‌است. اگر قبل از تکمیل این ۱۴ روز، پیام تایید را از کارفرما دریافت شود، ادامه‌ی پروژه انجام خواهد شد و چنانچه قبل از تکمیل ۱۴ روز، پیام رد گزارش دریافت شود، پروژه اصلاح شده و مجدداً گزارش پیشرفت تنظیم می‌شود. چنانچه از جانب کارفرما پیامی دریافت نشود و زمان ۱۴ روزه تکمیل شود، یک پیام یادآوری به کارفرما، جهت بررسی پروژه ارسال می‌شود.



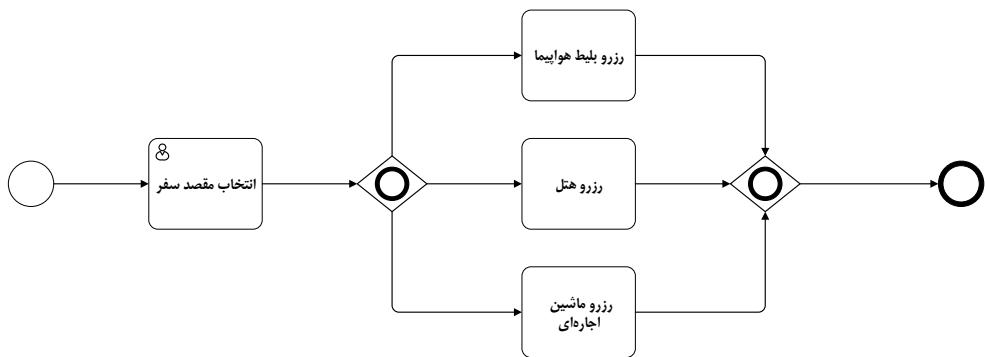
شکل ۲-۵: مثالی از روش به کارگیری دروازه‌ی انحصاری و مبتنی بر رویداد در مدل BPMN2.0

**دوازه‌های فصلی (جامع)** در حالت انشعابی، از این دروازه‌ها در شرایطی استفاده می‌شود که ادامه‌ی اجرای فرآیند، از طریق یک یا چند شاخه از بین شاخه‌های انشعابی، ممکن باشد. در حالت اتصالی، جهت ادامه‌ی جریان فرآیند لازم است تمامی شاخه‌های فعال شده، همگرا شوند. این دروازه، در مقایسه با دروازه‌ی انحصاری، در حالت انشعاب می‌تواند بیش از یک شاخه را انتخاب کند و در حالت اتصال نیز می‌تواند بیش از یک شاخه را همگرا نماید.

نمونه ۲-۶. نمونه‌ای از کاربرد دروازه‌های جامع، در حالت انشعابی و اتصالی در شکل ۲-۶ قابل مشاهده است. در این شکل، یک فرآیند رزرواسیون سفر، که می‌تواند شامل رزرو هتل، بلیط هوایپیما و ماشین اجاره‌ای باشد، قابل مشاهده است. برای داشتن یک سفر، حداقل یک وحدت اتصال شاخه‌های فعال سه رزرو می‌تواند انجام شود و در این صورت، فرآیند ادامه یافته و بعد از اتصال شاخه‌های فعال شده، به اتمام می‌رسد.

**(دوازه‌های عطفی)دوازه‌های موازی** در بسیاری از مواقع، در یک فرآیند کاری با چند مسیر موازی مواجه می‌شویم. در چنین موقعی، در صورتی که اجرای فرآیند به چنین نقطه‌ای برسد، تمامی شاخه‌های موازی می‌باشد به صورت همزمان اجرا شوند. برای ایجاد شاخه‌های موازی در فرآیند از نوع انشعابی این دروازه استفاده می‌شود.

از سوی دیگر، در موقعی لازم است که چند شاخه از فرآیند، همگرا شده و هم‌گام شوند؛ این بدان معنا خواهد بود : «تا زمانی که اجرای تمامی شاخه‌های منشعب شده تمام نشود، فرآیند از

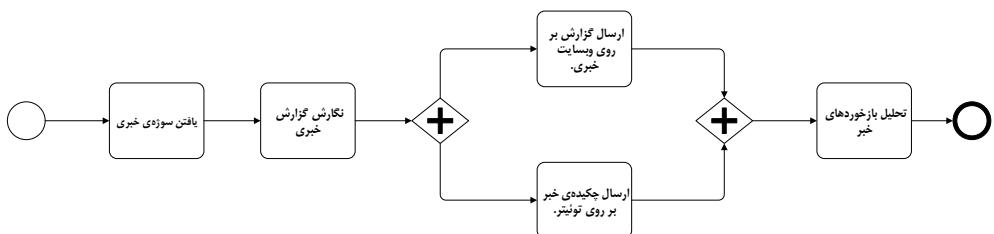


شکل ۲-۶: مثالی از روش به کارگیری دروازه‌ی جامع در مدل BPMN2.0

نقطه‌ی اتصال ادامه نخواهد یافت.» برای این منظور، از نوع اتصالی این دروازه استفاده می‌شود.

نوع اتصالی این دروازه، می‌تواند بیانگر نقطه‌ای برای همگام‌سازی شاخه‌های فرآیند باشد.

**نمونه ۷-۲-۲.** نمونه‌ای از کاربردهای دروازه‌های موازی، در حالت انشعابی و اتصالی در شکل ۲-۷ قابل مشاهده است. در این شکل، یک فرآیند برای انتشار یک گزارش خبری، ترسیم شده است. بعد از یافتن سوژه‌ی خبری و نگارش خبر، به طور همزمان هم بر روی سایت خبری منتشر شده و هم چکیده‌ی آن بر روی شبکه‌ی اجتماعی توئیتر منتشر می‌شود. تا زمانی که این خبر بر روی سایت خبری منتشر نشود و چکیده‌ی آن بر روی توئیتر قابل مشاهده نباشد، امکان تحلیل بازخوردهای خبر وجود نخواهد داشت؛ در نتیجه تا زمانی که هردو کار انجام نشود، فرآیند در نقطه‌ی اتصال منتظر خواهد ماند. بعد از انجام دو فعالیت موازی، فرآیند از نقطه‌ی اتصال ادامه خواهد یافت.

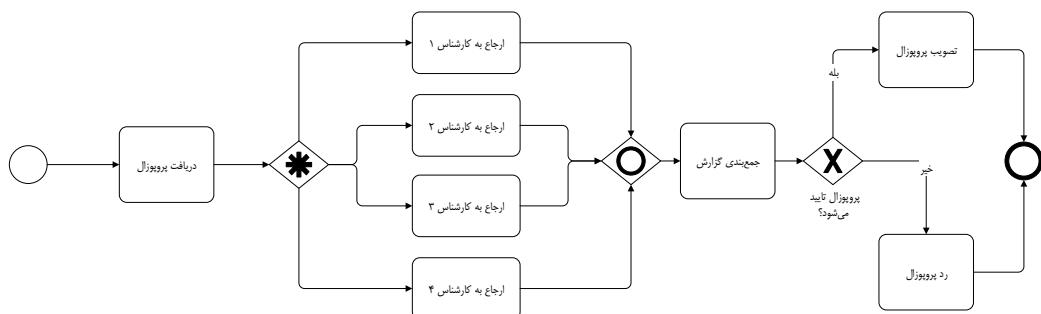


شکل ۷-۲: مثالی از روش به کارگیری دروازه‌ی موازی در مدل BPMN2.0

دوازه‌های پیچیده این دروازه‌ها، معمولاً در زمانی به کار می‌روند که دقیقاً مشخص نباشد که از کدام یک از دروازه‌های قبلی باید استفاده شود. عملکرد این دروازه‌ها می‌تواند به صورت هر کدام از

دروازه‌های قبلی و یا تلفیقی از آن‌ها باشد؛ معمولاً در زمانی که چندین شرط باید بررسی شود و بر مبنای نتیجه‌ی آن‌ها، یک یا چند خروجی می‌تواند انتخاب شود، از این دروازه استفاده می‌شود [۱۲]. معمولاً استفاده از این دروازه‌ها در طراحی یک فرآیند توصیه نمی‌شود؛ چرا که باعث ایجاد پیچیدگی بالا در فرآیند می‌شود.

**نمونه ۸-۲**. نمونه‌ای کاربرد دروازه‌های پیچیده، در حالت انشعابی در شکل ۸-۲ قابل مشاهده است. در این شکل، یک فرآیند برای تصویب یک پروپوزال ترسیم شده‌است؛ در ابتدا پروپوزال دریافت می‌شود و بر اساس موضوع، به یک یا چند کارشناس از بین چهار کارشناس ارجاع می‌شود. برای جلوگیری از اتلاف وقت کارشناسان، باید موضوع پروپوزال با تخصص آن‌ها مرتبط باشد. این تطبیق را در دروازه‌ی پیچیده انجام می‌دهیم. در این مثال، از دروازه‌ی موازی استفاده نکردیم؛ چرا نمی‌خواهیم هر پروپوزال به همه‌ی کارشناسان ارجاع شود. از دروازه‌ی انحصاری نیز استفاده نکردیم؛ چرا که ممکن است یک پروپوزال به بیش از یک کارشناس ارجاع شود.



شکل ۸-۲: مثالی از روش به کارگیری دروازه‌ی پیچیده در مدل BPMN2.0

## ۳-۲ شبکه‌های پتری

امروزه، با گسترش سیستم‌های صنعتی و تعامل پیچیده‌ای از عوامل انسانی، سختافزارها و نرمافزارها، چالش‌های متعددی برای ایجاد کنندگان و توسعه‌دهندگان آن‌ها در ایجاد هماهنگی بین این اجزا به وجود آمده‌است [۱۳]. شبکه‌های پتری، مدل‌هایی ریاضی / گرافیکی هستند که به ما، قابلیت‌های زیر را می‌دهند [۱۳]:

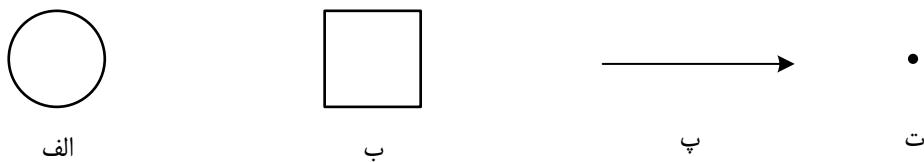
### ۱. مدل‌سازی سیستم‌ها و فرآیندهای کاری

## ۲. تجزیه و تحلیل صوری (فرمال) سیستم‌ها و فرآیندها

۳. طراحی سیستم‌ها با وقایع گسسته<sup>۳۹</sup>

اگرچه زبان مدل‌سازی BPMN2.0 یک زبان ساده و قابل فهم برای توصیف فرآیندهای کسب و کار است، اما همان‌طور که در ادامه خواهیم دید، در توصیف فرآیندهای کسب و کار می‌تواند ایجاد ابهام نماید [۴]؛ بنابراین، به یک زبان مدل‌سازی صوری<sup>۴۰</sup> به منظور توصیف فرآیندهای کسب و کار نیازمند خواهیم بود. شبکه‌های پتری، مدل‌هایی صوری هستند که امکان تحلیل صوری (تحلیل کمی) فرآیندهای کاری را برای ما فراهم می‌کنند. باید توجه داشته باشیم که فرآیند کاری<sup>۴۱</sup> مفهومی عام است و یک فرآیند کاری، می‌تواند یک فرآیند کسب و کار باشد. در این بخش قصد داریم به صورتی مختصر و اجمالی، به معرفی شبکه‌های پتری بپردازیم.

## ۱-۳-۲ اجزای شبکه‌های پتری



شکل ۹-۲: اجزای شبکه‌های پتری

در شکل ۹-۲ به اختصار، اجزای مورد استفاده در یک شبکه‌ی پتری، قابل مشاهده است. در ادامه توضیحاتی مختصر در خصوص این اجزا ارائه می‌شود.

**مکان‌ها** مکان‌ها<sup>۴۲</sup> با نماد دایره نشان داده می‌شوند (شکل ۹-۲ - الف). در مدل‌سازی فرآیندها، هر مکان می‌تواند نشان‌دهنده‌ی یک وضعیت معین باشد. هیچ دو مکانی به صورت مستقیم به‌هم متصل نمی‌شوند و هر مکان، می‌تواند به یک انتقال متصل گردد.

<sup>39</sup>Discrete event systems<sup>40</sup>Formal<sup>41</sup>Workflow<sup>42</sup>Places

**انتقال‌ها** انتقال‌ها<sup>۴۳</sup> با استفاده از نماد مربع (ویا مستطیل) نشان داده می‌شوند (شکل ۹-۲ - ب)؛ در مدل‌سازی فرآیندهای کاری، انتقال‌ها نشان‌دهنده‌ی عملی هستند که منجر به تغییر وضعیت یک مورد در فرآیند کاری می‌شود. هیچ دو انتقالی به صورت مستقیم بهم متصل نمی‌شوند و هر انتقال، می‌تواند به یک مکان متصل شود.

**کمان‌ها** کمان‌ها، فلش‌های جهت داری هستند که ارتباط بین یک مکان و یک انتقال را برقرار می‌کنند (شکل ۹-۲ - پ).

**نشانه‌ها** نشانه‌ها یا توکن‌ها<sup>۴۴</sup> بیان‌گر موردهایی هستند که در یک فرآیند کاری جریان دارند. توکن‌ها در مکان‌ها قرار می‌گیرند. توکن‌ها را با یک نقطه نشان می‌دهند (شکل ۹-۲ - ت).

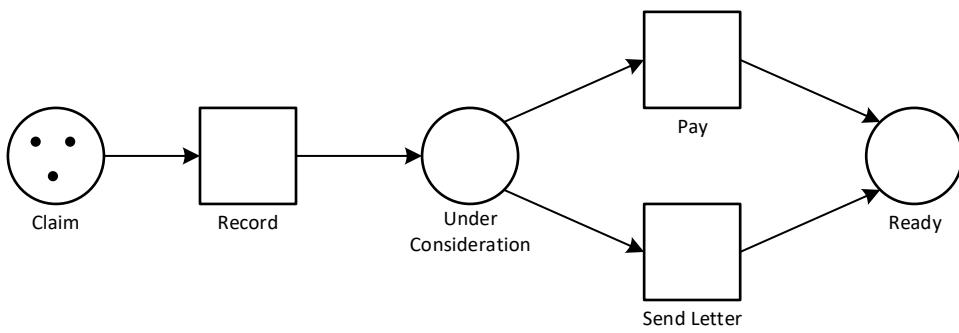
**نتیجه ۲-۳-۱.** مکان‌ها به واسطه‌ی انتقال‌ها به یک دیگر متصل می‌شوند و به طور متقابل، انتقال‌ها نیز به واسطه‌ی مکان‌ها به یک دیگر متصل می‌شوند.

**نمونه ۲-۳-۲.** به عنوان نمونه، فرآیندی را در نظر بگیرید که نحوه‌ی پرداخت خسارت از بیمه را نشان دهد. این فرآیند کاری در شکل ۱۰-۲ نشان داده شده‌است [۳]. در این فرآیند، هر توکن نشان‌دهنده‌ی یک درخواست برای دریافت خسارت از بیمه است. در ابتدا، درخواست در مکان Claim است و این بدان معناست که درخواست اولیه، در قالب یک ادعا قرار دارد که باید بررسی شود. سپس، درخواست دریافت می‌شود و با دریافت یک درخواست، توکن متناظر با آن درخواست از مکان Claim و به واسطه‌ی انتقال Under Consideration منتقل می‌شود. در این مرحله، درخواست در وضعیتی قرار Record می‌گیرد که در حال بررسی است. در نتیجه‌ی این بررسی، درخواست قبول و یا رد می‌شود و در هر دو حالت، درخواست آمادگی رسیدگی خواهد بود و به مکان Ready خواهد رفت. چنان‌چه درخواست قبول شود، به واسطه‌ی انتقال Pay به مکان Ready منتقل می‌شود و در صورت رد شدن درخواست، توکن متناظر با درخواست به واسطه‌ی انتقال Send Letter به مکان Ready منتقل می‌شود.

**نتیجه ۲-۳-۳.** مکان ورودی به یک انتقال، به معنای پیش‌شرط رخ دادن آن انتقال تلقی می‌شود و مکان خروجی از یک انتقال، به متابه‌ی پس‌شرط و حاصل آن انتقال، تلقی خواهد شد.

<sup>43</sup>Transitions

<sup>44</sup>Tokens



شکل ۲-۱۰: مدل‌سازی فرآیند دریافت خسارت با استفاده از شبکه‌های پتری

## ۲-۳-۲ فعال‌سازی و شلیک انتقال‌ها

در بخش‌های قبل، در خصوص شبکه‌های پتری و مفهوم اجزای آن، به صورت اجمالی بحث شد؛ اما در این خصوص بحث نشد که: «توکن‌ها در چه زمانی و تحت چه قوانینی در یک شبکه‌ی پتری به جریان می‌افتد؟». در این بخش، با دو عملی که در یک شبکه‌ی پتری، منجر به حرکت توکن‌ها می‌شود، آشنا می‌شویم. این دو عمل عبارتند از: فعال شدن<sup>۴۵</sup> و شلیک شدن<sup>۴۶</sup>.

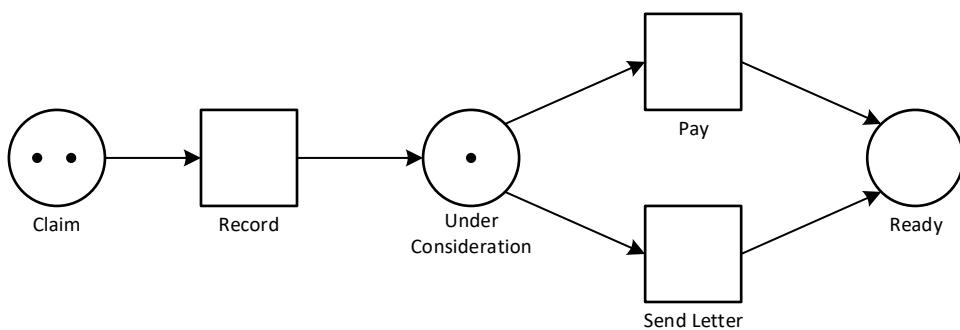
**تعریف ۲-۳-۴.** در یک شبکه‌ی پتری مانند  $P = (P, T, F)$  با مجموعه مکان‌های  $P$ ، مجموعه انتقال‌های  $T$  و مجموعه کمان‌های  $F$ ، انتقالی مانند  $t \in T$  فعال خواهد بود، اگر و تنها اگر در تمامی مکان‌های ورودی آن، حداقل یک توکن وجود داشته باشد.

**تعریف ۲-۳-۵.** در یک شبکه‌ی پتری مانند  $P = (P, T, F)$  انتقالی مانند  $t \in T$  می‌تواند شلیک شود، اگر و تنها اگر فعال باشد. با شلیک انتقال، از تمامی مکان‌های ورودی آن یک توکن کم می‌شود و به تمامی مکان‌های خروجی آن، یک توکن اضافه می‌شود.

**نمونه ۲-۳-۶.** در نمونه‌ی ۲-۳-۲ و در شکل ۲-۳-۲ انتقال Record فعال است و می‌تواند شلیک شود. با شلیک این انتقال، یک توکن از ورودی آن (یعنی مکان Claim) کم می‌شود و یک توکن به مکان خروجی آن، یعنی مکان Under Consideration وارد می‌شود (شکل ۱۱-۲). بعد از این اتفاق، دو انتقال Pay و Send Letter فعال می‌شوند؛ با این حال، توکن موجود در مکان Under Consideration فقط توسط یکی از دو انتقال می‌تواند شلیک شود. در صورت قبول شدن درخواست متناظر با توکن، انتقال Pay و

<sup>45</sup>Enabling

<sup>46</sup>Firing



شکل ۱۱-۲: مدل سازی فرآیند دریافت خسارت با استفاده از شبکه‌های پتری

در صورت رد درخواست، انتقال Send Letter شلیک خواهد شد و در نهایت، توکن در مکان Ready قرار خواهد گرفت.

## ۴-۲ شبکه‌های جریان کار

یکی از چالش‌هایی که در صحبت‌سنگی یک مدل فرآیندی ممکن است رخ دهد، وجود بیش از یک وضعیت شروع و یا وجود بیش از یک وضعیت پایانی است. حتی در بسیاری از پژوهش‌های پیشین، همانند پژوهش‌های [۱۴]، [۱۵] و [۱۶] وجود بیش از یک حالت پایانی برای یک مدل فرآیندی، به منزله‌ی یک خطای ساختاری در نظر گرفته شده‌است. بنابراین، یکی از مهم‌ترین مفروضاتی که در صحبت‌سنگی یک مدل فرآیندی باید در نظر گرفته شود، وجود یک حالت برای شروع و یک حالت برای پایان فرآیند است. این خاصیت، در قالب یک تعریف توسط [۳] و [۲] مطرح شده‌است:

**تعریف ۴-۲.** یک شبکه‌ی پتری مثل  $PN = (P, T, F)$  یک شبکه‌ی جریان کار نامیده می‌شود، اگر و تنها اگر شرایط زیر برقرار باشد:

۱. یک مکان متمایز مثل  $i$  از مجموعه  $P$  (که مکان اولیه نامیده می‌شود) وجود دارد، به طوری که هیچ یال ورودی ندارد؛ یعنی مجموعه ورودی‌های مکان  $i$  (که با  $i$  نشان داده می‌شود) تهی باشد.

۲. یک مکان متمایز مثل  $o$  از مجموعه  $P$  (که مکان نهایی نامیده می‌شود) وجود دارد که هیچ یال خروجی ندارد؛ یعنی  $o$ . تهی باشد.

۳. هر مکان و هر انتقالی در این شبکه، در مسیری از مکان اولیه تا مکان نهایی قرار دارد.

هر توکنی که در مکان <sup>۴۷</sup> قرار می‌گیرد، بیانگر موردنی است که در خصوص آن، فرآیند شروع نشده‌است و در صورتی که به مکان نهایی <sup>۰</sup> برسد، به این معنا خواهد بود که فرآیند در خصوص آن به اتمام رسیده‌است. برای هر کدام از انتقال‌ها در این شبکه، می‌توان یک قاعده تعريف کرد و بر اساس این قاعده، می‌توان هر کدام از موارد (که متناظر با هر کدام‌شان یک توکن در شبکه موجود است) را بر اساس ویژگی‌هایشان، از مسیر معینی عبور داد. به این قواعد، قاعده‌ی تصمیم <sup>۴۷</sup> گفته می‌شود.

## ۵-۲ شبکه‌های پتری جریان کار

شبکه‌های پتری، به تنها‌ی نمی‌توانند تمامی ویژگی‌های یک فرآیند کسب و کار را بازنمایی کنند و در بیان برخی ویژگی‌ها، دارای محدودیت هستند؛ از سوی دیگر، نشان دادن برخی ساختارها با استفاده از شبکه‌های پتری کلاسیک (در بخش ۳-۲ مورد بررسی قرار گرفت) می‌تواند پیچیده باشد؛ بنابراین در طی تحقیقات مختلف، این شبکه‌ها گسترش پیدا کردن و مفاهیمی مانند رنگ به آن‌ها وارد شد و در پی آن، انواع دیگری از شبکه‌های پتری مثل شبکه‌های پتری زمانی <sup>۴۸</sup>، شبکه‌های پتری با گسترش زمانی <sup>۴۹</sup> و شبکه‌های پتری با گسترش سلسله مراتب به وجود آمدند. به این شبکه‌های پتری، شبکه‌های پتری سطح بالا گفته می‌شود [۱۱]. در نهایت، می‌پردازیم به یکی از گسترش‌های شبکه‌های پتری که به وسیله‌ی آن، مدل‌سازی فرآیندهای کسب و کار، با پیچیدگی کمتری قابل بازنمایی و پیاده‌سازی است. این دسته از شبکه‌های پتری، به شبکه‌های پتری جریان کار <sup>۵۰</sup> موسوم هستند. یک شبکه‌ی پتری جریان کار نوعی شبکه‌ی پتری است که محدودیت‌های جدید و نیز، چهار نماد جدید در آن تعريف شده‌است و قواعد فعال‌سازی و شلیک، در آن به همان ترتیب قبل، تعريف می‌شود.

در شبکه‌های پتری جریان کاری، برای اتصال و انشعاب شاخه‌ها و اعمال قواعد تصمیمی، نمادهای جدیدی معرفی شده‌است که با استفاده از آن، می‌توان ساده‌تر از شبکه‌های پتری کلاسیک، مبادرت به توصیف یک فرآیند کسب و کار نمود.

<sup>47</sup>Decesion Rule

<sup>48</sup>Colored Petrinets

<sup>49</sup>Time Extended Petri nets

<sup>50</sup>Workflow Petrinet

## انشعاب و اتصال در شبکه‌های پتری جریان کار

همان‌طور که پیش‌تر دیدیم، در شبکه‌های پتری جریان کار، تعیین مسیر توکن‌ها (مورددها) بر اساس ویژگی‌های آن‌ها و بر اساس شرط‌های تصمیم‌گیری موجود بر روی انتقال‌ها (که در برخی متون به قاعده‌ی شلیک موسوم بود)، انجام می‌شد؛ بنابراین، برای سادگی در نمایش و وضوح بیشتر، برای نمایش انتقال‌هایی که موجب انشعب و یا اتصال می‌شوند، نمادهای جدیدی تعریف شده‌است که در شکل ۱۲-۲ قابل مشاهده هستند. در ادامه توضیحاتی مختصر در خصوص هر کدام از این انتقال‌ها ارائه می‌شود.

**انشعاب عطفی**<sup>۵۱</sup> این انتقال در شکل ۱۲-۲-الف نشان داده شده‌است. در صورت شلیک این انتقال، یک توکن از ورودی حذف شده و بر روی تمامی مکان‌های خروجی، یک توکن ظاهر می‌شود.

**اتصال عطفی**<sup>۵۲</sup> این انتقال در شکل ۱۲-۲-ب نشان داده شده‌است. در صورت شلیک این انتقال، یک توکن از تمامی ورودی‌ها حذف شده و بر روی تمامی مکان خروجی، یک توکن ظاهر می‌شود. در صورتی ادامه‌ی فرآیند میسر است که در تمامی ورودی‌ها، حداقل یک توکن وجود داشته باشد.

**انشعاب فصلی**<sup>۵۳</sup> در صورتی که این انتقال فعال شود، ادامه‌ی فرآیند، بر مبنای اطلاعات موجود در توکن و شرط تصمیم‌گیری انتقال، از روی حداقل یکی از شاخه‌های خروجی میسر است. (۱۲-۲-پ)

**اتصال فصلی**<sup>۵۴</sup> در صورتی که حداقل یکی از ورودی‌های این انتقال حاوی توکن باشد، این انتقال فعال می‌شود (۱۲-۲-ت)؛ با این حال می‌توان بر روی انتقال‌ها، قاعده‌ی تصمیم تعریف کرد و ادامه‌ی فرآیند را منوط به فعال‌سازی ورودی‌های خاصی از انتقال، نمود.

## ۶-۶ خطاهای معنایی در فرآیندهای کاری

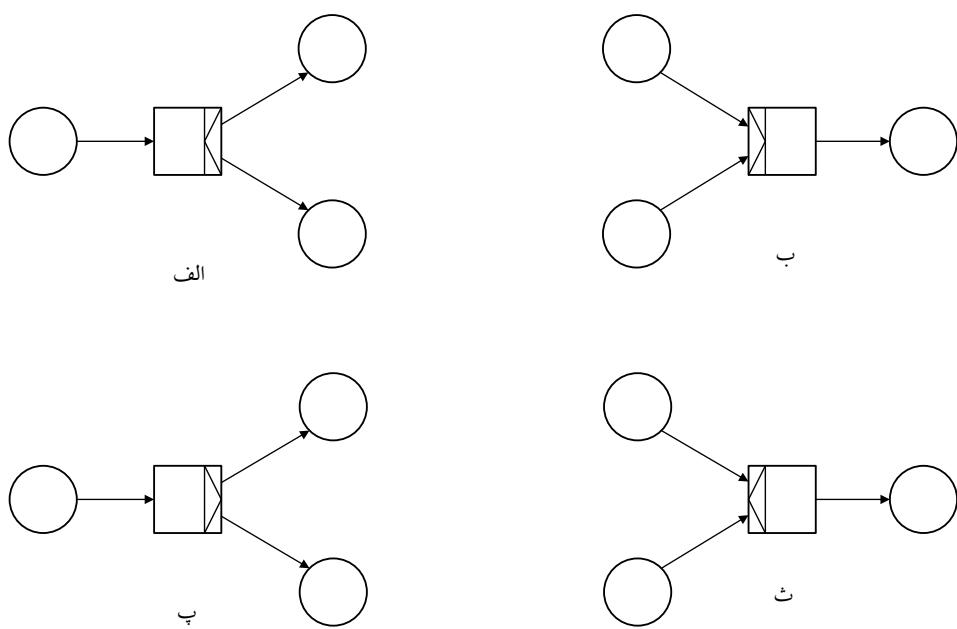
در این بخش، خطاهای معنایی را در فرآیندهای کاری، معرفی می‌کنیم. وقوع این خطاهای در فرآیندهای کاری، منجر به نقض صحت در فرآیند کاری می‌شود و سعی داریم که با صحت سنجی یک فرآیند کاری، وجود این خطاهای را در یک فرآیند، پیش‌بینی کنیم و یا از نبود آن‌ها در یک فرآیند، اطمینان حاصل

<sup>51</sup> And Split

<sup>52</sup> And Join

<sup>53</sup> OR Split

<sup>54</sup> OR Join



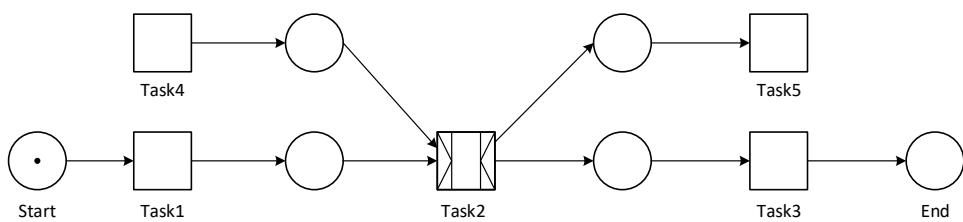
شکل ۱۲-۲: نمادهای جدید در شبکه‌های پتری جریان کار

کنیم. همان‌گونه که در ادامه مشاهده خواهد شد، برای کشف بسیاری از خطاهای معنایی، نیازی به دانستن جزئیات فرآیند کاری وجود ندارد و تحلیل و بررسی آن‌ها، صرفا بر مبنای ساختار جریان کنترلی فرآیند انجام می‌شود [۳].

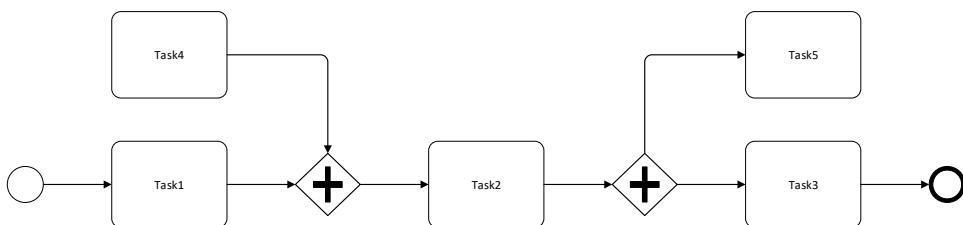
## ۱-۶-۲ وظایف بدون ورودی یا خروجی

زمانی که یک وظیفه، ورودی نداشته باشد مشخص نیست که چه زمانی اجرا می‌شود؛ همچنین، چنان‌چه وظیفه‌ای شرایطی برای خروج نداشته باشد، به تکمیل فرآیند کمکی نمی‌کند [۳]؛ چنان‌چه چنین مشکلی رخ دهد، شرایط حاکم بر شبکه‌ی جریان کاری نقض می‌شود؛ چون وظایفی وجود خواهند داشت که در مسیری از ابتدا تا انتهای فرآیند واقع نخواهند بود. مثالی از این موقعیت، در شکل ۱۳-۲ قابل مشاهده است [۳].

در یک مدل فرآیندی به زبان BPMN2.0 این خطا در صورتی رخ می‌دهد که فعالیت و یا وظیفه‌ای وجود داشته باشد که ورودی و یا خروجی نداشته باشد. مدل فرآیندی BPMN2.0 معادل با شکل ۱۳-۲ در شکل ۱۴-۲ نشان داده شده است. در این شکل، فعالیت‌های Task4 و Task5 فعالیت‌هایی هستند که با این خطا مواجه می‌شوند.



شکل ۲-۱۳: مثالی از وظایفی که ورودی Task4 یا خروجی Task5 ندارند.



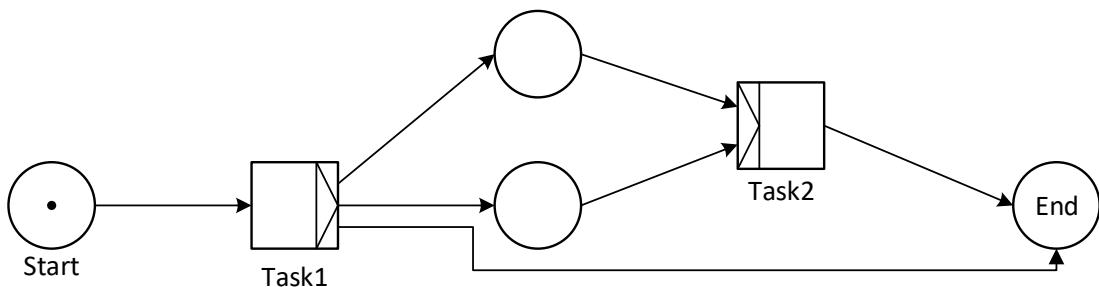
شکل ۲-۱۴: مثالی از وظایف مرده. وظیفه‌ی Task4 فاقد ورودی و وظیفه‌ی Task5 فاقد خروجی است.  
(در مدل فرآیندی BPMN2.0)

## ۲-۶-۲ کارهای مرده

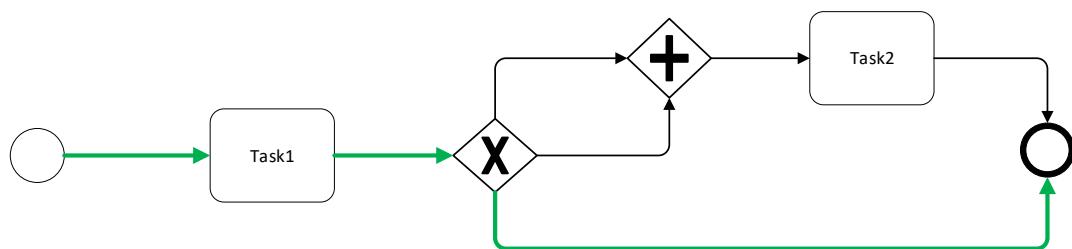
کارهای مرده<sup>۵۵</sup> به وظایفی اطلاق می‌شوند که هرگز انجام نخواهد شد.

نمونه ۲-۶-۱. به عنوان نمونه، در فرآید ایجاد شده در شکل ۲-۱۵، وظیفه‌ی Task2 هرگز اجرا نمی‌شود. چرا که توکن، فقط از یکی از مسیرهای بالایی یا پایینی می‌تواند عبور کند و برای اجرای این وظیفه، لازم است که مکان ورودی هر دو شاخه‌ی ورودی عطف اتصال به این وظیفه، حاوی حداقل یک توکن باشد. با توجه به این که این اتفاق رخ نمی‌دهد، در نتیجه این وظیفه نیز اجرا نخواهد شد [۳]. برای درک این خطا در مدل فرآیندی BPMN2.0 مجدداً این فرآیند را با زبان مدل‌سازی 2.0 BPMN ترسیم می‌کنیم (شکل ۲-۱۶). در این نمونه، تنها مسیری که منجر به پایان موفقیت‌آمیز اجرای فرآیند می‌شود، با رنگ سبز مشخص شده است. سایر مسیرها (که شامل وظیفه‌ی ۲ نیز می‌شود) به اتمام نمی‌رسند و فرآیند، قبل از دروازه‌ی موازی با بن‌بست مواجه شده و ادامه نمی‌یابد. در نتیجه، در این مدل فرآیندی، Task2 اجرا نخواهد شد.

<sup>۵۵</sup>Dead Tasks



شکل ۱۵-۲: مثالی از یک کار مرده در یک شبکه‌ی پتری جریان کار



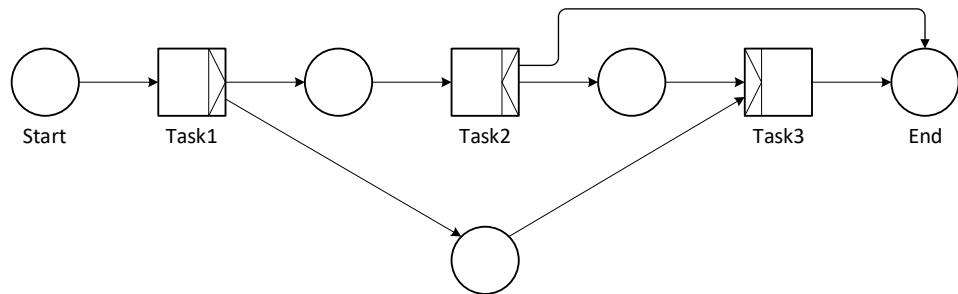
شکل ۱۶-۲: مثالی از یک کار مرده در زبان مدل‌سازی BPMN2.0

### ۳-۶-۲ بن‌بست

بن‌بست<sup>۵۶</sup> به وضعیتی اطلاق می‌شود که طی آن، توکن متناظر با یک مورد کاری، در یکی از مکان‌ها باقی بماند و هیچ‌گاه به مکان پایانی نرسد؛ به عبارتی، توکنی در یک مکان باقی بماند و توسط هیچ‌گاهی شلیک نشود؛ در چنین وضعیتی، توکن مورد بحث در یکی از مکان‌ها گیر می‌افتد و تحت هیچ شرایطی، امکان خروجی از آن مکان وجود نخواهد داشت [۳].

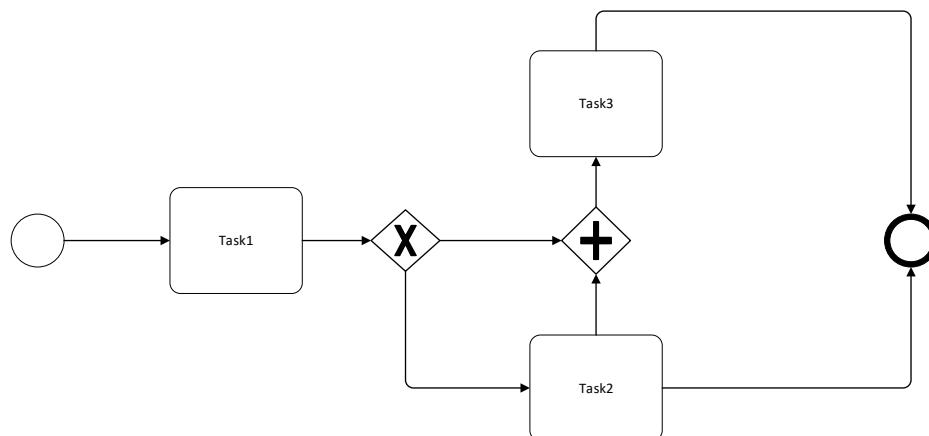
نمونه ۲-۶-۲. در شکل ۱۷-۲ نمونه‌ای از وقوع بن‌بست، در یک شبکه‌ی پتری جریان کار نشان داده شده است [۳]. در این شکل، در صورتی که توکنی از مکان شروع به مکان پایینی و توسط Task1 شلیک شود، نخواهد توانست توسط وظیفه‌ی Task3 شلیک شود؛ چرا که آن انتقال، از نوع اتصال عطفی است و در صورتی شلیک می‌شود که در تمامی مکان‌های ورودی آن، حداقل یک توکن وجود داشته باشد. مدل فرآیندی BPMN 2.0 معادل با این مدل فرآیندی، در شکل ۱۸-۲ ترسیم شده است؛ همان‌طور که مشاهده می‌شود، فرآیند در دروازه‌ی موازی با بن‌بست مواجه می‌شود؛ چرا که در صورتی گذر از این

<sup>۵۶</sup>Deadlock



شکل ۲-۱۷: مثالی از وقوع بن‌بست در یک مدل فرآیندی ایجاد شده با استفاده از شبکه‌های پتری جریان کار [۳]

دروازه انجام می‌شود که تمامی ورودی‌های دروازه‌ی موازی فعال باشد. حال آن‌که دروازه‌ی انحصاری واقع در قبل از آن، اجازه‌ی چنین اتفاقی را نمی‌دهد؛ در نتیجه، وظیفه‌ی Task3 به هیچ عنوان انجام نخواهد شد.



شکل ۲-۱۸: مثالی از وقوع بن‌بست در یک مدل فرآیندی ایجاد شده با استفاده از مدل فرآیندی BPMN 2.0

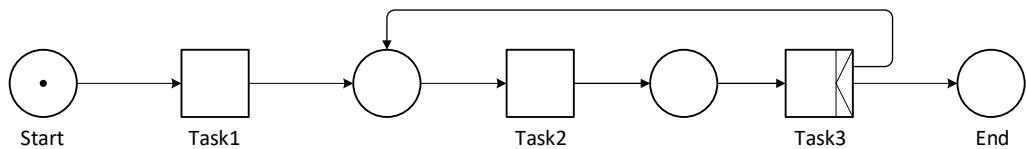
## ۴-۶-۲ بن‌بست حلقوی

بن‌بست حلقوی<sup>۵۷</sup> به پدیده‌ای اطلاق می‌شود که در آن، توکن‌هایی به مکان پایانی برسند، درحالی که توکن‌هایی در مکان‌های دیگری از فرآیند وجود دارند و در یک حلقه، دائما در حال چرخش هستند؛ به

<sup>57</sup>Livelock

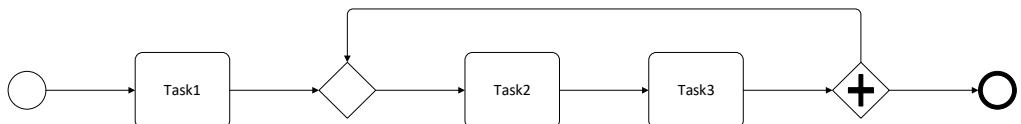
عبارت دیگر، با وجود یک بنبست حلقوی در فرآیند، یک مورد کاری (که در شبکه‌های پتری جریان کاری با یک توکن بازنمایی می‌شوند) در یک چرخه به دام می‌افتد و امكان خروج از چرخه را ندارند [۳].

نمونه ۶-۳. به عنوان نمونه، در فرآیند ترسیم شده در شکل ۱۹-۲ یک بنبست حلقوی رخ داده است. در این مثال، توکن از طریق Task1 به Task2 شلیک می‌شود. بعد از رسیدن به Task3 با توجه به این که به یک انشعاب عطفی می‌رسد، به تمامی مکان‌های خروجی Task3 منتقل می‌شود. درنتیجه، یک توکن در مکان پایانی و یک توکن، در مکان پیش از Task2 خواهیم داشت؛ درنتیجه، تعداد توکن‌های مکان پایانی تا بی‌نهایت می‌تواند رشد کند و یک توکن، دائماً وظایف Task2 و Task3 را تکرار کند و هیچ‌گاه از این حلقه خارج نشود [۳]. اگر همین فرآیند، با استفاده از زبان مدل‌سازی BPMN2.0 ترسیم شود، به شکل ۲۰-۲ می‌رسیم. در این شکل مشاهده می‌شود که دو وظیفه Task2 و Task3 دائم در حال



شکل ۱۹-۲: مثالی از وقوع بنبست حلقوی در یک مدل فرآیندی ایجاد شده با استفاده از مدل شبکه‌های پتری جریان کار [۳]

تکرار هستند. چرا که با رسیدن اجرای فرآیند به دروازه‌ی موازی، هر دو خروجی دروازه فعال می‌شود و یک شاخه‌ی آن به ابتدای حلقه و یک شاخه‌ی دیگر آن به انتهای آن به انتها می‌رسد. با توجه به این که بین یکی از خروجی‌های دروازه‌ی موازی و ورودی آن حداقل یک مسیر وجود دارد، یک حلقه‌ی نامتناهی در فرآیند تشکیل شده است.



شکل ۲۰-۲: مثالی از وقوع بنبست حلقوی در یک مدل فرآیندی ایجاد شده با استفاده از مدل فرآیندی BPMN

## ۵-۶-۲ اجرای برخی از فعالیت‌ها بعد از پایان فرآیند

یک فرآیند مناسب، شروع و پایان مشخصی دارد؛ بنابراین، بعد از اتمام اجرای فرآیند برای یک مورد کاری، نباید هیچ توکن مشتق شده از آن مورد کاری، در سطح فرآیند باقی بماند؛ در نتیجه، هیچ وظیفه‌ای که مرتبط با مورد کاری پایان یافته است، نباید انجام شود؛ فرآیند توصیف شده در نمونه‌ی ۳-۶-۲ از این قاعده پیروی نمی‌کند؛ چرا که با اجرای آن و به علت وجود یک عطف انشعاب در بعد از Task3 تعداد توکن‌های موجود در مکان پایانی، تا بی‌نهایت می‌تواند افزایش یابد [۳].

## ۶-۶-۲ باقی ماندن توکن بعد از اتمام فرآیند

بعد از آن‌که یک مورد کاری به مکان پایانی رسید، در سطح فرآیند نباید توکنی باقی مانده باشد که به آن مورد کاری پایان یافته، ارجاع داشته باشد [۳]. در غیر این صورت، احتمال وقوع ناهنجاری‌هایی مثل بنبست حلقوی در داخل فرآیند، محتمل است.

## ۷-۲ صحت (Soundness) مدل‌های فرآیندی

### ۱-۷-۲ مفاهیم و تعاریف اساسی صحت

یک شبکه‌ی گردش کار دارای یک نقطه‌ی شروع یکتا و یک نقطه‌ی پایانی یکتا می‌باشد. همچنین، در یک شبکه‌ی جریان کار، تمامی فعالیت‌ها در مسیری از ابتدای فرآیند تا انتهای فرآیند، واقع هستند. این شرط باعث می‌شود تا وظیفه‌ی مرده<sup>۵۸</sup> نداشته باشیم و تمامی فعالیت‌های تعریف شده در فرآیند، اجرا شود. چنان‌چه یک شبکه‌ی پتری جریان کار این نیازها را برآورده نماید، شبکه‌ی جریان کار<sup>۵۹</sup> نامیده می‌شود [۳].

تعريف ۲-۱. چنانچه سه ویژگی زیر در خصوص یک شبکه‌ی جریان کار برقرار باشد، آن شبکه‌ی جریان کار صحیح<sup>۶۰</sup> خواهد بود [۳]:

- برای هر توکنی که در ابتدای اجرای فرآیند در مکان شروع قرار می‌گیرد، در انتهای اجرای فرآیند یک (و تنها یک) توکن در مکان انتهایی قرار خواهد گرفت. این شرط تضمین می‌کند که هر مورد

<sup>58</sup>Dead Task

<sup>59</sup>Workflow net

<sup>60</sup>Sound

کاری، در یک دوره‌ی زمانی با موفقیت تکمیل می‌شود.

- در انتهای اجرای فرآیند، فقط یک توکن در مکان خروجی وجود داشته باشد و تمامی مکان‌های دیگر خالی باشند. این شرط تضمین می‌کند که زمانی که یک مورد کاری تکمیل می‌شود، در فرآیند، هیچ مرجعی به آن وجود نخواهد داشت.
- برای هر انتقال (وظیفه)، امکان انتقال از حالت اولیه، به حالتی که آن انتقال (وظیفه) فعال باشد ممکن است. این شرط تضمین می‌کند که تمامی وظایف اجرا می‌شوند.

پیش از تعیین صحت فرآیند، لازم است که بررسی کنیم که آیا فرآیند داده شده، یک شبکه‌ی جریان کار هست یا خیر؟ ساده‌ترین کار، استفاده از گراف دستیابی<sup>۶۱</sup> است؛ اگرچه این کار می‌تواند در تشخیص صحت فرآیند راهگشا باشد، اما دارای پیچیدگی محاسباتی بالایی را دارد؛ همچنین در یافتن اجزایی که صحت فرآیند را خدشه‌دار کرده‌اند، کمک چندانی نمی‌کنند. بنابراین، در فصل بعد، الگوهایی معرفی خواهد شد که به وسیله‌ی آن، می‌توان با استفاده از ترکیبیات، فرآیندهایی صحیح را ساخت و با کشف این الگوها در ساختار یک فرآیند کسب و کار، در خصوص صحت فرآیندهای کسب و کار قضاوت کرد.<sup>[۳]</sup>.

**نکته ۷-۲.** اگرچه در این بخش، مفهوم صحت بر مبنای شبکه‌های پتری جریان کار گفته شده است؛ اما این مفهوم در هر زبان مدل‌سازی فرآیندهای کاری، قابل بیان است [۲].

## ۷-۲-۲ ویژگی‌های اساسی یک فرآیند صحیح

بر طبق آنچه که از تعریف ۷-۲ در خصوص ویژگی‌های اساسی قابل برداشت است، آن است که: «فرآیندی صحیح است که در آن تمامی انتقال‌ها بتوانند شلیک شوند و در پایان آن، فقط یک توکن در مکان پایانی قرار گرفته و سایر مکان‌ها خالی باشند.» برای آن که در پایان فرآیند، فقط یک توکن در مکان پایانی وجود داشته باشد و در عین حال سایر مکان‌ها خالی باشند، لازم است که تعداد توکن‌هایی که در مکان‌ها قرار می‌گیرند، محدود باشد و در هیچ کدام از این مکان‌ها، تعداد توکن‌ها بی کران نباشد. در مقاله‌ی مفصلی که توسط ریچارد زوراسکی و همکارانش در خصوص شبکه‌های پتری نوشته شده است [۱۳]، ویژگی‌های مختلفی برای یک شبکه‌ی پتری برشمرده شده است. دو ویژگی گفته شده، یعنی تضمین شلیک تمامی انتقال‌ها و کران‌دار بودن تعداد توکن‌های موجود در هر مکان، در قالب دو ویژگی زیر بیان شده است:

<sup>۶۱</sup>Reachability Graph

۱. زنده بودن<sup>۶۲</sup>

۲. کراندار بودن<sup>۶۳</sup>

در ابتدا، تعاریف دقیق این دو ویژگی را مطرح می‌کنیم:

**تعریف ۲-۷-۳.** یک شبکه‌ی پتری مثل  $PN = (P, T, F)$  زنده<sup>۶۴</sup> است، اگر و تنها اگر هر انتقالی مانند  $T \in T$  بتواند به تعداد دفعات دلخواهی شلیک شود [۳].

**تعریف ۲-۷-۴.** یک شبکه‌ی پتری مثل  $PN = (P, T, F)$  کران‌دار است، اگر و تنها اگر تعداد توکن‌ها در هر مکان مثل  $P \in p$ ، از عدد صحیحی مثل  $k \in \mathbb{N}$  کم‌تر یا مساوی باشد. به چنین شبکه‌ای گفته می‌شود [۱۳] و در آن، تعداد توکن‌ها تا بی‌کران رشد نمی‌کند [۳].

به عبارت بهتر، فرآیندی صحیح است که زنده و کران‌دار باشد. در منابع دیگری همچون [۲] و [۳] نیز بر این ویژگی‌ها به عنوان ویژگی‌های یک فرآیند صحیح، تاکید شده‌است؛ بنابراین، می‌توان تعریف زیر را برای یک مدل فرآیندی صحیح ارائه نمود:

**تعریف ۲-۷-۵.** یک شبکه‌ی پتری (و یا یک شبکه‌ی پتری جریان کار) مثل  $PN = (P, T, F)$  صحیح است، اگر و تنها اگر کران‌دار و زنده باشد.

## ۸-۲ مدل‌های فرآیندی عام و بیان صوری آن‌ها

تا کنون از تحقیقات صورت گرفته در حوزه‌ی مدل‌های فرآیندی سخن گفته و ضرورت استفاده از آن‌ها را بیان کرده‌ایم. برای ارائه‌ی یک راهکار مستدل، لازم است یک مدل فرآیندی را به صورت عام و با زبانی صوری، تعریف نماییم؛ در تعریف ۲-۸-۱ یک مدل فرآیندی، به صورت علم و از نگاه ماتیاس و همکارانش [۲] تعریف شده‌است:

**تعریف ۲-۸-۱.** یک مدل فرآیندی<sup>۶۵</sup> توصیف‌کننده‌ی ساختار مجموعه‌ای از فرآیندها<sup>۶۶</sup> را بازنمایی می‌کند که ساختاری مشابه دارند. هر مدل فرآیندی، به صورت گرافی بازنمایی می‌شوند که دارای

<sup>62</sup>Liveness

<sup>63</sup>Boundness

<sup>64</sup>Live

<sup>65</sup>Process Model

<sup>66</sup>Process Instance

مجموعه‌ای از گره‌ها و یال‌های جهت‌دار می‌باشد. یال‌ها، ارتباط بین گره‌های مدل فرآیندی را بازنمایی می‌کنند. مجموعه‌ی گره‌ها، به سه زیرمجموعه افزار می‌شوند که هر کدام، بازنمایی‌کننده‌ی نوعی از اجزای یک مدل فرآیندی می‌باشند:

**مدل‌های فعالیتی** مدل‌های فعالیتی<sup>۶۷</sup>، واحدهای کاری را که در طی یک فرآیند کاری و یا یک فرآیند کسب و کار باید انجام شوند، توصیف می‌کند.

**مدل‌های رویدادی** رویدادها، اتفاقاتی هستند که در طی یک فرآیند کاری ممکن است رخ دهند.

**مدل‌های دروازه‌ای** دروازه‌ها، ساختارهای کنترلی و جریانی فرآیند را توصیف می‌کنند و کارهایی مثل انشعاب و اتصال شاخه‌های فرآیند را انجام می‌دهند.

با شروع هر مورد کاری در هر فرآیند، از فرآیند یک نمونه ساخته می‌شود و هر مورد کاری که به فرآیند وارد شود، بر اساس آن نمونه‌ی ایجاد شده از فرآیند، اجرا می‌شود. در ادامه، یک تعریف صوری برای یک مدل فرآیندی، توسط ماتیاس بیان شده‌است [۲].

**تعریف ۲-۸-۲.** فرض کنید که  $C$  مجموعه‌ای از ساختارهای جریان کنترلی باشد؛ همچنین فرض کنید که  $P = (N, E, type)$  یک مدل فرآیندی باشد، به طوری که  $N$  مجموعه‌ی گره‌ها،  $E$  مجموعه‌ی یال‌ها و  $type$  تابعی باشد که هر مدل دروازه‌ای را به یکی از ساختارها، نسبت می‌دهد. در این صورت [۲]:

•  $N = N_A \cup N_E \cup N_G$  به گونه‌ای که مجموعه‌ی مدل‌های فعالیتی،  $N_E$  مجموعه‌ی مدل‌های

رخدادی و  $N_G$  مجموعه‌ی مدل‌های دروازه‌ای باشد.

• مجموعه‌ی  $E$  مجموعه‌ای از یال‌های جهت‌دار بین گره‌ها است، به گونه‌ای که  $N \times N \subset E$  و یک جریان کنترلی را بازنمایی می‌کند.

• تابع  $type : N_G \rightarrow C$  به هر مدل دروازه‌ای، یک ساختار جریان کنترلی را نسبت می‌دهد.

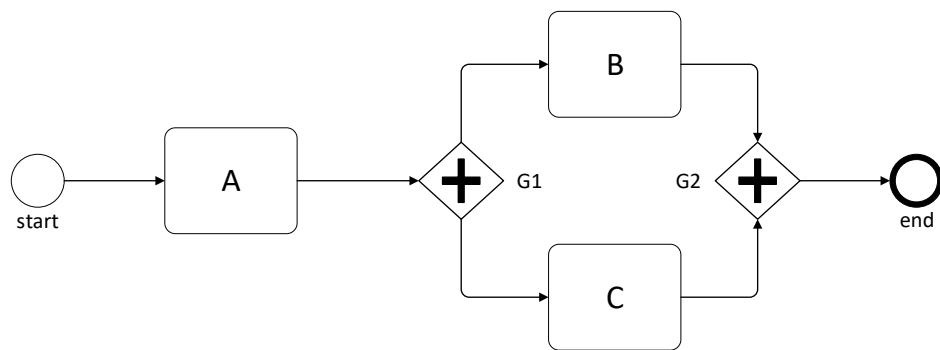
برای درک بهتر تعاریف گفته شده، نمونه‌ای از به کارگیری این تعاریف، تشریح خواهد شد.

**نمونه ۲-۳.** نمونه‌ای از یک مدل فرآیندی در شکل ۲۱-۲ نشان داده شده‌است. هر الگوی فرآیندی می‌تواند با زبان‌های مختلفی ترسیم و مدل‌سازی شود. اما در این مثال، از زبان مدل‌سازی BPMN 2.0 برای این منظور استفاده شده‌است. با توجه به مدل ارائه شده در شکل ۲۱-۲ خواهیم داشت:

<sup>67</sup>Activity Models

<sup>68</sup>Event Models

<sup>69</sup>Gateway Models



شکل ۲۱-۲: مثالی از یک مدل فرآیندی که با زبان مدل‌سازی BPMN 2.0 مدل‌سازی شده است.

$$N_A = \{A, B, C\} \quad N_E = \{start, end\} \quad N_G = \{G_1, G_2\}$$

$$E = \{(start, A), (A, G1), (G1, B), (G1, C), (B, G2), (C, G2), (G2, end)\}$$

$$type(G1) = \text{and split} \quad , \quad type(G_2) = \text{and join}$$

## ۹-۲ الگوهای جریان کنترلی

الگوهای جریان کنترلی، معیاری را به منظور هماهنگ‌سازی اجزای یک فرآیند فراهم می‌کنند. این الگوها، مستقل از زبان‌های توصیف و بیان فرآیندهای کاری (اعم از زبان مدل‌سازی BPMN 2.0 و شبکه‌های پتری جریان کار و...) هستند و می‌توانند در هر کدام از زبان‌های مدل‌سازی فرآیندهای کاری، مورد استفاده قرار گیرند [۲]. الگوهای جریان کنترلی شامل موارد زیر می‌باشند [۲]:

۱. الگوی دنباله‌ای<sup>۷۰</sup>

۲. الگوی انشعاب عطفی<sup>۷۱</sup>

۳. الگوی اتصال عطفی<sup>۷۲</sup>

۴. الگوی انشعاب انحصاری<sup>۷۳</sup>

<sup>70</sup>Sequence

<sup>71</sup>And Split Control Flow Pattern

<sup>72</sup>And Join Control Flow Pattern

<sup>73</sup>XOR Split Control Flow Pattern

۵. الگوی اتصال انحصاری<sup>۷۴</sup>

۶. الگوی انشعاب فصلی<sup>۷۵</sup>

۷. الگوی اتصال فصلی<sup>۷۶</sup>

نکته ۱-۹-۲. بر طبق گفته‌ی ماتیاس در کتاب مدیریت فرآیندهای کسب و کار [۲] الگوهای جریان کنترلی گفته شده در بالا، در تمامی زبان‌های مدل‌سازی فرآیندهای کاری، قابل استفاده بوده و موجود است.

نکته ۲-۹-۲. هر الگوی جریان کنترلی، با استفاده از یک مدل دروازه‌ای بازنمایی می‌شود. هر نمونه دروازه، دارای یک رخداد آغاز و یک رخداد پایانی است [۲]. آنچه که الگوهای جریان کنترلی را از یک دیگر متمایز می‌کند، ترتیب اجرای رخدادها در هر الگو می‌باشد و این ترتیب می‌تواند به واسطه‌ی زبان‌های مختلفی نشان داده شود.

در بخش قبلی، یک مدل مبتنی بر گراف به منظور نشان دادن ترتیب و تقدم و تاخر رخدادها در مدل‌های فرآیندی معرفی شد و از آن به عنوان گراف زنجیره‌ی رخدادی یا ساختار زنجیره‌ی رخدادی یاد شد. البته لازم به توضیح است که این نام‌گذاری، توسط نگارنده‌ی پژوهش حاضر انتخاب شده‌است و مدل، توسط [۲] ارائه شده‌است.

الگوهای جریان کنترلی<sup>۷۷</sup> معیاری را به منظور بیان هماهنگی فرآیندها، ارائه می‌دهند [۲]. همچنین، این الگوها معیارهایی را برای سنجش میزان گویایی زبان‌های مدل‌سازی، ارائه می‌دهند؛ طبق توضیحاتی که در خصوص معرفی مدل‌های فرآیندی BPMN 2.0 در بخش ۲-۲ داده شد، دروازه‌های انحصاری، نوع خاصی از دروازه‌های فصلی هستند (با این قید که فقط و فقط یک شاخه از آن فعال می‌شود) بنابراین، از لحاظ ساختاری، تمامی دروازه‌های عطفی و فصلی را در یک دسته در نظر خواهیم گرفت؛ در نتیجه، در زبان مدل‌سازی BPMN 2.0 الگوهای مربوط به اتصال و انشعاب انحصاری را به ترتیب با الگوهای اتصال و انشعاب عطفی، معدل و هم‌گروه در نظر خواهیم گرفت.

در هر الگوی جریان کنترلی، یک دروازه خواهیم داشت که تعیین کننده‌ی جریان کنترلی مربوطه می‌باشد و وجود آن، موجب به وجود آمدن آن الگوی جریان کنترلی می‌شود [۲].

<sup>74</sup>XOR Join Control Flow Pattern

<sup>75</sup>OR Split Control Flow Pattern

<sup>76</sup>OR Join Control Flow Pattern

<sup>77</sup>Control Flow Patterns

## ۱-۹-۲ گراف‌های زنجیره‌ی رخدادی

ماتیاس و همکارانش، در کتاب تالیفی خود در خصوص مدیریت فرآیندهای کسب و کار [۲] از رویکردی صوری به منظور بازنمایی الگوهای جریان کنترلی در فرآیندهای کاری، استفاده کرده‌اند. در این رویکرد، یک روش مدل‌سازی و مبتنی بر گراف‌ها، به منظور مدل‌سازی تقدم و تاخر فعال شدن، اجرا شدن و تمام شدن فعالیت‌ها، ارائه شده‌است.

در تعریف ۱-۸-۲ مدل‌های فرآیندی تعریف شده‌اند و همان‌طور که گفته شد، هر مدل فرآیندی شامل یک گراف است که گره‌های آن، به سه مجموعه‌ی مجزا افزار شدند:

۱. مدل‌های فعالیتی

۲. مدل‌های دروازه‌ای

۳. مدل‌های رخدادی

هر مدل فرآیندی یک ساختار کلی برای مجموعه‌ای از فرآیندها هستند. هر مورد کاری، بر روی یکی از این نمونه‌ها اجرا می‌شوند. بنابراین برای هر نمونه‌ی کاری، می‌توان از هر مدل فرآیندی، یک نمونه ایجاد کرد. به عنوان مثال، می‌توان از یک مدل فرآیندی مثل  $P$  یک نمونه فرآیند<sup>۷۸</sup> ایجاد نمود. فعالیت‌های یک مدل فرآیندی با حروف بزرگ انگلیسی نشان داده می‌شوند و نمونه‌های ایجاد شده از فعالیت‌های یک مدل فرآیندی، در یک نمونه فرآیندی با حروف کوچک انگلیسی نشان داده می‌شوند. در صورتی که از یک فعالیت، بیش از یک نمونه ایجاد شود، هر نمونه با یک زیرونده شماره گذاری می‌شود (همچون فعالیت‌های چند نمونه‌ای). معمولاً دروازه‌ها با حرف  $G$  نشان داده می‌شوند و در نمونه‌های ایجاد شده از یک مدل فرآیندی، دروازه‌ها با  $w$  نشان داده می‌شوند [۲].

### ساختار گراف زنجیره‌ی رخدادی

هر کدام از الگوهای جریان کنترلی، یک مدل فرآیندی هستند. در [۲] برای تحلیل صوری الگوهای جریان کنترلی، از ساختاری مبتنی بر گراف بهره برده شده‌است. با توجه به این‌که در این کتاب، این مدل در قالب یک تعریف ارائه نشده‌است، خواص آن در قالب یک تعریف گرددآوری می‌شود و از این پس در این پژوهش، از آن به عنوان گراف زنجیره‌ی رخدادی<sup>۷۹</sup> یاد خواهد شد. بر اساس [۲] یک گراف زنجیره‌ی رخدادی به صورت زیر تعریف می‌شود:

<sup>78</sup>Process Instance

<sup>79</sup>Event Chain Graph

**تعريف ۲-۹-۳.** برای یک نمونه فرآیندی مثل  $p$  از مدل فرآیندی  $P$ ، یک گراف زنجیره‌ی رخدادی مثل  $ECG = (\epsilon_p, E)$  شامل یال‌هایی جهت‌دار و گره‌هایی از تمام رخدادهای مدل فرآیندی  $P$  است؛ برای بدست آوردن این گراف، مدل‌های رخدادی، بدون تغییر از مدل فرآیندی  $P$  در مجموعه‌ی گره‌های این گراف ظاهر می‌شوند و مدل‌های فعالیتی، در قالب سه گره‌ی رخدادی (رخداد فعال شدن، رخداد اجرا شدن و رخداد اتمام) و به ترتیب، در این گراف ظاهر می‌شوند.

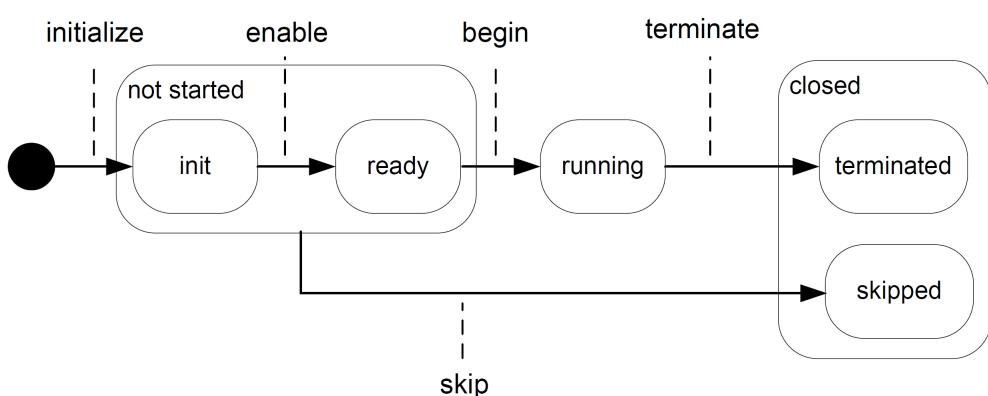
### رخدادهای یک مدل فعالیتی

مدل‌های فعالیتی، بخش مهمی از یک مدل فرآیندی هستند و واحدهای کاری آن را تشکیل می‌دهند. برای مدل‌سازی تقدم و تاخر هر کدام از فعالیتهای مدل فرآیندی در گراف زنجیره‌ی رخدادی، لازم است که هر کدام از فعالیت‌ها، به رخدادهایی مجزا تفکیک شوند تا بتوان آن‌ها را به عنوان رخدادهایی مجزا، در گراف زنجیره‌ی رخدادی به کار برد. این کار در [۲] انجام شده‌است. برای هر نمونه از مدل‌های فعالیتی مثل مدل فعالیتی  $a$  از فعالیت  $A$  در نمونه‌ی فرآیندی  $p$  از مدل فرآیندی  $P$  می‌توان سه رخداد را در نظر گرفت [۲] :

۱. رخداد فعال شدن<sup>۸۰</sup> نمونه‌ی مدل فعالیتی  $a$  که با  $e_a$  نشان داده می‌شود.

۲. رخداد شروع شدن<sup>۸۱</sup> نمونه‌ی مدل فعالیتی  $a$  که با  $b_a$  نشان داده می‌شود.

۳. رخداد پایان یافتن<sup>۸۲</sup> نمونه‌ی مدل فعالیتی  $a$  که با  $t_a$  نشان داده می‌شود.



شکل ۲-۲: نمودار انتقال حالت رخدادهای درونی یک نمونه مدل فعالیتی [۲]

<sup>80</sup>Enabling

<sup>81</sup>Begin

<sup>82</sup>Terminate

در شکل ۲۲-۲ مراحل اجرای یک نمونه از یک مدل فعالیتی (یک فعالیت) در یک نمونه از مدل فرآیندی مشاهده می‌شود؛ مراحل سه‌گانه‌ی گفته شده در این نمودار انتقال حالت قابل مشاهده است. در ادامه، قدری بیشتر در خصوص رخدادهای سه‌گانه‌ی گفته شده، بحث می‌شود:

**رخداد فعال شدن** بعد از مقداردهی اولیه‌ی نمونه‌ی مدل فعالیتی (فعالیت)، فعالیت آماده‌ی اجرا می‌شود. این رخداد با  $e_a$  نشان داده می‌شود. در طی این رخداد، فعالیت هنوز آغاز نشده‌است و جهت اجرا آماده‌سازی می‌شود. در صورتی که شرایط اجرای آن مهیا شود، فعالیت شروع به اجرا می‌کند و در غیر این صورت، از مرحله‌ی اجرا پرش<sup>۸۳</sup> می‌کند و به پایان رسیده و بسته می‌شود.

[۲]

**رخداد شروع شدن** بعد از فعال شدن یک نمونه از یک مدل فعالیتی، اجرای آن می‌تواند شروع شود و بدین‌وسیله، تکمیل می‌شود.

**رخداد پایان یافتن** بعد از اتمام اجرای نمونه مدل فعالیتی، اجرای فعالیت پایان یافته و نمونه مدل فعالیتی، بسته می‌شود.

**قرارداد ۴-۹-۲**. از این پس در برخی موضع، به جای عبارت «نمونه‌ای از مدل فعالیتی»، به اختصار از کلمه‌ی فعالیت استفاده خواهد شد.

### تعریف صوری تقدم و تاخر در گراف زنجیره‌ی رخدادی

در بخش‌های قبل، یک مدل فرآیندی به صورت صوری تعریف شد و سپس، مفهوم الگوهای جریان کنترلی مورد بررسی قرار گرفت. در بخش قبل، مدلی صوری به منظور بازنمایی تقدم و تاخر رخدادها، تحت عنوان گراف زنجیره‌ی رخدادی، بر اساس [۲] معرفی شده و در نهایت، هر فعالیت را به صورت دنباله‌ای از رخدادهای فعال‌سازی، شروع و پایان یافتن، تعریف کردیم.

هدف از بیان صوری به وسیله‌ی یک گراف زنجیره‌ی رخدادی، مدل‌سازی و بررسی تقدم و تاخر رخدادها نسبت به یک دیگر است؛ درنتیجه، لازم است تقدم و تاخر دو رخداد را نسبت به یک دیگر، به زبانی صوری تعریف نماییم. در [۲] تقدمها و تاخرها، بر اساس تعریف زیر در نظر گرفته شده‌است:

**تعریف ۵-۹-۲**. فرض کنید  $ECG = (\epsilon_p, E)$  یک گراف زنجیره‌ی رخدادی، برای یک نمونه‌ای مانند  $p$  از یک مدل فرآیندی به نام  $P$  باشد؛ به گونه‌ای که  $\epsilon_p$  مجموعه‌ی گره‌های گراف زنجیره‌ی رخدادی

<sup>83</sup>Skip

باشد و  $e_1$  و  $e_2$  دو رخداد از مجموعه‌ی  $\epsilon_p$  باشد؛ در این صورت : اگر و تنها اگر رخداد  $e_1$  زودتر از رخداد  $e_2$  رخ داده باشد و در این صورت، در گراف زنجیره‌ی رخدادها، یالی از  $e_2$  به  $e_1$  ترسیم خواهد شد.

## ۲-۹-۲ اساسی‌ترین الگوهای جریان کنترلی

در ادامه، به معرفی و بررسی الگوهای پایه‌ای جریان کنترلی می‌پردازیم. در بررسی آن‌ها و ترسیم گراف زنجیره‌ای رخدادی برای هر کدام از آن‌ها، به موارد زیر باید توجه شود:

- مدل‌های فعالیتی، با حروف بزرگ نشان داده می‌شوند؛ مثل حروف A، B، C و ... .
- نمونه‌های ایجاد شده از مدل‌های فعالیتی در نمونه‌های ایجاد شده از مدل‌های فرآیندی، با حروف کوچک نشان داده می‌شوند؛ مثل حروف a، b، c و ... .
- در مواردی که از یک مدل فعالیتی، بیش از یک نمونه ایجاد شود، هر نمونه با حروف کوچک و با زیروندهایی از یک دیگر متمایز می‌شوند. به عنوان مثال : فعالیت‌های  $a_1$ ،  $a_2$ ،  $a_3$  و ... نمونه‌هایی از مدل فعالیتی A هستند.

در ادامه و به منظور بررسی بهتر الگوهای جریان کنترلی، فرض می‌شود که P یک مدل فرآیندی و p یک نمونه‌ی ایجاد شده از آن می‌باشد و مجموعه‌ی رخدادهای آن مدل فرآیندی با  $\epsilon_p$  نشان داده می‌شود.

**یاداوری ۶-۹-۲.** هر کدام از فعالیت‌های موجود در مدل‌های فرآیندی را می‌توان به سه رخداد تقسیم کرد. این سه رخداد بر اساس شکل ۲-۲ شامل رخدادهای فعال شدن، شروع شدن و پایان یافتن هستند؛ به عنوان نمونه، فرض کنید که a یک نمونه‌ی فعالیتی از مدل فعالیتی A باشد. سه رخداد فعال شدن، شروع شدن و پایان یافتن، برای نمونه‌ی مدل فعالیتی a به ترتیب با  $e_a$ ،  $b_a$  و  $t_a$  نشان داده می‌شود.

## الگوی جریان کنترلی دنباله‌ای

الگوی جریان کنترلی دنباله‌ای، یکی از اساسی‌ترین و ساده‌ترین الگوهای جریان کنترلی می‌باشد و نشان‌دهنده‌ی ترتیب و توالی اجرای دو فعالیت در پی یک دیگر می‌باشد؛ برای هر دو مدل فعالیتی مثل a و b به گونه‌ای که فعالیت a پیش از فعالیت b انجام شود، یک الگوی جریان کنترلی دنباله‌ای به صورت زیر تعریف می‌شود:

**تعريف ۲-۹-۷.** برای دو فعالیت  $a$  و  $b$  از مجموعه مدل‌های فعالیتی یک مدل فرآیندی مثل  $p$ , الگوی جریان کنترلی دنباله‌ای بیان می‌کند که مدل فعالیتی  $b$  فعال می‌شود، اگر و تنها اگر مدل فعالیتی  $a$  به اتمام برسد (پایان یابد). به عبارت دیگر:

$$t_a \iff e_b$$

**نمونه ۲-۹-۸.** فرض کنید بر طبق تعریف ۲-۸-۲، مدل فرآیندی  $P = (N, E, type)$  دارای مدل‌های فعالیتی  $A$  و  $B$  بوده و دارای مدل دروازه‌ای  $G$  باشد؛ طبق همان تعریف موارد زیر در خصوص آن برقرار است:

$$A, B \in N_A$$

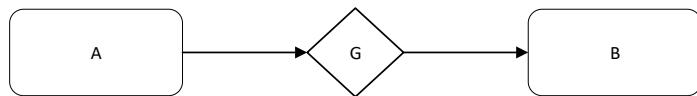
$$G \in N_G$$

$$\{(A, G), (G, B)\} \subset E \quad type(G) = sequence$$

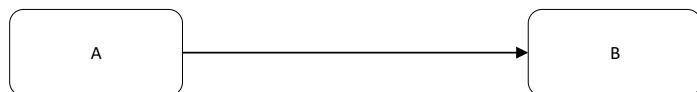
در عمل، برای هر مدل فعالیتی، سه رخداد فعال شدن، شروع شدن و پایان یافتن، بر اساس شکل ۲۲-۲ رخ می‌دهد. فرض کنید که  $p$  یک نمونه‌ی ایجاد شده از مدل فرآیندی  $P$  باشد؛ در این نمونه برای هر رخداد پایانی  $t_a \in \epsilon_p$  از نمونه مدل فعالیتی  $a$ ، رخدادی مثل  $e_b \in \epsilon_p$  وجود دارد که به موجب آن، مدل فعالیتی  $b$  فعال می‌شود؛ به گونه‌ای که داشته باشیم :

همان‌طور که گفته شد، متناظر با هر الگوی جریان کنترلی، یک مدل دروازه‌ای (دوازه) وجود دارد. متناظر با الگوی جریان کنترلی دنباله‌ای، یک دروازه با «نوع دنباله‌ای» در نظر گرفته شده است [۲]. در نمادگذاری فرآیندهای کسب و کار (مدل‌های فرآیندی BPMN 2.0) همانند شکل ۲۳-۲ از دروازه‌ی انحصاری و با درجه‌ی ورودی و خروجی ۱ برای نمایش الگوی جریان کنترلی دنباله‌ای استفاده می‌شود [۲]. با این حال مشابه با شکل ۲۴-۲ در مدل‌های فرآیندی BPMN 2.0 می‌توان این دروازه‌ها را حذف نمود و از یک کمان مستقیم به جای آن استفاده کرد [۲]. از این دروازه‌ی انحصاری درجه ۱، با نام دروازه‌ی دنباله‌ای نیز یاد شده و با نماد لوزی که داخل آن عبارت Seq نوشته شده است نیز نمایش داده می‌شود [۲].

فرض کنید که  $a$  و  $b$  نمونه‌هایی از مدل‌های فعالیتی  $A$  و  $B$  و در نمونه‌ی فرآیندی  $p$  از مدل فرآیندی  $P$  باشند. گراف زنجیره‌ی رخدادی این الگوی جریان کاری در شکل ۲۵-۲، به نقل از ماتیاس

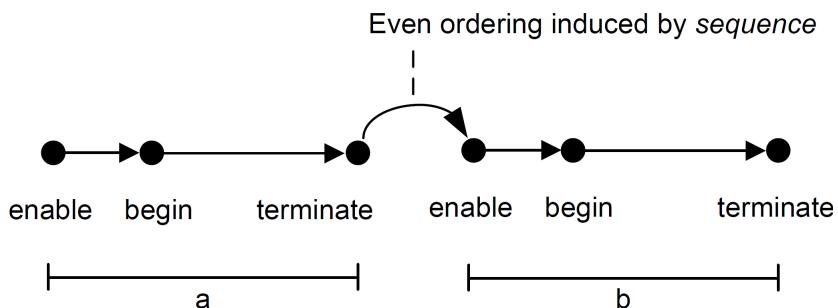


شکل ۲۳-۲: الگوی جریان کنترلی دنباله‌ای در مدل‌های فرآیندی BPMN 2.0 با استفاده از یک دروازه‌ی انحصاری درجه‌ی ۱ [۲]



شکل ۲۴-۲: الگوی جریان کنترلی با حذف دروازه‌ی انحصاری درجه‌ی ۱ (دوازه‌ی دنباله‌ای) در مدل‌های فرآیندی BPMN 2.0 [۲]

و همکارانش [۲] آورده شده‌است؛ همان‌طور که در این شکل مشاهده می‌شود، هر کدام از دو نمونه‌ی فعالیتی  $a$  و  $b$ ، به سه رخداد تقسیم شده‌اند: فعال شدن، شروع شدن و پایان یافتن؛ در نهایت، رخداد پایان یافتن نمونه مدل فعالیتی  $a$  پیش از فعال شدن نمونه مدل فعالیتی  $b$  آمده‌است و این امر بیان می‌دارد که فعالیت  $a$  پیش از فعالیت  $b$  انجام می‌شود.



شکل ۲۵-۲: گراف زنجیره‌ی رخدادی الگوی جریان کنترلی دنباله‌ای [۲]

### الگوی جریان کنترلی انشعاب عطفی

تعریف ۹-۹. یک انشعاب عطفی و یا یک انشعاب موازی<sup>۸۴</sup>، نقطه‌ای است که یک شاخه‌ی کنترلی به چندین شاخه‌ی کنترلی تقسیم شده و تمامی شاخه‌ها، به صورت هم‌روند اجرا می‌شوند [۲].

<sup>84</sup>Parallel Split

نمونه ۲-۹-۱۰. فرض کنید بر طبق تعریف ۲-۸-۲، مدل فرآیندی  $P = (N, E, type)$  دارای مدل‌های فعالیتی  $A$  و  $B$  بوده و دارای مدل دروازه‌ای  $G$  باشد؛ طبق همان تعریف موارد زیر در خصوص آن برقرار است:

$$A, B, C \in N_A$$

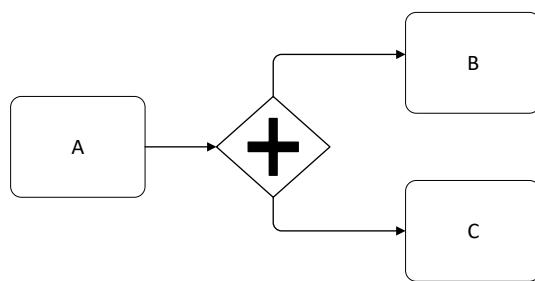
$$G \in N_G$$

$$\{(A, G), (G, B), (G, C)\} \subset E \quad type(G) = AndSplit$$

یک انشعباب عطفی، تضمین می‌کند که برای هر نمونه مدل فعالیتی مثل فعالیت  $a$  که پیش از انشعباب قرار دارد، با رخ دادن اتمام آن (یعنی رخ دادن  $t_a \in \epsilon_p$ ) تمامی فعالیت‌های بعد از انشعباب، مثل فعالیت‌های  $b$  و  $c$  به وقوع می‌پیوندند (یعنی رخدادهای  $e_b \in \epsilon_p$  و  $e_a \in \epsilon_p$  و  $e_c \in \epsilon_p$  رخ خواهند داد) [۲]. این بدان معنا خواهد بود که برای رخداد  $t_a \in \epsilon_p$  رخدادهای  $e_a, e_c \in \epsilon_p$  وجود خواهند داشت، به گونه‌ای که گزاره‌ی ۲-۱ برقرار باشد؛ مدل فرآیندی ۲.۰ BPMN بیان‌کننده‌ی این توصیف، در تصویر ۲۶-۲ آمده است.

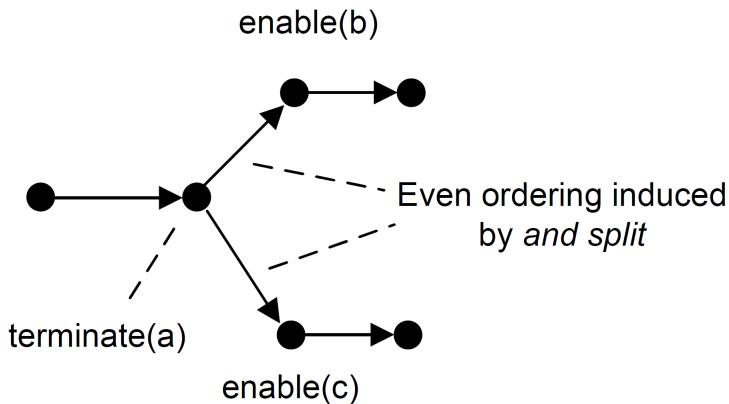
$$t_a < e_b \wedge t_a < e_c \quad (1-2)$$

با استفاده از استقراء می‌توان رابطه‌ی ۱-۲ را به بیش از دو شاخه‌ی کنترلی هم‌روند تعمیم داد.



شکل ۲-۲۶: نمونه‌ای از به کارگیری الگوی جریان کنترلی انشعباب عطفی در مدل فرآیندی BPMN ۲.۰ [۲]

همچنین، گراف زنجیره‌ی رخدادی حاصل از تحقق این الگوی جریان کنترلی بر روی مدل ارائه شده در تصویر ۲-۲۶، در تصویر ۲-۲۷ آمده است. همان‌طور که در این تصویر مشاهده می‌شود، با اتمام فعالیت  $a$ ، فعالیت‌های  $b$  و  $c$  به طور همزمان فعال می‌شوند.



شکل ۲۷-۲: گراف زنجیره‌ی رخدادی حاصل از تحقق این الگو در فرآیند ارائه شده در تصویر ۲۶-۲ [۲]

### الگوی جریان کنترلی اتصال عطفی

تعریف ۹-۱۱. یک اتصال عطفی و یا یک اتصال موازی<sup>۸۵</sup>، نقطه‌ای است که چندین شاخه‌ی کنترلی هم‌روند با یک دیگر ادغام کرده و هم‌گام می‌کند [۲]. در این الگوی جریان کنترلی فرض می‌شود که هر شاخه، فقط یک مرتبه اجرا می‌شود [۲].

نمونه ۹-۱۲. فرض کنید بر طبق تعریف ۲-۸-۲، مدل فرآیندی  $P = (N, E, type)$  دارای مدل‌های فعالیتی  $A$  و  $B$  بوده و دارای مدل دروازه‌ای  $G$  باشد؛ طبق همان تعریف موارد زیر در خصوص آن برقرار است:

$$B, C, D \in N_A$$

$$G \in N_G$$

$$\{(B, G), (C, G), (G, D)\} \subset E \quad type(G) = Andjoin$$

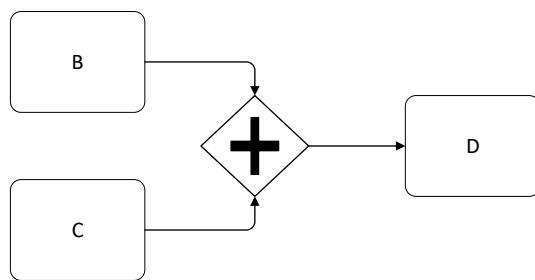
یک اتصال عطفی، تضمین می‌کند که برای هر فعالیتی مثل فعالیت‌های  $b$  و  $c$  که پیش از اتصال قرار دارند، با رخ دادن اتمام آن (یعنی رخ دادن توامان  $t_c \in \epsilon_p$  و  $t_b \in \epsilon_p$ ) فعالیت واقع در بعد از دروازه‌ی اتصالی، یعنی فعالیت  $d$  شروع به اجرا می‌شود. (یعنی رخدادهای  $e_d \in \epsilon_p$  رخ خواهد داد) [۲]. این بدان معنا خواهد بود که برای رخدادهای  $e_d \in \epsilon_p$  و  $t_c \in \epsilon_p$  رخداد  $t_b \in \epsilon_p$  وجود خواهد داشت، به گونه‌ای که

<sup>85</sup>Parallel Join

گزاره‌ی ۲-۲ برقرار باشد؛ مدل فرآیندی BPMN 2.0 معادل با این توصیف، در تصویر ۲۸-۲ آمده است.

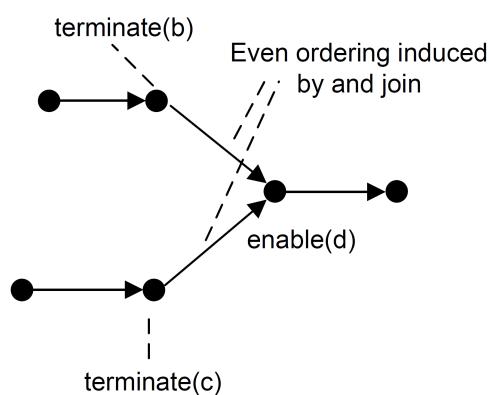
$$t_b < e_d \wedge t_c < e_d \quad (2-2)$$

با استفاده از استقرا، می‌توان رابطه‌ی ۲-۲ را به بیش از دو شاخه‌ی کنترلی هم‌روند تعمیم داد.



شکل ۲-۲: نمونه‌ای از به کارگیری الگوی جریان کنترلی اتصال عطفی در مدل فرآیندی BPMN 2.0 [۲]

همچنین، گراف زنجیره‌ی رویدادی حاصل از تحقق این الگوی جریان کنترلی بر روی مدل ارائه شده در تصویر ۲-۲، در تصویر ۲-۹ آمده است؛ همان‌طور که در این تصویر دیده می‌شود، آغاز فعالیت  $d$  منوط به پایان یافتن دو فعالیت  $b$  و  $c$  است و لازم است هر دو تمام شوند و چنان‌چه حداقل یکی از آن‌ها به اتمام نرسد، فعالیت  $d$  فعال نخواهد شد و در نتیجه، فرآیند ادامه نخواهد یافت.



شکل ۲-۹: گراف زنجیره‌ی رویدادی حاصل از تحقق این الگو در فرآیند ارائه شده در تصویر ۲-۲ [۲]

### الگوی جریان کنترلی انشعاب فصلی

تعريف ۲-۹-۱۳. یک انشعاب فصلی<sup>۸۶</sup>، نقطه‌ای در یک فرآیند است که تعدادی از شاخه‌های کنترلی انتخاب می‌شوند؛ برای این منظور، هر زیرمجموعه‌ی ناتهی از شاخه‌های خروجی انشعاب، می‌تواند برای اجرا انتخاب شود [۲].

نمونه ۲-۹-۱۴. فرض کنید بر طبق تعریف ۲-۸-۲، مدل فرآیندی  $P = (N, E, type)$  دارای مدل‌های فعالیتی  $A$  و  $B$  بوده و دارای مدل دروازه‌ای  $G$  باشد؛ طبق همان تعریف موارد زیر در خصوص آن برقرار است:

$$A, B, C \in N_A$$

$$G \in N_G$$

$$\{(A, G), (G, B), (G, C)\} \subset E \quad type(G) = OrSplit$$

یک انشعاب فصلی، تضمین می‌کند که برای هر نمونه مدل فعالیتی مثل فعالیت  $a$  که پیش از انشعاب قرار دارد، با رخ دادن اتمام آن (یعنی رخ دادن  $\epsilon_p \in \epsilon_p$ ) حداقل یکی از فعالیت‌های پس از آن، مثل فعالیت‌های  $b$  و  $c$  به فعال می‌شوند (یعنی رخدادهای  $e_b \in \epsilon_p$  و  $e_a \in \epsilon_p$  رخ خواهند داد) [۲]. این بدان معنا خواهد بود که برای رخدادهای  $t_a \in \epsilon_p$  وجود خواهند داشت، به گونه‌ای که گزاره‌ی ۳-۲ برقرار باشد؛ مدل فرآیندی BPMN 2.0 با این توصیف، در تصویر ۳۰-۲ آمده است.

$$t_a < e_b \vee t_a < e_c \quad (3-2)$$

با استفاده از استقراء، می‌توان رابطه‌ی ۳-۲ را به بیش از دو شاخه‌ی کنترلی نیز تعمیم داد.

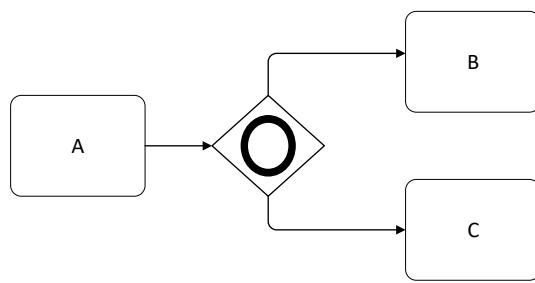
همچنین، گراف زنجیره‌ی رویدادی حاصل از تحقق این الگوی جریان کنترلی بر روی مدل ارائه شده در تصویر ۳۰-۲، در تصویر ۳۱-۲ آمده است. همان‌طور که در این تصویر مشاهده می‌شود، با اتمام فعالیت  $a$ ، ممکن است سه حالت رخ دهد:

۱. فقط فعالیت  $b$  فعال شود.

۲. فقط فعالیت  $c$  فعال شود.

---

<sup>86</sup>OR Split



شکل ۲-۳۰: نمونه‌ای از به کارگیری الگوی جریان کنترلی انشعاب فصلی در مدل فرآیندی BPMN [۲] 2.0

۳. هر دو فعالیت  $b$  و  $c$  فعال شوند.

### الگوی جریان کنترلی اتصال فصلی

تعریف ۲-۹. یک اتصال فصلی<sup>۸۷</sup>، نقطه‌ای است که چند شاخه‌ی کنترلی با یک دیگر ادغام می‌شوند [۲]. ادامه‌ی اجرای فرآیند از این نقطه، منوط به ادغام تمامی شاخه‌های فعال شده می‌باشد.

نمونه ۲-۹. فرض کنید بر طبق تعریف ۲-۸-۲، مدل فرآیندی  $P = (N, E, type)$  دارای مدل‌های فعالیتی  $A$  و  $B$  بوده و دارای مدل دروازه‌ای  $G$  باشد؛ طبق همان تعریف موارد زیر در خصوص آن برقرار است:

$$B, C, D \in N_A$$

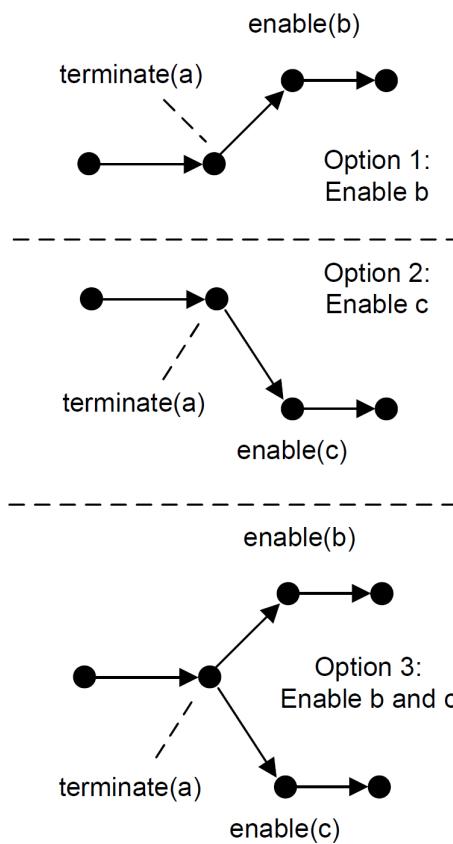
$$G \in N_G$$

$$\{(B, G), (C, G), (G, D)\} \subset E \quad type(G) = Orjoin$$

یک اتصال فصلی، تضمین می‌کند که برای هر فعالیتی مثل فعالیت‌های  $b$  و  $c$  که پیش از اتصال قرار دارند، با تکمیل اجرای رخدادهای موجود در شاخه‌هایی که فعال شده‌اند، فعالیت واقع در بعد از دروازه اتصالی، یعنی فعالیت  $d$  شروع به اجرا می‌شود. (یعنی رخدادهای  $\epsilon_p \in e_d$  رخ خواهد داد) [۲]. مدل فرآیندی BPMN 2.0 معادل با این توصیف، در تصویر ۲-۳۲ آمده است.

همچنین، گراف زنجیره‌ی رویدادی حاصل از تحقق این الگوی جریان کنترلی بر روی مدل ارائه شده در تصویر ۲-۳۲، در تصویر ۲-۳۳ آمده است؛ همان‌طور که در این تصویر دیده می‌شود، ادامه‌ی اجرای

<sup>87</sup>Parallel Join

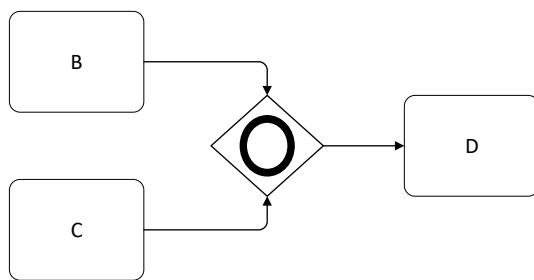


شکل ۲-۳۱-۲: گراف زنجیره‌ی رویدادی حاصل از تحقق این الگو در فرآیند ارائه شده در تصویر ۲-۳۰ [۲]

فرآیند، به سه طریق ممکن است:

۱. در صورتی که فعالیت  $b$  فعال باشد و به پایان برسد، به این معنا خواهد بود که شاخه‌ی بالایی فعال بوده و باید به اتمام برسد. در نتیجه، با پایان یافتن شاخه‌ی بالایی (که معادل با پایان یافتن رخداد  $b$  یا رخ دادن  $t_b$  است) فعالیت  $d$  فعال شده و فرآیند ادامه می‌یابد. با توجه به آن که شاخه‌ی پایینی فعال نشده‌است، فعال شدن فعالیت  $d$  وابسته به نمونه‌های مدل‌های رخدادی  $C$  نخواهد بود.

۲. در صورتی که فعالیت  $c$  فعال باشد و به پایان برسد، به این معنا خواهد بود که شاخه‌ی پایینی فعال بوده و باید به اتمام برسد. در نتیجه، با پایان یافتن شاخه‌ی بالایی (که معادل با پایان یافتن رخداد  $b$  یا رخ دادن  $t_c$  است) فعالیت  $d$  فعال شده و فرآیند ادامه می‌یابد. با توجه به آن که شاخه‌ی بالایی فعال نشده‌است، فعال شدن فعالیت  $d$  وابسته به نمونه‌های مدل‌های رخدادی  $B$  نخواهد بود.



شکل ۳۲-۲: نمونه‌ای از به کار گیری الگوی جریان کنترلی اتصال فصلی در مدل فرآیندی BPMN 2.0 [۲]

۳. در صورتی که فعالیت‌های  $b$  و  $c$  فعال باشند و به پایان برسند، به این معنا خواهد بود که شاخه‌های بالایی و پایینی فعال بوده و باید به اتمام برسند. در نتیجه، با پایان یافتن شاخه‌های بالایی (که معادل با پایان یافتن رخداد  $b$  یا رخ دادن  $t_b$  است) و پایینی (که معادل با پایان یافتن رخداد  $c$  یا رخ دادن  $t_c$  است) فعالیت  $d$  فعال شده و فرآیند ادامه می‌یابد. با توجه به آن که هردو شاخه‌های بالایی و پایینی فعال شده‌است، فعال شدن فعالیت  $d$  وابسته به به پایان یافتن هر دو فعالیت شاخه‌ها، خواهد شد.

### الگوی جریان کنترلی انشعاب انحصاری

تعریف ۲-۹-۱۷. یک انشعاب انحصاری<sup>۸۸</sup>، نقطه‌ای در یک فرآیند است که فقط یکی از شاخه‌های کنترلی، از بین چندین شاخه از انشعاب کنترلی، انتخاب می‌شوند؛ برای این منظور، فقط یکی از شاخه‌های خروجی انشعاب، می‌تواند برای اجرا انتخاب شود [۲].

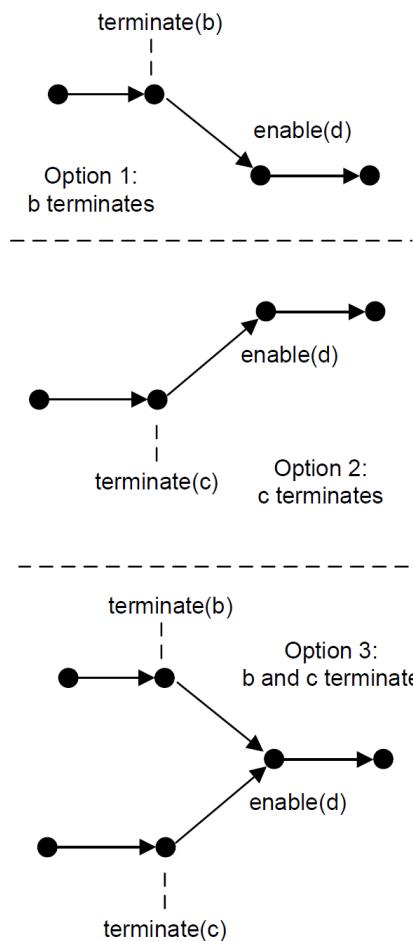
نمونه ۲-۹-۱۸. فرض کنید بر طبق تعریف ۲-۸-۲، مدل فرآیندی  $P = (N, E, type)$  دارای مدل‌های فعالیتی  $A$  و  $B$  بوده و دارای مدل دروازه‌ای  $G$  باشد؛ طبق همان تعریف موارد زیر در خصوص آن برقرار است:

$$A, B, C \in N_A$$

$$G \in N_G$$

$$\{(A, G), (G, B), (G, C)\} \subset E \quad type(G) = XorSplit$$

<sup>88</sup>Xor Split

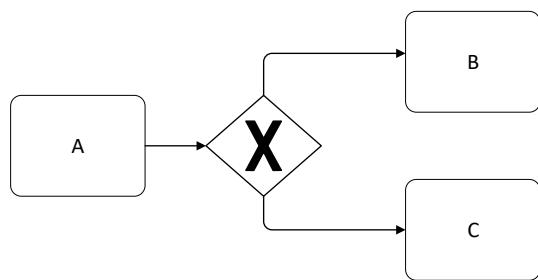


شکل ۹-۲: گراف زنجیره‌ی رویدادی حاصل از تحقق این الگو در فرآیند ارائه شده در تصویر ۳۲-۲ [۲]

یک انشعاب انحصاری، تضمین می‌کند که برای هر نمونه مدل فعالیتی مثل فعالیت  $a$  که پیش از انشعاب قرار دارد، با رخ دادن اتمام آن (یعنی رخ دادن  $e_p \in \epsilon_p$ ) فقط یکی از فعالیت‌های پس از آن، مثل فعالیت‌های  $b$  و  $c$  فعال می‌شوند (یعنی یکی از رخدادهای  $e_a \in \epsilon_p$  یا  $e_b \in \epsilon_p$  یا  $e_c \in \epsilon_p$  رخ خواهد داد) [۲]. این بدان معنا خواهد بود که برای رخداد  $t_a \in \epsilon_p$  در خصوص رخدادهای  $e_a, e_c \in \epsilon_p$  گزاره‌ی ۴-۲ برقرار خواهد بود:

$$(t_a < e_b \wedge (e_b \in \epsilon_p \iff e_c \notin \epsilon_p)) \vee (t_a < e_c \wedge (e_b \notin \epsilon_p \iff e_c \in \epsilon_p)) \quad (4-2)$$

مدل فرآیندی BPMN 2.0 با این توصیف، در تصویر ۳۴-۲ آمده است. با استفاده از استقراء، می‌توان رابطه‌ی ۳-۲ را به بیش از دو شاخه‌ی کنترلی نیز تعمیم داد.

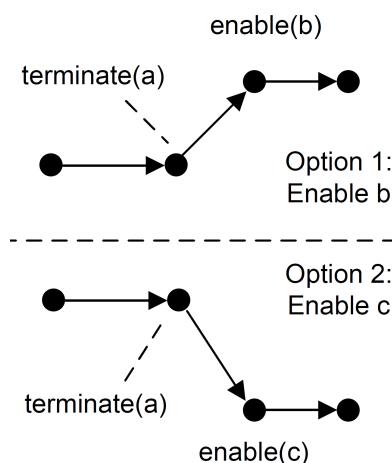


شکل ۲-۳۴: نمونه‌ای از به کارگیری الگوی جریان کنترلی انشعاب انحصاری در مدل فرآیندی BPMN [۲] 2.0

همچنین، گراف زنجیره‌ی رویدادی حاصل از تحقق این الگوی جریان کنترلی بر روی مدل ارائه شده در تصویر ۲-۳۴، در تصویر ۲-۳۵ آمده است. همان‌طور که در این تصویر مشاهده می‌شود، با اتمام فعالیت  $a$ ، ممکن است سه حالت رخ دهد:

۱. فقط فعالیت  $b$  فعال شود.

۲. فقط فعالیت  $c$  فعال شود.



شکل ۲-۳۵: گراف زنجیره‌ی رویدادی حاصل از تحقق این الگو در فرآیند ارائه شده در تصویر ۲-۳۴ [۲]

## الگوی جریان کنترلی اتصال انحصاری

تعریف ۲-۹-۱۹. یک اتصال انحصاری<sup>۸۹</sup>، نقطه‌ای در یک مدل فرآیندی است که چند شاخه‌ی کنترلی به یک دیگر می‌پیوندد و این اتصال، بدون هم‌گام‌سازی صورت می‌گیرد؛ این بدان معناست که ادامه‌ی

<sup>89</sup>Xor Join

اجرای فرآیند از نقطه‌ی اتصال، منوط به فعال شدن تمامی شاخه‌های کنترلی نمی‌باشد [۲].

در این الگوی جریان کنترلی، فرض می‌شود که فقط یکی از شاخه‌های ورودی، فعال بوده و متصل می‌شود.

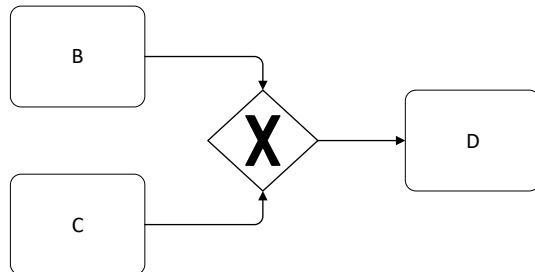
**نمونه ۲۰-۹.** فرض کنید بر طبق تعریف ۲-۸-۲، مدل فرآیندی  $P = (N, E, type)$  دارای مدل‌های فعالیتی  $A$  و  $B$  بوده و دارای مدل دروازه‌ای  $G$  باشد؛ بنابراین، موارد زیر در خصوص آن برقرار است:

$$B, C, D \in N_A$$

$$G \in N_G$$

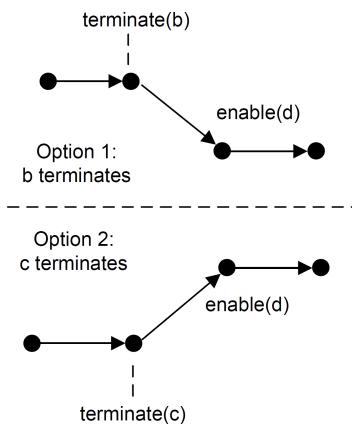
$$\{(B, G), (C, G), (G, D)\} \subset E \quad type(G) = XorJoin$$

یک انشعاب انحصاری، تضمین می‌کند که برای هر نمونه مدل فعالیتی مثل هرکدام از فعالیت‌های  $b$  یا  $c$  که پیش از اتصال قرار دارد، با رخ دادن اتمام دقیقاً یکی از آن (یعنی رخ دادن  $t_a \in \epsilon_p$  یا  $t_b \in \epsilon_p$  یا  $t_c \in \epsilon_p$ ) فعالیت موجود در بعد از اتصال مثل  $d$  فعال می‌شود (یعنی رخداد  $e_d \in \epsilon_p$  رخ خواهد داد) [۲].



شکل ۳۶-۲: نمونه‌ای از به کارگیری الگوی جریان کنترلی اتصال انحصاری در مدل فرآیندی BPMN [۲] 2.0

همچنین، گراف زنجیره‌ی رویدادی حاصل از تحقق این الگوی جریان کنترلی بر روی مدل ارائه شده در تصویر ۳۶-۲، در تصویر ۳۷-۲ آمده است. همان‌طور که در این تصویر مشاهده می‌شود، با اتمام دقیقاً یکی از فعالیت‌های  $b$  یا  $c$  فقط فعالیت  $d$  فعال می‌شود.



شکل ۲-۳۷-۲: گراف زنجیره‌ی رویدادی حاصل از تحقق این الگو در فرآیند ارائه شده در تصویر ۲ [۲]

## ۱۰-۲ تعاریف و مفاهیم اساسی در تضمین کیفیت نرم‌افزار

دغدغه‌ی تضمین کیفیت نرم‌افزار، در سطوح مختلفی قابل بررسی است. در سطح سازمان، نرم‌افزاری را با کیفیت می‌نامیم که با استانداردها و چارچوب سازمان مطابقت داشته باشد. در سطح یک پروژه‌ی نرم‌افزاری، نرم‌افزاری دارای سطح مطلوبی از کیفیت است که دارای ویژگی‌های زیر باشد [۸] :

- خروجی آن با استانداردهای تعیین شده مطابقت داشته باشد.
- بر طبق فرآیندی برنامه‌ریزی شده تولید شده و بر اساس فرآیندی برنامه‌ریزی شده، کیفیت آن تضمین شود.

بنابراین، تضمین کیفیت نرم‌افزار به صورت زیر قابل تعریف است [۸] :

تعریف ۲-۱۰-۱. تضمین کیفیت نرم‌افزار<sup>۹۰</sup> به معنای تعریف فرآیندها و استانداردهایی است که منجر به تولید محصولاتی با کیفیت بالا می‌شود و به دنبال آن، فرآیندهایی را برای تضمین کیفیت به خطوط تولیدی نرم‌افزارها، ارائه می‌دهد.

## ۱۰-۲ کیفیت نرم‌افزار

در کیفیت نرم‌افزار، به دنبال انطباق خروجی محصول با مشخصات آن هستیم. در تضمین کیفیت نرم‌افزار، قدری تحمل<sup>۹۱</sup> که منجر به ساخت محصولی تقریباً درست می‌شود قابل قبول است. این حد کم از خطأ

<sup>90</sup>Software Quality

<sup>91</sup>Tolerance

به علل زیر ممکن است رخ دهد [۸]:

- توصیف خیلی دقیق مشخصات سیستم ممکن نیست.
- الزامات سیستم، معمولاً حاصل توافق بین ذینفعان است و ممکن است که برخی از آن‌ها، با برخی از توافقات حاصله، موافق نباشند.
- برخی از ویژگی‌های نرم‌افزار، مثل ویژگی قابلیت مراقبت و نگهداری، قابل ارزیابی نیستند.

برای اظهار نظر در خصوص کیفیت نرم‌افزار، تیمی به منظور تضمین کیفیت نرم‌افزار باید تشکیل شود و تضمین کیفیت نرم‌افزار را بر مبنای مستندهای بدست آمده در فرآیند توسعهٔ نرم‌افزار و نتایج تست‌های صورت گرفته انجام دهند. در تضمین کیفیت نرم‌افزار، هم ویژگی‌های کارکردی و هم ویژگی‌های غیرکارکردی باید مورد ارزیابی قرار گیرد.

## ۲-۱۰-۲ فعالیت‌های تضمین کیفیت نرم‌افزار

در تضمین کیفیت نرم‌افزار، دو فعالیت اساسی وجود دارد [۸]:

۱. بازبینی<sup>۹۲</sup>

۲. بازررسی<sup>۹۳</sup>

بازبینی نرم‌افزار، به معنای بازبینی کیفی است که بر مبنای مستندهایی انجام می‌شود که در طول فرآیند توسعهٔ نرم‌افزار بدست آمده‌است. سپس، اطمینان حاصل می‌شود که استانداردهای کیفیت در طول فرآیند توسعهٔ نرم‌افزار، دنبال شده‌است. در طی این فعالیت، مستندها بررسی می‌شوند و مشکلات بالقوه شناسایی می‌شوند. با استفاده از بازبینی، مطمئن می‌شویم که استانداردهای کیفیت، در طول اجرای پروژه، در نظر گرفته شده‌است [۸].

در کنار فعالیت بازبینی، فعالیت دیگری در تضمین کیفیت نرم‌افزار وجود دارد که در طی آن، نقص‌های موجود در کد نهایی مورد کشف و بررسی قرار می‌گیرد. به این فعالیت، بازررسی گفته می‌شود. در طی این فعالیت، تست‌های تعریف شده انجام می‌شوند و به این وسیله، نقص‌های کد کشف می‌شود. این نقص‌ها، می‌توانند خطاهایی منطقی باشند که در شرایط خاصی خود را نشان می‌دهند. به گفتهٔ فاگان (۱۹۸۶)

<sup>92</sup>Review

<sup>93</sup>Inspection

به نقل از [۸] بیش از ۶۰ درصد از خطاهای یک برنامه را با استفاده از روش‌های غیر صوری، شناسایی کرد. از سوی دیگر، مایلز و همکاران (۱۹۸۷) پیشنهاد می‌کنند که با استفاده از یک رویکرد صوری، بیش از ۹۰ درصد از خطاهای را شناسایی نمود [۸].

### ۳-۱۰-۲ شاخص‌های کمی نرم‌افزار

شاخص‌های کیفی نرم‌افزار<sup>۹۴</sup> به شاخص‌هایی از نرم‌افزار اطلاق می‌شوند که به هر کدام از اجزای نرم‌افزار، یک مقدار عدد را نسبت می‌دهد. با استفاده از این مقادیر بدست آمده، می‌توانیم در خصوص کیفیت نرم‌افزار قضاوت کنیم [۸]. مثال‌هایی از معیارهای اندازه‌گیری نرم‌افزار:

۱. شاخص فاگ<sup>۹۵</sup> که میزان خوانایی کد را مشخص می‌کند.

۲. تعداد خطاهای گزارش شده.

۳. تعداد نفر/روز مورد نیاز برای پیش‌برد پروژه.

از سوی دیگر، اندازه‌گیری بسیاری از ویژگی‌های دیگر نرم‌افزار، مثل قابلیت اطمینان، عملاً ممکن نیست.

### ۱۱-۲ پلتفرم‌های توسعه‌ی کم کد

همواره یکی از معضلاتی که موفقیت یک ایده را تهدید می‌کند، طولانی شدن زمان مورد نیاز برای ورود محصول به بازار، به دلیل زمانبز بودن تولید سامانه‌های اطلاعاتی مورد نیاز می‌باشد؛ [۱۷]. مهم‌ترین عامل این تأخیر، طی کردن مراحل معمول ایجاد نرم‌افزار است که از مرحله‌ی شناخت نیازمندی‌ها، تا استقرار و مراقبت و نگهداری را در بر می‌گیرد. برنامه‌سازی با استفاده از پلتفرم‌های توسعه‌ی کم کد، به ایجاد کننده‌ی نرم‌افزار این امکان را می‌دهد که زمان کمتری را صرف تفکر در مورد رعایت قوانین نحوی زبان‌های برنامه نویسی کند و در عوض، تاکید بیشتری را بر جنبه‌های زیبایی شناختی، طراحی و کارکرد برنامه داشته باشد؛ در نتیجه، استفاده از پلتفرم‌های توسعه‌ی کم کد، مدت زمان صرف شده را برای عیب‌یابی و پیاده‌سازی، کاهش می‌دهد. پلتفرم‌های توسعه‌ی کم کد، مجموعه‌ای از ابزارها هستند که هم برای برنامه نویسان و هم برای افراد غیر برنامه نویس، قابل استفاده هستند؛ استفاده از این پلتفرم‌ها، تولید سریع نرم‌افزارها و تحويل برنامه‌های کاربردی تجاری را با حداقل تلاش برای برنامه نویسی در

<sup>94</sup>software Measure

<sup>95</sup>Fog

یک زبان برنامه‌نویسی، ممکن می‌سازد و نیازمند کمترین تلاش ممکن برای نصب، پیکربندی محیط‌ها، آموزش و پیاده‌سازی (به روش‌های سنتی) است. با رشد سریع تعداد شرکت‌ها، استفاده از پلتفرم‌های توسعه کم کد می‌تواند گام مهمی در جهت ایجاد نرم افزارهای تجاری باشد. این روش، تاکنون سریع‌ترین و احتمالاً ارزان‌ترین روش ایجاد و توسعه نرم‌افزار بوده است [۱۷]. پلتفرم‌های توسعه‌ی کم کد، برخلاف تکنیک‌های برنامه‌نویسی سنتی که مبتنی بر کدنویسی هستند، مبتنی بر واسطه‌های کاربری گرافیکی هستند و طراحی، ایجاد و یا توسعه‌ی نرم‌افزار، با استفاده از اشکال بصری، انجام می‌شود. در پلتفرم‌های توسعه‌ی کم کد و مبتنی بر زبان مدل‌سازی BPMN2.0 فرآیندهای کسب و کار، با استفاده از این زبان مدل‌سازی و ترسیم می‌شوند؛ سپس بدون نیاز به کدنویسی، این فرآیندها به نرم‌افزارهای قابل اجرا تبدیل خواهند شد و در بخش‌های مختلف سازمان و نیز، توسط مشتریان سازمان قابل استفاده خواهد بود.

## ۱۱-۱. اهمیت و ضرورت استفاده از پلتفرم‌های توسعه‌ی کم کد

گسترش پلتفرم‌های توسعه‌ی کم کد، می‌تواند تأثیر بالایی را در اشتغال متخصصان حوزه فناوری اطلاعات و ارتباطات داشته باشد؛ به عنوان مثال، تخمین زده می‌شود که در حدود ۲۷۵۰۰۰ فرصت شغلی خالی در بخش فناوری اطلاعات و ارتباطات در اتحادیه اروپا وجود دارد و تا سال ۲۰۲۰ این کمبود به ۱ میلیون فرصت شغلی کاهش خواهد یافت؛ طبق گفته‌ی شرکت تحقیقاتی گارتنر، تا سال ۲۰۲۱، تقاضا برای سیستم‌های اطلاعاتی افزایش خواهد یافت و رشد این تقاضا، پنج برابر سریع‌تر از توانایی بخش‌های فناوری اطلاعات، برای ارائه‌ی آن‌ها خواهد شد؛ زیرا تعداد کارکنان این حوزه، با سرعت کافی در حال افزایش نیست. این وضعیت بر تحويل به موقع نرم‌افزارها تاثیر می‌گذارد؛ علاوه بر این، همه‌ی راه حل‌های نوین فناوری اطلاعات، لزوماً مبتنی بر روش‌های استاندارد نیستند؛ چرا که امروزه، ذینفعان تجارت‌ها، به طور فزاینده‌ای از راه حل‌های سفارشی سازی شده توسط خود استفاده می‌کنند [۱۷] [۱۸] [۱۹]. پلتفرم‌های توسعه‌ی کم کد به این ذینفعان اجازه می‌دهند تا برنامه‌های کاربردی مورد نظر خود را بدون نیاز به کد نویسی و متناسب با راه حل‌های مورد نظرشان ایجاد کنند، در حالی که به سرعت بتوانند تغییرات مورد نظر خود را در آن لحاظ کرده و سیستم را با شرایط جدید، تطبیق دهند. برخی از پلتفرم‌های توسعه‌ی کم کد که در دنیا معروف هستند عبارتند از: Google，Microsoft Power Apps، Salesforce، Mendix، App Maker، TrackVia و Appian؛ همچنین در بازار ایران، تعدادی از شرکت‌ها، پلتفرم‌های توسعه‌ی کم کدی را، همانند ابزارهای فروشگاه‌ساز و سایت‌ساز و... ارائه کرده‌اند. استفاده از پلتفرم‌های توسعه‌ی کم کد، برای خودکار سازی فرآیندهای کسب و کار در سازمان‌ها، یک چالش دشوار و سخت است که

قابلیت حل مشکلات را برای بسیاری از مسائل حوزه‌ی تحقیقات و فناوری، فراهم می‌سازد و برای این منظور، به تحقیقات پایه‌ای، کاربردی و اکتشافی نیاز دارد.

رت راموس<sup>۹۶</sup>، مدیر منطقه‌ای عملیات فناوری اطلاعات آلگرو - میکروسیستمز، بیان می‌کند که امروزه، در یک زمینه‌ای که توسط مرزهای محدود و توأم با رقابت شدید در هزینه‌ها تعریف شده‌اند، سازمان‌ها به طور مداوم نیاز دارند که برای باقی ماندن در عرصه‌ی رقابت و داشتن عملکرد موثر، تا حد امکان در استفاده از روش‌های نوآورانه در خودکارسازی، پیش‌تاز باشند؛ چرا که برای اداره‌ی یک سازمان سودآور، از کاهش هزینه‌های مربوط به نیروی کار و تعمیر و نگهداری گرفته تا اطمینان از بهبود کیفیت و عملکرد بهتر کالاهای ساخته‌شده، خودکارسازی امری کلیدی است [۱۷]. متاسفانه، بسیاری از مدیران سازمان‌ها، مسئله خودکارسازی را جزو مسائل مهم نمی‌دانند. امروزه، لازم است که بسیاری از شرکت‌ها و سازمان‌ها، به دنیای کاملاً جدیدی از ایجاد نرم‌افزار و تحلیل داده‌ها وارد شوند؛ در این صورت، با چالش‌هایی کاملاً جدید، مانند موارد زیر مواجه خواهند شد:

- تحقیق و هزینه، بابت مهارت‌های تخصصی ایجاد نرم‌افزار.
- درک تجربه‌های کاربری<sup>۹۷</sup> در جهت ساخت برنامه‌های کاربردی کاربرپسند.
- نگهداری دستی برنامه‌های کاربردی و فرآیندهای کسب و کار خودکارسازی شده، برای اطمینان از حداقل شدن مجموع زمان‌هایی که نرم‌افزار، در دسترس نیست<sup>۹۸</sup>.

این چالش‌های به وجود آمده در حوزه‌های فنی، همانند چالش‌های ایجاد و نگهداری نرم‌افزار، دلیل عدم تصمیم شرکت‌ها برای خودکار کردن فرآیندهای کسب و کار بود؛ این دیدگاه، زمانی تغییر کرد که گزینه‌ی جدیدی پدیدار شد که می‌توانست تمام آن چالش‌ها را ساده کند و حتی به کسانی که دارای حداقل تجربه در ایجاد نرم‌افزار هستند، فرصت ایجاد نرم‌افزارهای کاربردی و سیستم‌های خودکار را بدهد. این امکانات جدید، توسط پلتفرم‌های توسعه‌ی کم کد فراهم شدند [۱۷]. ایجاد برنامه‌های کاربردی و براساس نمودارهای فرآیندهای کسب‌وکار، به روش‌های مدل‌سازی خاصی نیاز دارد. برخی ساختارهای پیچیده‌تر وجود دارند که می‌توانند در نمودارهای BPMN 2.0 رسم شوند، اما برای خودکارسازی مناسب نیستند. بنابراین، یک نمودار فرآیندی، باید به گونه‌ای تهیه شود که هر گونه درخواست در فرآیند، به طور منحصر به فردی به عنوان مجموعه‌ای از فرم‌ها، پیوست‌ها و اقدامات کاربر<sup>۹۹</sup> شناخته شود و با

<sup>96</sup>Rhett Ramos

<sup>97</sup>User Experience

<sup>98</sup>Down Time

<sup>99</sup>User Action

این کار، پس از واگذاری فعالیت به مسئول انجام آن، اطلاعات و ابزارهای لازم برای اجرای آن فعالیت، برای او فراهم شده و در اختیار او قرار گیرد؛ برای این منظور، ابزارهایی برای تهیه نمودارهای فرآیند کسب‌وکار به این روش، وجود دارد؛ یکی از این ابزارها، توسط [۱۷] طراحی و ایجاد شده است که نام آن، Auera BPM System است. همچنین، واتزکوسکی و کووالسکی، در مقاله‌ای تحت عنوان «تحلیل مقایسه‌ای چارچوب‌های مدیریت فرآیندهای کسب و کار» در سال ۲۰۱۷ [۲۰]، به مقایسه‌ی برخی از نرم‌افزارهای مشابه پرداختند. بدیهی است که نمودارهای فرآیندی ساخته شده، برای تولید برنامه‌های کاربردی کافی نیستند؛ چرا که داده‌های فرآیند مدل‌سازی شده، به اندازه‌ی فرآیند مدل‌سازی شده، مهم هستند؛ چرا که براساس مدل‌های داده‌ای، اشکال صفحه نمایش و عناصر واسط کاربری ایجاد می‌شوند و تعامل کاربر با نرم‌افزار، از همین طریق شکل می‌گیرد. تولید خودکار برنامه‌های کاربردی، از طریق تعریف فرآیندهای کسب و کار صورت می‌گیرد و اجرای برنامه‌های کاربردی، با تکمیل «اقدامات ۱۰۰» تعریف شده در فرآیند کسب و کار صورت می‌گیرد. این اقدامات را می‌توان به دو نوع استاندارد تقسیم نمود: اول، اقدامات عمومی که برای تمامی وظایف تعریف شده در فرآیندها، باید اجرا شود؛ دوم، اقداماتی که برای کاربرانی خاص باید صورت بگیرد. در یک پلتفرم توسعه‌ی کم کد، الگوریتم‌های تولید مولفه‌های رابط کاربری، باید قادر به آماده‌سازی یک برنامه‌ی کاربردی در سطوح مختلف پیچیدگی، از یک گردش کار ساده تا یک راه حل پیچیده و مرتبط با پایگاه‌های داده، حاوی داشبوردهای کاربر و مشتری، مدیریت محظوا، اتوماسیون کار، تعادل بار، رابط تلفن همراه، نظارت تجاری، بررسی پس از عمل و شبیه‌سازی فرآیند باشد؛ در نتیجه، مدل فرآیندی ایجاد شده با زبان مدل‌سازی BPMN 2.0، با نرم‌افزار تولید شده، یکپارچه خواهد بود؛ بنابراین، هر گونه تغییر در مدل‌های فرآیندی بر روش اجرای نرم‌افزار تولید شده و ارائه‌ی آن به کاربران نهایی، تاثیرگذار خواهد بود و هرگونه تغییر در مدل، معادل با تغییر در سیستم و عمل کرد نرم‌افزار تولید شده خواهد بود [۱۷]. با استفاده از پلتفرم‌های توسعه‌ی کم کد، نمودارهای فرآیندی به نرم‌افزارهایی قابل اجرا تبدیل می‌شوند و این نرم‌افزارها، نه تنها بر روی وب، بلکه بر بسترها دیگری که مبتنی بر وب نیستند نیز، می‌توانند اجرا شوند. متناظر با هر وظیفه‌ی <sup>۱۰۱</sup> تعریف شده در مدل فرآیندی، سیستم، فرم مربوطه را برای ورود داده تولید کرده و همچنین، مجموعه‌ای از داده‌های مورد نیاز را برای اجرای وظیفه، ایجاد می‌کند؛ این داده‌ها شامل داده‌های رابطه‌ای یا XML و انواع مختلف فایل‌ها به عنوان پیوست می‌باشند [۱۷].

<sup>100</sup>Activities<sup>101</sup>Task

## ۲-۱۱-۲ پلتفرم‌های توسعه‌ی کم کد و مبتنی بر BPMN 2.0

پلتفرم‌های توسعه‌ی کم کد، انواع مختلفی دارند و هر کدام، می‌توانند از روش مدل‌سازی منحصر به فردی برای مدل‌سازی استفاده کنند. در این تحقیق، پلتفرم‌های توسعه‌ی کم کدی مدنظر ما هستند که مبتنی بر زبان مدل‌سازی BPMN 2.0 کار می‌کنند؛ این دسته از پلتفرم‌های توسعه‌ی کم کد، مدل‌های فرآیندی 2.0 BPMN را به نرم‌افزارهای قابل اجرا تبدیل می‌کنند؛ این دسته از پلتفرم‌های توسعه‌ی کم کد، ویژگی‌های منحصر به فردی را دارند که در ادامه، مروری بر آن‌ها خواهیم داشت.

**یکپارچه بودن مدیریت فرآیندهای سازمانی** در یک پلتفرم توسعه‌ی کم کد، طراحی، اجرا و کنترل تمام فرآیندهای کسب‌وکار در یک مخزن مرکزی و یکپارچه رخ می‌دهد؛ این امر باعث می‌شود که بخش‌های مختلف یک سازمان، همانند بخش‌های مدیریتی، بخش‌های عملیاتی، فناوری اطلاعات و محققان مالی، در یک محیط یکپارچه کار کنند. این امر باعث جلوگیری از بروز ناسازگاری در تراکنش‌ها و فرآیندهای سازمان می‌شود و به مدیریت یکپارچه‌ی فرآیندهای سازمانی، کمک می‌کند [۱۷].

قابلیت دسترسی از راه دور این قابلیت، به کاربران اجازه می‌دهد که از هرجایی و از طریق یک مرورگر وب، به منابع و مخازن مورد نیازشان، دسترسی داشته باشند. معمولاً این پلتفرم‌ها، بر روی بیشتر سیستم‌های عامل و سخت‌افزارهای متداول، قابل اجرا هستند و می‌توانند توسط انواع مرورگرهای وب، پشتیبانی شوند (همانند اینترنت اکسپلورر، فایرفاکس، موزیلا، سافاری و اپرا) [۱۷].

امنیت زیرسیستم‌های امنیتی، مانع از دسترسی کاربرانی به سیستم می‌شوند که احراز هویت نشده‌اند. لازم است که امنیت، در لایه‌های مختلفی بررسی شود. این لایه‌ها، تمامی لایه‌ها را، اعم از از لایه‌ی پایگاه داده گرفته تا لایه‌ی سرور برنامه‌ی کاربردی را می‌بایست دربر گیرد؛ چرا که این پلتفرم‌ها و نرم‌افزارهای مشتق شده از آن‌ها، می‌توانند در سطح وب و یا در سطح سازمان‌ها و یا حتی در کانال‌های مختلف دیگر (مانند تلفن همراه و یا دستگاه‌های کارتخوان و...)، مورد استفاده واقع شوند [۱۷].

قابلیت اطمینان استفاده از تکنولوژی‌های تایید شده‌ی فناوری اطلاعات، امکان مدیریت روند صحیح فرآیندهای تولید شده را فراهم می‌آورد؛ علاوه بر آن، زیرسیستمی برای گزارش استثناهای سطح سیستم، می‌بایست وجود داشته باشد و با این کار، امکان شناسایی هرگونه مشکل در سیستم، ممکن می‌شود [۱۷].

قابلیت گزارش‌دهی در پلتفرم‌های توسعه‌ی کم کد و مبتنی بر زبان مدل‌سازی BPMN 2.0، به طور همزمان، تمام رویدادهای مرتبط با اجرای فرآیند ثبت می‌شوند (رویدادهایی مثل زمان شروع و توقف فرآیند و...). به این ترتیب، بجز گزارش‌های استاندارد موجود در سیستم، شما می‌توانید گزارش‌هایی تک منظوره را با استفاده از ابزارهای مبتنی بر SQL و یا ابزارهای تخصصی و مبتنی بر دیگر فناوری‌ها، ایجاد کنید [۱۷].

مدل سازی فرآیندهای کسب و کار وجه تمایز هر کسب و کار با سایر کسب و کارها، بر پایه‌ی فرآیندهای سازمانی آن کسب و کار استوار است. سیستم مدل سازی فرآیندهای کسب و کار در پلتفرم‌های توسعه‌ی کم کد و مبتنی بر زبان مدل سازی BPMN 2.0، مبتنی بر واسط کاربری بصری است که از استاندارد مدل سازی BPMN 2.0 پشتیبانی می‌کند و کاربران احراز هویت شده، می‌توانند فرآیندهای کسب و کار را بهبود دهند [۱۷].

نگاه سازمانی یکی از مهم‌ترین ریسک‌های بهبود سازمان‌ها، ریسک فناوری است. امکانات پلتفرم توسعه‌ی کم کد یوبی پراسس، تأمین دسترسی به اطلاعات کسب و کار را میسر کرده و امکان ادغام این اطلاعات را با دانش بیرونی، از طریق یکپارچگی با ابزارهای پیشرفته‌ی گزارش‌سازی و هوش سازمانی، ممکن می‌سازد. [۲۱]

اجرای فرآیندهای کسب و کار در پلتفرم‌های توسعه‌ی کم کد و مبتنی بر مدل‌های فرآیندی، مولفه‌ی موتور اجرای فرآیند (موتور جریان کار<sup>۱۰۲</sup>)، مسئول اجرای فرآیندهای کسب و کار تعریف شده و مدل سازی شده است. زمانی که نوع خاصی از یک فرآیند شروع می‌شود، سیستم به طور خودکار یک نمونه‌ی جدید از فرآیند را تولید کرده و کارها را به کاربران مناسب، برای تکمیل فهرست Flowable وظایف خود محول می‌کند [۲۲]. یکی از معروف‌ترین موتورهای جریان، توسط شرکت Flowable تولید شده است. موتور جریان Flowable، یک موتور اجرا کننده‌ی فرآیندهای کسب و کار است که با زبان برنامه نویسی جاوا ایجاد شده است. از مزایای این موتور، سبک وزن بودن آن است. این موتور به شما این امکان را می‌دهد که فرآیندهای کسب و کار خود را، مبتنی بر زبان مدل سازی BPMN2.0 تعریف کرده و سپس از روی تعریف، نمونه‌هایی را از فرآیند بسازید و کارهایی دیگر، همانند پرس و جو و [...] را بر روی فرآیندهای کسب و کار انجام دهید. این موتور جریان، بسیار منعطف است و می‌تواند در برنامه‌های کاربردی و سرویس‌ها، بصورت نهفته استفاده شود. با توجه

<sup>102</sup>Workflow Engine

به این که این موتور جریان، بصورت Jar file موجود است، می‌توان آن را به برنامه‌های کاربردی ایجاد شده با زبان جاوا، اضافه کرد و مورد استفاده قرار داد [۲۲].

**قابلیت جمع‌آوری گزارش‌ها و اطلاعات آماری** در پلتفرم‌های توسعه‌ی کم کد، مولفه‌ی نظارت بر فعالیت‌های تجاری، مجموعه‌ای از گزارش‌های از پیش تعریف شده را در خصوص فرآیندهای ایجاد شده و وضعیت آن‌ها، تولید می‌کند و امکان رصد کردن وضعیت فرآیندهای کسب و کار را فراهم می‌سازد؛ این ابزار، برای صاحبان فرآیند، مدیران و کاربران دیگر در نظر گرفته شده است. گزارش‌ها از داده‌های جمع‌آوری شده و دریافت شده، توسط موتور جریان بارگیری می‌شود؛ همچنین، امکان ایجاد گزارش‌های تک منظوره، با استفاده از ابزارهای موجود دیگر که از SQL پشتیبانی می‌کنند، وجود دارد [۱۷].

**واسط کاربری** در پلتفرم‌های توسعه‌ی کم کد و مبتنی بر زبان مدل‌سازی BPMN 2.0 و یوبی پراسس، واسط کاربری نرم افزارهای ایجاد شده، نه تنها فهرست وظایف کاربران را نمایش می‌دهد، بلکه اطلاعات مربوط به فرآیندهایی را که در حال حاضر اجرا شده‌اند و فرآیندهایی را که تکمیل شده‌اند، نشان می‌دهد [۱۷].

مدیریت سیستم مدیریت کاربران و موارد مرتبط به آن‌ها، همانند سطح دسترسی و □ موضوع مهمی است که در یک پلتفرم توسعه‌ی کم کد می‌باشد مورد توجه قرار گیرد [۱۷]. به عنوان مثال، در پلتفرم توسعه‌ی کم کد ایرانی یوبی پراسس، امکان اتصال سایر سامانه‌ها به یوبی پراسس از طریق Single Sign On وجود دارد؛ مهمترین امکانات این پلترم توسعه‌ی کم کد که مدیریت سیستم را تسهیل می‌کند، به شرح زیر است: [۲۳]

- مدیریت پیشرفته‌ی کاربران
- گروه‌بندی و مدیریت دسترسی کاربران
- ایجاد کارپوشه برای دسترسی به امور محله و سابقه‌ی فرآیندهای انجام شده توسط هر کاربر
- مدیریت جانشینی کاربران
- طراح ساختار سازمانی پیشرفته، با امکان ایجاد چندین ساختار چند درختی
- امکان ارجاع کار به افراد مختلف بر اساس ویژگی‌هایی همچون ساختار، گروه، ویژگی‌های فردی و یا سلسله مراتب سازمانی

• امکان استفاده از اطلاعات جغرافیایی در فرآیندهای ایجاد شده

**چند کanalه بودن** سازمان‌ها همواره به دنبال افزایش سطح ارتباط کسب‌وکار خود با مشتریان خود هستند؛ یکی از مؤثرترین روش‌های افزایش سطح ارتباط کسب و کار با مشتریان، ایجاد تنوع در کanalهای ارتباطی با مشتریان است. اگر در گذشته، داشتن یک مرکز تماس با خطوط تلفنی متعدد، برای تمایز یک سازمان از رقبای خود کافی بود، امروزه داشتن اپلیکیشن‌های وب، موبایل، کدهای دستوری USSD و ارائه خدمت از طریق پیام‌رسان‌هایی نظیر گپ، تلگرام، اینستاگرام و یا حتی کanalهای خاصی همچون دستگاه‌های کارت‌خوان و یا خودپردازها، می‌تواند موجب افزایش رضایتمندی مشتریان شود. بنابراین، پلتفرم‌های توسعه‌ی کم کد و مبتنی بر فرآیندهای کسب و کار، جهت افزایش سطح ارتباط خود با مشتریان، می‌بایست در معماری خود، الزاماتی را برای چند کanalه بودن فراهم سازند. در ادامه، به بررسی قابلیت‌های مورد انتظار در یک پلتفرم توسعه‌ی کم کد و چند کanalه، برای ارتباط چند کanalه با مشتریان، می‌پردازیم [۲۱].

**تجربه‌ی کاربری یکپارچه و هویت واحد کاربران** تجربه‌ی کاربری یکپارچه، به معنای آن است که یک کاربر با هویت منحصر به فرد خود، می‌تواند فرآیندی را از یک کanal شروع کرده و آن را در کanalهای دیگر ادامه دهد، به نحوی که از نظر ظاهر و منطق تعامل، احساس کند که با یک سامانه‌ی واحد در تعامل است. به عنوان مثال، بخشی از فرآیند را با استفاده از اپلیکیشن موبایل انجام دهد و ادامه‌ی آن را با یک دستگاه کارت‌خوان انجام داده و در نهایت، آن را در یک کanal دیگری مانند اپلیکیشن تحت وب، به اتمام برساند. همچنین، در یک سیستم چند کanalه، هویت یک کاربر، از نظر تمامی کanal‌ها باید واحد باشد [۲۱].

**تجربه‌ی کاربری آسان و مناسب** برای تحقق این منظور، می‌بایست یک معماری انحصاری و ابزار طراحی فرم چند کanalه در این پلتفرم‌ها تعییه شده باشد که فرم‌های آن، قابلیت ارائه بر روی انواع کanalهای ارتباطی را، با در نظر گرفتن ملاحظات زیر، داشته باشند [۲۱]:

۱. برای تسهیل تجربه‌ی کاربری در کanalهای غنی که شامل وب و موبایل هستند، از مدل‌های طراحی مناسب و ترجیحاً متداول، که به طور وسیع در برنامه‌های کاربردی مانند تلگرام و واتس‌اپ به کار گرفته شده‌اند استفاده شود تا کاربران در کمترین زمان و به آسانی، بر اساس تجربیات پیشین خود بتوانند از ابزارهای تولید شده، استفاده نمایند.
۲. همچنین برای کanalهای مبتنی بر تعاملات متنی، نظام تعاملی مشابه با اپلیکیشن‌های پرکاربرد در نظر گرفته شود تا یادگیری و استفاده از آن برای کاربران نهایی آسان باشد.

۳. به طور پیش فرض، سامانه چند زبانه باشد.

**پشتیبانی از رایانش ابری و معماری چند مستاجری** در رایانش ابری، از معماری چند مستاجری <sup>۱۰۳</sup> به صورتی گستردۀ استفاده می‌شود؛ زیرا با استفاده از این معماری، مدل‌های جدیدی از خدمات به مشتریان، قابل ارائه هستند. کاربرد این معماری در توانایی شرکت‌ها برای ارائه خدمات با کیفیت و ارزان قیمت، در چارچوب خدمات نرم‌افزاری و یا نرم‌افزار به عنوان سرویس <sup>۱۰۴</sup>، به نحوی است که آن را به یکی از محبوب‌ترین معماری‌ها، برای علاقه‌مندان به ارائه این خدمات در فضای ابری، تبدیل کرده است. بیزوتاب <sup>۱۰۵</sup>، نسخه‌ای از پلتفرم توسعه‌ی کم کد یوبی پراسس است که از این معماری، پشتیبانی می‌کند. در فضای چند مستاجری، چندین سازمان یا کسب‌وکار می‌توانند از یک نمونه‌ی منفرد <sup>۱۰۶</sup> از نرم‌افزار استفاده کنند؛ یعنی این نمونه، بر روی سرور اجرا می‌شود و به چندین سازمان سرویس می‌دهد. در این معماری، هر سازمان را یک مستاجر <sup>۱۰۷</sup> می‌نامیم. هر مستاجر، فضایی کاملاً اختصاصی داشته و از سایر مستاجرهای ایزوله بوده و مطلقاً هیچ یک از منابع یک کسب و کار بین مستاجرهای مختلف به اشتراک گذاشته نمی‌شود؛ به نحوی که گویا هر کدام از کسب و کارها، یک نسخه‌ی انحصاری از نرم‌افزار را برای خود نصب کرده‌اند. نرم‌افزارهای چند مستاجر، بسته به میزان انعطاف و توجهی که به مشتریان خود دارند، به مستاجرهای امکان اعمال طیفی از تغییرات و شخصی‌سازی را در بخش‌های مختلف اپلیکیشن می‌دهند؛ این تغییرات، معمولاً شامل رابط کاربری و یا قوانین کسب‌وکار هستند؛ اما شامل امکان تغییر کدهای اپلیکیشن اصلی نخواهد بود. بنابراین، استفاده از معماری چند مستاجری، مقرن به صرفه و اقتصادی است؛ زیرا هزینه‌های ایجاد، توسعه و نگهداری نرم‌افزار و زیرساخت‌های سخت‌افزاری، تقسیم می‌شود [۲۱].

<sup>103</sup> Multi-Tenant

<sup>104</sup> Software as a Service (SaaS)

<sup>105</sup> Bizotop

<sup>106</sup> Single Instance

<sup>107</sup> Tenant

## فصل سوم

### پیشینه‌ی پژوهش

## ۱-۳ مروری بر تحقیقات مرتبط با صحبت‌سنجی مدل‌های فرآیندی

### ۱-۱-۳ صحبت‌سنجی با استفاده از شبکه‌های پتری کلاسیک

همان‌طور که در بخش‌های قبل اشاره شد، مدل‌های فرآیندی BPMN2.0 می‌توانند حاوی خطاهای منطقی، همانند بن‌بست و بن‌بست حلقوی (انتظار مشغول) باشند. در این مدل‌های فرآیندی، گنگ بودن نمادها مشکل‌ساز است؛ حتی اگر قوانین نحوی، به طور جامع در جدول‌های استاندارد BPMN2.0 شرح داده شود، باز هم ممکن است منطق مدل ایجاد شده به درستی منتقل نشود. در سال ۲۰۰۸، تحقیقاتی توسط رمکو و داکمن [۴] انجام شد که در آن، بر تعریف یک مدل صوری<sup>۱</sup> برای دسته‌ی بزرگی از مدل‌های فرآیندی BPMN2.0 تمرکز دارد؛ این صوری‌سازی، بر اساس نگاشت مدل‌های فرآیندی BPMN 2.0 به شبکه‌های پتری کلاسیک انجام می‌شود. بسیاری دیگر از مدل‌های فرآیندی که با دیگر زبان‌های مدل‌سازی پیاده‌سازی شده‌اند، می‌توانند با استفاده از شبکه‌های پتری، صوری‌سازی شوند؛ در [۲۴] مروری بر نگاشت مدل‌های فرآیندی دیگر به شبکه‌های پتری، انجام شده است.

#### نقاط قوت و ضعف

شبکه‌های پتری، به عنوان ابزاری گرافیکی و مدلی ریاضی، محیطی را برای تحلیل صوری و مدل‌سازی فرآیندهای کاری و سیستم‌های صنعتی ارائه می‌کنند. یکی از مزایای مهمی که شبکه‌های پتری برای طراحان فرآیندها فراهم می‌کنند، امکان تحلیل مشخصات رفتاری، کارایی و شبیه‌سازی فرآیندها است [۱۳]. تمرکز روش ارائه شده در پژوهش رمکو و داکمن [۴] بر ترتیب فعالیتها<sup>۲</sup> و رخدادها<sup>۳</sup> در مدل فرآیندی BPMN است و اجزای زیر در تحلیل ایستای مدل فرآیندی، در نظر گرفته نمی‌شود(برای آشنایی با اجزای مدل‌های فرآیندی BPMN2.0 به ۲-۲ مراجعه شود):

- اجزای غیر رفتاری<sup>۴</sup> مثل مصنوعات و گروه‌ها.
- اجزای سازمانده مثل استخراجها و خطوط شناوری.

<sup>1</sup>Formal Model

<sup>2</sup>Activity

<sup>3</sup>Events

<sup>4</sup>Non Functional

## توضیحات اجمالی پیرامون روش پیشنهادی

در ادامه، به صورت اجمالی به ارائه‌ی توضیحاتی پیرامون راه حل به کار گرفته شده در [۴] می‌پردازیم.

پیش‌پردازش اولین کاری که انجام می‌شود، پیش‌پردازش مدل فرآیندی است. این کار به منظور ساخت یک مدل فرآیندی یکتا و خوش تعریف است. در پژوهش پایان‌نامه نیز از این روش‌ها استفاده خواهد شد و در بخش مربوطه، تشریح خواهد شد. بر اساس پژوهش رمکو و داکمن، یک مدل فرآیندی خوش تعریف را تعریف می‌کنیم:

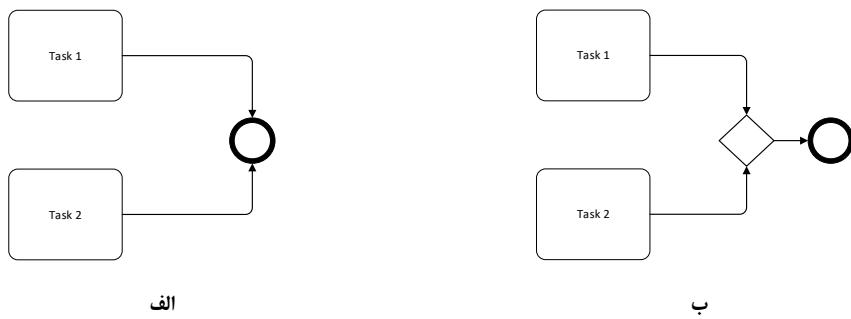
**تعریف ۱-۱-۳.** یک مدل فرآیندی خوش تعریف است، اگر و تنها اگر، دارای ویژگی‌های زیر باشد:

- رخداد شروع و رخداد استثناء، فقط باید یک یال خروجی داشته باشد.
- رخداد پایانی فقط دارای جریان ورودی است و جریان خروجی ندارد.
- فعالیتها و رخدادهای میانی، فقط یک ورودی و یک خروجی دارند.
- دروازه‌های انشعابی، دارای یک ورودی و بیش از یک خروجی باشد.
- دروازه‌های اتصالی، فقط یک جریان خروجی دارند و حاوی بیش از یک جریان ورودی هستند.

در ادامه، برای رسیدن به یک مدل فرآیندی خوش تعریف و تحلیل آن، پیش‌پردازش‌هایی بر روی مدل فرآیندی صورت می‌گیرد. در ادامه، به بخشی از مهم‌ترین این پیش‌پردازش‌ها می‌پردازیم.

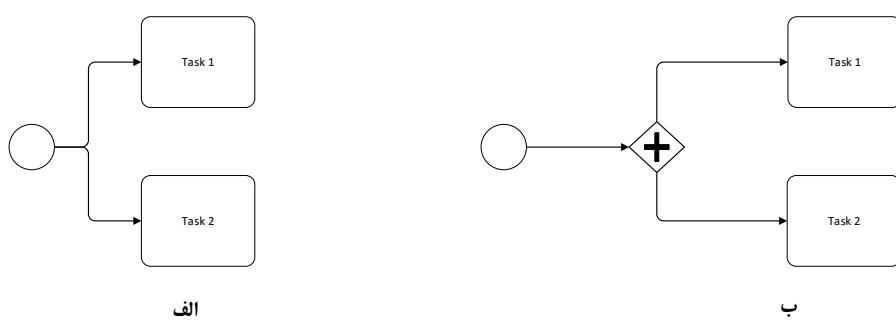
۱. جریان‌های چندگانه‌ی ورودی به هر کدام از اجزا (مثل رخدادها یا فعالیتها) به واسطه‌ی یک دروازه‌ی اتصال انحصاری، متصل شود. نمونه‌ای از این پیش‌پردازش، در شکل ۱-۳ آمده است. در شکل ۱-۳-الف دو ورودی به یک رخداد پایانی وجود دارد. بعد از اعمال این پیش‌پردازش، به شکل ۱-۳-ب می‌رسیم. همان‌طور که مشاهده می‌شود، در این شکل، اتصال ورود به رخداد پایانی به واسطه‌ی یک دروازه‌ی انحصاری انجام شده است.

۲. جریان‌های چندگانه‌ی خروجی از هر کدام از اجزا (مثل فعالیتها و رخدادها) به واسطه‌ی یک دروازه‌ی انشعاب موازی، منشعب شود. نمونه‌ای از این پیش‌پردازش، در شکل ۲-۳ آمده است. در شکل ۲-۳-الف دو خروجی از یک رخداد شروع، منشعب شده است. بعد از اعمال این پیش‌پردازش، به شکل ۲-۳-ب می‌رسیم. همان‌طور که مشاهده می‌شود، در



شکل ۳-۱: سازماندهی به جریان‌های ورودی چندگانه.

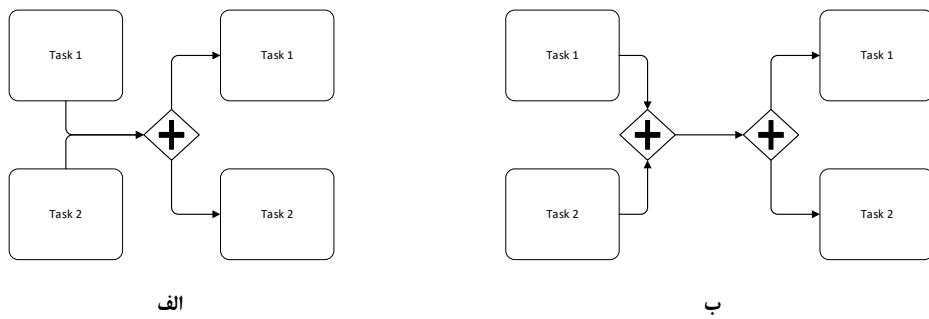
این شکل، اتصال خروجی‌ها از رخداد شروع به فعالیت‌ها، به واسطه‌ی یک دروازه‌ی انشعاب موازی انجام شده است.



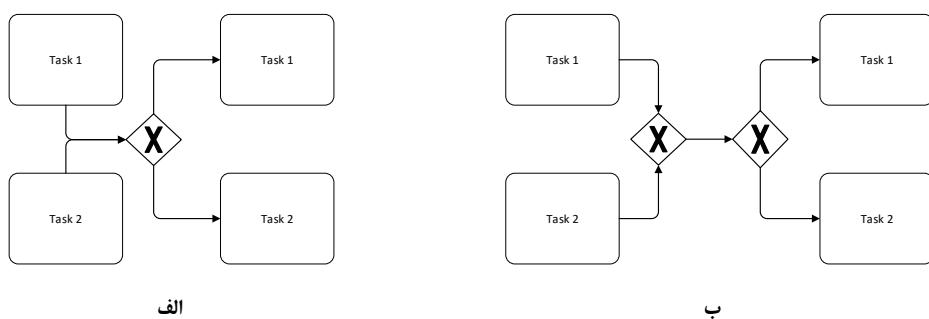
شکل ۳-۲: سازماندهی به جریان‌های خروجی چندگانه.

۳. در صورتی که دروازه‌های انحصاری و یا موازی، دارای چند ورودی و چند خروجی هستند، آن‌ها را به دو دروازه‌ی مشابه تبدیل کنید که یکی از آن‌ها ورودی‌ها را دریافت کند و دیگری، خروجی‌ها را تحويل دهد. نمونه‌هایی از این پیش‌پردازش‌ها، در شکل‌های ۳-۳ و ۴-۳ آمده است. در هردو شکل، شکل الف مربوط به وضعیت مدل فرآیندی پیش از اعمال پیش‌پردازش و شکل ب، مربوط به وضعیت مدل فرآیندی، بعد از اعمال پیش‌پردازش است.

۴. در صورتی که فرآیند دارای رخداد شروع و پایانی نیست، فعالیتی را که از آن کمانی خارج نشده، به یک رخداد پایانی متصل شود و به فعالیتی که به آن کمانی وارد نشده، یک رخداد آغازین متصل شود.



شکل ۳-۳: پیش‌پردازش ورودی‌ها و خروجی‌های چندگانه در دروازه‌های موازی



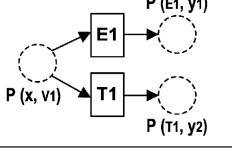
شکل ۳-۴: پیش‌پردازش ورودی‌ها و خروجی‌های چندگانه در دروازه‌های انحصاری

نگاشت مدل فرآیندی به شبکه‌های پتری به طور خلاصه، نگاشت ارائه شده در این پژوهش، در شکل ۳-۵ آمده است. در خصوص نگاشت موارد دیگری مثل کارهای چندنمونه‌ای<sup>۵</sup>، حلقه‌ها و زیرفرآیندها، به صورت زیر عمل می‌کنیم:

حلقه‌ها به جای حلقه، با استفاده از دروازه‌های انحصاری و یک فعالیت، یک حلقه می‌سازیم. بسته به این که شرط حلقه در ابتدا و یا انتهای حلقه بررسی شود، شرط دروازه‌های انحصاری حلقه تغییر می‌کند. در شکل‌های ۶-۳ - الف و ۶-۳ - ب به ترتیب دو نوع حلقه، نحوه پیش‌پردازش و در نهایت، نحوه نگاشت به شبکه‌های پتری نشان داده شده است.

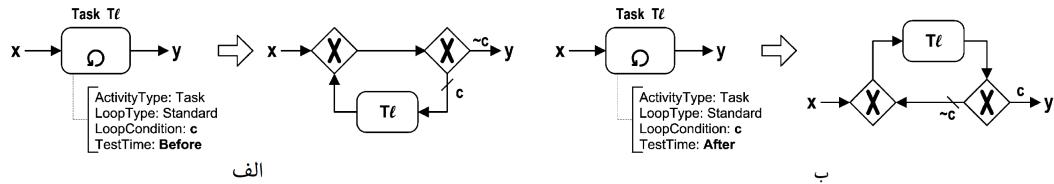
وظایف چندنمونه‌ای با توجه به این که در این اجزا چند فرآیند به صورت موازی انجام می‌شوند، می‌توان فعالیتها را با استفاده از دروازه‌ی موازی، به صورت موازی با سایر فعالیتها آغاز کرد و به پیش برد (شکل ۷-۳). در صورتی که قصد کشف بن‌بست را داشته باشیم، می‌توانیم

<sup>5</sup>Multi instance Tasks

BPMN Object	Petri-net Module	BPMN Object	Petri-net Module	BPMN Object
 $y$	 $P_s \xrightarrow{t_s} P(s, y)$	$x$ 	 $P(x, e) \xrightarrow{t_e} P(e, y)$	 $y$
$x \rightarrow$  $y$	$P(x, E1) \xrightarrow{E1} P(E1, y)$	$x \rightarrow T \rightarrow y$	$P(x, T1) \xrightarrow{T1} P(T1, y)$	 $y_1$
$x \rightarrow$  $y_1$ $y_2$	$P(x, F1) \xrightarrow{t_{F1}} P(F1, y_1)$ $P(x, F1) \xrightarrow{t_{F1}} P(F1, y_2)$	$x_1 \rightarrow$  $y$	$P(x_1, J1) \xrightarrow{t_{J1}} P(J1, y)$ $P(x_2, J1) \xrightarrow{t_{J1}} P(J1, y)$	 $y_1$ $y_2$
$x \rightarrow$  $y_1$ $y_2$ (Data-based) Condition: $c$	$P(x, D1) \xrightarrow{t_{(D1, y1)}} P(D1, y1)$ $P(x, D1) \xrightarrow{t_{(D1, y2)}} P(D1, y2)$	$x_1 \rightarrow$  $y$	$P(x_1, M1) \xrightarrow{t_{(M1, x1)}} P(M1, y)$ $P(x_2, M1) \xrightarrow{t_{(M1, x2)}} P(M1, y)$	 $E1 \xrightarrow{P(E1, y1)} P(y1)$ $P(x, v1) \xrightarrow{T1} P(T1, y2)$

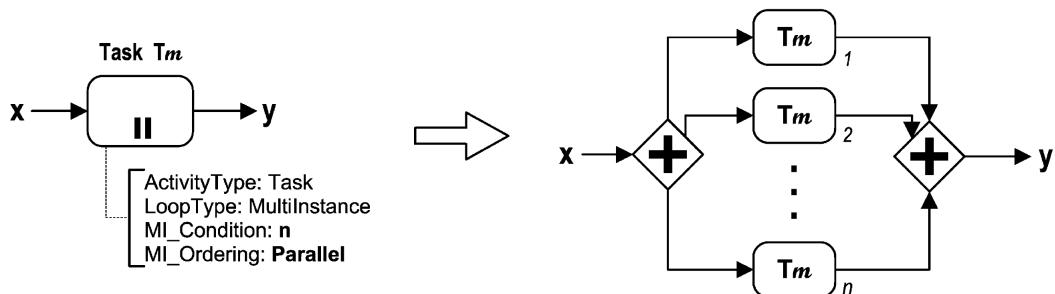
[ Note ]:  
 $x, x_1$  or  $x_2$  represents an input object, and  $y, y_1$  or  $y_2$  represents an output object.

شکل ۳-۵: نگاشت ارائه شده توسط [۴] به منظور تبدیل یک مدل فرآیندی BPMN2.0 به یک شبکه پتری مورد استفاده قرار می‌گیرد



شکل ۳-۶: پیش‌پردازش و نگاشت ارائه شده توسط [۴] به منظور تبدیل یک مدل فرآیندی به یک شبکه‌ی پتری

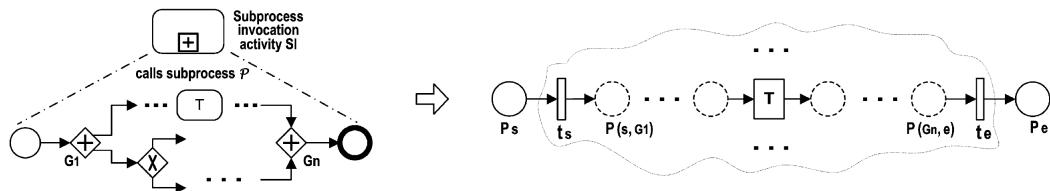
به جای یک وظیفه‌ی چندنمونه‌ای، از یک فعالیت استفاده کنیم [۴].



شکل ۳-۷: پیش‌پردازش و نگاشت ارائه شده توسط [۴] به منظور تبدیل یک وظیفه‌ی چندنمونه‌ای به یک شبکه‌ی پتری

زیرفرآیندها به منظور مدل‌سازی زیرفرآیندها، کافیست به جای آن‌ها یک فعالیت بگذاریم و فرآیند داخلی زیرفرآیند را جداگانه بررسی کنیم (تصویر ۳-۸). البته لازم است که پیش از

شروع فرآیند موجود در زیرفرآیند، مکانی را به عنوان مکان شروع فرآیند و پس از پایان آن، مکانی را به عنوان مکان پایانی فرآیند، درنظر بگیریم.



شکل ۱-۳: پیش‌پردازش و نگاشت ارائه شده توسط [۴] به منظور تبدیل یک زیرفرآیند به یک شبکه پتری

روش پیشنهادی این پژوهش به منظور تحلیل ایستا در این پژوهش، از ابزاری به نام ProM به منظور کشف خطا استفاده شده‌بود؛ با استفاده از این ابزار، موارد زیر قابل بررسی هستند:

**نبوت وظایف مرده** در خصوص وظایف مرده، در بخش قبل به تفصیل بحث گشت. اطمینان از عدم وجود این ناهنجاری، با نبوت انتقال‌های مرده (یعنی انتقال‌هایی که شلیک نمی‌شوند) ممکن خواهد بود.

اطمینان در تکمیل فرآیند این ویژگی، تضمین می‌کند که مدل فرآیندی داده‌شده، قطعاً تکمیل خواهد شد. یک مدل فرآیندی در صورتی تکمیل می‌شود که هر مورد کاری، به رخداد پایانی برسد و بعد از رسیدن به آن، هیچ فعالیتی در طول فرآیند، فعال نباشد [۴]. در شبکه پتری معادل با یک مدل فرآیندی، در صورتی یک فرآیند به اتمام می‌رسد که توکن متناظر با مورد کاری، به مکان پایانی برسد و هیچ توکن منشعب شده‌ای از این توکن، در فرآیند وجود نداشته باشد و هیچ انتقالی به واسطه‌ی آنها فعال نباشد [۳]. در خصوص شبکه‌های پتری، این وضعیت در صورتی به وجود می‌آید که صرفاً مکان پایانی دارای توکن باشد و در سایر مکان‌ها، توکنی وجود نداشته باشد [۴]. بنابراین، شرایط نامطلوب از دیدگاه شبکه‌های پتری، در دو صورت رخ می‌دهد:

- اگر در شبکه پتری، وضعیتی رخ دهد که به عنوان وضعیت پایانی تلقی شود و در آن، مکان پایانی حاوی توکن نباشد، یک بن‌بست رخ داده [۴].
- نشانه‌گذاری وجود داشته باشد که در آن، علاوه بر مکان پایانی، سایر مکان‌ها نیز دارای توکن باشند [۴].

## چالش‌های انتقال مدل BPMN به شبکه‌های پتری

در پژوهش [۲۴]، داکمن و همکارانش، روش ارائه شده در پژوهش قبلی خود [۴] را مورد نقد قرار داده‌اند. در این پژوهش، مهم‌ترین چالشی که در خصوص نگاشت مدل‌های فرآیندی BPMN2.0 به شبکه‌های پتری کلاسیک مطرح شده‌است، زیاد بودن تعداد نمادهای BPMN2.0 و نیز، کم بودن تعداد نمادهای قابل استفاده در شبکه‌های پتری است؛ این امر باعث می‌شود که پس از انتقال، هر کدام از نمادهای زبان BPMN2.0 به صورت ترکیب‌های مختلفی از مکان‌ها، انتقال‌ها و کمان‌ها نشان داده شوند؛ در نتیجه، بیشتر شدن نمادها و از همه مهم‌تر، افزایش پیچیدگی ارتباطات بین نمادهای جدید، می‌تواند مشکل ساز بوده و پیچیدگی را افزایش دهد. با این وجود، در پژوهش‌های بعدی، بازهم از شبکه‌های پتری کلاسیک، برای تجزیه و تحلیل مدل‌های فرآیندی BPMN2.0 استفاده شده‌است.

### ۲-۱-۳ صحبت‌سنجدی تراکنش‌های مبتنی بر مدل‌های فرآیندی BPMN

یکی از مواردی که در بسیاری از کسب و کارها و تجارت‌ها رخ می‌دهد، لغو یک معامله و یا یک فرآیند است؛ بنابراین لازم است یک مکانیزم مناسب، برای مدیریت لغوها، ایجاد شود. برای این منظور، لازم است به بخش‌هایی از فرآیند کسب و کار، به عنوان یک تراکنش<sup>۶</sup> نگاه کرد. در صورتی که بخشی از تراکنش دچار مشکل شود، کل تراکنش به حالت اول باز می‌گردد. در پژوهشی که توسط تاکمورا و همکارانش صورت گرفته‌است [۶]، سعی شده‌است که تراکنش‌های مدل‌سازی شده را با استفاده از زبان BPMN صحبت‌سنجدی شوند. این کار، با استفاده از شبکه‌های پتری انجام شده‌است.

### ۳-۱-۳ صحبت‌سنجدی صوری مدل‌های فرآیندی کسب و کار با محدودیت‌های

#### زمانی و منبع

فرآیندهای کسب و کار، در زمان و استفاده از منابع، می‌توانند دچار محدودیت‌های متعددی شوند. این محدودیت‌ها ممکن است باعث مشکلات جدی مانند بروز بن‌بست‌ها شوند. در [۲۵]، یک رویکرد مبتنی بر تکنیک وارسی مدل‌ها<sup>۷</sup> برای ارزیابی صوری مدل‌های فرآیند کسب و کار ارائه شده‌است؛ در این رویکرد، محدودیت‌های زمانی و منابع در نظر گرفته شده است. در آن پژوهش، برای این منظور، انجام مراحل زیر پیشنهاد شده‌است:

<sup>6</sup>Transaction

<sup>7</sup>Model Checking

۱. در ابتدا، زبان مدل‌سازی BPMN2.0 به منظور رسیدگی به این محدودیت‌ها گسترش داده شده است.

۲. سپس، یک نگاشت از مدل‌های فرآیند کسب‌وکار توصیف شده با این BPMN توسعه یافته بر روی اتوماتای زمان‌دار ارائه شده است که می‌تواند توسط کنترل‌کننده مدل UPPAAL تایید شود.

این رویکرد، به از بین بردن مشکلات ناشی از محدودیت‌های زمانی و منابع در مراحل اولیه ی ایجاد یک فرآیند کسب و کار، کمک می‌کند و تضمین کیفیت این دسته از مدل‌های فرآیند کسب‌وکار را ممکن می‌سازد.

### نقاط قوت و ضعف

اگرچه در روش‌های قبلی، روش‌هایی برای صوری‌سازی مدل‌های فرآیندی BPMN2.0 ارائه شده است، اما آن‌ها با محدودیت‌های زمانی، منابع و همزمانی به اندازه کافی مقابله نکردند. اما این روش، محدودیت‌هایی دارد؛ در این مدل، اجزای مدل فرایندی باید ایستا باشد و امكان مدیریت پارامترهایی که در طول فرآیند به صورت پویا تغییر می‌کنند، میسر نیست [۲۵].

### ۴-۱-۳ صحت‌سنجی با استفاده از وارسی مدل

در سال‌های ۲۰۱۲ و ۲۰۱۳، اسامه محمد خربوش و همکارانش طی پژوهش‌های [۱۴] و [۱۵] روشی را بر مبنای وارسی مدل ارائه کردند؛ در این روش، از دو روش برای تحلیل معنایی مدل‌های فرآیندی BPMN2.0 استفاده شده است:

- ساختارهای کریپکی<sup>۸</sup>

- منطق زمان خطی<sup>۹</sup>

ایده‌ی اصلی به کار رفته در این روش، تبدیل مدل فرآیندی BPMN2.0 به یک ساختار کریپکی است؛ بعد از این نگاشت، صحت ویژگی‌های اساسی، شامل بن‌بست، بن‌بست حلقوی و وجود پایان‌های چندگانه، با استفاده از یک مدل منطقی زمان خطی، مورد بررسی قرار می‌گیرد [۱۴]. اعمال این انطباق می‌تواند صحت مدل فرآیندی را تایید کند؛ در غیر این صورت، منجر به تولید یک مثال نقض می‌شود [۱۴].

<sup>8</sup>Kripke

<sup>9</sup>Linear Temporal Logic (LTL)

از مزایای این روش، می‌توان به مرتبه‌ی زمان اجرای مناسب و امکان یافتن مسیری که منجر به وقوع خطا گشته است اشاره نمود؛ بنابراین، اجرا در زمان چند جمله‌ای و نیز امکان ردیابی خطا، از مزایای به کارگیری این روش بوده‌اند [۱۴].

### ۵-۱-۳ صحت‌سنجی با استفاده از گراف‌های گردش کار

در سال ۲۰۱۴، توماس پرینز و همکارانش، پژوهشی را در خصوص صحت‌سنجی مدل‌های فرآیندی BPMN2.0 ارائه کردند [۵]. هدف از این پژوهش، برداشتن یک گام به سمت ساخت یک کامپایلر برای مدل‌های فرآیندی BPMN2.0 بود. در این پژوهش، یک ساختار میانی برای بازنمایی فرآیندهای کاری ارائه شده‌است. از این ساختار، با نام گراف گردش کار<sup>۱۰</sup> نام برد شده‌است. در ادامه، تعریف گراف گردش کار را از منظر این پژوهشگران، ارائه می‌دهیم.

**تعریف ۳-۱-۲.** یک گراف گردش کار، گرافی جهت‌دار مانند  $WFG = (N, E)$  است؛ به گونه‌ای که مجموعه‌ی  $N$  حاوی گره‌ها و مجموعه‌ی  $E$  حاوی یال‌های متصل‌کننده می‌باشد. مجموعه‌ی گره‌ها، شامل گره‌هایی مثل فعالیت‌ها، انشعاب‌ها<sup>۱۱</sup>، تقسیم‌ها<sup>۱۲</sup>، ادغام‌ها<sup>۱۳</sup>، اتصال‌ها<sup>۱۴</sup> و یک نقطه‌ی شروع و یک نقطه‌ی پایانی می‌باشد.

هر کدام از گره‌های فعالیتی، تقسیمی، انشعابی و گرهی پایانی، دارای دقیقاً یک یال ورودی است؛ همچنین، گره‌های فعالیتی، اتصالی، ادغامی و گرهی آغازین، دقیقاً یک یال خروجی دارند؛ همچنین، گره‌هایی که منجر به همگرایی شاخه‌های فرآیند می‌شوند، حداکثر از درجه‌ی ورودی ۲ و گره‌هایی که منجر به واگرایی فرآیند می‌شوند، حداکثر از درجه‌ی خروجی ۲ هستند. نمونه‌ای از این شبکه‌ی جریان کاری در شکل ۹-۳ آمده‌است. در این شکل، نوع گره‌هایی به کار گرفته شده، ذیل هر گره نوشته شده‌است.

**نکته ۳-۱-۳.** در تمامی گره‌ها، در صورت فعل شدن حداقل یکی از ورودی‌ها، جریان فرآیند ادامه خواهد یافت. تنها گره‌ای که از این خاصیت تبعیت نمی‌کند، گرهی اتصالی است. برای ادامه‌ی جریان فرآیند از یک گرهی اتصالی، لازم است که تمامی شاخه‌های ورودی آن فعل باشد.

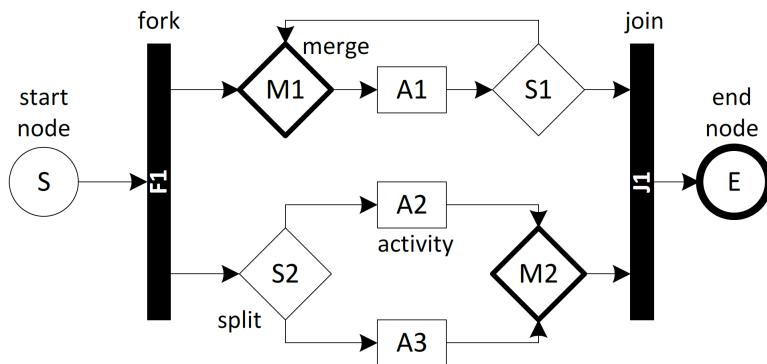
<sup>10</sup>Workflow Graph

<sup>11</sup>forks

<sup>12</sup>Splits

<sup>13</sup>Merges

<sup>14</sup>Joins

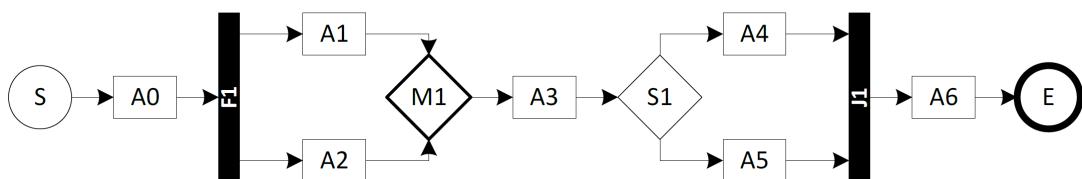


شکل ۳-۹: نمونه‌ای از گراف جریان کار با نمادگذاری ابداع شده توسط [۵]

نکته ۳-۴. خروجی گره‌های تقسیمی، به صورت غیرقطعی فعال می‌شوند؛ این بدان معنا است که حداقل یکی و حداقل، تمامی شاخه‌های خروجی آن می‌تواند فعال شود؛ اما تمامی خروجی‌های گره‌های انشعابی، به طور همزمان فعال می‌شوند و این گره‌ها، عمل موازی‌سازی را انجام می‌دهند.

در اثر استفاده‌ی نادرست از گره‌های اتصالی، ممکن است بن‌بست رخ دهد؛ در شرایطی این اتفاق رخ می‌دهد که تمامی شاخه‌های ورودی گره‌ی اتصالی فعال نباشد.

نمونه ۳-۵. نمونه‌ای از بروز بن‌بست، در شکل ۳-۱۰ توسط [۵] به تصویر کشیده شده است. در این نمونه، در اتصال J1 بن‌بست رخ می‌دهد.



شکل ۳-۱۰: نمونه‌ای از رخداد بن‌بست در گراف جریان کار با نمادگذاری ابداع شده توسط [۵]

در این پژوهش، برای تشخیص بن‌بست و عدم همزمانی، بررسی فرآیند از نقاط مختلفی شروع می‌شود؛ انتخاب این نقاط به گونه‌ای انجام می‌شود که تایید صحت هر کدام از آن‌ها، منجر به تایید صحت فرزندان آن‌ها بشود.

این روش، بر روی یکی از ابزارهای تحلیل فرآیندهای کسب و کار به نام Activity که بر روی نرم‌افزار Eclipse اجرا می‌شود، پیاده‌سازی شده است و نتایج مطلوبی را داشته است. در این نرم‌افزار، مدل فرآیندی ایجاد شده با زبان BPMN 2.0 به یک گراف گردش کار تبدیل می‌شود و سپس، صحبت‌سنجی می‌شود.

با این وجود، این روش دارای یک محدودیت می‌باشد: «گراف‌های جریان کاری، باید گراف‌هایی ساده باشند؛ یعنی بر روی هر یال آن، نباید بیش از یک فعالیت قرار گیرد.»

### ۳-۱-۶ صحت‌سنجی فرآیندهای مشارکتی و بین سازمانی

در بین سازمان‌های مختلف، همکاری بین‌سازمانی، قابلیتی بسیار مهم و ضروری است و سیاست‌های حاکم بر ارتباطات بین سازمانی، می‌تواند نقشی مهم و اساسی در موفقیت یک سازمان داشته باشد. بنابراین، عدم امکان همکاری در بین سازمان‌ها، می‌تواند منجر به عدم موفقیت آن‌ها بشود. برای برقراری ارتباط در بین سازمان‌ها، لازم است مجموعه‌ای از الزامات در آن‌ها تعیین شود. یکی از این الزامات، فرآیندی است که تعامل بین سازمان‌ها را تشکیل می‌دهد. این فرآیندهای کاری می‌تواند با زبان مدل‌سازی BPMN 2.0 ایجاد شوند [۲۶].

#### کشف بن‌بست و بن‌بست حلقوی برای مدیریت عمومی فرآیندهای بین سازمانی

مدیریت عمومی<sup>۱۵</sup> در بین دو یا چند سازمان مختلف، به مجموعه‌ای از فعالیت‌های مرتبط و متولی اطلاق می‌شود که توسط دو یا چند کارمند از چند سازمان به اشتراک گذاشته شده و اجرا می‌شوند تا به یک هدفی تجاری منجر شوند که از نظر خدمات برای شهروندان یا شرکت‌ها ارزشمند باشد. یک فرآیند کسب‌وکار بین سازمانی، از ادغام غیر بدیهی فرآیندهای کسب و کار داخلی آن‌ها بدست می‌آید؛ در چنین فرآیندهایی، مشکلات ساختاری مانند بن‌بست‌ها و یا بن‌بست‌های حلقوی، به راحتی می‌توانند رخ دهند. اگرچه آن‌ها می‌توانند توسط کارمندان سازمان‌ها حل شوند، اما با گسترش دولت الکترونیک، باید به فکر راهکارهایی هوشمند برای حل این مشکل بود؛ بنابراین در خصوص فرآیندهای بین سازمانی، پژوهش‌های مختلفی انجام شده‌است.

در سال ۲۰۱۲، پژوهشی توسط فالسیونی و همکارانش در این خصوص صورت گرفت [۲۷]. در این پژوهش، یک رویکرد مناسب برای شناسایی بن‌بست‌ها و بن‌بست‌های حلقوی در فرآیند کسب‌وکار بین سازمانی ارائه شده‌است. به طور خاص، روشی به منظور بررسی مستقیم یک فرآیند کسب‌وکار مدل‌سازی شده با استفاده از زبان مدل‌سازی BPMN<sup>۲۰</sup> معرفی شده است. در این پژوهش، روشی به منظور صحت‌سنجی صوری فرآیندهای کسب و کار بین سازمانی در مدیریت عمومی، ارائه شده‌است. این رویکرد از تکنیک‌های ارزیابی حالت با یک الگوریتم unfolding استفاده می‌کند. به این ترتیب، پس از

<sup>15</sup>Public Administration (PA)

مدل‌سازی فرآیند کسب و کار با استفاده از BPMN ۲.۰، تحلیلگر می‌تواند الگوریتم را اجرا کند تا بررسی کند که آیا فرآیند کسب و کار مورد نظر او، شامل اثرات بد است یا خیر. در چنین مواردی او می‌تواند فرآیند کسب و کار را مهندسی مجدد کند تا اثرات بد را از بین ببرد و بدین‌وسیله، فرآیند کسب و کار را بهبود بخشد [۲۷].

### استفاده از وارسی مدل به منظور تحلیل فرآیندهای بین سازمانی

یکی دیگر از این تحقیقات، در سال ۲۰۱۵ توسط ملک و همکارانش صورت گرفته است [۲۶]. در این تحقیق، چنین گفته شده است: «برای ارتباطات بین سازمانی، وجود یک سری از الزامات همکاری بین سازمانی، لازم است.» الزامات همکاری‌های بین سازمانی، با استفاده از زبان مدل‌سازی BPMN 2.0 می‌تواند بیان شود. برای تحلیل صحت ساختاری این مدل‌ها، استفاده از روش‌های صوری ارجح است. بنابراین از وارسی مدل به منظور کشف خطاها ساختاری، استفاده می‌شود؛ برای این منظور از یک ابزار وارسی‌کننده مدل به نام UPPAL استفاده می‌شود. مدلی که توسط این ابزار تحلیل می‌شود، شبکه‌ی آتماتای زمانی<sup>۱۶</sup> است؛ بنابراین، به منظور تحلیل ساختاری مدل فرآیندی، مراحل زیر پیشنهاد شده است [۲۶] :

۱. در ابتدا لازم است که مدل فرآیندی BPMN 2.0 که نشان‌دهنده‌ی فرآیند مشارکتی است، به یک شبکه‌ی آتماتای زمانی نگاشته شود تا توسط UPPAL قابل بررسی باشد. یک آتماتای زمانی، می‌تواند رفتار تعاملی بین فرآیندها را به خوبی مدل‌سازی کند. بنابراین، لازم است که مدل فرآیندی BPMN 2.0 به این مدل، نگاشته شود.

۲. آتماتای زمانی، اگرچه می‌تواند همکاری بین سازمان‌ها را مدل‌سازی کند، اما برای وارسی ملزمات همکاری بین سازمانی، لازم است که این الزامات را با استفاده از یک زبان صوری دیگری بیان شوند. برای این منظور در این پژوهش، از زبان اختصاصی UPPAL یعنی زبان منطق زمان خطی (TCTL) استفاده شده است.

روش ارائه شده در این پژوهش، دارای محدودیت‌هایی است. یکی از محدودیت‌های اساسی آن، فضای حالت بزرگ برای بررسی مسیرهای فرآیند است. با به کارگیری روش ارائه شده در این پژوهش، در بهترین حالت صحت فرآیند تایید می‌شود و در بدترین حالت، صحیح بودن و یا صحیح نبودن فرآیند، به صورت قطعی مشخص نمی‌شود. همچنین، انطباق الزامات همکاری با فرآیند، عملی دشوار است [۲۶].

<sup>۱۶</sup>Network of Timed Automata

### ۷-۱-۳ صحت‌سنجی با استفاده از روش جبری

در کنار تمامی روش‌های صوری، روش‌های مبتنی بر ریاضیات و جبر به علت دقت بالا، مورد توجه برخی محققین قرار گرفته است. در حوزه‌ی صحت‌سنجی فرآیندهای کسب و کار و مدل‌های فرآیندی ایجاد شده با زبان BPMN 2.0 نیز این رویکر مورد توجه قرار گرفته است، به عنوان نمونه در سال ۲۰۱۵ طی پژوهشی که توسط عثمان الحجمی صورت گرفت [۱۶]، از روشی جبری به منظور صحت‌سنجی مدل‌های فرآیندی BPMN 2.0 استفاده شده است. در این پژوهش، از نمادگذاری جبری مکس پلاس<sup>۱۷</sup> به منظور صحت‌سنجی مدل‌های فرآیندی، استفاده شده است. این نمادگذاری جبری، روشی کارا برای ارزیابی عملکرد تعامل و قابلیت همکاری در ترکیب فرآیندها می‌باشد [۱۶]. این نمادگذاری جبری، برای بررسی پدیده‌هایی مثل همزمانی<sup>۱۸</sup>، همروندی و موازی‌سازی، بسیار مناسب می‌باشد.

با استفاده از این رویکرد، می‌توان با به کارگیری معادلات مکس پلاس، یک معناشناسی مناسبی را از الگوهای به کار گرفته شده در مدل‌های فرآیندی BPMN 2.0 انجام داد [۱۶].

### ۸-۱-۳ صحت‌سنجی با استفاده از روش‌های احتمالاتی

روش‌هایی که تا کنون بررسی شده‌اند، مبتنی بر انتقال مدل فرآیندی BPMN 2.0 به یک مدل صوری دیگر بوده است و در هیچ کدام از روش‌های پیشین، مدل فرآیندی به تنها‌یی بررسی نمی‌شده است. انتقال مدل فرآیندی به مدل دیگر، معمولاً عملی هزینه‌بر است و براساس مدل صوری انتخابی، می‌تواند با چالش‌های خاصی مواجه شود. بنابراین، تمایل به یافتن روش‌های تحلیل ساختاری، بدون نیاز به تبدیل آن به مدل‌های صوری، در بین محققین زیاد شد. در سال ۲۰۱۶ طی پژوهشی که توسط محمد نوم و عثمان الحجمی و همکارانشان صورت گرفت، از رویکردی مبتنی بر تئوری احتمالات، به منظور تحلیل ساختاری مدل‌های فرآیندی BPMN 2.0 استفاده شد [۲۸]. در این پژوهش، به ورودی و خروجی هر کدام از اجزای مدل فرآیندی، یک احتمال تخصیص داده می‌شود. این احتمال‌ها نشان‌دهنده‌ی احتمال گذر جریان فرآیند از آن جزء می‌باشد؛ در نهایت، احتمال وقوع بن‌بست و یا بن‌بست حلقوی، بر مبنای احتمال فعال شدن ورودی و یا خروجی هر کدام از اجزای مسیر، بدست می‌آید. در ادامه، به اختصار احتمالات مربوط به هر کدام از اجزا مورد بررسی قرار می‌گیرد: [۲۸]:

**رخداد شروع** این رخداد، حتماً اتفاق می‌افتد؛ بنابراین، احتمال ورود به آن و خروج از آن، برابر با ۱

<sup>17</sup>Max + Algebra

<sup>18</sup>Synchronization

است.

**رخداد میانی** برای رخدادهای میانی احتمال وقوع، پنجاه درصد در نظر گرفته شده است؛ چرا که یک رخداد میانی ممکن است انجام بشود و یا انجام نشود [۲۸]؛ با این حال، نگارنده‌ی پایان‌نامه‌ی حاضر با این نظر موافق نیست و معتقد است که برای یافتن احتمال ورودی و خروجی یک رخداد میانی، باید جایگاه آن رخداد را در فرآیند دانست.

**رخداد پایانی** احتمال وقوع رخداد پایانی، برابر است با احتمال خروج آخرین جزئی که قبل از آن قرار دارد.

**فعالیت‌ها** احتمال وقوع یک فعالیت مثل  $T$  با جریان ورودی  $i$  و جریان خروجی  $o$  به صورت زیر محاسبه می‌شود [۲۸]:

$$P(T \cap i) = P(T | i) \times P(i) = P(i) = P(o) \quad (1-3)$$

دروازه‌های انشعاب عطفی (انشعاب موازی) با توجه به این‌که جریان فرآیند از تمامی شاخه‌های خروجی بیرون می‌رود، احتمال خروج از هر شاخه، با احتمال ورود به دروازه برابر است. بنابراین برای شاخه‌ی خروجی  $j$ ام دروازه‌ی موازی  $(o_j)$  و شاخه‌ی ورودی دروازه‌ی موازی  $(i)$  و گذر جریان فرآیند از خود دروازه‌ی موازی  $(T)$ ، این احتمال به صورت زیر محاسبه می‌شود [۲۸]:

$$P(T \cap i) = P(T | i) \times P(i) = P(i) = P(o_j) \quad (2-3)$$

دروازه‌های انشعاب فصلی (دروازه‌های جامع انشعابی) با توجه به این‌که جریان فرآیند از هر کدام از شاخه‌های خروجی می‌تواند بیرون برود و یا بیرون نرود، احتمال خروج از هر شاخه، برابر است با نصف احتمال ورود به دروازه. بنابراین برای شاخه‌ی خروجی  $j$ ام دروازه‌ی جامع انشعابی  $(o_j)$  و شاخه‌ی ورودی دروازه‌ی جامع انشعابی  $(i)$  و گذر جریان فرآیند از خود دروازه‌ی جامع انشعابی  $(T)$ ، این احتمال به صورت زیر محاسبه می‌شود [۲۸]:

$$P(T \cap i) = P(T | i) \times P(i) = P(i) = P(o_j) \quad (3-3)$$

دروازه‌های انشعاب انحصاری<sup>۱۹</sup> همان‌طور که در بخش ۲-۲ مشاهده شد، در دروازه‌های انشعاب انحصاری، فقط یکی از خروجی‌ها فعال می‌شود. این به این معناست که ممکن است تعداد  $k$  خروجی، مردود شوند. بنابراین در تحلیل احتمالی این جزء باید بدین صورت عمل کرد: «احتمال آن که خروجی زام بعد از مردود شدن  $k$  خروجی فعال شود چقدر است؟» این احتمال از طریق رابطه‌ی زیر بدست می‌آید [۲۸]:

$$P(T \cap o_j \cap \bar{o}_k) = P(T) \times P(j | T) \times P\left(\bigcap_{k < j} \bar{o}_k\right) \quad (4-3)$$

دروازه‌های اتصال موازی (اتصال عطفی) این دروازه، به منزله‌ی همگام‌سازی شاخه‌های فرآیند به کار می‌رود؛ بنابراین، تنها در صورتی ادامه‌ی جریان فرآیند از آن میسر خواهد بود که تمامی شاخه‌های ورودی آن، فعال باشد. بنابراین، احتمال مربوط به این جزء را محاسبه‌ی احتمال این اجزا با فرض وجود  $m$  شاخه‌ی ورودی، از طریق رابطه‌ی زیر بدست می‌آید [۲۸]:

$$P(T \cap o_j \bigcap_{n=1}^m i_n) = P(T) \times \prod_{n=1}^m P(i_n) \quad (5-3)$$

دورازه‌های جامع اتصالی (اتصال فصلی) و اتصال انحصاری این دروازه‌ها، به منظور ادغام شاخه‌های جریانی فرآیندها به کار می‌روند؛ برای ادامه‌ی جریان فرآیند از این دروازه‌ها، لازم است حداقل یکی از شاخه‌های ورودی فعال باشد؛ برای محاسبه‌ی احتمال این اجزا با فرض وجود  $m$  شاخه‌ی ورودی، از رابطه‌ی زیر استفاده می‌کنیم [۲۸]:

$$P(T \cap o_j \bigcup_{n=1}^m i_n) = P(T) \times \bigcup_{i=1}^m P(i_m) \quad (6-3)$$

نتیجه ۳-۱-۶. بر طبق رابطه‌ی ۳-۱ وجود فعالیت‌ها، تاثیری بر جریان فرآیند (از نظر وقوع بنست) ندارد؛ چراکه این اجزا جریان ورودی را بدون تغییر به خروجی خود منتقل می‌کنند [۲۸].

<sup>19</sup>XOR Split

### احتمال وقوع بنبست

بنبست در موقعیتی رخ می‌دهد که یک دروازه‌ی ادغامی، منتظر فعال شدن برخی از ورودی‌های خود بماند، درحالی که این اتفاق اصلاً رخ ندهد. این اتفاق در حین همگامسازی و در دروازه‌های اتصالی (اتصال موازی) رخ می‌دهد. در دروازه‌های اتصال عطفی (دوازه‌های اتصال موازی)، در صورتی بنبست رخ می‌دهد که زیرمجموعه‌ای غیرتنهی از یال‌های ورودی آن، نامعین و یا غیرقابل فعال‌سازی باشند. برای یک دروازه مانند  $G$  با تعداد  $n$  تا کمان ورودی ( $k$  امین کمان ورودی را با  $i_k$  نمایش می‌دهیم) احتمال وقوع بنبست به صورت زیر محاسبه می‌شود:

$$P(D) = P\left(\bigcap_{k=1}^n i_k \cap \bigcap_{k=1}^n \bar{i}_k\right) = 1 - P\left(\bigcap_{k=1}^n i_k \cup \bigcap_{k=1}^n \bar{i}_k\right) \quad (7-3)$$

### ارزیابی تحقیق

بزرگ‌ترین مزیت این کار، عدم نیاز به نگاشت مدل فرآیندی BPMN 2.0 به یک مدل صوری است. با استفاده از این روش‌ها، می‌توان بدون نگاشت مدل فرآیندی به یک مدل صوری، در خصوص صحت آن تا حد خیلی مناسبی تصمیم‌گیری نمود.

## ۲-۳ سایر تحقیقات صورت گرفته در خصوص بازنمایی صوری

### مدل‌های فرآیندی

در کنار تحقیقات صورت گرفته بر روی روش‌های صحتسنجی مدل‌های فرآیندی BPMN 2.0، در تحقیقات دیگری به روش‌های بازنمایی و صوری‌سازی این مدل‌های فرآیندی پرداخته شده‌است. عمده‌ی تمرکز این تحقیقات، بر روی صوری‌سازی این مدل‌های فرآیندی است و کمتر بر روی الگوریتم‌های صحتسنجی ساختاری مدل‌های فرآیندی متمرکز شده‌اند. با توجه به این‌که ادامه‌ی این تحقیقات می‌تواند منجر به توسعه‌ی روش‌های تحلیل ایستای مدل‌های فرآیندی BPMN 2.0 شود، بررسی اجمالی آن‌ها، خالی از لطف نخواهد بود.

## ۱-۲-۳ بازنمایی بصری شبکه‌های پتری حاصل از مدل‌های فرآیندی BPMN2.0

در سال ۲۰۱۸، پژوهشی توسط لی و همکارش صورت گرفت که در آن، یک نگاشت از شبکه‌های مدل‌های فرآیندی BPMN 2.0 به شبکه‌های پتری پیشنهاد شد [۲۹]. در این پژوهش، علاوه بر ارائه‌ی یک نگاشت، تعدادی پیش‌پردازش<sup>۲۰</sup> نیز پیشنهاد شد. با اعمال این پیش‌پردازش‌ها بر روی مدل فرآیندی BPMN 2.0 می‌توان به یک مدل فرآیندی خوش‌تعریف<sup>۲۱</sup> رسید و با استفاده از آن، ساختار مدل فرآیندی، ساختاری استاندارد و یکتا خواهد بود؛ در این پژوهش بعد از اجرای پیش‌پردازش و نیز نگاشت به شبکه‌های پتری، روشی به منظور ترسیم شبکه‌ی پتری متناظر با مدل فرآیندی ارائه شده‌است. این ترسیم به گونه‌ای انجام می‌شود که ساختار و چینش اجزای شبکه‌ی پتری، تا حد زیادی با ساختار و چینش مدل فرآیندی مطابقت داشته باشد و با این کار، تحلیل مدل فرآیندی با استفاده از شبکه‌های پتری، از نظر بصری (و نه با استفاده از الگوریتم‌های صحبت‌سنگی) راحت‌تر می‌شود.

## ۲-۲-۳ انتقال مدل‌های فرآیندی BPMN 2.0 به شبکه‌های پتری رنگی

در تحقیقات پیشین، تحقیقات گستردۀ‌ای بر روی انتقال مدل‌های فرآیندی به مدل‌های صوری، صورت گرفته‌بود. یکی از چالش‌هایی که در انتقال مدل فرآیندی به سایر مدل‌های صوری مطرح می‌شود، صحت انتقال است؛ صحت انتقال، تضمین می‌کند که مدل مقصد، معادل با مدل مبدا است. در پژوهش اخیر صورت گرفته توسط سعید مغزیلی و همکارانش، یک انتقال از مدل‌های فرآیندی BPMN 2.0 به شبکه‌های پتری رنگی<sup>۲۲</sup> صورت گرفته‌است [۳۰]. در این پژوهش، کارهای زیر انجام شده‌است:

۱. یک رویکرد برای انتقال مدل‌های فرآیندی BPMN 2.0 به شبکه‌های پتری ارائه شده‌است. این انتقال با استفاده از ابزار EMF/Xpand و Groove graph صورت می‌گیرد.
۲. رویکردی ارائه شده‌است که به وسیله‌ی آن، صحت انتقال صورت گرفته بررسی شود؛ به عبارت دیگر، شط خاتمه‌ی فرآیند انتقال بررسی می‌شود.

این پژوهش، با استفاده از یک مطالعه‌ی موردی بررسی شده‌است.

<sup>20</sup>Pre process

<sup>21</sup>Well Defined

<sup>22</sup>Colored Petrinets

### ۳-۳ نمایه‌ی پژوهش‌های صورت گرفته

صحت‌سنجی مدل‌های فرآیندی BPMN 2.0 از سال ۲۰۰۸ تا کنون در حال توسعه است. در ادامه، یک نمایه از تحقیقات صورت گرفته در این حوزه، در جدول ۱-۳ قابل مشاهده است. در سطر پایانی جدول، مشخصات پژوهش جاری مورد بررسی قرار گرفته است.

### ۴-۳ ارتباط پژوهش‌های پیشین با پژوهش جاری

این پژوهش نیز همانند سایر کارهای پژوهشی، مزایایی را نسبت به روش‌های ارائه شده‌ی پیشین دارد و نسبت به آن‌ها دارای مزایایی است؛ از جمله:

۱. اگرچه در پژوهش‌های پیشین در خصوص صحت‌سنجی مدل‌های فرآیندی BPMN2.0 راهکارهایی ارائه شده‌است، اما در هیچ کدام از پژوهش‌های پیشین، تعریف جامع و مشخصی درباره‌ی یک مدل فرآیندی BPMN2.0 صحیح ارائه نشده‌است؛ بنابراین لازم است در ابتداء، بر اساس یک پایه‌ی نظری مناسب، ویژگی‌هایی برای یک مدل فرآیندی BPMN 2.0 صحیح ارائه شود تا بتوان به وسیله‌ی آن‌ها، یک مدل فرآیندی صحیح را از یک مدل فرآیندی ناصحیح، بازناسی نمود. در پژوهش جاری، بر پایه‌ی شبکه‌های پتری جریان کار، مفهوم صحت به مدل‌های فرآیندی BPMN2.0 تعمیم پیدا کرده‌است و بدین‌وسیله، می‌توان مدل‌های فرآیندی صحیح را از مدل‌های فرآیندی ناصحیح بازناخت.

۲. عمدۀ‌ی پژوهش‌های صورت گرفته در این حوزه، بر لزوم تبدیل مدل‌های فرآیندی BPMN2.0 به مدل‌هایی صوری تاکید دارند؛ این درحالی است که به کارگیری روش‌های صوری در صحت‌سنجی این مدل‌های فرآیندی، منجر به طولانی شدن عملیات صحت‌سنجی می‌شوند. از مزایای روش ارائه شده در پژوهش جاری، امکان تصمیم‌گیری در خصوص صحت مدل‌های فرآیندی BPMN2.0 بدون نیاز به انتقال آن‌ها به مدل‌های صوری است و برای این منظور یک روش پیشنهاد شده‌است که می‌تواند بدون نیاز به رایانه و یا به وسیله‌ی رایانه مورد استفاده قرار گیرد.

۳. تعمیم مفهوم صحت به مدل‌های فرآیندی BPMN2.0 و نیز ارائه‌ی روش صحت‌سنجی، بر مبنای یک نگاشت یک به یک بین اجزای مدل‌های فرآیندی BPMN2.0 و اجزای شبکه‌های پتری جریان کار صورت گرفته‌است. این درحالی است که بر اساس ارزیابی صورت گرفته در پژوهش داکمن،

### فصل سوم: پیشینه‌ی پژوهش

مرجع	سال انتشار	شماره مرجع در فهرست مراجع	مفروضات	اهداف تحقیق	راه حل ارائه شده
جلد		[۲۳]	عدم وجود عناصر غیر رفتاری	کشف الگوهای رفتاری در فرآیند	نگاشت به شبکه‌های پتری
۲۰۲۱	۲۰۲۰	[۲۹]	وجود محدودیت زمانی	کنشل ملزومات همکاری	تحلیل اختلالات
۲۰۲۰	۲۰۱۸	[۲۸]	امکان همکاری بین سازمانی	کشف گلوگاهها	استفاده از روش جبری (LTL)
۲۰۱۸	۲۰۱۷	[۲۹]	وجود محدودیت در منابع فرآیند	کشf دسترسی پذیری	استفاده از مدل بازار (UPPAL)
۲۰۱۷	۲۰۱۵	[۲۸]	وجود یک رخداد شروع و پایانی	کشف حلقه‌های می‌نهایت	نگاشت به شبکه‌ی پتری رنگی
۲۰۱۵	۲۰۱۴	[۲۶]	عدم وجود استخراج خطوط شناسوی	کشف بن‌بست حلقوی	استفاده از ساختار کپیک (Model Checking)
۲۰۱۴	۲۰۱۳	[۵]	کشف بن‌بست	صوت‌سنگی مدل‌های فرآیندی	وارسی مدل (Model Checking)
۲۰۱۳	۲۰۱۲	[۲۷]	کنشل ملزومات همکاری	صوت‌سنگی مدل‌های فرآیندی	نگاشت به شبکه‌های پتری
۲۰۱۲	۲۰۱۱	[۲۵]	کشف گلوگاهها	صوت‌سنگی مدل‌های فرآیندی	تحلیل اختلالات
۲۰۱۱	۲۰۱۰	[۲۴]	کشf دسترسی پذیری	صوت‌سنگی مدل‌های فرآیندی	استفاده از روش جبری (LTL)
۲۰۱۰	۲۰۰۹	[۳۲]	کشف حلقه‌های می‌نهایت	صوت‌سنگی مدل‌های فرآیندی	نگاشت به شبکه‌ی پتری رنگی
۲۰۰۹	۲۰۰۸	[۲۳]	کشف بن‌بست حلقوی	صوت‌سنگی مدل‌های فرآیندی	استفاده از ساختار کپیک (Model Checking)
۲۰۰۸	۲۰۰۷	[۲۴]	صوت‌سنگی مدل‌های فرآیندی	صوت‌سنگی مدل‌های فرآیندی	نگاشت به شبکه‌های پتری
۲۰۰۷	۲۰۰۶	[۲۷]	صوت‌سنگی مدل‌های فرآیندی	صوت‌سنگی مدل‌های فرآیندی	تحلیل اختلالات
۲۰۰۶	۲۰۰۵	[۵]	صوت‌سنگی مدل‌های فرآیندی	صوت‌سنگی مدل‌های فرآیندی	استفاده از روش جبری (LTL)
۲۰۰۵	۲۰۰۴	[۲۶]	صوت‌سنگی مدل‌های فرآیندی	صوت‌سنگی مدل‌های فرآیندی	نگاشت به شبکه‌ی پتری رنگی
۲۰۰۴	۲۰۰۳	[۱۶]	صوت‌سنگی مدل‌های فرآیندی	صوت‌سنگی مدل‌های فرآیندی	استفاده از ساختار کپیک (Model Checking)
۲۰۰۳	۲۰۰۲	[۲۸]	صوت‌سنگی مدل‌های فرآیندی	صوت‌سنگی مدل‌های فرآیندی	نگاشت به شبکه‌های پتری
۲۰۰۲	۲۰۰۱	[۱۹]	صوت‌سنگی مدل‌های فرآیندی	صوت‌سنگی مدل‌های فرآیندی	تحلیل اختلالات
۲۰۰۱	۲۰۰۰	[۲۹]	صوت‌سنگی مدل‌های فرآیندی	صوت‌سنگی مدل‌های فرآیندی	استفاده از روش جبری (LTL)

جدول ۳-۱: نمایه‌ی تحقیقات پیشین پیرامون موضوع پژوهش

در بسیاری از تبدیل‌های صورت گرفته بین مدل‌های فرآیندی BPMN2.0 به سایر مدل‌های صوری، مثل شبکه‌های پتری، ممکن است تعداد اجزای مدل مقصود نسبت به تعداد اجزای مدل فرآیندی مبدا بیشتر شود؛ از سوی دیگر درخصوص نگاشت بین مدل‌های فرآیندی BPMN2.0 و شبکه‌های پتری جریان کار تا کنون پژوهشی صورت نگرفته است؛ این در حالی است که اجزای شبکه‌های پتری جریان کار، شbahت بیشتری را نسبت به سایر مدل‌های صوری به اجزای مدل‌های فرآیندی BPMN2.0 دارند و سازگاری بیشتری را با آن‌ها دارند.

۴. در پژوهش جاری، به منظور صحتسنجی مدل‌های فرآیندی BPMN2.0 بخشی از آن مورد ارزیابی قرار می‌گیرد و بخش مهمی از اجزای متنوع آن از فرآیند صحتسنجی کنار گذاشته می‌شوند. این رویکرد منجر به پیشنهاد داده‌ساختاری به نام گراف دروازه‌ای<sup>۲۳</sup> شد. در رویکرد پیشنهادی، آن‌چه که به عنوان ویژگی اساسی در صحت و یا عدم صحت یک مدل فرآیندی می‌بایست مورد تحلیل قرار گیرد، جریان کنترلی فرآیند است و جریان اصلی فرآیند بر اثر ارتباط بین دروازه‌ها ایجاد می‌شود؛ در نتیجه ترکیب اتصال بین دروازه‌های مدل فرآیندی مورد بررسی قرار گرفته است. این رویکرد شbahت زیادی به روش پیشنهاد شده در [۵] دارد، با این تفاوت که در روش جاری، مدل فرآیندی به یک مدل میانی مثل گراف جریان کار منتقل نمی‌شود، بلکه بخشی از مدل فرآیندی کنونی مورد تحلیل و بررسی قرار می‌گیرد.

از سوی دیگر، نتایج مهمی از پژوهش‌های پیشین، تشکیل دهنده‌ی بخش مهمی از پایه‌های نظری پژوهش جاری می‌باشد و یا برای پژوهش جاری، الهام‌بخش هستند. از جمله:

۱. پیش‌پردازش‌هایی که به منظور خوش‌تعريف کردن مدل‌های فرآیندی BPMN2.0 در پژوهش‌های [۴] و [۲۹] ارائه شده‌است، در پژوهش جاری نیز به عنوان بخشی از فرآیند پیش‌پردازش، مورد استفاده قرار گرفته است.

۲. داده‌ساختار پیشنهادی در پژوهش جاری، شbahت زیادی به گراف جریان کار ارائه شده در پژوهش [۵] دارد، اما نسبت به آن مزیتی دارد: «گراف جریان کار، یک زبان میانی است که مدل فرآیندی BPMN2.0 به آن نگاشته شده و تحلیل می‌شود؛ اما گراف دروازه‌ای، بخشی از مدل فرآیندی جاری است و با حداقل پیش‌پردازش، قابل دستیابی است.»

۳. همان‌طور که گفته شد، بخشی از اجزای بی‌تأثیر در تحلیل ساختاری اجزای مدل فرآیندی

<sup>23</sup>Gateway Graph (GG)

BPMN2.0 از فرآیند صحت‌سنجدگی کنارگذاشته شده‌اند. میزان تاثیر هرکدام از اجزا بر فرآیند صحت‌سنجدگی، بر اساس نتایج بدست آمده از تحلیل احتمالاتی هرکدام از اجزا بدست آمده‌است. این تحلیل‌های احتمالی در اثر پژوهش [۲۸] بدست آمده‌اند.

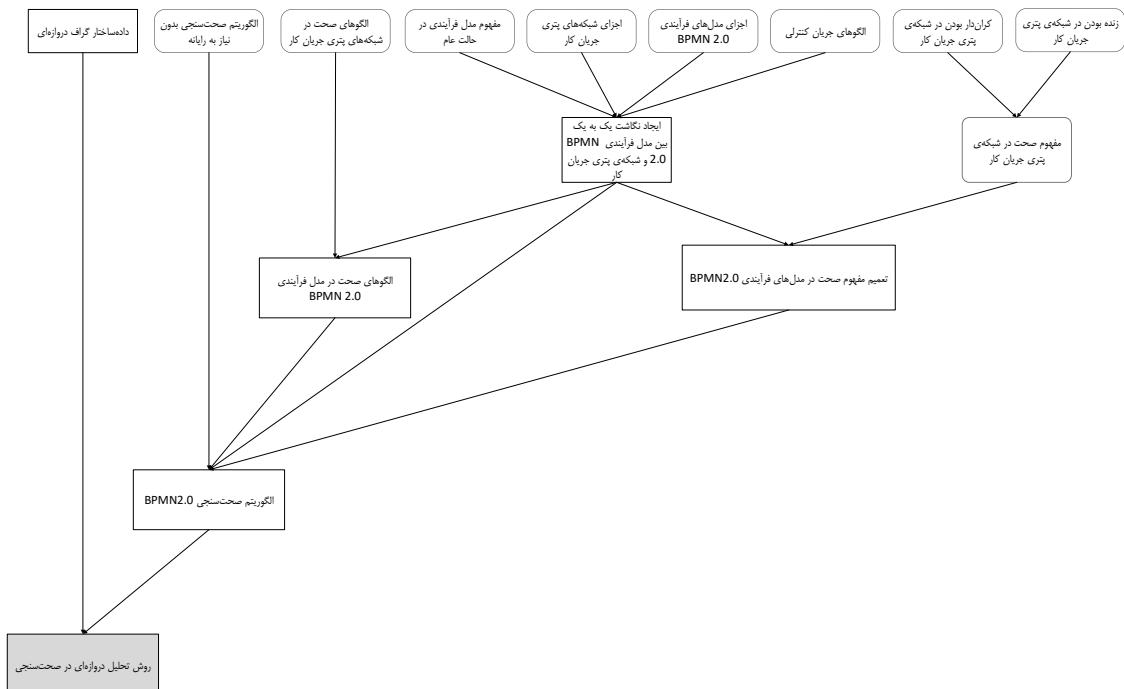
## فصل چهارم

## روش پیشنهادی

## ۱-۴ چارچوب مفهومی راه حل پیشنهادی

در این فصل به ارائه‌ی راه حلی به منظور تحلیل ایستای مدل‌های فرآیندی BPMN 2.0 پرداخته می‌شود. راهکار پیشنهادی این پژوهش، همانند راهکارهای ارائه شده توسط بسیار دیگری از پژوهش‌ها، بر مبنای تعدادی از مفاهیم پایه‌ای بنا شده است؛ این مفاهیم پایه‌ای، به عنوان مفروضات اولیه‌ی پژوهش در نظر گرفته می‌شوند و بر اساس آن‌ها، نتایجی در میانه‌ی پژوهش به وجود می‌آیند. این نتایج، راه حل نهایی پژوهش نیستند، اما بخش‌هایی از راه حل نهایی را تشکیل می‌دهند و در نهایت، با به کارگیری این نتایج میانی، راه کار نهایی پژوهش بدست خواهد آمد.

برای ارائه‌ی یک نقشه‌ی راه به منظور مشاهده‌ی مفروضات، نتایج میانی و راهکار نهایی، مفاهیم مورد نیاز و نتایج بدست آمده، در شکل ۱-۴ نشان داده شده است. در این شکل، مفروضات اساسی (بر اساس مفاهیم پایه‌ای) با مستطیل‌هایی گوش‌گرد نشان داده شده است. نتایج بدست آمده از مجموعه‌ای از مفروضات، با استفاده از یک کمان جهت دار و از جهت مفروضات به نتیجه‌ی بدست آمده از آن، ترسیم شده است. در بسیاری از مواقع، نتایج میانی به عنوان مفروضاتی برای نتایج دیگر مورد استفاده قرار می‌گیرند و در نهایت، تمامی آن‌ها منجر به ایجاد راه حل نهایی می‌شوند. در این شکل، نتایج میانی با مستطیل‌هایی با گوش‌های تیز و نتیجه‌ی نهایی، با یک مستطیل تیره نشان داده شده‌اند.



شکل ۱-۴: نقشه‌ی راه حل تا رسیدن به راهکار پیشنهادی

در ادامه‌ی پژوهش، با استفاده از مدل‌های صوری، نتایج میانی را بدست خواهیم آورد و درنهایت، به نتیجه‌ی نهایی خواهیم رسید. پیش از ورود به بخش‌های بعد، کلیاتی در خصوص هرکدام از نتایج بدست آمده، بیان می‌شود:

ایجاد نگاشت یک به یک بین مدل‌های فرآیندی **BPMN2.0** و شبکه‌های پتری جریان کار یکی از اساسی‌ترین نتایج میانی این پژوهش، ایجاد یک نگاشت بین شبکه‌های پتری جریان کار و مدل‌های فرآیندی **BPMN2.0** می‌باشد. طبق شکل ۱-۴ از این نگاشت به منظور تعمیم مفهوم صحت به مدل‌های فرآیندی **BPMN2.0**، یافتن الگوهایی از صحت در آن‌ها و در نهایت ارائه یک روش صحت‌سنجی برای مدل‌های فرآیندی **BPMN2.0** استفاده می‌شود. این نگاشت، بر مبنای مفهوم عام مدل‌های فرآیندی (تعریف ۲-۸-۱) و مفهوم جریان کنترلی مطرح می‌شود. برای تشکیل این نگاشت، قدم‌های زیر اجرا شده‌است:

۱. در ابتدا، مدل فرآیندی **BPMN2.0** ساده‌سازی می‌شود؛ طی این عمل، تمامی گره‌هایی که دارای جریان کنترلی یکسانی هستند، همارز درنظر گرفته می‌شوند. با این کار، تنوع و تعدد اجزای مدل‌های فرآیندی **BPMN2.0** کنترل می‌شود. این روش ساده‌سازی، در بخش ۴-۲-۱ مورد بررسی قرار گرفته است.

۲. سپس، نگاشتی یک به یک، بین مدل‌های فرآیندی 2.0 **BPMN** و مدل‌های فرآیندی (در حالت عام و بر اساس تعاریف ۲-۸-۱ و ۲-۸-۲) ایجاد شده است.

۳. سپس، نگاشتی یک به یک، بین شبکه‌های پتری جریان کار و مدل‌های فرآیندی (در حالت عام و بر اساس تعاریف ۲-۸-۱ و ۲-۸-۲) ایجاد شده است.

۴. در نهایت با ترکیب دو نگاشت بدست آمده، یک نگاشت نهایی بین اجزای مدل‌های فرآیندی **BPMN2.0** و شبکه‌های پتری جریان کار، ایجاد شده است.

بعد از ایجاد این نگاشت، قوانینی به منظور حفظ سازگاری در قواعد ترسیم این مدل‌ها پس از نگاشت، ارائه شده است. تمامی این موارد در بخش ۴-۲-۲ مورد بررسی قرار گرفته است.

تعمیم مفهوم صحت به مدل‌های فرآیندی **BPMN2.0** همان‌طور که در بخش ۷-۲ دیده شده، صحت یک مدل فرآیندی توصیف شده با شبکه‌های پتری و شبکه‌های پتری جریان کار، به دو ویژگی مهم وابسته است:

۱. کران دار بودن، یعنی تضمین وجود یک کران بالا برای تعداد توکن هایی که در یک مکان ذخیره می شدند.

۲. زنده بودن، یعنی تضمین شلیک شدن تمامی انتقال های شبکه.

ویژگی های گفته شده، مختص شبکه های پتری جریان کار هستند و برای تعمیم مفهوم صحت به مدل های فرآیندی BPMN2.0 لازم است که دو ویژگی گفته شده، یعنی کران دار بودن و زنده بودن، برای مدل های فرآیندی BPMN2.0 نیز تعریف شوند. برای تعمیم این دو مفهوم به مدل های فرآیندی BPMN 2.0 لازم است که با استفاده از نگاشت داده شده، این دو مفهوم به مدل های فرآیندی BPMN 2.0 نیز تعمیم داده شوند تا به وسیله‌ی آن بتوان بین مدل های فرآیندی BPMN2.0 صحیح و مدل های غیر صحیح، تمایز قائل شد.

صحت سنجی بر مبنای الگوهای صحت روش پیشنهادی بر مبنای یکی از روش های صحت سنجی شبکه های پتری جریان کار، عمل می کند. این روش مبتنی بر یافتن بلوک های ساختاری صحت <sup>۱</sup> عمل می کند. برای آن که بتوان این روش را برای صحت سنجی مدل های فرآیندی BPMN 2.0 به کار برد، گام های زیر طی شده است:

۱. در ابتدا، با استفاده از نگاشت ارائه شده، بلوک های ساختاری صحت شبکه های پتری جریان کار به بلوک های ساختاری صحت در مدل های فرآیندی BPMN 2.0 تبدیل شده است. این روند در تصویر ۱-۴ مشخص است.

۲. بعد از مشخص شدن بلوک های ساختاری صحت، با استفاده از آن ها، نگاشت بدست آمده و روش صحت سنجی شبکه های پتری جریان کار ( این روش در قضیه ۴-۲-۲ مطرح شده است)، روشی برای صحت سنجی مدل های فرآیندی BPMN2.0 ارائه شده است. این روش، با استفاده از تعمیم روش اجرا شده بر روی شبکه های پتری جریان کار، بدست آمده است. این بخش از راه حل پیشنهادی، در یکی دیگر از مستطیل های شکل ۱-۴ نشان داده شده است.

در پژوهش جاری، بلوک های ساختاری صحت در دو بخش دسته بندی شده است:

۱. الگوهای صحت

---

<sup>۱</sup> Soundness Building Blocks

## ۲. پادالگوهای صحت

جزئیات مربوط به گام اول (بلوک‌های ساختار صحت در مدل‌های فرآیندی BPMN2.0) در بخش ۴-۲-۴ و پادالگوهای صحت در بخش ۵-۲-۴ مورد بررسی قرار گرفته‌اند؛ همچنین، روش صحتسنجی شبکه‌های پتری جریان کار در بخش ۶-۲-۴ و به نقل از [۳] مورد بررسی قرار گرفته و سپس در بخش ۳-۲-۴ به مدل‌های فرآیندی BPMN2.0 تعمیم پیدا کرده‌است.

**داده‌ساختار گراف دروازه‌ای و روش تحلیل دروازه‌ای** بعد از تعمیم روش صحتسنجی شبکه‌های پتری جریان کار به مدل‌های فرآیندی BPMN2.0 و بدست آوردن یک روش جدید برای صحتسنجی این مدل‌های فرآیندی، داده‌ساختار و الگوریتمی به منظور اجرای روش پیشنهادی ارائه شده‌است. داده‌ساختار پیشنهادی، به داده‌ساختار گراف دروازه‌ای<sup>۲</sup> موسوم است؛ چرا که در آن، صرفاً اطلاعات ارتباط بین دروازه‌های مدل فرآیندی BPMN2.0 به عنوان اسکلت جریان کنترلی آن، استخراج می‌شود؛ سپس الگوریتمی برای کشف الگوها و پادالگوهای صحت، بر مبنای یک روش تحلیل پسروی زیرگراف‌ها ارائه شده‌است. بدین منظور، کارهای انجام شده‌اند:

۱. گراف‌های دروازه‌ای متناظر با ساختارهای بلوکی صحت ارائه شده در بخش‌های قبل (طبق تصویر ۱-۴) بدست آمده‌است.

۲. زوج گره‌های مجاور در گراف دروازه‌ای نهایی مدل فرآیندی، با الگوها و پادالگوهای بدست آمده تطبیق داده می‌شوند و بر مبنای روش صحتسنجی تعمیم یافته، صحتسنجی مدل فرآیندی BPMN2.0 انجام می‌شود.

جزئیات این داده‌ساختار و الگوییتم، در بخش ۳-۴ ارائه شده‌است.

## ۲-۴ بیان ریاضی (صوری) روش پیشنهادی

### ۱-۲-۴ ساده‌سازی مدل‌های فرآیندی BPMN 2.0

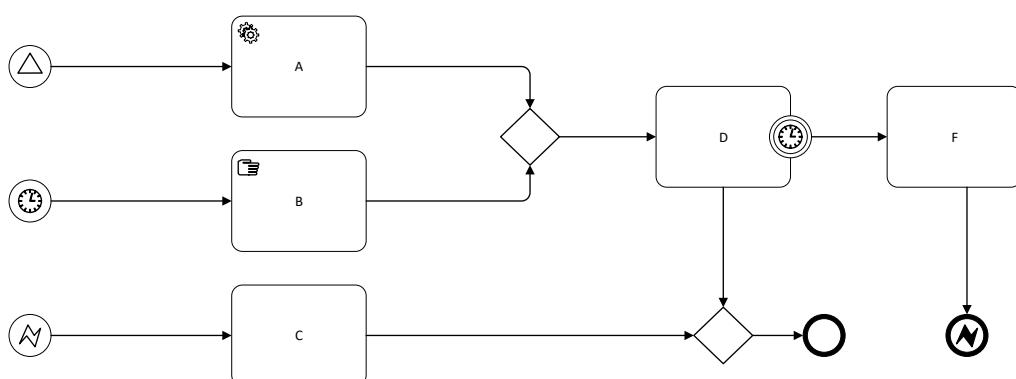
همان‌گونه که در بخش ۲-۲ مشاهده شد، مدل‌های فرآیندی BPMN 2.0 دارای اجزای متعدد و متنوعی هستند و همین تنوع اجزا و قابلیت استفاده از آن در موقعیت‌های متنوع، باعث شده‌است تا مورد توجه علاقه‌مندان زیادی قرار گیرد؛ چرا که در ک مدل‌های فرآیندی ایجاد شده با آن‌ها، راحت‌تر و گویاتر است.

<sup>2</sup>Gateway Graph

در تحلیل ساختاری این مدل‌های فرآیندی، نقش بسیاری از اجزا در هدایت جریان‌های کاری فرآیند، یکسان است. به عنوان مثال، انواع مختلفی از فعالیت‌ها، در بخش ۲-۲ معرفی شدن و هرکدام، در موقعیت خاصی مورد استفاده قرار می‌گرفتند؛ اما از نظر جریان کنترلی، تمام این فعالیت‌ها، فارغ از این‌که به چه نحوی انجام می‌شوند، صرفاً جریان کنترلی را از خود عبور می‌دهند و انشعابی درست نمی‌کنند که موجب خطاهای منطقی شود. بنابراین می‌توان تمام نمادهای فعالیت‌ها را با یک فعالیت عادی، همارز دانست؛ در نتیجه می‌توان دامنه‌ی اجزای مورد بررسی را در تحلیل ایستای مدل‌های فرآیندی BPMN 2.0 کاهش داد؛ بنابراین اجزایی از مدل فرآیندی BPMN 2.0 که از نظر رفتار جریان کنترلی، مشابه هستند در یک گروه قرار گرفته و با یک دیگر همارز خواهند بود و استفاده از یکی از آن‌ها به جای بقیه‌ی آن‌ها (در هر دسته) تغییری را در جریان کنترلی فرآیند اصلی نخواهد داد. با به کارگیری این اجزا به جای گروهی از اجزای همارز، به یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده می‌رسیم و پیش از اعمال تحلیل ایستا بر روی مدل فرآیندی مورد نظر، آن را به یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده تبدیل کرده و بعد بر اساس روش پیشنهادی، آن را مورد تحلیل ایستا قرار می‌دهیم.

نکته ۲-۴-۱. پیش از اعمال هرگونه همارزی بر روی مدل فرآیندی BPMN 2.0 لازم است مدل فرآیندی را به یک مدل فرآیندی خوش‌تعریف تبدیل نماییم. برای این منظور، تمامی پیش‌پردازش‌های مورد استفاده در بخش ۳-۱-۱ می‌بایست بر روی مدل فرآیندی اعمال گردد.

برای درک بهتر، یک مدل فرآیندی BPMN 2.0 و خوش‌تعریف در شکل ۲-۴ آمده‌است و موارد مربوط به ساده‌سازی را بر روی آن اجرا می‌کنیم.

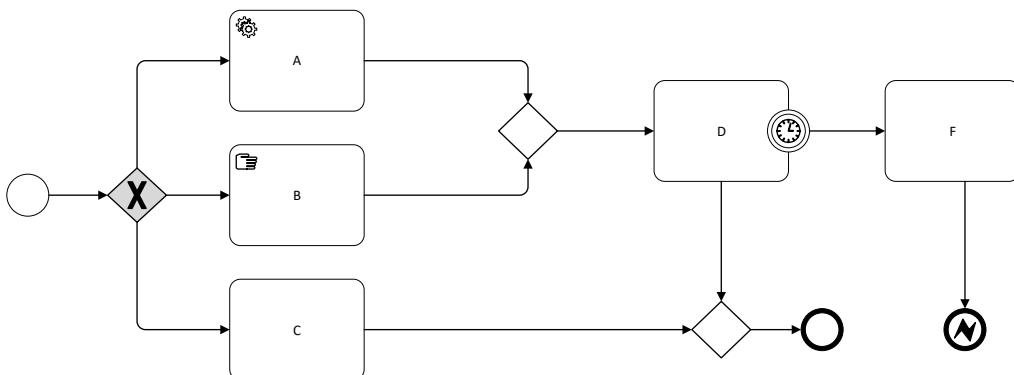


شکل ۲-۴: یک نمونه مدل فرآیندی BPMN 2.0 خوش‌تعریف به منظور ساده‌سازی

### BPMN 2.0 مدل‌های رخدادی در مدل‌های فرآیندی ساده‌سازی شده

برای آن که بتوان صحت‌سنجی را برای یک مدل فرآیندی انجام داد، لازم است که آن مدل فرآیندی دارای خاصیت شبکه‌های جریان کاری باشد؛ برای این منظور، لازم است تا تمامی فعالیت‌های داخل مدل فرآیندی، در مسیری از ابتدا تا انتهای فرآیند قرار بگیرد و نیز، فرآیند دارای یک نقطه‌ی شروع و یک نقطه‌ی پایانی باشد.

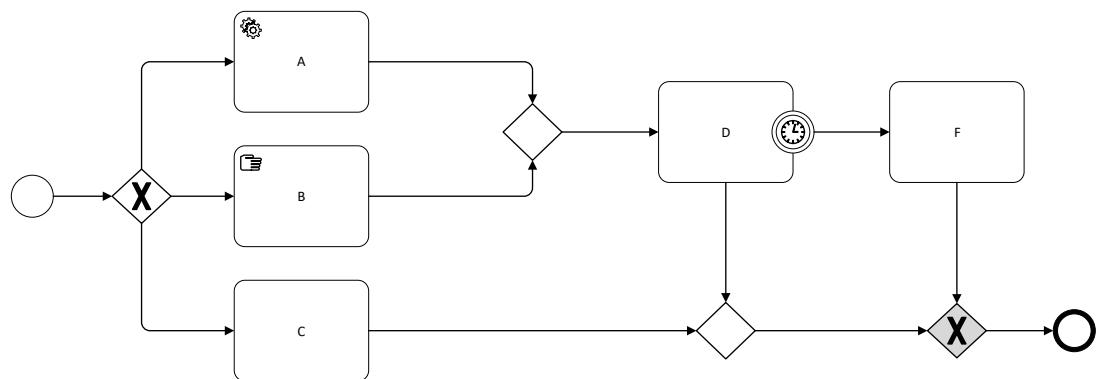
در مدل‌هایی که دارای بیش از یک رخداد شروع هستند، برای ساخت یک نقطه‌ی شروع یکتا کافیست تمامی رخدادهای شروع را حذف کرده و شاخه‌های آن‌ها به یک انشعاب انحصاری متصل شود و ابتدای انشعاب انحصاری، به یک رخداد شروع معمولی متصل شود؛ با این کار، بعد از آغاز به کار یک مورد کاری، شروع مورد کاری فقط از طریق یکی از شاخه‌های دروازه‌ی انشعاب انحصاری صورت می‌گیرد. این کار، تغییری در جریان فرآیند کاری ایجاد نمی‌کند؛ چرا که یک فرآیند فقط از یک طریق شروع می‌شود و دقیقاً از یکی از شاخه‌های متصل به رخدادهای شروع‌کننده، شروع به اجرا می‌شود. نحوه‌ی اعمال این ساده‌سازی در شکل ۳-۴ نشان داده شده‌است.



شکل ۳-۴: تبدیل ورودی‌های چندگانه‌ی شکل ۲-۴ به یک ورودی

در صورتی که مدل فرآیندی دارای بیش از یک رخداد پایانی است، کافیست تمامی رخدادهای پایانی حذف شوند و به واسطه‌ی یک اتصال انحصاری به یک پایان معمولی متصل گردند. این کار نیز، تغییری در جریان فرآیند به وجود نمی‌آورد؛ چرا که یک فرآیند، صرفاً از یک طریق و یک بار می‌تواند پایان یابد. در شکل ۴-۴ این ساده‌سازی بر روی مدل فرآیندی مطرح شده در شکل ۳-۴، اعمال گشته است.

نتیجه ۲-۴. با اعمال ساده‌سازی بر روی رخدادهای شروع چندگانه و بر روی رخدادهای پایانی چندگانه، مدل فرآیندی به یک شبکه‌ی جریان کار تبدیل خواهد شد؛ چرا که دارای یک نقطه‌ی ورودی



شکل ۴-۴: تبدیل خروجی‌های چندگانه‌ی شکل ۳-۴ به یک ورودی

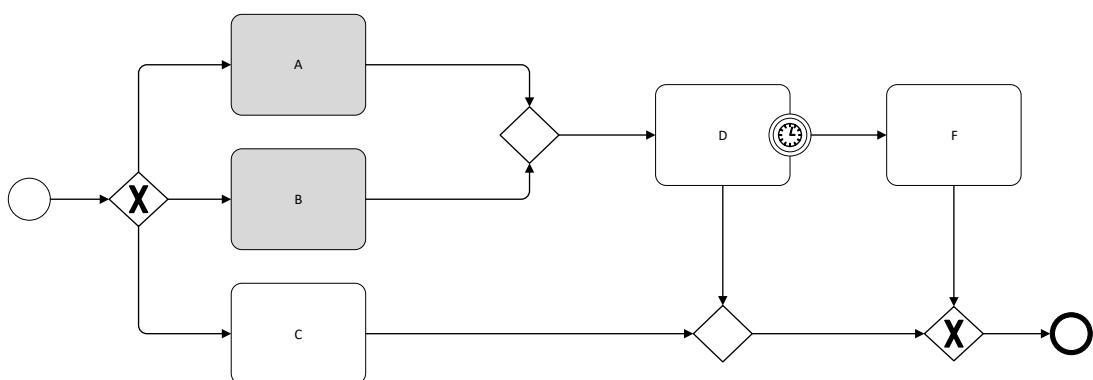
و یک نقطه‌ی خروجی است و تمامی فعالیت‌ها نیز باید در مسیری از رخداد شروع و رخداد پایان قرار گیرند.

مدل‌های فرآیندی، می‌توانند شامل یک یا چند رخداد میانی نیز باشند که این رخدادها، می‌توانند انواع مختلفی داشته باشند. پیش‌تر، انواع مختلف رخدادها در بخش ۲-۲ و در شکل ۲-۲ معرفی شده‌اند. رخدادهای میانی می‌توانند تغییری نکنند و همگی از نظر الگوی جريان کنترلی، از الگوی جريان کنترلی دنباله‌ای تبعیت می‌کنند.

### مدل‌های فعالیتی در مدل‌های فرآیندی BPMN2.0 ساده‌سازی شده

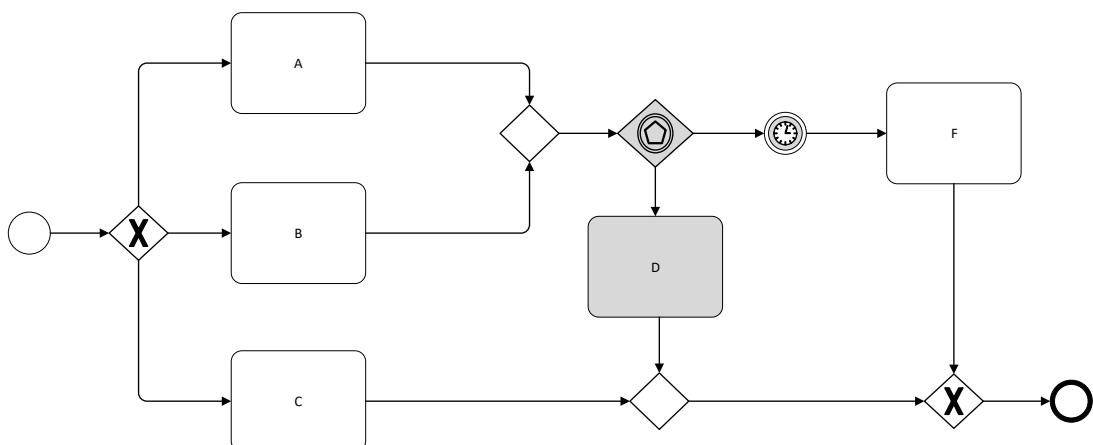
در بخش ۲-۲ با فعالیت‌ها در مدل‌های فرآیندی 2.0 BPMN آشنا شدیم و مشاهده کردیم که طبق تصویر ۲-۱ انواع مختلفی از فعالیت‌ها وجود دارند. تمام این فعالیت‌ها، فارغ از این‌که در کدامین موقعیت به کار می‌روند، صرفاً جريان کاری فعالیت را از خود عبور می‌دهند و انشعابی ایجاد نمی‌کنند. بنابراین به منظور ساده‌سازی مدل‌های فرآیندی، به جای تمامی فعالیت‌ها، از یک فعالیت معمولی استفاده می‌شود؛ به عنوان نمونه، در تصویر ۴-۵ تمامی فعالیت‌های غیرساده‌ی به کار رفته در تصویر ۴-۴ به فعالیت‌هایی ساده تبدیل شده‌اند.

یک مدل فرآیندی 2.0 BPMN می‌تواند دارای رخدادهای کرانی باشد. این رخدادها، به مرز یک فعالیت متصل می‌شوند و در صورتی که بروز پیدا کنند، ادامه‌ی جريان فرآیند از مسیر متصل به آن‌ها ادامه پیدا می‌کند و در غیر این صورت، ادامه‌ی جريان فرآیند از مسیر متصل به فعالیت ادامه پیدا می‌کند. به عنوان مثال، در تصویر ۴-۵ یک رخداد کرانی به فعالیت D متصل شده‌است؛ در صورتی که رخداد



شکل ۴-۵: تبدیل فعالیت‌های سرویسی و دستی به کار رفته در شکل ۴-۴ به یک فعالیت عادی

میانی زمانی، زمان معینی را سپری کند و به پایان برسد، ادامه‌ی جریان فرآیند از مسیر فعالیت F ادامه خواهد یافت و در غیر این صورت، از مسیر دیگر، یعنی مسیر متصل به فعالیت D ادامه خواهد یافت. چنین جریانی، در عمل یک جریان انحصاری است و می‌توان یک دروازه‌ی انشعاب انحصاری را به جای آن نشاند. البته در زبان مدل‌سازی BPMN 2.0 در صورتی که بعد از دروازه، یک رویداد وجود داشته باشد، از دروازه‌ی رویدادی استفاده خواهد شد. چنان‌چه بر روی مدل فرآیندی ارائه شده در تصویر ۴-۵ این ساده‌سازی اعمال شود، مدل فرآیندی ترسیم شده در تصویر ۴-۶ به وجود خواهد آمد.



شکل ۴-۶: تبدیل رخداد کرانی به ترکیبی از دروازه‌ی انحصاری ۴-۵ به یک فعالیت عادی

در تحلیل ایستای مدل‌های فرآیندی، به گفته‌ی داکمن در پژوهش [۴] می‌توان زیرفرآیندها را همانند فعالیت‌های معمولی مورد تحلیل ایستا قرار داد؛ بدیهی است که صحتسنجی کل مدل فرآیندی، منوط به صحتسنجی فرآیندهای تعریف شده در زیرفرآیندها خواهد بود.

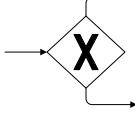
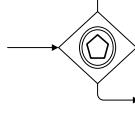
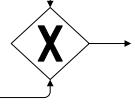
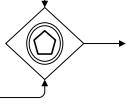
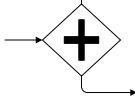
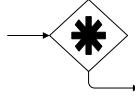
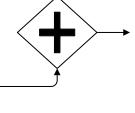
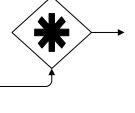
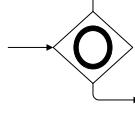
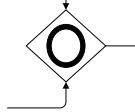
## مدل‌های دروازه‌ای در مدل‌های فرآیندی ساده‌سازی شده BPMN 2.0

مدل‌های دروازه‌ای، نقش مهمی را در انشعاب و اتصال شاخه‌های فرآیند را در مدل‌های فرآیندی دارند. استفاده‌ی صحیح و چینش دروازه‌ها، نقشی اساسی را در صحت ساختاری مدل‌های فرآیندی دارد. همان‌طور که در بخش ۲-۲ مشاهده شد، انواع متعددی از دروازه‌ها در مدل‌های فرآیندی BPMN 2.0 به کار می‌روند.

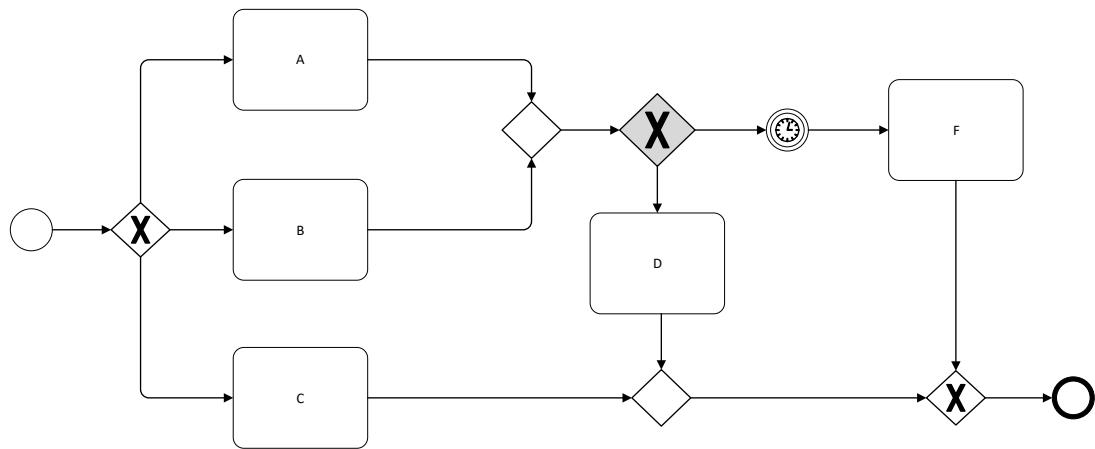
تنوع دروازه‌ها در این مدل‌های فرآیندی باعث شده‌است تا زبان مدل‌سازی BPMN 2.0 بسیار گویا‌تر شده و در موقعیت‌های متعدد، قابل استفاده باشد. اما از نظر جریان‌های کنترلی فرآیندهای کاری، بسیاری از دروازه‌های معرفی شده در مدل‌های فرآیندی BPMN 2.0 از یک الگوی جریان کنترلی پیروی می‌کنند و در نتیجه، از نظر ساختاری دارای یک تاثیر بر روی فرآیند کسب و کار خواهند داشت؛ بنابراین می‌توان دروازه‌های معادل و دارای یک الگوی جریان کار را در گروه‌هایی تقسیم‌بندی کرد و همه‌ی اعضای آن گروه را دارای یک نوع تاثیر بر جریان کنترلی فرآیند دانست. این بدان معنا خواهد بود که در صورت استفاده از یکی از اعضای گروه به جای تمام اعضای دیگر، تغییری در جریان کنترلی فرآیند ایجاد نمی‌شود. دروازه‌هایی که از نظر الگوهای جریان کنترلی، همانند یک دیگر عمل می‌کنند و دارای یک گراف زنجیره‌ی رخدادی مشترک هستند، در جدول ۴-۱ سازمان‌دهی شده‌اند. دروازه‌های پیچیده، دارای عملکرد مشخصی نیستند و در مواردی می‌توانند به صورت انشعاب و یا اتصال موازی، عطفی، فصلی و انحصاری عمل کنند. نظر به این‌که بیشترین احتمال بروز خطاها در ساختاری در خصوص دروازه‌های موازی رخ می‌دهد [۲۸] در خصوص دروازه‌های پیچیده بدترین فرض ممکن در نظر گرفته شده و در دسته‌ی دروازه‌های موازی دسته‌بندی شده‌است. \*

نکته ۴-۲-۳. با توجه به این که دروازه‌های انشعاب و اتصال انحصاری، حالت خاصی از دروازه‌های انشعاب و اتصال فصلی نیز می‌باشند، می‌توانند با یک دیگر در یک گروه جریان کنترلی قرار گیرند.

نمونه ۴-۲-۴. به عنوان مثال، در شکل ۷-۶ نتیجه‌ی اعمال ساده‌سازی مدل‌های دروازه‌ای بر روی مدل فرآیندی تصویر ۶-۴ قابل مشاهده است. مشاهده می‌شود که در این مدل فرآیندی، بدون تغییر در جریان کنترلی فرآیند، دروازه‌ی انحصاری به جای دروازه‌ی انحصاری رویدادی جایگزین شده‌است؛ چرا که در هر صورت، یکی از شاخه‌های خروجی فعال خواهد شد و این جایگزینی، باعث تغییر در جریان کنترلی فرآیند نمی‌شود.

نام الگوی جریان کنترلی کنترلی	جزء کنترلی نماینده	اجزای کنترلی معادل
الگوی جریان کنترلی انشعاب انحصاری	دروازه‌ی انشعاب انحصاری	 
الگوی جریان کنترلی اتصال انحصاری	دروازه‌ی اتصال انحصاری	 
الگوی جریان کنترلی انشعاب موازی	دروازه‌ی انشعاب موازی	 
الگوی جریان کنترلی اتصال موازی	دروازه‌ی اتصال موازی	 
الگوی جریان کنترلی انشعاب انحصاری	دروازه‌ی انشعاب عطفی	
الگوی جریان کنترلی اتصال موازی	دروازه‌ی اتصال موازی	
الگوی جریان کنترلی دنباله‌ای	دروازه‌ی دنباله‌ای	

جدول ۱-۴: دسته‌بندی دروازه‌های مشابه بر اساس الگوهای جریان کنترلی



شکل ۷-۴: تبدیل رخداد دروازه‌ی انحصاری رویدادی به دروازه‌ی انحصاری، در فرآیند شکل ۶-۴

## ۲-۲-۴ نگاشت بین مدل‌های فرآیندی BPMN2.0 و شبکه‌های پتری جریان کار

### نگاشت بین مدل‌های فرآیندی BPMN2.0 ساده شده و مدل‌های فرآیندی عام

بنابر تعریف ۲-۱-۸ یک مدل فرآیندی، گرافی جهت‌دار مثل  $P = (N, E)$  است؛ به طوری که مجموعه‌ی یال‌ها و نیز  $N$  مجموعه‌ی گره‌های آن است. مجموعه‌ی گره‌ها نیز به سه زیر مجموعه افزایش دارند:

$$N = N_A \cup N_E \cup N_G \quad (1-4)$$

به گونه‌ای که  $N_A$  مجموعه‌ی فعالیت‌ها،  $N_E$  مجموعه‌ی رخدادها و  $N_G$  مجموعه‌ی دروازه‌ها هستند. در ادامه، لازم است نمادهای مدل‌های فرآیندی 2.0 BPMN ساده‌سازی شده را بر اساس این تعریف دسته‌بندی نماییم. یادآوری می‌شود که در مدل‌های فرآیندی BPMN 2.0 ساده‌سازی شده، اجزای دارای یک الگوی جریان کار مشترک دسته‌بندی شده و یکی از آن‌ها به عنوان سردسته انتخاب شده و به جای بقیه‌ی اجزای دسته، مورد استفاده قرار می‌گیرد. هدف از این کار، کاهش نمادهای درگیر مدل فرآیندی BPMN 2.0 در تحلیل ایستا بوده است و بدین منظور صورت گرفته است که بتوان یک مدل فرآیندی ساده‌تر و با جریان کار یکسان را از مدل فرآیندی اصلی استخراج نمود تا بتوان تحلیل ایستا را به راحتی بر روی آن اجرا کرد. یک مدل فرآیندی BPMN 2.0 و ساده‌سازی شده را به صورت زیر تعریف می‌کنیم:

تعريف ۴-۲-۵. یک مدل فرآیندی  $BPMN 2.0$  ساده‌سازی شده مانند  $SBPMN = (B, E)$ ، گرافی جهت‌دار با مجموعه‌ای از گره‌ها ( $B$ ) و مجموعه‌ای از یال‌ها ( $E$ ) است و یک مدل فرآیندی است که مطابق با تعریف ۲-۸-۱ بیان می‌شود و طبق تعریف ۲-۸-۲ دارای مجموعه‌ای از مدل‌های رخدادی، فعالیتی و دروازه‌ای می‌باشد که مجموعه‌ی گره‌ها را افزایش می‌کند؛ یعنی:

$$B = B_A \cup B_E \cup B_G \quad (2-4)$$

بر طبق جدول ۱-۴ مدل‌های دروازه‌ای مجاز در این مدل‌های فرآیندی (مجموعه‌ی  $B_G$ ) عبارتند از:

۱. دوازه‌های انشعباب و اتصال انحصاری

۲. دوازه‌های انشعباب و اتصال موازی

۳. دوازه‌های انشعباب و اتصال فصلی

۴. دوازه‌های دنباله‌ای

همچنین مدل‌های رخدادی مجاز (مجموعه‌ی  $B_E$ ) عبارتند از :

۱. رخداد شروع ( فقط یک مرتبه می‌تواند مورد استفاده قرار گیرد)

۲. رخداد پایان ( فقط یک مرتبه می‌تواند مورد استفاده قرار گیرد)

۳. رخدادهای میانی (طبق شکل ۲-۲ )

همچنین مجموعه‌ی رخدادهای فعالیتی (مجموعه‌ی  $B_A$ )، صرفاً حاوی فعالیت‌ها می‌باشد.

لم ۶-۲-۴. اگر مجموعه‌ی  $B$  مجموعه‌ی تمامی مدل‌های فرآیندی  $BPMN 2.0$  ساده‌سازی شده و مجموعه‌ی  $P$  مجموعه‌ی تمامی مدل‌های فرآیندی باشد، نگاشتی یک به یک مانند  $P \xrightarrow{f} W$  وجود دارد، به طوری که هر مدل فرآیندی را به یک مدل  $BPMN 2.0$  بنگارد (و بالعکس).

برهان. هر مدل فرآیندی  $BPMN 2.0$  ساده شده، دارای مجموعه‌ای از یال‌ها و مجموعه‌ای از گره‌ها است. گره‌ها، به سه زیر مجموعه‌ی مدل‌های فعالیتی، مدل‌های رخدادی و مدل‌های دروازه‌ای افزایش می‌شوند. فرض کنید مجموعه‌ی  $B$  مجموعه‌ی تمام مدل‌های فرآیندی  $BPMN 2.0$  ساده‌سازی شده باشد.

هر مدل فرآیندی BPMN 2.0 ساده‌سازی شده مانند  $b = (B_b, E_b) \in B$  بیانگر یک مدل فرآیندی مثل  $(N_p, E_p)$  در مجموعه مدل‌های فرآیندی در  $P$  است. بنابراین، نگاشتی مانند  $f$  وجود دارد که یک مدل فرآیندی مانند  $p \in P$  را به یک مدل فرآیندی مانند  $b \in B$  می‌نگارد. اگر چنین نباشد، مدل  $b$  یک مدل فرآیندی نخواهد بود؛ چرا که مدل‌های فرآیندی BPMN 2.0 ساده‌سازی شده، نوع خاصی از مدل‌های فرآیندی BPMN 2.0 هستند، در صورتی که  $b$  مدل فرآیندی نباشد، به این معنا خواهد بود که یک مدل فرآیندی BPMN 2.0 وجود دارد که مدل فرآیندی نیست و این با مفروضات مسئله و مفروضات [۲] در تناقض است.

برای آن که نشان دهیم که چنین نگاشت دوسویه‌ای وجود دارد، نگاشت  $f$  را به سه نگاشت مجزا افزای می‌کنیم. هر کدام از این سه نگاشت، ارتباطی را بین مدل‌های فعالیتی، مدل‌های دروازه‌ای و مدل‌های رخدادی در بین مدل‌های فرآیندی و مدل‌های فرآیندی BPMN 2.0 ساده‌سازی شده، برقرار می‌کند.

**نگاشت بین مدل‌های فعالیتی و فعالیت‌ها** فرض کنید مدل فرآیندی  $(N_p, E_p) \in P$  بیان صوری و کلی مدل فرآیندی  $b = (B_b, E_b) \in B$  باشد (طبق تعریف ۲-۸-۱) در مدل فرآیندی BPMN 2.0 ساده‌سازی شده  $b \in B$ ، برای هر فعالیتی مثل  $a \in B_A \subset N_b$  یک مدل فعالیتی مثل  $a' \in N_A \subset N_p$  در مدل فرآیندی  $p \in P$  وجود خواهد داشت، به طوری که هردو مدل فعالیتی، یک فعالیت را اجرا خواهند کرد. بنابراین، نگاشتی مثل  $f_A$  وجود خواهد داشت، به گونه‌ای که یک فعالیت را از مجموعه فعالیت‌های  $b$  به یک مدل فعالیتی در مجموعه مدل‌های فعالیتی  $b$  بنگارد؛

$$f_A : B_A \longrightarrow N_A \quad (3-4)$$

از سوی دیگر، بر طبق تعریف ۴-۵-۲ هر مدل فعالیتی در مدل فرآیندی  $p$ ، در مدل‌های فرآیندی BPMN 2.0 ساده‌سازی شده، در قالب یک فعالیت قابل بازنمایی است؛ بنابراین، نگاشتی از فعالیت‌های یک مدل فرآیندی، به مجموعه فعالیت‌های یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده وجود خواهد داشت، به گونه‌ای که هر دو، یک فعالیت را بازنمایی کنند.

$$f_A^{-1} : N_A \longrightarrow B_A \quad (4-4)$$

بر اساس دو رابطه‌ی ۳-۴ و ۴-۴ نگاشت دوسویه است و بر اساس آن می‌توان

نتیجه گرفت که هر فعالیت، دارای یک مدل فعالیتی در مدل فرآیندی معادلش است و هر مدل فعالیتی موجود در یک مدل فرآیندی، می‌تواند با استفاده از یک فعالیت، در یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده، به کار رود.

**نگاشت بین مدل‌های رخدادی و رخدادها** فرض کنید مدل فرآیندی  $p = (N_p, E_p) \in P$  بیان صوری و کلی مدل فرآیندی  $b = (B_b, E_p) \in B$  باشد (طبق تعریف ۲-۸-۱) در مدل فرآیندی BPMN 2.0 ساده‌سازی شده‌ی  $b \in B$ ، برای هر رخدادی مثل  $e \in B_E \subset N_b$  یک مدل فعالیتی مثل  $e' \in N_E \subset N_p$  وجود خواهد داشت، به طوری که هردو مدل رخدادی، منجر به بروز یک رخداد می‌شوند. بنابراین، نگاشتی مثل  $f_E$  وجود خواهد داشت، به گونه‌ای که یک رخداد را از مجموعه‌ی رخدادهای  $b$  (یعنی مجموعه‌ی  $B_E$ ) به یک مدل رخدادی در مجموعه‌ی مدل‌های رخدادی  $p$  (یعنی مجموعه‌ی  $N_E$ ) بنگارد؛

$$f_E : B_E \longrightarrow N_E \quad (5-4)$$

از سوی دیگر، بر طبق تعریف ۴-۵-۲ هر مدل رخدادی در مدل فرآیندی  $p$ ، در مدل‌های فرآیندی BPMN 2.0 ساده‌سازی شده، در قالب یک رخداد قابل بازنمایی است؛ تمامی رخدادها، بجز رخدادهای یکتای شروع و پایان، به یک مدل رخدادی میانی نگاشته می‌شوند. بنابراین، نگاشتی از رخدادهای یک مدل فرآیندی، به مجموعه‌ی رخدادهای یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده وجود خواهد داشت، به گونه‌ای که هر دو، یک رخداد را بازنمایی کنند.

$$f_E^{-1} : N_E \longrightarrow B_E \quad (6-4)$$

بر اساس دو رابطه‌ی ۵-۴ و ۶-۴ نگاشت  $f_E$  یک نگاشت دوسویه است و بر اساس آن می‌توان نتیجه گرفت که به ازای هر رخداد در یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده، یک مدل رخدادی در مدل فرآیندی معادلش وجود دارد و به ازای هر مدل رخدادی موجود در یک مدل فرآیندی، می‌تواند با استفاده از یک رخداد میانی (مگر برای رخدادهای آغازین و پایانی که باید با رخداد آغازین و پایانی مدل شوند)، در یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده، به کار رود.

نگاشت بین مدل‌های دروازه‌ای و دروازه‌ها فرض کنید مدل فرآیندی  $P = (N_p, E_p) \in P$  بیان صوری و کلی مدل فرآیندی  $b = (B_b, E_b) \in B$  باشد (طبق تعریف ۲-۸-۱)

در تعریف ۲-۸-۲ مشاهده می‌شود که به هر مدل دروازه‌ای، یک نوع جریان کنترلی یکتا تخصیص داده شده‌بود و این نوع جریان کنترلی، به وسیله‌یتابع  $type : N_G \longrightarrow C$  مشخص می‌شود؛ بنابراین تابع  $type$  به هر مدل دروازه‌ای، یک ساختار را از مجموعه‌ی ساختارهای کنترلی، یعنی مجموعه‌ی  $C$ ، تخصیص می‌دهد. از سوی دیگر، در مدل‌های فرآیندی BPMN 2.0 ساده‌سازی شده، دروازه‌های دارای یک جریان کنترلی مشترک در دسته‌هایی دسته‌بندی شده‌اند و از هر دسته، یک دروازه به عنوان نماینده انتخاب شده و در صورتی که به جای هر کدام از دروازه‌های هم‌گروهش به کار برود، تغییری در جریان کنترلی فرآیند ایجاد نمی‌شود. بنابراین می‌توان برای مدل‌های فرآیندی BPMN 2.0 تابع  $type_{BPMN}$  را به صورت زیر تعریف نمود:

$$type_{BPMN} : B_G \longrightarrow C \quad (7-4)$$

این تابع، نگاشت را بر اساس نگاشتهای تعریف شده در جدول ۴-۱ انجام می‌دهد.

در مدل فرآیندی BPMN 2.0 ساده‌سازی شده‌ی  $B$ ، برای هر دروازه‌ای مثل  $b \in B$ ، یک مدل دروازه‌ای مثل  $g \in B_G \subset B_b$  در مدل فرآیندی  $p \in N_p \subset N_G$  وجود خواهد داشت، به طوری که هر دو دروازه، یک ساختار جریان کنترلی را اجرا می‌کنند؛ به عبارت دیگر :

$$\forall g \in B_G \exists g' \in N_G \quad s.t \quad type_{BPMN}(g) = type(g') \quad (8-4)$$

بنابراین، نگاشتی مثل  $f_G$  وجود خواهد داشت، به گونه‌ای که یک دروازه را از مجموعه‌ی دروازه‌های  $b$  (یعنی مجموعه‌ی  $B_G$ ) به یک مدل دروازه‌ای در مجموعه‌ی مدل‌های دروازه‌ای  $p$  (یعنی مجموعه‌ی  $N_G$ ) بنگارد؛

$$f_E : B_G \longrightarrow N_G \quad (9-4)$$

از سوی دیگر، بر طبق تعریف ۴-۵-۲ هر مدل دروازه‌ای در مدل فرآیندی  $p$ ، در مدل‌های فرآیندی BPMN 2.0 ساده‌سازی شده، در قالب یک دروازه قابل بازنمایی است، مشروط بر آن که هر دو دارای

یک نوع الگوی جریان کنترلی باشند؛ بنابراین خواهیم داشت:

$$\forall g' \in N_G \exists g \in B_G \quad s.t \quad type_{BPMN}(g) = type(g') \quad (10-4)$$

بنابراین، معکوس رابطه‌ی قبل به شکل زیر قابل تعریف خواهد بود:

$$f_G^{-1} : N_G \longrightarrow B_G \quad (11-4)$$

بر اساس دو رابطه‌ی ۹-۴ و ۱۱-۴ نگاشت  $f_G$  یک نگاشت دوسویه است و بر اساس آن می‌توان نتیجه گرفت که به ازای هر رخداد در یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده، یک مدل رخدادی در مدل فرآیندی معادلش وجود دارد و به ازای هر مدل رخدادی موجود در یک مدل فرآیندی، می‌تواند با استفاده از یک رخداد میانی (مگر برای رخدادهای آغازین و پایانی که باید با رخداد آغازین و پایانی مدل شوند)، در یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده، به کار رود.

با توجه به این‌که یکایک گره‌های یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده، مثل  $B = (B_b, E_b) \in B$  را به صورت دو سویه به گره‌های مدل‌های فرآیندی  $P = (N_p, E_p) \in P$  نگاشت دادیم، می‌توانیم نگاشتی مانند  $f$  را بین گره‌های مدل فرآیندی BPMN 2.0 ساده‌سازی شده را با گره‌های یک مدل فرآیندی، به صورت رابطه‌ای دوسویی و به صورت زیر تعریف کنیم :

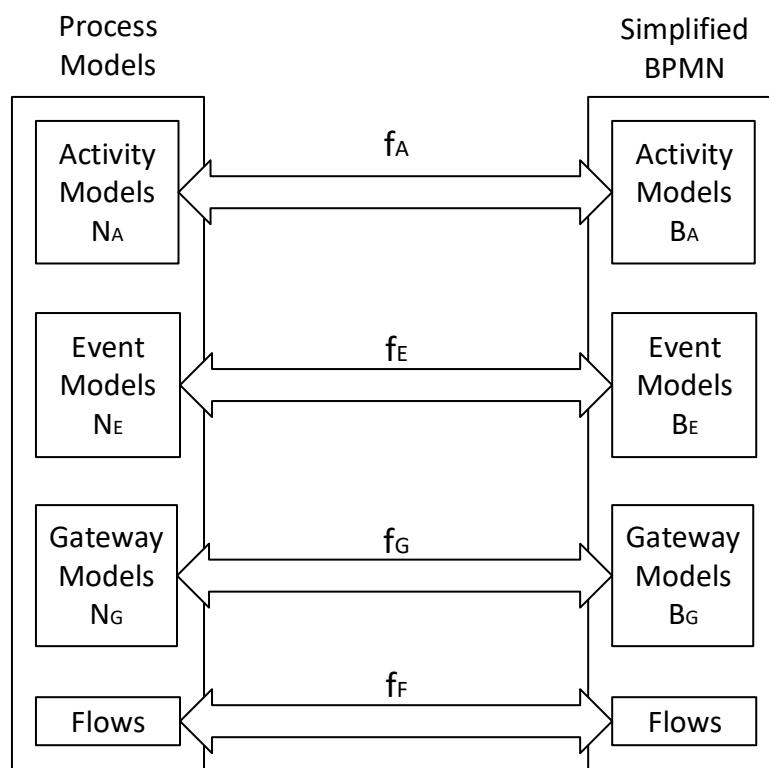
$$f : N_p \longrightarrow B_b \quad s.t \quad f(n) = \begin{cases} f_A(n) & n \in N_A \\ f_E(n) & n \in N_E \\ f_G(n) & n \in N_G \end{cases} \quad (12-4)$$

همچنین، برای نگاشت بین کمان‌های یک مدل فرآیندی BPMN 2.0 ساده‌سازی شده و کمان‌های مدل‌های فرآیندی معادل آن‌ها، نگاشت  $E_p \longrightarrow E_b$  را می‌توان به صورت زیر تعریف نمود:

$$f_F = \{(f(x), f(y)) | (x, y) \in E_p \wedge x, y \in N_p\} \quad (13-4)$$

$$f_F^{-1} = \{(f(x)^{-1}, f(y)^{-1}) | (x, y) \in E_b \wedge x, y \in B_b\} \quad (14-4)$$

بنابراین، می‌توان با نگاشت دوسویه‌ی  $f$  گره‌های یک مدل فرآیندی را به گره‌های معادلش و در مدل فرآیندی 2.0 BPMN ساده‌سازی شده‌ی معادلش تبدیل نمود (و بالعکس). در خصوص یال‌ها نیز می‌توان با استفاده از تابع دوسویه‌ی  $f_F$  این نگاشت یک به یک را انجام داد. بنابراین، هر مدل فرآیندی 2.0 ساده‌سازی شده به یک مدل فرآیندی قابل تبدیل است و بالعکس. نگاشت ارائه شده در این اثبات،  $\square$  به طور خلاصه در تصویر ۴-۸ آمده‌است.



شکل ۴-۸: نگاشت یک به یک مدل‌های فرآیندی 2.0 BPMN و مدل‌های فرآیندی

### نگاشت بین شبکه‌های پتری جریان کار با مدل‌های فرآیندی عام

بنابر تعریف ۱-۸-۲ یک مدل فرآیندی، گرافی جهتدار مثل  $P = (N, E)$  است؛ به طوری که مجموعه‌ی یال‌ها و نیز  $N$  مجموعه‌ی گره‌های آن است. مجموعه‌ی گره‌ها نیز به سه زیر مجموعه افزای

می‌شدن:

$$N = N_A \cup N_E \cup N_G \quad (15-4)$$

به گونه‌ای که  $N_A$  مجموعه‌ی فعالیت‌ها،  $N_E$  مجموعه‌ی رخدادها و  $N_G$  مجموعه‌ی دروازه‌ها هستند. در ادامه، لازم است نمادهای شبکه‌های پتری جریان کار را بر اساس این تعریف دسته‌بندی نماییم.

**مجموعه‌ی مدل‌های فعالیتی** در شبکه‌های پتری جریان کار، تمام فعالیت‌ها و نیز، زیرفرآیندها به وسیله‌ی انتقال‌های عادی نشان داده می‌شوند. بنابراین، بر طبق تعریف مدل فرآیندی، مجموعه‌ی مدل‌های فعالیتی برابر است با مجموعه‌ی انتقال‌های غیرانشعابی و غیر رویدادی.

**مجموعه‌ی مدل‌های رخدادی** در شبکه‌های پتری جریان کار، طبق نمادگذاری مطرح شده در [۲] فعال شدن برخی از انتقال‌ها، منوط به رخ دادن رخداد معینی است. این دسته از انتقال‌ها در مجموعه‌ی مدل‌های رویدادی واقع می‌شوند؛ همچنین، مکان اولیه (مکان ؟) و مکان نهایی (مکان ۰) نیز در مدل‌های رویدادی در نظر گرفته می‌شوند. چرا که بازنمایی کننده‌ی رخداد شروع و رخداد پایانی هستند.

**مدل‌های دروازه‌ای** در شبکه‌های پتری جریان کار، دروازه‌ها می‌توانند به صورت صریح و یا ضمنی به کار گرفته شوند. دروازه‌های صریح، انتقال‌هایی هستند که در شکل ۱۲-۲ نشان داده شده‌اند و ۴ نوع هستند. دروازه‌های غیر صریح، مکان‌هایی هستند که دارای درجه‌ی ورودی و یا خروجی بزرگ‌تر از ۱ هستند. بر طبق نمادگذاری پیشنهادی توسط [۴] این دروازه‌های ضمنی، به سه شکل به کار می‌روند:

۱. مکان‌هایی که دارای درجه‌ی ورودی ۱ و درجات خروجی بزرگ‌تر از ۱ هستند، به عنوان دروازه‌های انشعاب انحصاری به کار گرفته می‌شوند و انتقال‌هایی که در ادامه‌ی آن‌ها و یا قبل از آن‌ها می‌آیند، می‌توانند مدل‌های فعالیتی و یا رخدادی باشند.

۲. مکان‌هایی که دارای درجه‌ی ورودی بیش از ۱ و درجه‌ی خروجی ۱ هستند، به عنوان دروازه‌های اتصال فصلی غیرصریح (ضمنی) به کار گرفته می‌شوند و انتقال‌هایی که در ادامه‌ی آن‌ها و یا قبل از آن‌ها می‌آیند، می‌توانند مدل‌های فعالیتی و یا رخدادی باشند.

۳. مکان‌هایی که دارای درجه‌ی ورودی ۱ و درجه‌ی خروجی ۱ هستند، می‌توانند به عنوان

مدل‌های دروازه‌ای دنباله‌ای [۲] به کار گرفته شوند؛ اما برای سهولت، می‌توان آن‌ها را به عنوان مسیر مستقیم تفسیر نمود. در [۲] نیز برای توصیف الگوی جریان کنترلی دنباله‌ای، چنین کاری صحیح در نظر گرفته شده است.

**نتیجه ۴-۲-۷.** بر طبق توضیحات بالا، در یک فرآیند مدل‌سازی شده با زبان شبکه‌های پتری جریان کاری مثل  $PN = (P, T, F)$ ، سه نوع انتقال وجود دارد که هر کدام از آن‌ها، به طور صریح در این زبان تعریف شده و دارای نمادهایی منحصر به فرد و یکتا هستند. این انتقال‌ها شامل انتقال‌های زیر هستند:

۱. انتقال‌های منشعب‌کننده و متصل‌کننده (شکل ۱۲-۲)

۲. انتقال‌هایی که بازنمایی‌کننده‌ی یک فعالیت در فرآیند هستند (انتقال‌های معمولی)

۳. انتقال‌های رخدادی که بر اساس نماد رخداد، رخداد خاصی را بازنمایی می‌کنند.

با توجه به این‌که این انتقال‌ها در جای دیگری به‌جز مقصود خود به کار نمی‌روند، می‌توان مجموعه‌ی انتقال‌ها را به سه زیرمجموعه افزای نمود. انتقال‌های منشعب‌کننده و متصل‌کننده در مجموعه‌ی  $T_G$ ، انتقال‌های فعالیتی در مجموعه‌ی  $T_A$  و انتقال‌های رخدادی در مجموعه‌ی  $T_E$  واقع می‌شوند. بنابراین خواهیم داشت:

$$T = T_A \cup T_E \cup T_G \quad (16-4)$$

**نتیجه ۴-۲-۸.** بر اساس تعریف ۱-۸-۲ و تعریف ۲-۸-۲، برای هر فرآیند ایجاد شده با زبان شبکه‌های پتری جریان کار مثل  $PN = (P, T, F)$  یک مدل فرآیندی مثل  $PM = (N, E)$  وجود دارد، به طوری که

$$N = N_A \cup N_E \cup N_G \quad (17-4)$$

**نتیجه ۴-۲-۹.** بر طبق نتیجه‌ی ۴-۲-۸ می‌توان برای هر فرآیند مدل‌سازی شده با شبکه‌های پتری جریان کار، یک مدل فرآیندی قائل شد.

**لم ۴-۲-۱۰.** اگر مجموعه‌ی  $WF$  مجموعه‌ی تمامی فرآیندهای توصیف شده با استفاده از شبکه‌های پتری جریان کار و مجموعه‌ی  $P$  مجموعه‌ی تمامی مدل‌های فرآیندی باشد، نگاشتی یک به یک مانند  $g$  وجود دارد، به طوری که هر مدل فرآیندی را به یک شبکه‌ی پتری جریان کار بنگارد (و بالعکس).

برهان. هر فرآیند ایجاد شده با استفاده از شبکه‌های پتری جریان کار، دارای مجموعه‌ای از یال‌ها و مجموعه‌ای از گره‌ها است. در حالت کلی، گره‌های شبکه‌های پتری جریان کار به دو بخش مکان‌ها و انتقال‌ها تقسیم می‌شوند. بر اساس مفاهیم اجزای مدل‌های فرآیندی، مجموعه‌ی گره‌ها به سه دسته تقسیم می‌شوند:

۱. مجموعه‌ی گره‌های فعالیتی (مجموعه‌ی  $W_A$ )

۲. مجموعه‌ی گره‌های رخدادی (مجموعه‌ی  $W_E$ )

۳. مجموعه‌ی گره‌های دروازه‌ای (مجموعه‌ی  $W_G$ )

پیش از اثبات وجود نگاشت، دو فرض اساسی را بر اساس [۲] در نظر می‌گیریم:

۱. هر فرآیند توصیف شده با استفاده از شبکه‌های پتری جریان کار، در مفهوم عام مدل‌های فرآیندی (تعریف ۲-۸-۱) صدق می‌کند.

۲. شبکه‌های پتری جریان کار، زبانی برای توصیف فرآیندها هستند و می‌توانند مدل‌های فرآیندی را توصیف کنند.

هر شبکه‌ی پتری جریان کاری مانند  $PN = (P, T, F)$  در  $WF$  بیان‌گر یک مدل فرآیندی مثل  $p = (N_p, E_p)$  در مجموعه‌ی مدل‌های فرآیندی در  $P$  است. بنابراین، نگاشتی مانند  $g$  وجود دارد که یک مدل فرآیندی مانند  $P \in p$  را به یک مدل فرآیندی مانند  $PN \in B$  می‌نگارد. اگر چنین نباشد، مدل  $PN$  یک مدل فرآیندی نخواهد بود و این با مفروضات قضیه در تنافض خواهد بود. برای آن که نشان دهیم که چنین نگاشت دوسویه‌ای وجود دارد، نگاشت  $g$  را به سه نگاشت مجزا افزای می‌کنیم. هر کدام از این سه نگاشت، ارتباطی را بین مدل‌های فعالیتی، مدل‌های دروازه‌ای و مدل‌های رخدادی در بین مدل‌های فرآیندی و شبکه‌های پتری جریان کار، برقرار می‌کند.

فرض کنید مدل فرآیندی  $p = (N_p, E_p) \in P$  بیان صوری و کلی مدل فرآیندی  $pn = (P_p, T_p, F_p) \in WF$  باشد. برای نگاشت فرآیندهای مدل‌سازی شده با شبکه‌های پتری جریان کار به مدل‌های فرآیندی، گره‌های یک شبکه‌ی پتری جریان کار را به سه مجموعه افزای می‌کنیم. این مجموعه‌های مجزا، دربرگیرنده‌ی مدل‌های فعالیتی، رخدادی و دروازه‌ای در شبکه‌های پتری جریان کار هستند. بنابراین خواهیم داشت:

$$T_p \cup P_p = W_A \cup W_E \cup W_G \quad (۱۸-۴)$$

به گونه‌ای که  $W_A$  حاوی مدل‌های فعالیتی در شبکه‌های پتری جریان کار،  $W_E$  حاوی مدل‌های رخدادی شبکه‌های پتری جریان کار و  $W_G$  حاوی مدل‌های دروازه‌ای در شبکه‌های پتری جریان کار خواهد بود. از سوی دیگر، طبق تعریف ساختارهای کنترلی (تعریف ۲-۸-۲) مجموعه‌ی گره‌های مدل‌های فرآیندی را می‌توان به همین صورت افزار نمود؛ بنابراین، مجموعه‌ی گره‌های مدل فرآیندی معادل با شبکه‌ی پتری جریان کار  $pn$  به صورت زیر خواهد بود:

$$N_p = N_A \cup N_E \cup N_G \quad (۱۹-۴)$$

به گونه‌ای که  $N_A$  حاوی مدل‌های فعالیتی،  $N_E$  حاوی مدل‌های رخدادی و  $N_G$  حاوی مدل‌های دروازه‌ای در مدل فرآیندی معادل با شبکه‌های پتری جریان کار  $pn$  خواهد بود. حال، مجموعه‌های  $W_A$ ،  $W_E$  و  $W_G$  را به صورت زیر تعریف می‌کنیم:

$$W_A = T_A \quad (۲۰-۴)$$

$$W_E = T_E \cup \{i, o\} \quad (۲۱-۴)$$

$$W_G = T_G \cup [P - \{i, o\}] \quad (۲۲-۴)$$

نگاشت بین مدل‌های فعالیتی و فعالیت‌ها (طبق تعریف ۲-۸-۱) در شبکه‌ی پتری جریان کار  $\in pn$ ، برای هر انتقال فعالیتی مثل  $t \in W_A \subset T_p$  یک مدل فعالیتی مثل  $a' \in N_A \subset N_p$  در مدل فرآیندی  $P \in p$  وجود خواهد داشت، به طوری که هردو مدل فعالیتی، یک فعالیت را اجرا خواهند کرد. بنابراین، نگاشتی مثل  $g_A$  وجود خواهد داشت، به گونه‌ای که یک انتقال را از مجموعه‌ی انتقال‌های  $W_A$  به یک مدل فعالیتی در مجموعه‌ی مدل‌های فعالیتی  $N_A$  بنگارد؛

$$g_A : W_A \longrightarrow N_A \quad (۲۳-۴)$$

از سوی دیگر، هر مدل فعالیتی در مدل فرآیندی  $pn$ ، در شبکه‌ی پتری جریان کار، در قالب یک

انتقال فعالیتی، قابل بازنمایی است؛ بنابراین، نگاشتی از مدل‌های فعالیتی یک مدل فرآیندی، به مجموعه‌ی انتقال‌های فعالیتی یک شبکه‌ی پتری جریان کار وجود خواهد داشت، به گونه‌ای که هر دو، یک فعالیت را بازنمایی کنند.

$$g_A^{-1} : N_A \longrightarrow W_A \quad (24-4)$$

بر اساس دو رابطه‌ی ۲۳-۴ و ۲۴-۴ نگاشت  $g_A$  یک نگاشت دوسویه است و بر اساس آن می‌توان نتیجه گرفت که هر انتقال، دارای یک مدل فعالیتی در مدل فرآیندی معادلش است و هر مدل فعالیتی موجود در یک مدل فرآیندی، می‌تواند با استفاده از یک انتقال، در یک شبکه‌ی پتری جریان کار، به کار رود.

**نگاشت بین مدل‌های رخدادی و رخدادها** فرض کنید مدل فرآیندی  $P = (N_p, E_p) \in P$  بیان صوری و کلی مدل فرآیندی  $pn = (P_p, T_p, F_p) \in WF$  باشد (طبق تعریف ۲-۸-۱) در شبکه‌ی پتری جریان کار  $pn \in WF$ ، برای هر رخدادی مثل  $e \in W_e$  یک مدل رخدادی مثل  $e' \in N_E \subset N_p$  در مدل فرآیندی  $p \in P$  وجود خواهد داشت، به طوری که هر دو مدل رخدادی، یک رخداد را اجرا خواهند کرد. بنابراین، نگاشتی مثل  $g_E$  وجود خواهد داشت، به گونه‌ای که یک رخداد را از مجموعه‌ی رخدادهای  $pn$  (یعنی مجموعه‌ی  $W_E$ ) به یک مدل رخدادی در مجموعه‌ی مدل‌های رخدادی  $p$  (یعنی مجموعه‌ی  $N_E$ ) بنگارد؛

$$g_E : W_E \longrightarrow N_E \quad (25-4)$$

از سوی دیگر، بر طبق تعریف ۴-۲-۵ هر مدل رخدادی در یک شبکه‌ی پتری جریان کار، به یک مدل رخدادی در مدل فرآیندی معادلشان تبدیل می‌شوند.

$$g_E^{-1} : N_E \longrightarrow W_E \quad (26-4)$$

بر اساس دو رابطه‌ی ۲۵-۴ و ۲۶-۴ نگاشت  $g_E$  یک نگاشت دوسویه است و بر اساس آن می‌توان نتیجه گرفت که به ازای هر گرهی رخداد در یک شبکه‌ی پتری جریان کار، یک مدل رخدادی در مدل فرآیندی معادلش وجود دارد و به ازای هر مدل رخدادی موجود در یک مدل

فرآیندی، می‌تواند با استفاده از یک انتقال رخدادی (مگر برای رخدادهای آغازین و پایانی که باید با یک مکان در ابتدای فرآیند و یک مکان در انتهای فرآیند مدل شوند)، در یک شبکه‌ی پتری جریان کار، به کار رود.

**نگاشت بین مدل‌های دروازه‌ای و دروازه‌ها** فرض کنید مدل فرآیندی  $p = (N_p, E_p) \in P$  بیان

صوری و کلی مدل فرآیندی  $pn = (P_p, T_p, F_p) \in WF$  باشد (طبق تعریف ۲-۸-۲)

در تعریف ۲-۸-۲ مشاهده می‌شود که به هر مدل دروازه‌ای، یک نوع جریان کنترلی یکتا تخصیص داده شده‌بود و این نوع جریان کنترلی، به وسیله‌یتابع  $type : N_G \rightarrow C$  مشخص می‌شود؛ بنابراین تابع  $type$  به هر مدل دروازه‌ای، یک ساختار را از مجموعه‌ی ساختارها، یعنی مجموعه‌ی  $C$  تخصیص می‌دهد.

در شبکه‌های پتری جریان کار، مدل‌های دروازه‌ای (اجزایی که باعث همگرایی و یا واگرایی شاخه‌های فرآیند می‌شوند) در جدول ۴-۲ اجزایی از شبکه‌های پتری جریان کار که باعث ایجاد انشعاب و اتصال شاخه‌های فرآیند می‌شوند، آورده شده‌است. هر کدام از این اجزاء، از یک الگوی جریان کنترلی معینی تبعیت می‌کند؛ بنابراین، می‌توان تابع  $type_{WF}$  را به صورت زیر تعریف نمود:

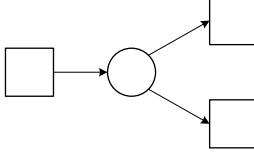
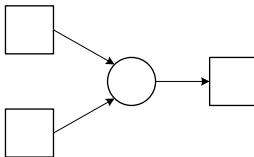
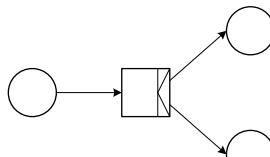
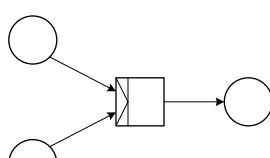
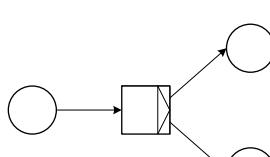
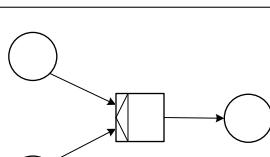
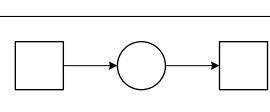
$$type_{WF} : W_G \rightarrow C \quad (27-4)$$

این تابع، نگاشت را بر اساس نگاشتهای تعریف شده در جدول ۲-۴ انجام می‌دهد.

در یک شبکه‌ی پتری جریان کار مثل  $pn \in WF$ ، برای هر دروازه‌ای مثل  $g \in W_G$  یک مدل دروازه‌ای مثل  $g' \in N_G \subset N_p$  در مدل فرآیندی  $p \in P$  وجود خواهد داشت، به طوری که هر دو دروازه، یک ساختار جریان کنترلی را اجرا می‌کنند؛ به عبارت دیگر :

$$\forall g \in W_G \exists g' \in N_G \quad s.t \quad type_{WF}(g) = type(g') \quad (28-4)$$

بنابراین، نگاشتی مثل  $g_G$  وجود خواهد داشت، به گونه‌ای که یک دروازه را از مجموعه‌ی دروازه‌های  $(W_G)$  به یک مدل دروازه‌ای در مجموعه‌ی مدل‌های دروازه‌ای  $p$  (یعنی  $pn$ )

نام الگوی جریان کنترلی کنترلی	نام جزء کنترلی	شماتی گرافیکی دروازه
الگوی جریان کنترلی انشعاب انحصاری	مکان انشعاب انحصاری	
الگوی جریان کنترلی اتصال انحصاری	مکان اتصال انحصاری	
الگوی جریان کنترلی انشعاب عطفی	انتقال انشعاب عطفی	
الگوی جریان کنترلی اتصال عطفی	انتقال اتصال عطفی	
الگوی جریان کنترلی انشعاب فصلی	انتقال انشعاب فصلی	
الگوی جریان کنترلی اتصال فصلی	دروازه‌ی اتصال فصلی	
الگوی جریان کنترلی دنباله‌ای	مکان عادی	

جدول ۴-۲: دسته‌بندی الگوهای دروازه‌ای بر مبنای الگوهای جریان کنترلی در شبکه‌های پتری جریان کاری

مجموعه‌ی  $N_G$ ) بنگارد؛

$$g_G : W_G \longrightarrow N_G \quad (۲۹-۴)$$

از سوی دیگر، هر مدل دروازه‌ای در مدل فرآیندی  $p$ ، در شبکه‌ی پتری جریان کاری  $pn$ ، در قالب یک دروازه قابل بازنمایی است، مشروط بر آن که هردو دارای یک نوع الگوی جریان کنترلی باشند؛ بنابراین خواهیم داشت:

$$\forall g' \in N_G \exists g \in W_G \quad s.t \quad type_{WF}(g) = type(g') \quad (۳۰-۴)$$

بنابراین، معکوس رابطه‌ی قبل به شکل زیر قابل تعریف خواهد بود:

$$f_E^{-1} : N_G \longrightarrow W_G \quad (۳۱-۴)$$

بر اساس دو رابطه‌ی ۲۹-۴ و ۳۱-۴ نگاشت  $g_G$  یک نگاشت دوسویه است و بر اساس آن می‌توان نتیجه گرفت که به ازای هر دروازه در یک شبکه‌ی پتری جریان کار (طبق جدول ۲-۴)، یک مدل دروازه‌ای در مدل فرآیندی معادلش وجود دارد که از الگوی جریان کنترلی مشابه، پیروی می‌کند.

با توجه به این‌که یکایک گره‌های یک شبکه‌ی پتری جریان کار، مثل  $pn = (P_p, T_p, F_p) \in WF$  را به صورت دو سویه به گره‌های مدل‌های فرآیندی  $p = (N_p, E_p) \in P$  نگاشت دادیم، می‌توانیم نگاشتی مانند  $g$  را بین گره‌های یک شبکه‌ی پتری جریان کار را با گره‌های یک مدل فرآیندی، به صورت رابطه‌ای دوسویی و به صورت زیر تعریف کنیم :

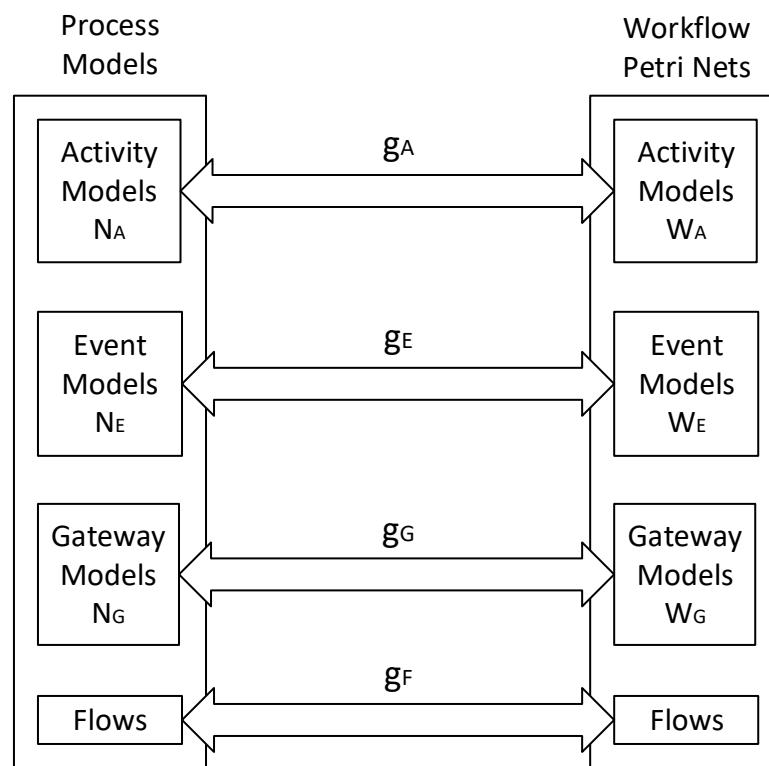
$$g : N_A \longrightarrow P_A \cup T_A \quad s.t \quad g(n) = \begin{cases} g_A(n) & n \in N_A \\ g_E(n) & n \in N_E \\ g_G(n) & n \in N_G \end{cases} \quad (۳۲-۴)$$

همچنین، برای نگاشت بین کمان‌های یک شبکه‌ی پتری جریان کار و کمان‌های مدل‌های فرآیندی معادل آن‌ها، نگاشت  $g_F : E_p \longrightarrow F_p$  را می‌توان به صورت زیر تعریف نمود:

$$g_F = \{(g(x), g(y)) | (x, y) \in E_p \wedge x, y \in P_A \cup T_A\} \quad (33-4)$$

$$g_F^{-1} = \{(g(x)^{-1}, g(y)^{-1}) | (x, y) \in P_A \cup T_A \wedge x, y \in N_b\} \quad (34-4)$$

بنابراین، می‌توان با نگاشت دوسویه‌ی  $g$  گره‌های یک مدل فرآیندی را به گره‌های معادلش و درشبکه‌ی پتری جریان کار معادلش تبدیل نمود (و بالعکس). در خصوص یال‌ها نیز می‌توان با استفاده از تابع دوسویه‌ی  $g_F$  این نگاشت یک به یک را انجام داد. بنابراین، هر شبکه‌ی پتری جریان کار، به یک مدل فرآیندی قابل تبدیل است و بالعکس. نگاشت ارائه شده در این اثبات، به طور خلاصه در تصویر ۹-۴ آمده است.  $\square$



شکل ۹-۴: نگاشت یک به یک بین شبکه‌های پتری جریان کار و مدل‌های فرآیندی

### BPMN2.0 نگاشت نهايی دوسویه بين شبکه‌های پتری جريان کار و مدل‌های فرآيندي

در بخش‌های قبل، نگاشتی را بين شبکه‌های پتری جريان کار و مدل‌های فرآيندي و نيز بين مدل‌های فرآيندي BPMN 2.0 و مدل‌های فرآيندي برقرار کردیم و توانستیم بين هر کدام از اين مدل‌های فرآيندي و مفهوم عمومی مدل‌های فرآيندي (بر اساس تعریف ۱-۸-۲) نگاشتی را برقرار کنیم. در ایجاد نگاشت بين مدل‌های BPMN 2.0 و شبکه‌های پتری جريان کار به مفهوم عام مدل‌های فرآيندي، آن‌چه که تضمین می‌کند که مدل‌های نگاشت یافته، يك فرآيند واحد را اجرا بکنند، يکسان بودن الگوهای جريان کنترلی در بين اجزای نگاشت یافته است. در اين بخش، نگاشتی دوسویه بين گره‌های يك شبکه‌ی پتری جريان کار و يك مدل فرآيندي BPMN 2.0 تعریف می‌شود. فرض کنید نگاشت  $f$ ، نگاشت تعریف شده در ۱۲-۴ باشد و نگاشت  $g$ ، نگاشت تعریف شده در ۳۲-۴ باشد. با توجه به اين که هر دو نگاشت، نگاشت‌هایي دوسویه هستند، اگر  $x$  يك گره در يك مدل فرآيندي BPMN 2.0 باشد، معادلش در مدل فرآيندي BPMN 2.0 به صورت زير بدست می‌آيد:

$$y = gof^{-1}(x) \quad (35-4)$$

و در صورتی که  $y$  يك گره در يك شبکه‌ی پتری جريان کاري باشد، گرهی معادل آن در مدل فرآيندي BPMN 2.0 به صورت زير بدست می‌آيد:

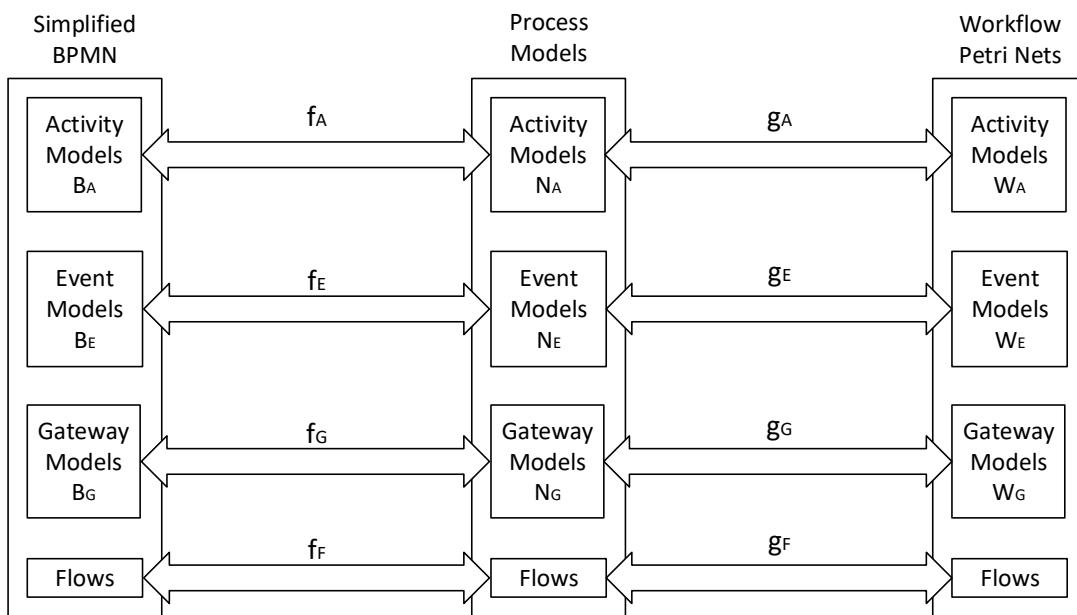
$$x = fog^{-1}(y) \quad (36-4)$$

بنابراین، با استفاده از هرکدام از اين دو رابطه، می‌توان گره‌های مدل‌های فرآيندي BPMN 2.0 و گره‌های شبکه‌های پتری جريان کار را به يك دیگر تبدیل نمود. استفاده از اين نگاشت‌ها، به اختصار و بر مبنای نگاشت‌های ارائه شده در بخش‌های قبل، در تصویر ۱۰-۴ قابل مشاهده است.

در تبدیل يك مدل فرآيندي BPMN 2.0 به يك شبکه‌ی پتری جريان کار (و بالعکس) دو نوع قاعده باید رعایت شود:

۱. قواعدی که در ضمن نگاشت، ضمن حفظ جريان کنترلی اجزا، صرفاً نمادگذاری را تغییر می‌دهند.

۲. قواعدی که بر روی مدل مبداء لحاظ می‌شوند تا مدل مقصد، به درستی و بر طبق قواعدهش ترسیم شود.



شکل ۴-۱۰: نگاشت ترکیبی اجزای مدل فرآیندی 2.0 BPMN به شبکه‌های پتری جریان کار و بالعکس

آنچه که در نگاشت ارائه شده در بخش‌های قبل ارائه شد، اولین نوع از قواعد را پوشش می‌دهد و در این بخش، نگاشتهای دوسویه را در بین اجزا برقرار می‌کنیم و سپس، قواعدی را برای حفظ قواعد ترسیمی در هر کدام از انواع مدل‌های فرآیندی، ارائه می‌کنیم. همان‌طور که در بخش‌های قبل مشاهده شد، گره‌های هر مدل فرآیندی را می‌توان به سه دسته تقسیم‌بندی نمود:

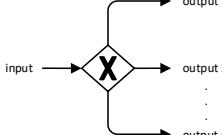
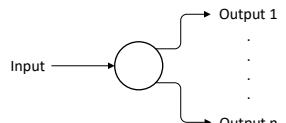
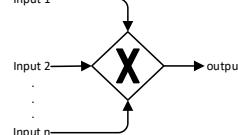
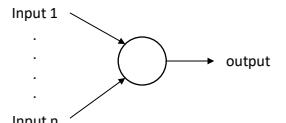
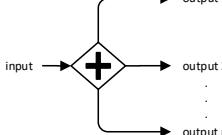
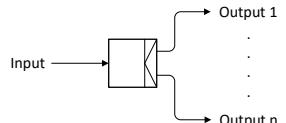
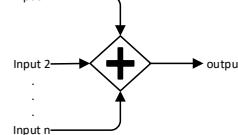
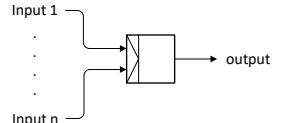
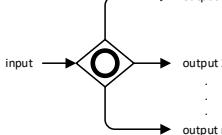
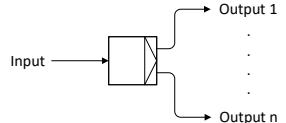
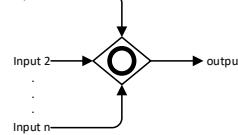
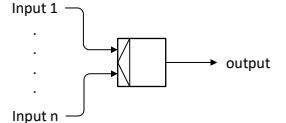
### ۱. مدل‌های فعالیتی

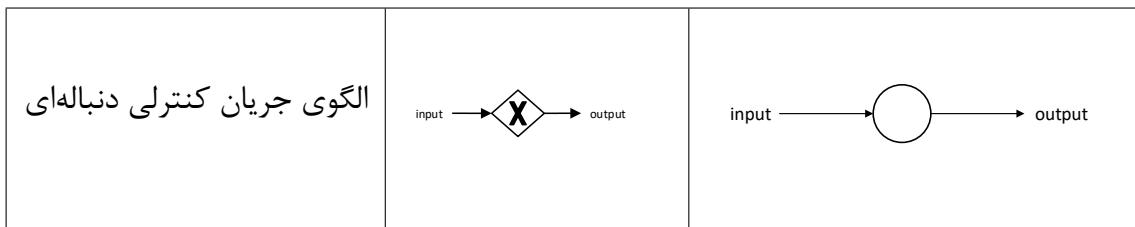
### ۲. مدل‌های رخدادی

### ۳. مدل‌های دروازه‌ای

در جدول ۳-۴ مدل‌های دروازه‌ای معادل با یک دیگر، بر اساس نگاشتهای مطرح شده در بخش‌های قبل و روابط ۴-۳۶ و ۴-۳۵ ترسیم شده‌اند. این اجزا در هر دو زبان مدل‌سازی، می‌توانند به جای یک دیگر به کار روند؛ چرا که هر کدام از آن‌ها، بازنمایی‌کننده‌ی یک الگوی جریان کنترلی هستند و همان‌طور که در [۲] گفته شد، این الگوهای جریان کنترلی مستقل از زبان مدل‌سازی هستند.

جدول ۴-۳: نگاشت مدل‌های دروازه‌ای

الگوی جریان کنترلی	BPMN 2.0	شبکه‌ی پتری جریان کار
الگوی جریانی انشعاب انحصاری		
الگوی جریانی اتصال انحصاری		
الگوی جریانی انشعاب عطفی		
الگوی جریانی اتصال عطفی		
الگوی جریانی انشعاب فصلی		
الگوی جریانی اتصال فصلی		



جدول ۴-۴: نگاشت مدل‌های رخدادی

مدل رخدادی	BPMN 2.0	شبکه‌ی پتری جریان کار
مدل‌های رخدادی میانی		
مدل رخدادی آغازین (شروع)		
مدل رخدادی پایانی		

همچنین جدول مدل‌های رخدادی در جدول ۴-۴ آمده است. الگوهای رخدادی میانی، از جریان کنترلی دنباله‌ای تبعیت می‌کنند؛ چرا که یک ورودی و یک خروجی دارند و برخلاف فعالیتها که دارای سه رخداد مجزا هستند، به خودی خود، یک رخداد مجزا هستند و در گراف زنجیره‌ی رویدادی، با یک راس نشان داده می‌شوند.

تمامی مدل‌های فعالیتی، بر طبق بخش از الگوی جریان کنترلی دنباله‌ای تبعیت می‌کنند و در جدول ۴-۵ نشان داده شده‌اند. خاطرنشان می‌گردد که بر طبق تحقیق داکمن در [۴] به منظور تصدیق صحت ساختاری، می‌توان زیرفرآیندها را نیز به صورت مدل‌های فعالیتی در نظر گرفت.

جدول ۴-۵: نگاشت مدل‌های رخدادی

الگوی جریان کنترلی	BPMN 2.0	شبکه‌ی پتری جریان کار
الگوی جریان کنترلی دنباله‌ای		

## حفظ قواعد ترسیم در تبدیل مدل‌های فرآیندی BPMN2.0 و شبکه‌های پتری جریان کار به یکدیگر

همان‌طور که در بخش‌های قبل گفته شد، برای تبدیل مدل‌های فرآیندی BPMN 2.0 به یک شبکه‌ی پتری جریان کار، علاوه بر نگاشت تک به تک گره‌ها، لازم است قواعد ترسیمی در مدل مقصد رعایت شود. مهم‌ترین قاعده‌ای که در مدل مقصد (یعنی شبکه‌های پتری جریان کاری) وجود دارد، عدم اتصال مستقیم دو مکان و یا دو انتقال می‌باشد. برای آن‌که بتوان ضمن استفاده از نگاشت ارائه شده، قواعد ترسیمی مدل مقصد رعایت شود، چیش گره‌های مدل فرآیندی مبداء (یعنی مدل فرآیندی BPMN 2.0) به گونه‌ای تنظیم شود که بتوان بدون نیاز به حذف اجزای آن، مدل جاری را به مدل مقصد تبدیل نمود (بر طبق نگاشت‌های ارائه شده در جداول ۳-۴، ۴-۴ و ۵-۴). برای رعایت قواعد نگاشت در مدل مقصد، لازم است موارد زیر بر روی مدل مبداء اجرا شود:

۱. در خصوص دروازه‌های انشعبان انصاری، پیش از شاخه‌ی ورودی دروازه و پس از شاخه‌های خروجی دروازه، فعالیت‌هایی قرار داده شوند. این فعالیت‌ها، فعالیت‌هایی بی اثر هستند و در اصطلاح به آن‌ها، فعالیت‌های ساکت<sup>۳</sup> گفته می‌شود. دلیل محاصره کردن دروازه با این فعالیت‌های ساکت را می‌توان با نگاشت ارائه شده توسط داکمن [۴] (تصویر ۳-۵) توجیه نمود؛ چرا که دروازه‌ی انشعبان انصاری، به صورت یک مکان با درجه‌ی خروجی بزرگ‌تر از ۱ بازنمایی می‌شود و در صورتی که با فعالیت‌های ساکت محاصره نشود، ممکن است در مدل مقصد با مکان دیگری ارتباط مستقیم برقرار کند و قواعد شبکه‌های پتری را نقض نماید.

۲. مشابه با مورد قبل، برای شاخه‌های اتصال انصاری نیز باید با فعالیت‌های ساکت محاصره شود.

۳. ورودی رخدادهای پایانی نیز، باید به یک فعالیت ساکت متصل گردد.

<sup>3</sup>Silent Activity

۴. خروجی یک رخداد شروع نیز باید به یک دروازه‌ی ساکت، متصل گردد.

وجود و یا عدم وجود فعالیت‌های ساکت، تغییری را در جریان کنترلی فرآیند کاری ایجاد نمی‌کند و صرفاً تضمین می‌کند که هیچ دو مکانی به یک دیگر متصل نمی‌شوند. در مدل فرآیندی 2.0 BPMN اتصال بین هر دو جزء (هر دو جزئی از مدل‌های فعالیتی، رخدادی و دروازه‌ای) با یک کمان صورت می‌گیرد. بر طبق گفته‌های ماتیاس در کتاب مدیریت فرآیندهای کسب و کار [۲] برای هر کمان مستقیم بین اجزای یک مدل فرآیندی 2.0 BPMN می‌توان یک دروازه قائل شد که درجه‌ی ورودی و درجه‌ی خروجی آن برابر با ۱ است. از این دروازه به عنوان دروازه‌ی دنباله‌ای یاد می‌شود و در نهایت می‌توان آن‌ها را از نمایش نهایی حذف نمود. برای تبدیل یک مدل فرآیندی 2.0 BPMN به یک شبکه‌ی پتری جریان کاری، لازم است این دروازه‌ها را مجدداً در نظر گرفت؛ چرا که این دروازه‌ها بر طبق جدول ۳-۴ به یک مکان در شبکه‌ی پتری مقصد تبدیل می‌شوند. بنابراین لازم خواهد بود که بین هر دو جزء جریانی مدل فرآیندی 2.0 BPMN یک دروازه‌ی دنباله‌ای (که می‌تواند با یک دروازه‌ی انحصاری و از درجه‌ی ۱ بازنمایی شود) قرار داده شود تا با ایجاد مکان‌ها در بین انتقال‌ها، قواعد ترسیمی در مدل مقصد رعایت شود. در این خصوص، سه مورد استثنای وجود دارد؛ در بخش‌های زیر، نباید هیچ دروازه‌ی دنباله‌ای قرار بگیرد:

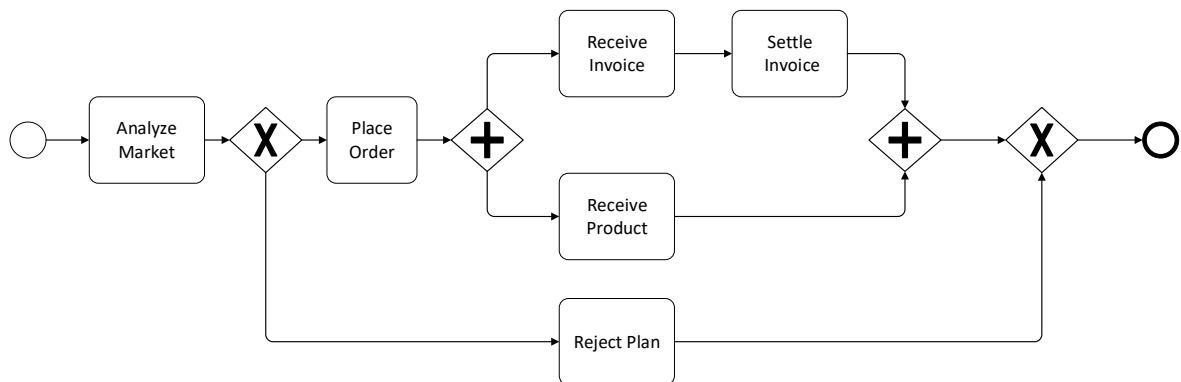
۱. در مبادی ورودی و خروجی دروازه‌های اتصال و انشعباب‌های انحصاری.

۲. پس از رخداد شروع.

۳. پیش از رخداد پایانی.

پس از اعمال این پیش‌پردازش‌ها بر روی مدل فرآیندی 2.0 BPMN می‌توان مدل مبداء را بر اساس نگاشته‌های ارائه شده در جداول ۳-۴، ۴-۴ و ۵-۴ به مدل مقصد، تبدیل نمود.

نمونه ۴-۱۱. روش گفته شده در این بخش را در قالب یک مثال، دنبال می‌کنیم. برای این منظور، فرآیند ترسیم شده در شکل ۱۱-۴ را در نظر بگیرید. در ابتدا، فعالیت‌های ساکت را به فرآیند اضافه



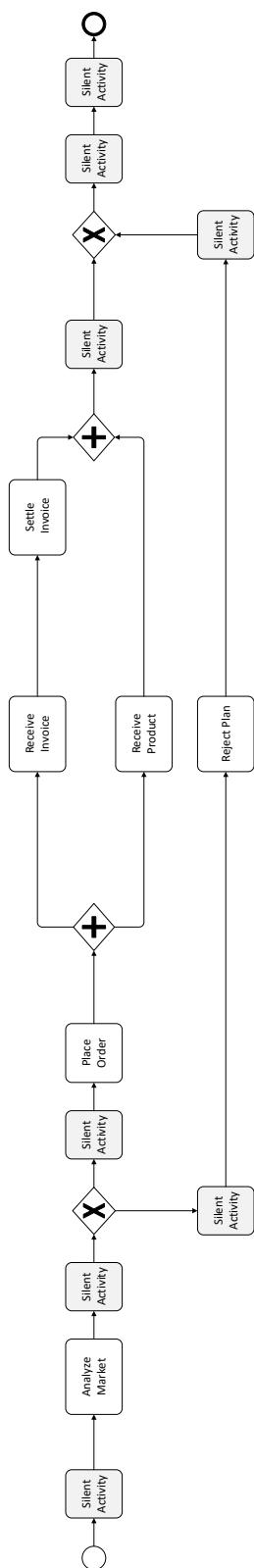
شکل ۱۱-۴: یک فرآیند مدل‌سازی شده با زبان BPMN 2.0 به منظور تبدیل به شبکه‌ی پتری جریان کار معادل [۳]

می‌کنیم و فرآیند شکل ۱۲-۴ بدست می‌آید. در گام بعد، دروازه‌های دنباله‌ای را با رعایت استثناهای گفته شده، در مدل فرآیندی قرار می‌دهیم. در نهایت، با اجرای جانشینی نظیر به نظیر اجزاء، به شبکه‌ی پتری جریان کار معادل آن، در شکل می‌رسیم.

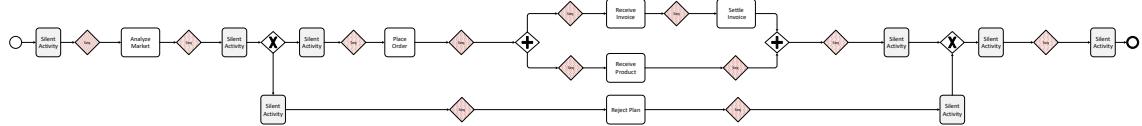
در ادامه، راهکاری به منظور تبدیل یک مدل فرآیندی ترسیم شده با شبکه‌ی پتری جریان کاری به مدل فرآیندی BPMN 2.0 معادل، ارائه می‌شود. برخلاف بخش قبل، با توجه به این‌که محدودیت‌های ترسیمی بر روی مدل‌های فرآیندی BPMN 2.0 باشد شبکه‌های پتری جریان کار برقرار نیست، نگاشت مدل‌های فرآیندی ترسیم شده با شبکه‌های پتری جریان کار به مدل‌های فرآیندی BPMN 2.0 دارای مراحل کمتری است.

پیش از تبدیل شبکه‌ی پتری جریان کار به مدل فرآیندی BPMN 2.0 لازم است پیش‌برداششای زیر بر روی آن انجام شود:

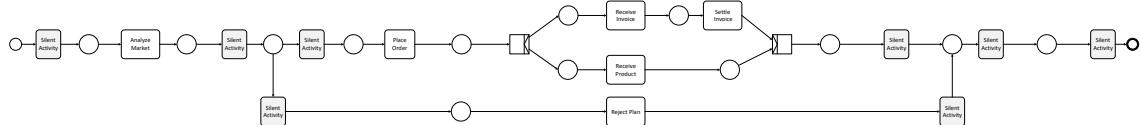
- در صورتی که فعالیتی به یکی از انتقال‌های انشعابی الصاق شده باشد، باید از آن انتقال جدا شده و در قالب یک انتقال مستقل و به واسطه‌ی یک مکان، پیش از انتقال انشعابی قرار گیرد.
- در صورتی که فعالیتی به یکی از انتقال‌های اتصالی الصاق شده باشد، باید از آن انتقال جدا شده و در قالب یک انتقال مستقل و به واسطه‌ی یک مکان، پس از انتقال انشعابی قرار گیرد.



شکل ۴-۱۲: گام اول : افزودن فعالیت‌های ساكت به مدل فرآيندي شکل ۱۱-۴



شکل ۱۳-۴: گام دوم: افزودن دروازه‌های دنباله‌ای به مدل فرآیندی شکل ۱۲-۴

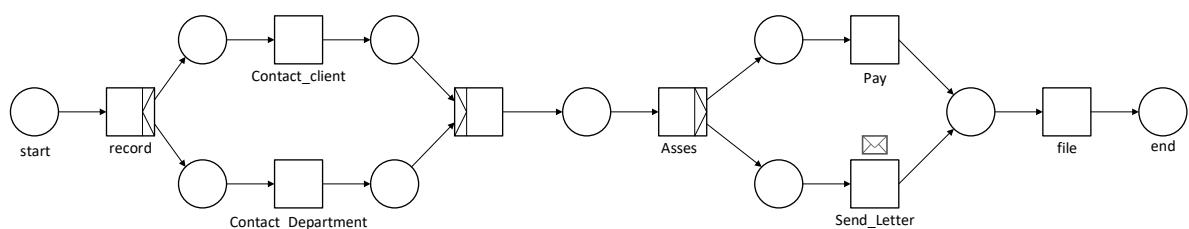


شکل ۱۴-۴: گام سوم: تبدیل مدل مبدا به شبکه‌ی پتری جریان کاری معادل ۱۳-۴

۳. در خصوص انتقال‌هایی که باید منتظر یک تحریک<sup>۴</sup> از یک رخداد باشند، بسته به این که فعالیت تعریف شده در آن‌ها باید پیش از وقوع رخداد و یا بعد از آن باید انجام شود، تبدیل به دو انتقال مجزا می‌شوند و به واسطه‌ی یک انتقال، به یکدیگر متصل می‌شوند.

پس از اجرای پیش‌پردازش‌های قبل، می‌توان با استفاده از جداول ۳-۴، ۴-۴ و ۵-۴ مدل مبداء را به مدل مقصد، تبدیل نمود.

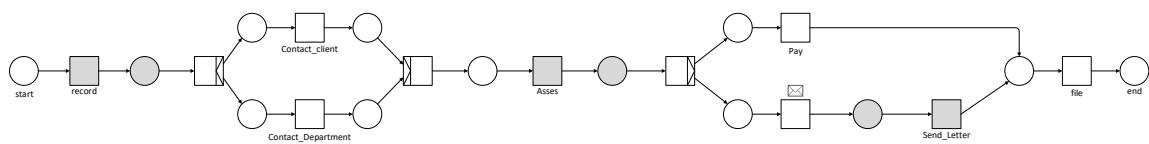
نمونه ۱۲-۴. به عنوان نمونه، یک فرآیند کاری را از یک شبکه‌ی پتری جریان کار، به یک مدل فرآیندی BPMN 2.0 تبدیل می‌کنیم. در ابتدا، پیش‌پردازش‌های گفته شده بر روی انتقال‌های رخدادی،



شکل ۱۵-۴: یک فرآیند مدل‌سازی شده با استفاده از شبکه‌های پتری جریان کار

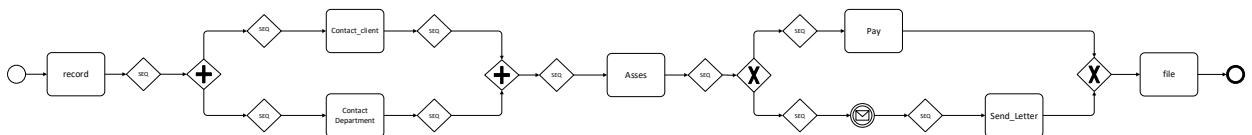
انشعابی و اتصالی انجام داده می‌شود و شکل ۱۶-۴ به وجود می‌آید. انتقال‌های تولید شده در اثر جداسازی، با رنگ خاکستری نشان داده شده‌اند. سپس، اجزای معادل را در مدل مبدا جایگزین می‌کنیم. با این کار، مدل BPMN 2.0 معادل، در تصویر ۱۷-۴ بدست آمده‌است؛ در نهایت، بر طبق نظر ماتیاس در کتاب مدیریت فرآیندهای کسب و کار [۲] می‌توان دروازه‌های دنباله‌ای را حذف کرد و جریان‌های

<sup>4</sup>Trigger

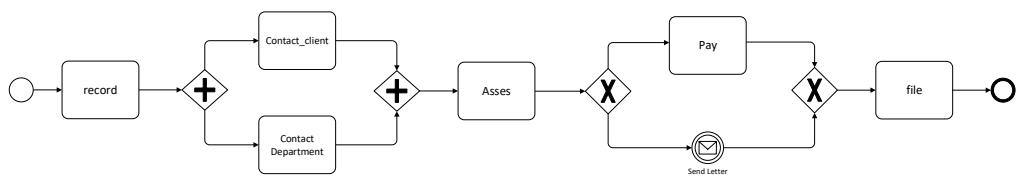


شکل ۱۵-۴: نتیجه‌ی اجرای پیش‌پردازش‌های گفته شده بر روی شکل ۴-۱۵

کنترلی دنباله‌ای را با فلش‌ها نشان داد. با این کار، به مدل فرآیندی BPMN 2.0 معادل در تصویر ۴-۱۸ می‌رسیم.



شکل ۱۶-۴: جانشینی اجزای مدل‌های فرآیندی BPMN 2.0



شکل ۱۸-۴: مدل فرآیندی BPMN 2.0 نهایی

### ۳-۲-۴ تعمیم مفهوم صحت در مدل‌های فرآیندی

در بخش‌های قبل مشاهده شد که مدل‌های فرآیندی، می‌توانند در زبان‌های مختلفی بازنمایی شوند و مدل‌های فرآیندی ارائه شده در زبان‌های مدل‌سازی مختلف، می‌توانند بر مبنای مفهوم عام مدل‌های فرآیندی (تعاریف ۱-۸-۲ و ۲-۸-۲) به یک دیگر تبدیل شوند. در بخش قبل، با تعریف نگاشته‌ای ثابت کردیم که مدل‌های فرآیندی BPMN 2.0 و مدل‌های فرآیندی ترسیم شده با استفاده از شبکه‌های پتری جریان کار، به یک دیگر قابل تبدیل هستند.

در شبکه‌های پتری جریان کار، صحت (Soundness) بر مبنای دو ویژگی تعریف شده است:

۱. زنده بودن (Liveness)

۲. کراندار بودن (Boundness)

در یک شبکه‌ی پتری جریان کار، ویژگی اول تضمین می‌کند که تمامی فعالیت‌های موجود در یک فرآیند، اجرا خواهد شد و در یک فرآیند، فعالیتی وجود ندارد که اجرا نشود. این امر، نبود بن‌بست و وظایف مرده را، در یک فرآیند تضمین می‌کند؛ هم‌چنین، ویژگی دوم تضمین می‌کند که در یک شبکه‌ی پتری جریان کار، تعداد توکن‌ها در یک مکان، تا اندازه‌ی مشخصی رشد می‌کند و تعداد آن‌ها، تا بی‌نهایت رشد نخواهد کرد. این امر تضمین می‌کند که هر فعالیت، با تعداد دفعات مشخص و معینی اجرا می‌شود (انتقال واقع در قبل از مکان، به تعداد دفعات معینی شلیک می‌شود و در مکان بعدی خود، به تعداد محدودی توکن می‌سازد).

برای تعمیم مفهوم صحت به مدل‌های فرآیندی BPMN 2.0 لازم است در ابتدا دو مفهوم زنده بودن و کراندار بودن به این مدل‌های فرآیندی تعمیم داده شوند.

**تعريف ۴-۲-۲.** یک مدل فرآیندی BPMN 2.0 زنده (Live) است، اگر و تنها اگر فاقد هرگونه بن‌بست و یا وظیفه‌ی مرده باشد و در نتیجه، تمامی فعالیت‌های آن، اجرا شود.

**قضیه ۴-۲-۴.** یک مدل فرآیندی BPMN 2.0 فاقد فعالیت‌های مرده و بن‌بست است، اگر و تنها اگر شبکه‌ی پتری جریان کار معادل با آن، زنده باشد.

برهان. اگر یک مدل فرآیندی BPMN 2.0 فاقد بن‌بست و یا وظایف مرده باشد، تمامی فعالیت‌های آن در مسیری از رخداد شروع تا رخداد پایانی واقع هستند. بنابراین، فعالیتی وجود نخواهد داشت که در فرآیند، اجرا نشود؛ هم‌چنین، در چنین فرآیندی، بن‌بستی وجود نخواهد داشت؛ برای اثبات این قضیه، فرض کنید که یک مدل فرآیندی BPMN 2.0 مثل  $(V, E)$  وجود دارد که دارای بن‌بست (و یا یک وظیفه‌ی مرده) است و شبکه‌ی پتری جریان کار  $(P, T, F)$  شبکه‌ی پتری جریان کار معادل با این مدل فرآیندی باشد که بر طبق روش ارائه شده در بخش ۲-۲-۴ ارائه شده باشد و بر طبق تعریف ۲-۸-۲ داشته باشیم:

$$V = N_E \cup N_A \cup N_G \quad (37-4)$$

به عنوان فرض خلف، فرض کنید که شبکه‌ی پتری جریان کار آن، زنده باشد. در  $b$ ، فعالیتی مثل  $a \in N_A$  وجود دارد که اجرا نمی‌شود؛ در نتیجه، بر طبق نگاشت ۳۶-۴ به ازای فعالیت  $a$ ، انتقالی مانند  $t \in T$  وجود خواهد داشت. هر بار اجرای فعالیت  $a$  در مدل فرآیندی  $b$ ، معادل است با یک مرتبه شلیک انتقال  $t$  در شبکه‌ی پتری جریان کار  $pn$ . با توجه به این‌که شبکه‌ی پتری جریان کار  $pn$  زنده است،

تمامی انتقال‌های موجود در آن نیز فعال شده و شبیک می‌شوند؛ اما با توجه به وقوع بنبست در  $a$  و در نتیجه شبیک نشدن  $t$ ، شبکه‌ی پتری جریان کار  $pn$  زنده نخواهد بود و این یک تناقض است.

از سوی دیگر، فرض کنید که شبکه‌ی  $(P, T, F)$ ، شبکه‌ی پتری جریان کار معادل با مدل فرآیندی BPMN 2.0 مثل  $b$  باشد و فرض کنید این شبکه، زنده باشد؛ در این صورت، در آن، انتقالی وجود نخواهد داشت که فعال نشده و شبیک نشود و در نتیجه، در مدل فرآیندی  $b$  نیز فعالیتی وجود نخواهد داشت که اجرا نشود؛ در نتیجه، تمامی فعالیت‌های  $b$  اجرا خواهد شد و  $b$  نیز زنده خواهد بود.  $\square$

**نتیجه ۲-۱۵.** یک مدل فرآیندی BPMN 2.0 زنده است؛ اگر و تنها اگر شبکه‌ی پتری جریان کار معادل با آن، زنده باشد.

**تعریف ۲-۱۶.** یک مدل فرآیندی BPMN 2.0 را کران‌دار می‌گویند، اگر و تنها اگر قادر به نبودن حلقوی باشد.

**قضیه ۲-۱۷.** یک مدل فرآیندی BPMN 2.0 قادر به نبودن حلقوی است اگر و تنها اگر شبکه‌ی پتری جریان کار معادل با آن، کران‌دار باشد.

**برهان.** فرض کنید مدل فرآیندی  $(V, E)$  یک مدل فرآیندی BPMN 2.0 باشد و شبکه‌ی پتری جریان کار معادل با  $b$  باشد که بر اساس نگاشت ارائه شده در بخش ۲-۴ بدست آمده باشد.

به عنوان فرض خلف، فرض کنید که  $b$  دارای بنبست حلقوی بوده و  $pn$ ، کران‌دار است. با توجه به این که  $b$  دارای بنبست حلقوی است، یک فعالیت مثل  $V \in b$  در آن وجود خواهد داشت، به گونه‌ای که با تعداد دفعات نامتناهی، اجرا شود. بر اساس نگاشت ۳۶-۴ به ازای فعالیت  $V \in b$  یک انتقال مثل  $t \in T$  در شبکه‌ی پتری جریان کار  $pn$  وجود خواهد داشت. هر بار اجرای فعالیت  $a$  معادل است با یک بار شبیک انتقال  $t$ ؛ بنابراین با توجه به این که فعالیت  $a$  با تعداد دفعات نامتناهی اجرا می‌شود، انتقال  $t$  نیز به تعداد دفعات نامتناهی شبیک می‌شود؛ در نتیجه، مکانی مانند  $p \in P$  وجود خواهد داشت که بعد از انتقال  $t$  واقع است و با شبیک انتقال  $t$  با تعداد دفعات نامتناهی، حاوی تعدادی نامتناهی توکن خواهد شد؛ در نتیجه، تعداد توکن‌ها به صورتی نامحدود در  $p$  افزایش خواهد یافت و  $pn$  کران‌دار نخواهد بود و این تناقض است.

از سوی دیگر، فرض کنید که شبکه‌ی پتری جریان کاری  $pn$  کران‌دار باشد؛ بنابراین، تعداد توکن‌ها در هیچ‌کدام از مکان‌های آن تا بینهایت رشد نخواهد کرد؛ در نتیجه، در آن انتقالی وجود نخواهد داشت

که با تعداد دفعات نامتناهی، شلیک شود. بنابراین، در مدل فرآیندی BPMN 2.0 معادل با آن، فعالیتی وجود نخواهد داشت که با تعداد دفعات نامتناهی اجرا شود و در نتیجه، فاقد بنبست حلقوی خواهد بود.

□

**نتیجه ۴-۲-۱۸.** یک مدل فرآیندی BPMN 2.0 کران دار است، اگر و تنها اگر شبکه‌ی پتری جریان کار آن، کران دار باشد.

با توجه به آن که مفهوم زنده بودن و کران دار بودن، برای مدل‌های فرآیندی BPMN 2.0 بازتعریف شده است، مفهوم صحت را نیز می‌توان به صورت زیر برای این مدل‌های فرآیندی تعریف نمود:

**تعریف ۴-۲-۱۹.** یک مدل فرآیندی BPMN 2.0 صحیح است، اگر و تنها اگر کران دار و زنده باشد.

**قضیه ۴-۲-۲۰.** یک مدل فرآیندی BPMN 2.0 صحیح است، اگر و تنها اگر شبکه‌ی پتری جریان کار معادل با آن، صحیح باشد.

برهان. اگر یک مدل فرآیندی BPMN 2.0 صحیح باشد، بدین معنا خواهد بود که زنده و کران دار است. بنابراین، بر طبق قضیه‌های ۴-۲-۱۴ و ۱۷-۲-۱۶ مدل فرآیندی شبکه‌ی پتری جریان کار حاصل از آن‌ها نیز زنده و کران دار بوده و در نتیجه، صحیح است. عکس قضیه به صورت مشابه برقرار است.

□

#### ۴-۲-۴ الگوهای صحت در مدل‌های فرآیندی

در ادامه، مدل‌هایی فرآیندی را معرفی خواهیم کرد که مدل‌هایی صحیح<sup>۵</sup> هستند. این مدل‌های فرآیندی صحیح می‌توانند با یک دیگر ترکیب شده و مدل‌هایی صحیح و بزرگ‌تر را به وجود آورند. در صورتی که یک مدل فرآیندی بزرگ، از تلفیق این مدل‌های فرآیندی کوچک‌تر و صحیح به وجود آید، مدل فرآیندی صحیحی خواهد بود؛ اما عکس این ادعا لزوماً برقرار نیست؛ یعنی ممکن است یک مدل فرآیندی صحیح باشد، در حالی که متشكل از این مدل‌های فرآیندی صحیح نباشد؛ با این حال، شناسایی این مدل‌های فرآیندی در یک فرآیند، می‌تواند در تحلیل ایستای آن و صحت‌سنجی آن، کمک شایانی را بکند. این مدل‌های فرآیندی کوچک و صحیح، توسط [۳] معرفی شده‌اند و از آن‌ها، به عنوان بلوک‌های سازه‌ای صحت<sup>۶</sup> یاد شده است. در [۳] این بلوک‌ها برای مدل‌های فرآیندی ایجاد شده با شبکه‌های پتری جریان کار ارائه شده‌اند. در ادامه، هرکدام از این بلوک‌ها را تحت نگاشت ۴-۳۵ به مدل‌های فرآیندی

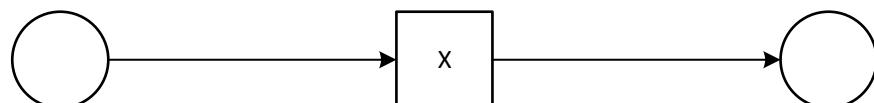
<sup>5</sup>Sound

<sup>6</sup>Soundness Building Blocks

BPMN2.0 معادلشان تبدیل می‌کنیم. برای این منظور، علاوه بر نگاشت، از قواعد ترسیمی گفته شده در بخش ۲-۲-۴ نیز استفاده می‌کنیم تا سازگاری مدل‌های مبدا و مقصد حفظ شود. در ذیل هر ساختار بلوکی صحت (که از این پس از آن‌ها به عنوان الگوهای صحت یاد خواهد شد) به سبب اختصار، از بیان جزئیات انتقال هر کدام از الگوها به مدل‌های فرآیندی BPMN2.0 معادلشان، خودداری شده‌است.

### الگوی پایه

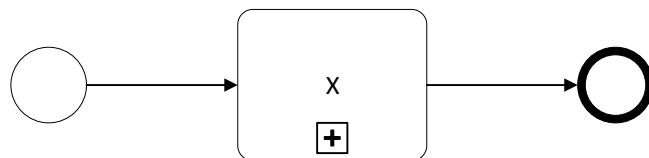
الگوی پایه‌ای یا سازه‌ی بلوکی پایه‌ای<sup>۷</sup>، یک مدل فرآیندی است که حاوی یک فعالیت است و این فعالیت می‌تواند با فعالیت‌های دیگر و یا زیر فرآیندهای دیگر جانشین شود. بنابر گفته‌ی [۳] این مدل فرآیندی، یک مدل فرآیندی صحیح است و در صورتی که زیرفرآیندهایی صحیح با فعالیت موجود در آن جایگزین شوند، فرآیند نهایی نیز صحیح خواهد بود. در [۳] این مدل فرآیندی به عنوان یک سازه‌ی بلوکی صحت، ارائه شده‌است و در شکل ۱۹-۴ قابل مشاهده است. در این شکل، چنان‌چه به جای انتقال X یک زیرفرآیند صحیح قرار بگیرد، مدل حاصل بازهم صحیح خواهد بود.



شکل ۱۹-۴: بلوک سازه‌ای پایه به زبان مدل‌سازی شبکه‌های پتری جریان کار [۳]

اگر این مدل فرآیندی صحیح را در زبان 2.0 BPMN به کار گرفته شود، به جای مکان‌های شروع و پایان، رخدادهای شروع و پایانی می‌تواند قرار گیرد؛ این کار بر طبق نگاشت ارائه شده توسط رمکو و داکمن در [۴] صحیح است. همچنین، طبق این تحقیق، می‌توان فعالیت X را با یک انتقال جایگزین نمود. این انتقال می‌تواند هم به عنوان یک فعالیت و هم به عنوان یک زیرفرآیند مورد بررسی قرار گیرد. با توجه به این که در این بلوک‌های سازه‌ای، امکان جایگزینی انتقال X با سایر زیر فرآیندها وجود دارد، ترجیح بر آن است که به یک زیرفرآیند در مدل فرآیندی 2.0 BPMN نگاشته شود. در نهایت، بلوک سازه‌ای پیشنهادی در شکل ۲۰-۴ ترسیم شده‌است.

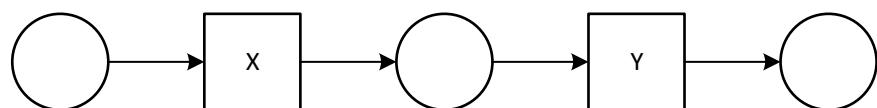
<sup>7</sup>Basic Building Block



[۳] شکل ۲۰-۴: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای پایه به زبان مدل‌سازی ۲.۰ BPMN

### الگوی دنباله‌ای

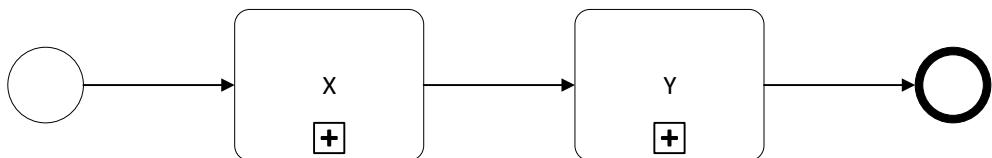
الگوی دنباله‌ای و یا سازه‌ی بلوکی دنباله‌ای<sup>۸</sup> یک مدل فرآیندی است که حاوی تعدادی فعالیت بوده که در پی یکدیگر اجرا می‌شوند و با اتمام هر فعالیت، فعالیت بعدی اجرا می‌شود. بنابرگفته‌ی [۳] این مدل فرآیندی، یک مدل فرآیندی صحیح است و در صورتی که زیرفرآیندهایی صحیح با هر کدام از فعالیت‌های موجود در آن جایگزین شود، مدل فرآیندی حاصل نیز صحیح خواهد بود؛ بنابراین در [۳] این مدل فرآیندی نیز به عنوان یک بلوک سازه‌ای صحیح معرفی شده‌است و در شکل ۲۱-۴ قابل مشاهده است؛ بنابراین، در این مدل فرآیندی چنانچه به جای انتقال X یک زیرفرآیند صحیح قرار گیرد، مدل فرآیندی حاصل نیز صحیح خواهد بود. اگر این مدل در زبان ۲,۰ BPMN به کار گرفته شود، مشابه با



[۳] شکل ۲۱-۴: بلوک سازه‌ای دنباله‌ای به زبان مدل‌سازی شبکه‌های پتری جریان کاری

الگوی پایه‌ای مورد استفاده قرار خواهد گرفت؛ در این الگوی صحت، به جای انتقال دوم نیز، یک زیرفرآیند قرار خواهد گرفت؛ در نتیجه، شکل ۲۲-۴ به عنوان معادلی برای مدل فرآیندی الگوی دنباله‌ای پیشنهاد می‌شود. در این نگاشت نیز به جای انتقال‌ها، از زیرفرآیندها استفاده شده‌است تا الگو قابل تعمیم باشد؛ بر طبق تحقیق داکمن [۴] در تحلیل ساختاری مدل‌های فرآیندی به منظور کشف خطاهاست ساختاری، در برخورد با زیر فرآیندها، می‌توان آن‌ها را با یک فعالیت معمولی، معادل دانست.

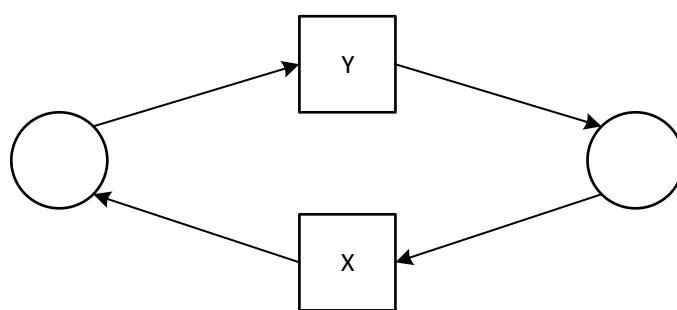
<sup>8</sup> Sequential Building Block



شکل ۲۲-۴: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای دنباله‌ای به زبان مدل‌سازی BPMN 2.0

### الگوی تکرار

الگوی تکرار و یا سازه‌ی بلوکی تکرار<sup>۹</sup> یک مدل فرآیندی است که حاوی یک یا چند فعالیت است که به صورت تکراری اجرا می‌شوند. بنابرگفته‌ی [۳] این مدل فرآیندی، یک مدل فرآیندی صحیح است و در صورتی که زیرفرآیندهایی صحیح با هر کدام از فعالیت‌های موجود در آن جایگزین شود، مدل فرآیندی حاصل نیز صحیح خواهد بود؛ بنابراین در [۳] این مدل فرآیندی نیز به عنوان یک بلوک سازه‌ای صحیح معرفی شده‌است و در شکل ۲۳-۴ قابل مشاهده است؛ بنابراین، در این مدل فرآیندی چنانچه به جای انتقال‌های X و Y زیرفرآیندهای صحیح قرار گیرد، مدل فرآیندی حاصل نیز صحیح خواهد بود. چنان‌چه یک الگوی صحت تکرار حاوی یک فعالیت باشد، یک انتقال دیگر برای بازگرداندن توکن‌ها به مکان اولیه مورد نیاز خواهد بود؛ در برخی از پژوهش‌ها مانند پژوهش داکمن، از این انتقال‌های بی‌اثر به عنوان قدم‌های ساکت<sup>۱۰</sup> نام برده شده‌است. به منظور به کارگیری این مدل فرآیندی در زبان مدل‌سازی

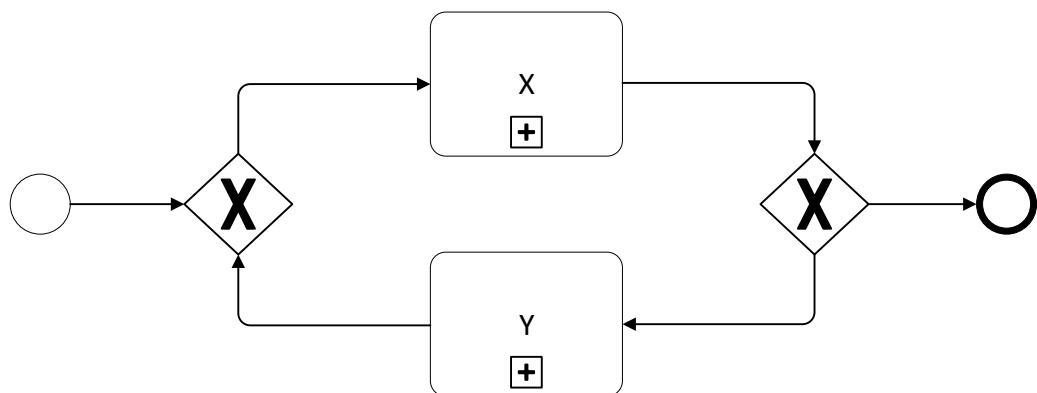


شکل ۲۳-۴: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای تکرار به زبان مدل‌سازی شبکه‌های پتری جریان کاری [۳]

<sup>9</sup>Iteration Building Block

<sup>10</sup>Silent Steps

BPMN 2.0 الگوی ارائه شده در تصویر ۲۴-۴ پیشنهاد می‌شود. طبق پژوهش صورت گرفته توسط داکمن و همانند الگوهای پیشین، به جای انتقال‌های قابل جایگزینی با مدل‌های فرآیندی دیگر، زیر فرآیندهایی را جایگذاری خواهیم نمود [۴] و طبق همان پژوهش، به جای نقطه‌ی پایانی تکرار و بر طبق الگوی جریان کنترلی اتصال انحصاری، از دروازه‌ی انحصاری استفاده می‌کنیم و برای ابتدای الگوی تکرار، از دروازه‌ی اتصال انحصاری استفاده می‌کنیم.

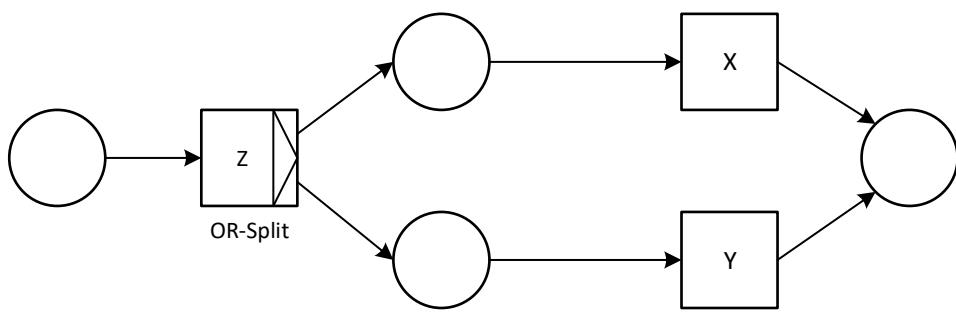


شکل ۲۴-۴: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای تکرار به زبان مدل‌سازی BPMN 2.0

### الگوی انشعاب فصلی غیرصریح

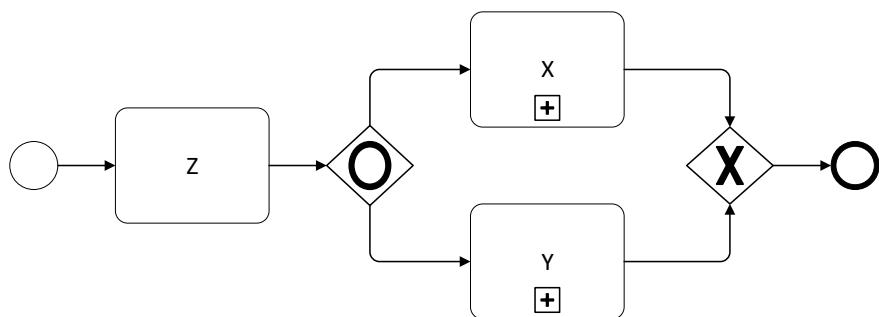
الگوی انشعاب فصلی غیرصریح و یا سازه‌ی بلوکی انشعاب فصلی ضمنی<sup>۱۱</sup> یک مدل فرآیندی است که حاوی یک یا چند فعالیت است که حداقل یکی از آن‌ها و حداقل همه‌ی آن‌ها اجرا می‌شوند. بنابرگفته‌ی [۳] این مدل فرآیندی، یک مدل فرآیندی صحیح است و در صورتی که زیرفرآیندهایی صحیح با هر کدام از فعالیت‌های موجود در آن جایگزین شود، مدل فرآیندی حاصل نیز صحیح خواهد بود؛ بنابراین در [۳] این مدل فرآیندی نیز به عنوان یک بلوک سازه‌ای صحیح معرفی شده‌است و در شکل ۲۵-۴ قابل مشاهده است؛ بنابراین، در این مدل فرآیندی چنانچه به جای انتقال‌های X و Y زیرفرآیندهای صحیح ۲۵-۴ قرار گیرد، مدل فرآیندی حاصل نیز صحیح خواهد بود. چنان‌چه سعی شود مدل فرآیندی شکل ۲۵-۴ را با زبان مدل‌سازی BPMN 2.0 بیان شود، بر طبق نگاشت ارائه شده به شکل ۲۶-۴ می‌رسیم؛ این شکل بیان می‌کند که همانند مدل ارائه شده به زبان شبکه‌های پتری جریان کار، بعد از اتمام فعالیت Z (بعد از شلیک شدن انتقال Z در شکل ۲۵-۴) حداقل یکی از دو شاخه‌ی منتهی به زیرفرآیندهای Z و Y

<sup>11</sup>Implicit OR-split Building Block



شکل ۴-۲۵: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای فصلی صریح به زبان مدل‌سازی شبکه‌های پتری جریان کاری [۳]

فعال می‌شود و بخش اتصالی نیز مطابق با نگاشت ارائه شده توسط داکمن و نگاشت ارائه شده در بخش ۴-۲-۴، قابل توجیه است.

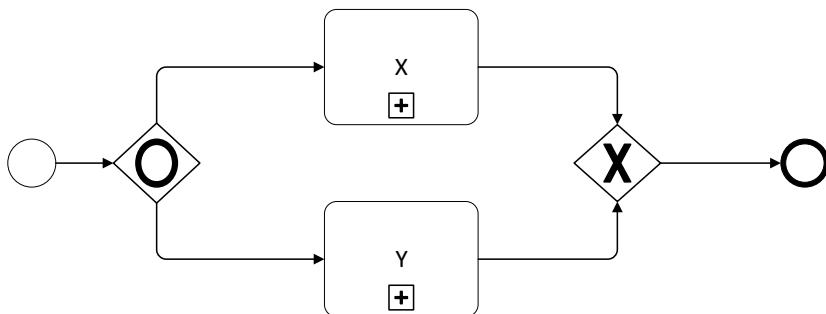


شکل ۴-۲۶: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای فصلی صریح به زبان مدل‌سازی BPMN 2.0

باز می‌توان الگو را ساده‌تر نمود و در نتیجه، قابلیت تعمیم‌پذیری آن را بالاتر برد. مدل فرآیندی ارائه شده در تصویر ۴-۲۶ می‌تواند با استفاده از الگوی دنباله‌ای و مدل فرآیندی ارائه شده در تصویر ۴-۲۷ ساخته شود. این کار با طی مراحل زیر قابل اجرا است:

۱. در ابتدا، در الگوی دنباله‌ای ارائه شده در شکل ۴-۲۲ به جای زیرفرآیند X یک فعالیت به نام Z قرار داده شود.
۲. همچنان، در همان الگوی دنباله‌ای، به جای زیرفرآیند Y الگوی صحت ارائه شده در تصویر ۴-۲۷ قرار داده شود و در نتیجه، الگوی صحت صحیح ۴-۲۶ بدست می‌آید. در این الگوی صحت، استفاده از دروازه‌ی اتصال فصلی به جای اتصال انحصاری نیز قابل قبول خواهد بود.

با توجه به این که الگوی صحت ارائه شده در شکل ۲۶-۴ صحیح است (بر اساس نگاشت انجام شده از مدل فرآیندی BPMN 2.0) و شامل ترکیب الگوی صحت دنباله‌ای، پایه و مدل فرآیندی شکل ۲۷-۴ است، در نتیجه مدل فرآیندی ارائه شده در تصویر ۲۷-۴ نیز صحیح خواهد بود؛ در غیر این صورت و با توجه به ترکیب ارائه شده، الگوی صحت ارائه شده در شکل ۲۶-۴ نا صحیح خواهد شد که برخلاف فرض ، مبني بر صحیح بودن آن‌ها است. در این نگاشت نیز به جای انتقال‌ها، از زیرفرآیندها استفاده شده است تا الگو قابل تعمیم باشد؛ بر طبق تحقیق داکمن [۴] در تحلیل ساختاری مدل‌های فرآیندی به منظور کشف خطاهای ساختاری، در برخورد با زیر فرآیندها، می‌توان آن‌ها را با یک فعالیت معمولی، معادل دانست.

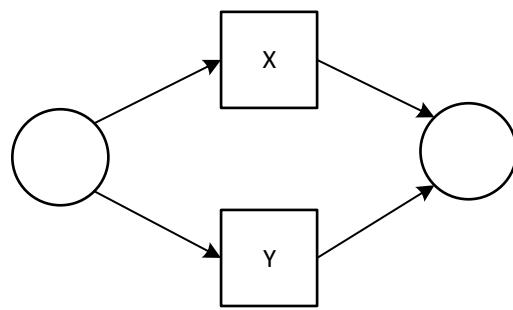


شکل ۲۷-۴: بلوک سازه‌ای پیشنهادی و ساده‌سازی معادل با بلوک سازه‌ای فصلی صریح به زبان مدل‌سازی BPMN 2.0

### الگوی انحصاری

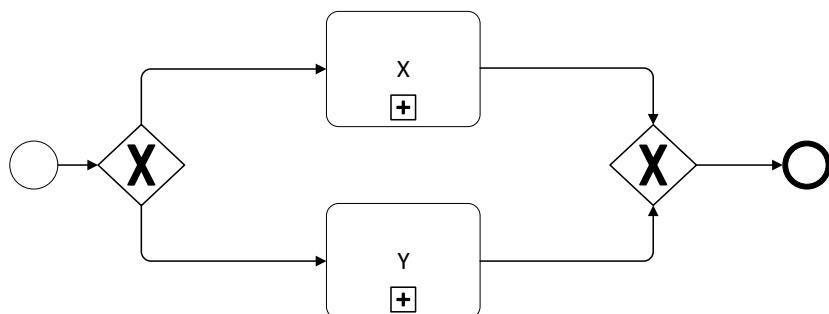
الگوی انشعباب فصلی غیرصریح و یا سازه‌ی بلوکی فصلی غیر صریح (که از این پس آن را الگوی انحصاری می‌نامیم)<sup>۱۲</sup> یک مدل فرآیندی است که حاوی یک یا چند فعالیت است که فقط یکی از آن‌ها اجرا می‌شوند. بنابرگفته‌ی [۳] این مدل فرآیندی، یک مدل فرآیندی صحیح است و در صورتی که زیرفرآیندهایی صحیح با هر کدام از فعالیت‌های موجود در آن جایگزین شود، مدل فرآیندی حاصل نیز صحیح خواهد بود؛ بنابراین در [۳] این مدل فرآیندی نیز به عنوان یک بلوک سازه‌ای صحیح معرفی شده است و در شکل ۲۸-۴ قابل مشاهده است؛ بنابراین، در این مدل فرآیندی چنانچه به جای انتقال‌های X و Y زیرفرآیندهای صحیح قرار گیرد، مدل فرآیندی حاصل نیز صحیح خواهد بود. بر طبق نگاشت ارائه شده توسط داکمن در [۴] که در شکل ۳-۵ آمده است، مدل فرآیندی BPMN 2.0 معادل با این مدل فرآیندی، در تصویر ۲۹-۴

<sup>12</sup>Explicit OR-split Building Block



شکل ۲۸-۴: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای انحصاری به زبان مدل‌سازی شبکه‌های پتری جریان کاری [۳]

آمده است. در این نگاشت نیز به جای انتقال‌ها، از زیرفرآیندها استفاده شده است تا الگو قابل تعمیم باشد؛ بر طبق تحقیق داکمن [۴] در تحلیل ساختاری مدل‌های فرآیندی به منظور کشف خطاها در ساختاری، در برخورد با زیر فرآیندها، می‌توان آن‌ها را با یک فعالیت معمولی، معادل دانست.



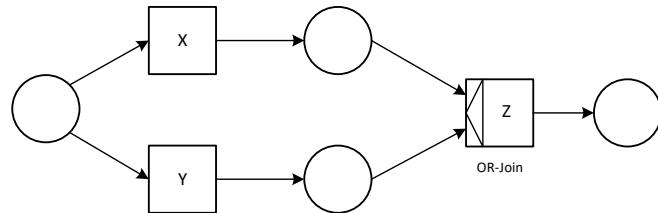
شکل ۲۹-۴: بلوک سازه‌ای پیشنهادی معادل با بلوک انحصاری به زبان مدل‌سازی BPMN 2.0

### الگوی اتصال فصلی غیر صریح

الگوی اتصال فصلی صریح و یا سازه‌ی بلوکی اتصال فصلی صریح<sup>۱۳</sup> یک مدل فرآیندی است که حاوی یک یا چند فعالیت است که حداقل یکی از آن‌ها اجرا می‌شوند. بنابرگفته‌ی [۳] این مدل فرآیندی، یک مدل فرآیندی صحیح است و در صورتی که زیرفرآیندهایی صحیح با هر کدام از فعالیت‌های موجود در آن جایگزین شود، مدل فرآیندی حاصل نیز صحیح خواهد بود؛ بنابراین در [۳] این مدل فرآیندی نیز به عنوان یک بلوک سازه‌ای صحیح معرفی شده است و در شکل ۳۰-۴ قابل مشاهده است؛ بنابراین، در این

<sup>13</sup>Explicit OR-split Building Block

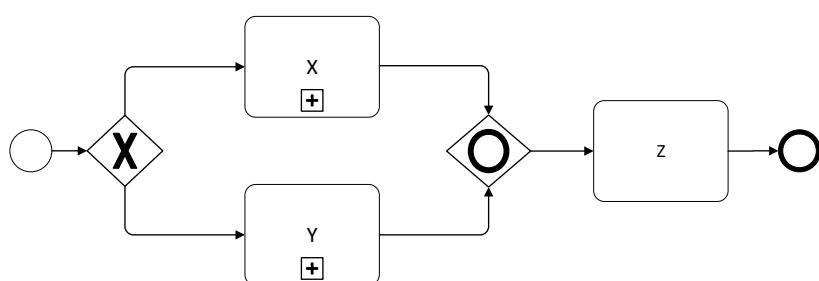
مدل فرآیندی چنانچه به جای انتقال‌های X و Y زیرفرآیندهای صحیح قرار گیرد، مدل فرآیندی حاصل نیز صحیح خواهد بود. نگاشت و تحلیل این مدل فرآیندی به زبان مدل‌سازی BPMN 2.0 همانند الگوی



شکل ۴-۳۰: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای تکرار به زبان مدل‌سازی شبکه‌های پتری جریان کاری [۳]

انشعاب فصلی غیرصریح توجیه می‌شود، با این تفاوت که در قسمت اتصال، می‌توان هم از دروازه‌ی اتصال فصلی و هم دروازه‌ی اتصال انحصاری استفاده نمود؛ چرا که در این ساختار، فقط یکی از شاخه‌های بالایی و پایینی می‌توانند فعال شوند و در نهایت، فقط یکی از شاخه‌ها به ادامه‌ی جریان کنترلی فرآیند متصل می‌شود.

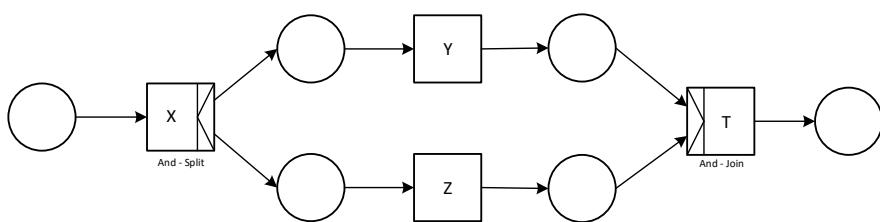
این الگو می‌تواند با ترکیب دو الگوی دنباله‌ای و الگوی انحصاری به وجود بیاید؛ بدین صورت که به جای زیرفرآیند اول دنباله، الگوی انحصاری و به جای زیرفرآیند دوم آن، از یک فعالیت استفاده شود. بنابراین، این الگو نیز یک الگوی فرآیندی صحیح خواهد بود؛ چرا که از ترکیب سازه‌های بلوکی صحیح به وجود آمده است.



شکل ۴-۳۱: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای تکرار به زبان مدل‌سازی BPMN 2.0

### الگوی عطفی

الگوی عطفی و یا سازه‌ی بلوکی عطفی<sup>۱۴</sup> یک مدل فرآیندی است که حاوی یک یا چند فعالیت است که تمامی آن‌ها به طور موازی اجرا می‌شوند. بنابرگفته‌ی [۳] این مدل فرآیندی، یک مدل فرآیندی صحیح است و در صورتی که زیرفرآیندهایی صحیح با هر کدام از فعالیت‌های موجود در آن جایگزین شود، مدل فرآیندی حاصل نیز صحیح خواهد بود؛ بنابراین در [۳] این مدل فرآیندی نیز به عنوان یک بلوک سازه‌ای صحیح معرفی شده‌است و در شکل ۳۲-۴ قابل مشاهده است؛ بنابراین، در این مدل فرآیندی چنانچه به جای انتقال‌های X و Y زیرفرآیندهای صحیح قرار گیرد، مدل فرآیندی حاصل نیز صحیح خواهد بود. به منظور نگاشت مدل فرآیندی به زبان مدل‌سازی BPMN 2.0 می‌توان طبق نگاشت ارائه



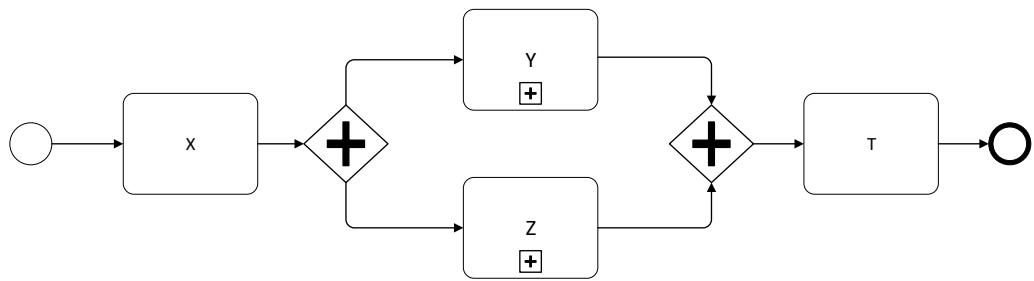
شکل ۳۲-۴: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای عطفی به زبان مدل‌سازی شبکه‌های پتری جریان کاری [۳]

شده در بخش ۲-۲-۴ عمل نمود، یعنی می‌توان به جای انتقال‌ها، از فعالیت‌ها و زیرفرآیندها استفاده کرد. به جای انتقال‌هایی که قابل جایگزینی با زیرفرآیندهای دیگری هستند، از زیرفرآیندها استفاده می‌کنیم تا این الگو، قابلیت استفاده‌ی مجدد باشد؛ همچنین به منظور موازی‌سازی و هم‌گام سازی، به ترتیب از دروازه‌های انشعاب عطفی و اتصال عطفی استفاده می‌کنیم. با رسیدن جریان کنترلی به دروازه‌ی انشعاب موازی، جریان کنترلی از تمامی شاخه‌های منشعب شده ادامه می‌یابد و به منظور ادامه‌ی جریان از دروازه‌ی اتصال موازی، لازم است اجرای تمامی شاخه‌های موازی به اتمام رسیده باشد؛ در نهایت، الگوی پیشنهادی در شکل ۳۳-۴ بدست می‌آید.

### ۵-۲-۴ پادالگوهای صحت در مدل‌های فرآیندی

در بخش قبل، الگوهایی در قالب بلوک‌های سازه‌ای، برای به کارگیری در مدل‌های فرآیندی BPMN 2.0 پیشنهاد شد. در ادامه، بر اساس چارچوب پژوهش که در ابتدای این فصل تشریح شد، در این بخش

<sup>14</sup> And Building Block



شکل ۴-۳۳: بلوک سازه‌ای پیشنهادی معادل با بلوک سازه‌ای عطفی به زبان مدل‌سازی BPMN 2.0

پادالگوهایی<sup>۱۵</sup> پیشنهاد خواهد شد که در صورت به کارگیری در مدل‌های فرآیندی ۲.۰ BPMN می‌تواند منجر به بروز خطاهای معنایی، همچون بن‌بست و بن‌بست حلقوی شود و در مدل‌های فرآیندی، باید مراقب بود که این پادالگوها بروز پیدا نکنند.

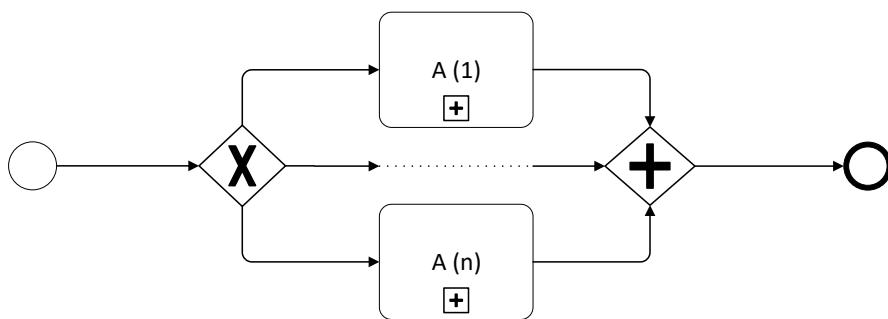
### اتصال توام دروازه‌ی انشعباب انحصاری و اتصال موازی

در صورتی که تمامی شاخه‌های منشعب شده از یک دروازه‌ی انشعباب انحصاری، به یک دروازه‌ی انشعباب موازی متصل شود، فرآیند قطعاً با بن‌بست مواجه خواهد شد. این موقعیت در تصویر ۴-۳۴ نشان داده شده است. در صورتی که یکی از وضعیت‌های زیر رخ دهد، احتمال وقوع بن‌بست وجود خواهد داشت، اما وقوع آن قطعی نخواهد بود؛ نتیجه‌های بدست آمده از پژوهش صورت گرفته در حوزه‌ی تحلیل احتمالاتی صحت، یعنی پژوهش [۲۸] نیز این موضوع را تایید می‌کند:

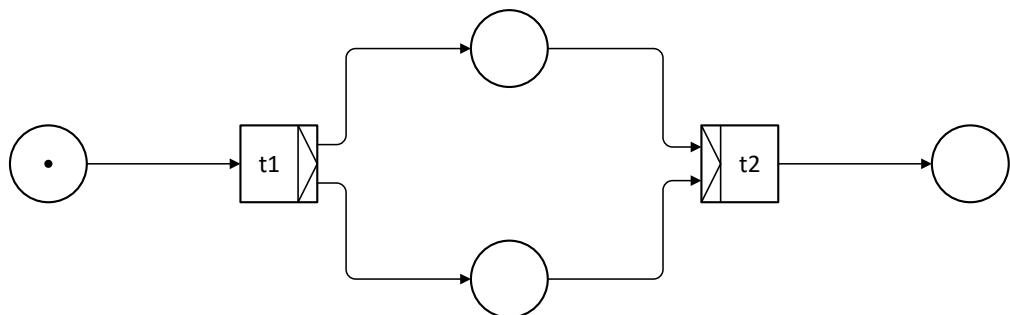
۱. در صورتی که حداقل یکی از ورودی‌های دروازه‌ی اتصال عطفی، به حداقل یکی از خروجی‌های یک انشعباب انحصاری و یا یک انشعباب فصلی متصل شود.
۲. در صورتی که تمامی خروجی‌های یک انشعباب فصلی به ورودی‌های یک اتصال عطفی (دوازه‌ی موازی) متصل شود.

در [۲] این موقعیت در شبکه‌های پتری جریان کار نیز بیان شده است و در تصویر ۴-۳۴ قابل مشاهده است؛ در این شکل، در صورتی که قانون شلیک انتقال  $t_1$  به صورت انحصاری تعریف شده باشد، این انتقال به صورت یک انشعباب انحصاری عمل خواهد کرد و در یکی از خروجی‌های خود، توکن خواهد گذاشت. از سوی دیگر، انتقال  $t_2$  در صورتی فعال می‌شود که در هردو مکان ورودی آن، حداقل یک توکن وجود داشته باشد؛ درنتیجه، شلیک نخواهد شد و دچار بن‌بست خواهد شد [۲].

<sup>15</sup> Antipattern



شکل ۴-۴: پاد الگوی اتصال توام انشعاب انحصاری و اتصال موازی و بروز بنبست در مدل فرآیندی BPMN 2.0



شکل ۴-۵: پاد الگوی اتصال توام انشعاب انحصاری و اتصال موازی و بروز بنبست در یک شبکه پتری جریان کار [۲]

#### ایجاد بنبست حلقوی با دروازه‌ی انشعاب موازی

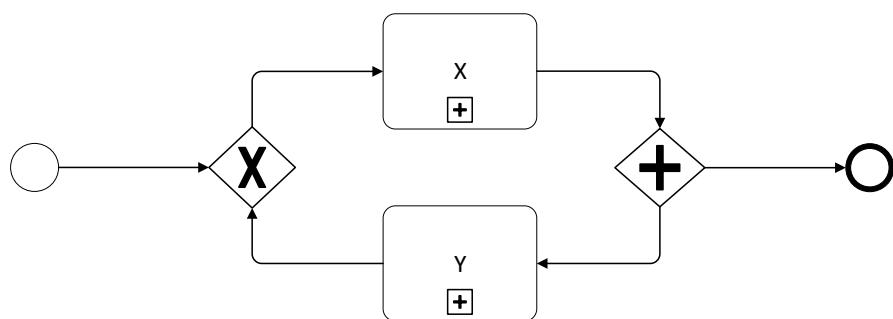
در صورتی که در یک مدل فرآیندی مدل‌سازی شده با زبان BPMN 2.0 یک دور وجود داشته باشد که در خروجی آن، یک دروازه‌ی انشعاب موازی وجود داشته باشد، این دروازه باعث ایجاد بنبست حلقوی خواهد شد؛ چرا که یکی از شاخه‌های خروجی آن منجر به ادامه‌ی جریان فرآید در داخل حلقه خواهد شد و دیگری، جریانی را به بیرون حلقه هدایت می‌کند. این پادالگو در حالت کلی، در تصویر ۴-۳۶ نشان داده شده است.

همان‌طور که وقوع بنبست می‌تواند منجر به عدم اجرای برخی از فعالیت‌ها در طول فرآیند شود، بنبست حلقوی باعث می‌شود که مجموعه‌ای از فعالیت‌ها، در یک حلقه‌ی اجرایی با تعداد دفعات تکرار نامتناهی، گرفتار شوند؛ عواملی که بنبست حلقوی می‌شود به شامل موارد زیر می‌شود [۲] :

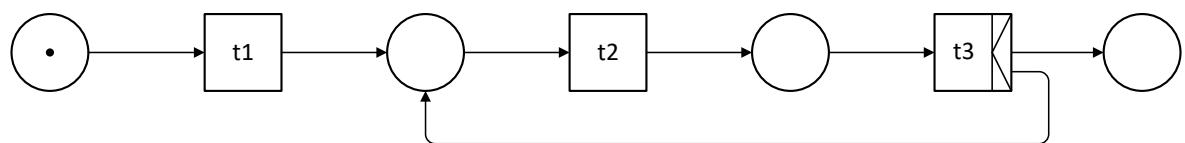
۱. وجود شرط‌های نامناسب در گره‌های تصمیم‌گیری.

## ۲. وجود ساختارهای نامناسب کنترلی در مدل فرآیندی.

آنچه که در این بخش به عنوان پادالگو مطرح شد، در دسته‌ی دوم عوامل خطا‌ساز واقع می‌شود و نمونه‌ای از وقوع آن در شبکه‌های پتری جریان کار، در شکل ۳۷-۴ نشان داده شده است. در این نمونه، یک انشعاب انحصاری در موقعیت خروجی یک حلقه واقع شده است و منجر به بروز بنبست حلقوی شده است.



شکل ۳۶-۴: پاد الگوی ایجاد بنبست حلقوی با قرارگیری یک دروازه‌ی موازی در یک دور در مدل فرآیندی BPMN 2.0



شکل ۳۷-۴: پاد الگوی ایجاد بنبست حلقوی با قرارگیری در یک شبکه‌ی پتری جریان کار [۲]

## ۶-۲-۴ صحت‌سنجی با روش جایگذاری

در بخش ۴-۲-۴ به معرفی الگوهای صحت در شبکه‌های پتری جریان کار پرداخته شد و سپس، به مدل‌های فرآیندی BPMN 2.0 تعمیم داده شدند. در کتاب مدیریت جریان‌های کاری [۳] روشی به منظور صحت‌سنجی ارائه شده است که بر اساس ترکیب الگوهای صحت با یک دیگر و با استفاده از شبکه‌های پتری جریان کار، اقدام به صحت‌سنجی مدل‌های فرآیندی می‌نماید. در ابتدا به توضیح این الگوریتم پرداخته و سپس، این روش را به مدل‌های فرآیندی BPMN 2.0 تعمیم می‌دهیم.

## الگوریتم صحتسنجی تلفیقی در شبکه‌های پتری جریان کار

همان‌طور که در بخش ۲-۷-۱ شرح داده شد، یک مدل فرآیندی در صورتی صحیح است که دارای دو ویژگی باشد:

۱. زنده باشد؛ یعنی برای هر کدام از انتقال‌های آن، وضعیتی وجود داشته باشد که آن انتقال فعال شده و شلیک شود؛

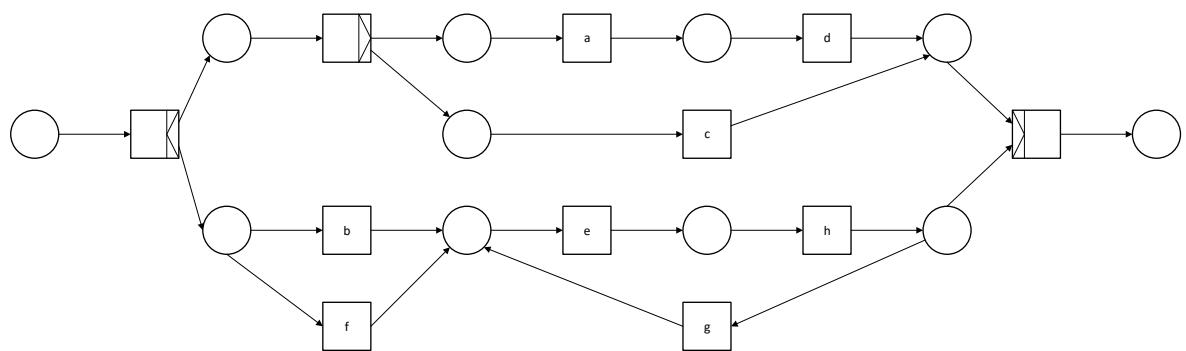
۲. کران‌دار باشد؛ یعنی تعداد توکن‌های موجود در هر مکان، تا حد معینی رشد کند و بی‌نهایت نباشد.

به طور غیررسمی می‌توان گفت که شرط اول، تضمین‌کننده‌ی عدم وجود بن‌بست و شرط دوم، تضمین‌کننده‌ی عدم وجود بن‌بست حلقوی است و با وجود آن، می‌توان مطمئن بود که در نهایت، فرآیند در یک وضعیت، پایان می‌یابد.

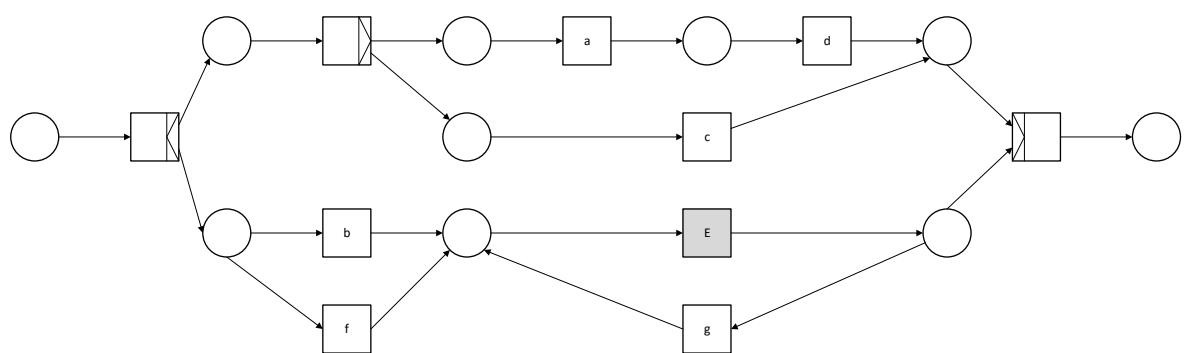
در کتاب مدیریت جریان‌های کاری [۳] دو روش برای صحتسنجی پیشنهاد شده است: یک روش مبتنی بر رایانه و یک روش، بدون نیاز به رایانه. در روش صحتسنجی و بدون نیاز به رایانه، الگوهای مطرح شده در بخش ۴-۲-۴ مطرح شدند و سعی شده که بدون نیاز به رایانه، الگوهای مطرح شده در این بخش استخراج شوند. در صورتی که یک فرآیند کاری، صرفاً با ترکیب این الگوها ساخته شود، آن فرآیند قطعاً صحیح است و هر دو شرط گفته شده در تعریف ۲-۷-۱ برای آن صادق است؛ اما اگر بخش‌هایی از این فرآیند کاری، مطابق با این الگوها نباشد، الزاماً فرآیند کاری داده شده، فرآیندی غیرصحیح نخواهد بود و صحت آن، منوط به صحت زیرفرآیندها در سازه‌های بلوکی شناسایی شده، می‌باشد.

در ادامه، در قالب یک مثال ساده، یک مدل فرآیندی صحیح را در نظر گرفته و صحت آن را با استفاده از سازه‌های بلوکی صحیح، تایید می‌کنیم.

**نمونه ۲-۴**. در شکل ۳۸-۴ یک نمونه مدل فرآیندی به زبان شبکه‌های پتری جریان کار ترسیم شده است [۳]. قصد داریم روش ارائه شده در [۳] را بر روی این مدل فرآیندی اعمال نماییم. در ابتدا، مشاهده می‌کنیم که دو انتقال  $e$  و  $h$  به دنبال یکدیگر آمده‌اند و تشکیل یک الگوی دنباله‌ای (بخش ۴-۲-۴) را می‌دهند. بنابراین، حذف آن و جانشینی آن با یک انتقال، وضعیت صحت مدل فرآیندی را تغییر نمی‌دهد. بنابراین آن را حذف کرده و به جای آن، انتقال  $E$  را قرار می‌دهیم. شکل فرآیند بعد از حذف این الگو در شکل ۳۹-۴ قابل مشاهده است. در شکل ۳۹-۴ مشاهده می‌شود که دو انتقال  $E$  و  $g$  تشکیل الگوی تکرار (طبق توضیحات بخش ۴-۲-۴) را می‌دهند. بنابراین، می‌توان سازه‌ی بلوکی تکرار را از آن حذف کرد و به جای آن، انتقال  $E$  را قرار داد. با این کار، به صحت کل مدل فرآیندی خللی وارد

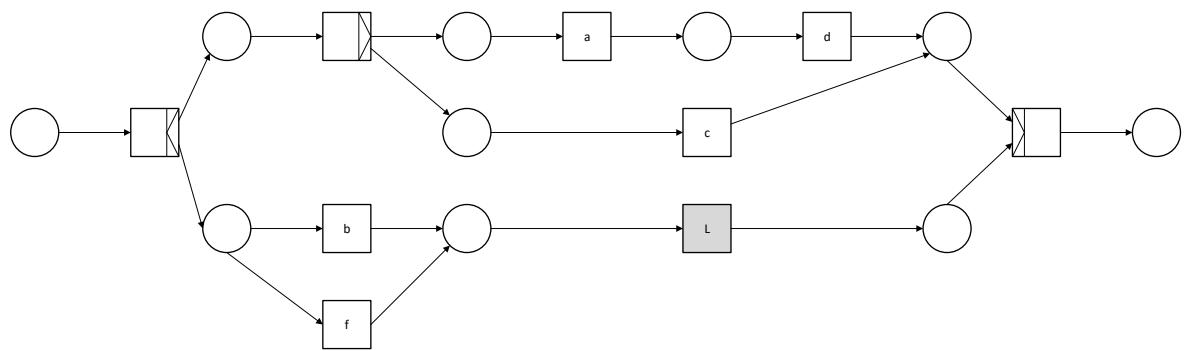


شکل ۴-۳۸: الگوی فرآیندی مورد نظر برای صحبت‌سنجی



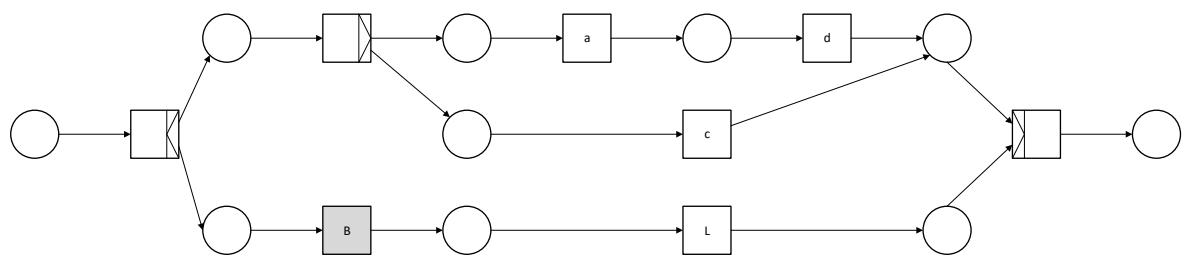
شکل ۴-۳۹: اعمال الگوی دنباله‌ای بر روی مدل فرآیندی

نخواهد شد؛ در نتیجه شکل ۴۰-۴ به وجود می‌آید. در شکل ۴۰-۴ مشاهده می‌شود که دو انتقال b و f



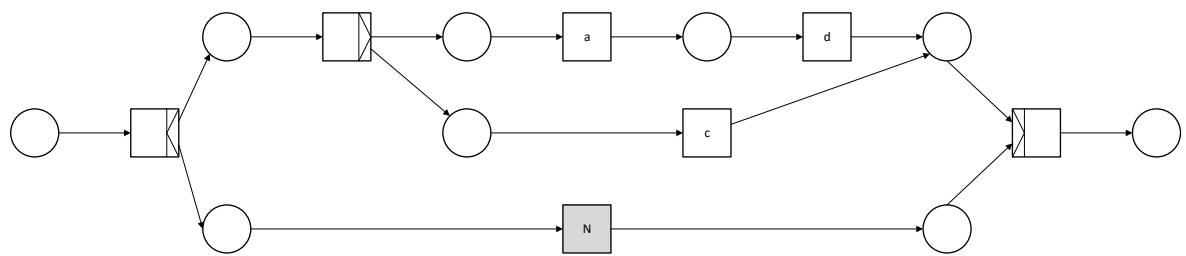
شکل ۴-۴۰: کشف الگوی تکرار بر روی مدل فرآیندی

الگوی انحصاری را تشکیل می‌دهند (طبق توضیحات بخش ۴-۲-۴)؛ بنابراین حذف آن و جایگزینی آن با یک انتقال، خللی به صحت مدل فرآیندی وارد نمی‌کند؛ در نتیجه، آن‌ها را حذف و با انتقال B جانشین می‌کنیم و شکل ۴۱-۴ بدست می‌آید. با این کار مشاهده می‌شود که در آخرین شاخه‌ی فرآیند، یک



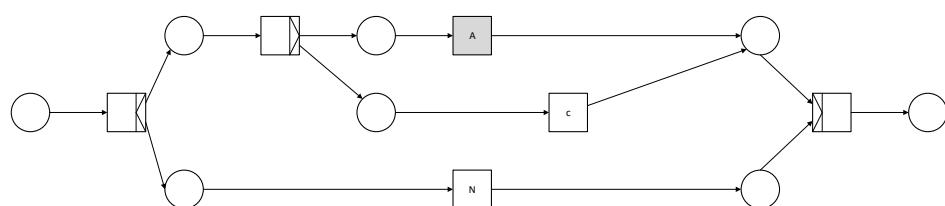
شکل ۴-۴: کشف الگوی فصل غیر صریح بر روی مدل فرآیندی

الگوی دنباله‌ای، توسط دو انتقال  $B$  و  $L$  به وجود می‌آید؛ بنابراین می‌توان این الگوی صحت را استخراج نمود و یک انتقال را به جای آن نشاند. با جانشینی انتقال  $N$  به جای آن، شکل ۴-۴ بدست می‌آید.



شکل ۴-۵: کشف الگوی دنباله‌ای بر روی مدل فرآیندی

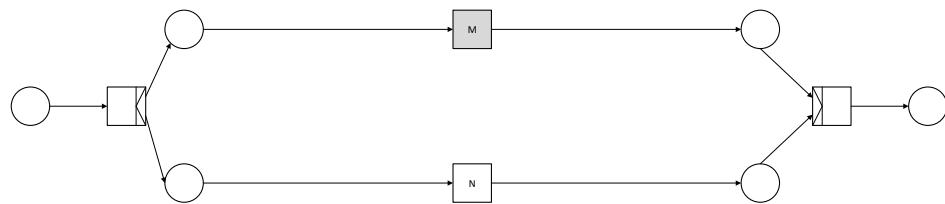
در ادامه مشاهده می‌شود که دو انتقال  $a$  و  $d$  با یکدیگر الگوی دنباله‌ای را تشکیل می‌دهند؛ بنابراین همانند مراحل قبل، حذف آن و جانشینی آن با یک انتقال، وضعیت صحت مدل فرآیندی را تغییر نخواهد داد. بنابراین می‌تواند حذف شده و با انتقال  $A$  جانشین شود. در نتیجه، تصویر ۴-۳ می‌داند مدل فرآیندی در اینجا تغییر نموده است.



شکل ۴-۶: کشف الگوی فرآیندی دنباله‌ای بر روی مدل فرآیندی

را تشکیل می‌دهند و به روش گام‌های قبل، می‌توان آن را با یک انتقال جانشین‌سازی کرد. در شکل ۴-۴ مشاهده می‌شود که این الگو با انتقال  $M$  جانشین شده‌است و در نهایت، یک الگوی عطفی به وجود

آمده است. در نهایت، تمام الگوی عطفی را می‌توان با انتقالی مانند  $R$  جانشین کرد و در نهایت به شکل



شکل ۴-۴: کشف الگوی انشعاب فصلی صریح بر روی مدل فرآیندی

۴-۵ رسید؛ در این شکل مشاهده می‌شود که به پایین‌ترین سطح الگوها رسیده‌ایم و مدل فرآیندی نهایی، طبق الگوی پایه‌ای، الگویی صحیح خواهد بود. با توجه به این‌که موفق شدیم تمام فرآیند را با ترکیب الگوهای صحت ایجاد کنیم، نتیجه می‌گیریم که مدل فرآیندی ارائه شده در تصویر ۳۸-۴ یک مدل فرآیندی صحیح می‌باشد.



شکل ۴-۵: کشف الگوی عطفی بر روی مدل فرآیندی

### تعمیم روش تلفیقی به مدل‌های فرآیندی BPMN 2.0

پیش از تعمیم روش‌های صحت‌سنجی در شبکه‌های جریان کاری به مدل‌های فرآیندی BPMN 2.0 لازم است بیانی صوری را از روش بخش قبل ارائه کرده و آن را تعمیم دهیم. روش بخش قبل، در ضمیمه‌ی کتاب مدیریت جریان‌های کاری [۳] به صورتی که در ادامه می‌آید، بیان شده است.

فرض کنید شبکه‌ی جریان کاری  $(P_1, T_1, F_1)$  و  $(P_2, T_2, F_2)$  دو شبکه‌ی جریان کاری، با مکان‌های شروع  $i$  و مکان‌های پایانی  $o$  باشند که روابط زیر در خصوص آن‌ها برقرار باشد:

$$T_1 \cap T_2 = \emptyset \quad (38-4)$$

$$P_1 \cap P_2 = \{i, o\} \quad (39-4)$$

و همچنین، انتقالی به نام  $t^+$  وجود داشته باشد، به طوری که:  $t^+ \in T_1$ ؛ در ادامه، فرض کنید که شبکه‌ی

پتری جریان کار  $P_3 = (P_3, T_3, F_3)$  در اثر جایگذاری شبکه‌ی پتری جریان کار  $PN_2$  به جای انتقال  $t^+$  در شبکه‌ی پتری  $PN_1$  بدست می‌آید. در این صورت، موارد زیر در خصوص مکان‌های این شبکه‌ی پتری جریان کار (یعنی مجموعه‌ی  $P_3$ ) برقرار خواهد بود.

$$P_3 = P_1 \cup P_2 \quad (40-4)$$

در خصوص انتقال‌های شبکه‌ی پتری جریان کار  $PN_3$  (یعنی مجموعه‌ی  $T_3$ ) موارد زیر برقرار خواهد بود.

$$T_3 = (T_1 - \{t^+\}) \cup T_2 \quad (41-4)$$

در نهایت، در خصوص کمان‌های جریانی شبکه‌ی پتری جریان کار  $PN_3$  (یعنی مجموعه‌ی  $F_3$ ) موارد زیر برقرار خواهد بود:

$$F_3 = \{(x, y) \in F_1 | x \neq t^+ \wedge y \neq t^+\} \quad (42-4)$$

$$\cup \{(x, y) \in F_2 | \{x, y\} \cap \{i, o\} = \emptyset\} \quad (43-4)$$

$$\cup \{(x, y) \in P_1 \times T_2 | (x, t^+) \in F_1 \wedge (i, y) \in F_2\} = \emptyset \quad (44-4)$$

$$\cup \{(x, y) \in T_2 \times P_1 | (t^+, y) \in F_1 \wedge (x, o) \in F_2\} = \emptyset \quad (45-4)$$

در نهایت، در خصوص این جانشینی، یک قضیه‌ی دو شرطی توسط نگارندگان کتاب مدیریت جریان‌های کاری [۳] ارائه شده‌است:

**قضیه ۲-۴-۲۲.** شبکه‌های پتری جریان کاری  $PN_1$  و  $PN_2$  امن و صحیح است، اگر و تنها اگر شبکه‌ی پتری جریان کاری  $PN_3$  امن و صحیح باشد.

قضیه‌ی ۲-۴-۲۲ بر این ادعا که می‌توان بجای انتقال‌ها، ساختارهای بلوکی صحیح را قرار داد و این کار، خللی به صحت وارد نمی‌کند، صحه می‌گذارد. اگر بتوان این قضیه را به مدل‌های فرآیندی BPMN 2.0 تعمیم داد، امکان اعمال این روش بر روی مدل‌های فرآیندی BPMN 2.0 نیز ممکن خواهد بود. قضیه‌ی ۲-۴-۲۲ بیان می‌دارد که اگر در یک شبکه‌ی پتری جریان کار، به جای هر انتقال، یک شبکه‌ی پتری جریان کار صحیح قرار گیرد، شبکه‌ی حاصل نیز صحیح خواهد بود. با توجه به جدول ۴-۵ و

نگاشت ۴-۳۵ به ازای هر انتقال وظیفه‌ای در یک شبکه‌ی پتری جریان کار، یک فعالیت در مدل فرآیندی BPMN 2.0 معادلش وجود خواهد داشت. از سوی دیگر، مفهوم صحت را در تعریف ۴-۲-۱۹ و بر اساس قضایای ۴-۲-۱۷ و ۴-۲-۱۴ به مدل‌های فرآیندی BPMN 2.0 تعمیم دادیم. در ادامه تلاش خواهیم کرد تا بر اساس این موارد، قضیه‌ی ۴-۲-۲۲ را نیز به مدل‌های فرآیندی BPMN 2.0 تعمیم دهیم. فرض کنید شبکه‌ی جریان کاری  $(P_1, T_1, F_1)$  و  $(P_2, T_2, F_2)$  دو شبکه‌ی جریان کاری، با مکان‌های شروع  $i$  و مکان‌های پایانی  $o$  باشند که روابط زیر در خصوص آن‌ها برقرار باشد:

$$T_1 \cap T_2 = \emptyset \quad (46-4)$$

$$P_1 \cap P_2 = \{i, o\} \quad (47-4)$$

بنابراین، بر طبق نگاشت ۴-۳۵ و بر مبنای تعریف ۴-۲-۵ دو مدل فرآیندی BPMN 2.0 مثل  $b_1 = (B_1, E_1)$  و  $b_2 = (B_2, E_2)$  وجود خواهد داشت که با این دو شبکه‌ی پتری جریان کار، معادل هستند؛ همچنین، انتقالی مثل  $t^+$  را در نظر می‌گیریم، به طوری که:  $t^+ \in T_1$ ؛ در ادامه، فرض کنید که شبکه‌ی پتری جریان کار  $(P_3, T_3, F_3)$  در اثر جایگذاری شبکه‌ی پتری جریان کار  $PN_2$  به جای انتقال  $t^+$  در شبکه‌ی پتری  $PN_1$  بدست می‌آید. در این صورت، موارد زیر در خصوص مکان‌های این شبکه‌ی پتری جریان کار (یعنی مجموعه‌ی  $P_3$ ) برقرار خواهد بود.

$$P_3 = P_1 \cup P_2 \quad (48-4)$$

به ازای انتقال  $t^+$  در شبکه‌ی پتری  $PN_1$ ، بر طبق نگاشت ۴-۳۵ یک فعالیت مانند  $a^+$  در مدل فرآیندی BPMN 2.0 معادل آن، یعنی  $b_1$  وجود خواهد داشت، به گونه‌ای که  $a^+ \in B_1$ . همچنین، فرض کنید که مدل فرآیندی BPMN 2.0 معادل با شبکه‌ی پتری جریان کار  $PN_3$  مدل  $(N_3, B_3)$  باشد. در خصوص انتقال‌های شبکه‌ی پتری جریان کار  $PN_3$  (یعنی مجموعه‌ی  $T_3$ ) موارد زیر برقرار خواهد بود.

$$T_3 = (T_1 - \{t^+\}) \cup T_2 \quad (49-4)$$

در نهایت، در خصوص کمان‌های جریانی شبکه‌ی پتری جریان کار  $PN_3$  (یعنی مجموعه‌ی  $F_3$ ) موارد

زیر برقرار خواهد بود:

$$F_3 = \{(x, y) \in F_1 | x \neq t^+ \wedge y \neq t^+\} \quad (50-4)$$

$$\cup \{(x, y) \in F_2 | \{x, y\} \cap \{i, o\} = \emptyset\} \quad (51-4)$$

$$\cup \{(x, y) \in P_1 \times T_2 | (x, t^+) \in F_1 \wedge (i, y) \in F_2\} = \emptyset \quad (52-4)$$

$$\cup \{(x, y) \in T_2 \times P_1 | (t^+, y) \in F_1 \wedge (x, o) \in F_2\} = \emptyset \quad (53-4)$$

با فرض اجرای پیش‌پردازش‌های مورد نیاز، پیش از انتقال هر کدام از مدل‌ها به مدل مقصد، موارد بالا در مدل فرآیندی 2.0 BPMN معادل با شبکه‌ی پتری جریان کار  $PN_3$  بر اساس نگاشت ۳۵-۴ قابل بازنمایی خواهد بود. همان‌طور که فرض شد،  $b_3$  مدل فرآیندی 2.0 BPMN معادل با شبکه‌ی پتری جریان کار  $PN_3$  می‌باشد. جایگذاری شبکه‌ی پتری جریان کار  $PN_2$  به جای انتقال  $t^+$  بر طبق نگاشت ۳۵-۴ معادل است با جایگذاری مدل فرآیندی  $b_2$  به جای فعالیت  $a^+ \in B_1$  در مدل فرآیندی  $b_1$  و در نتیجه، مدل فرآیندی  $b_3$  به وجود خواهد آمد؛ در نتیجه، موارد زیر در خصوص این مدل فرآیندی 2.0 برقرار خواهد بود:

$$B_3 = (B_1 - \{a^+\}) \cup B_2 \quad (54-4)$$

و در خصوص کمان‌ها، موارد زیر برقرار خواهد بود ( $i$  رخداد شروع و  $o$  رخداد پایانی است):

$$E_3 = \{(x, y) \in E_1 | x \neq a^+ \wedge y \neq a^+\} \quad (55-4)$$

$$\cup \{(x, y) \in E_2 | \{x, y\} \cap \{i, o\} = \emptyset\} \quad (56-4)$$

$$\cup \{(x, y) \in E_1 \times E_2 | (x, a^+) \in B_1 \wedge (i, y) \in B_2\} = \emptyset \quad (57-4)$$

$$\cup \{(x, y) \in B_2 \times E_1 | (a^+, y) \in B_1 \wedge (x, o) \in B_2\} = \emptyset \quad (58-4)$$

بنابراین با تکیه بر نگاشتهای ارائه شده در روابط ۴-۳۶ و ۴-۳۵، شرایط قضیه‌ی ۴-۲۲ را بر اساس مدل‌های فرآیندی 2.0 BPMN معادل با شبکه‌های پتری جریان کار  $PN_1$ ،  $PN_2$  و  $PN_3$  بازگو کردیم. در نهایت، قضیه‌ی ۴-۲۲ را بر اساس این موارد، بازگو و اثبات می‌کنیم.

**قضیه ۴-۲-۳.** فرض کنید  $(B_1, E_1)$  و  $(B_2, E_2)$  دو مدل فرآیندی  $BPMN 2.0$  باشند. این دو مدل فرآیندی صحیح هستند، اگر و تنها اگر  $b_3$  صحیح باشد.

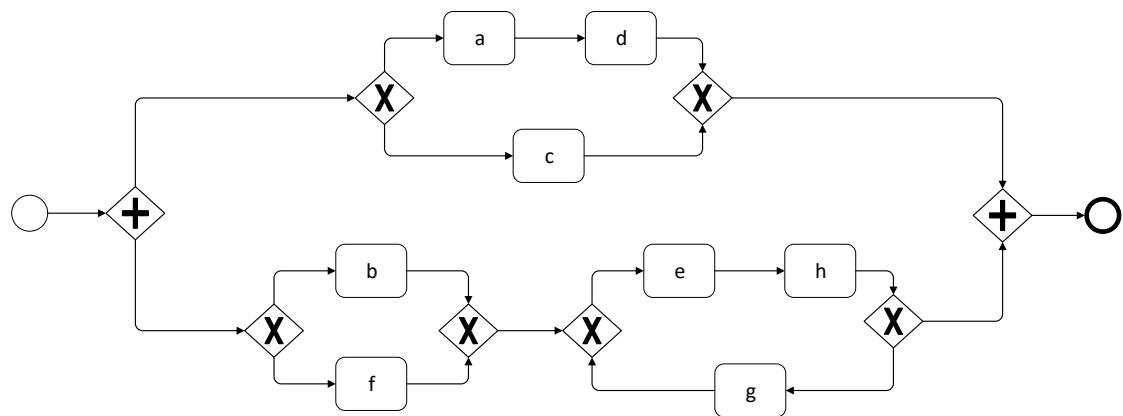
برهان. فرض کنید که دو شبکه‌ی پتری جریان کاری  $(T_1, P_1, F_1)$  و  $PN_1 = (T_1, P_1, F_1)$  و  $PN_2 = (T_1, P_1, F_1)$  شبکه‌های پتری معادل با دو مدل فرآیندی  $BPMN 2.0$  ،  $b_1$  و  $b_2$  باشند که با استفاده از نگاشت ۴-۳۶ بdst آمدند. همچنین فرض کنید که از طریق این نگاشت، شبکه‌ی پتری جریان کار معادل با  $b_3$ ، یعنی شبکه‌ی پتری  $PN_3$  بdst آمدند.

بر طبق مفروضات گفته شده، مدل فرآیندی  $b_2$  به جای فعالیت  $a^+ \in B_1$  جایگذاری شده‌است و بر طبق مفاهیم گفته شده، این امر معادل است با جایگذاری شبکه‌ی پتری جریان کار  $PN_2$  به جای انتقال  $t^+$  به گونه‌ای که این انتقال، تبدیل یافته‌ی فعالیت  $a^+$  تحت نگاشت ۴-۳۶ می‌باشد.

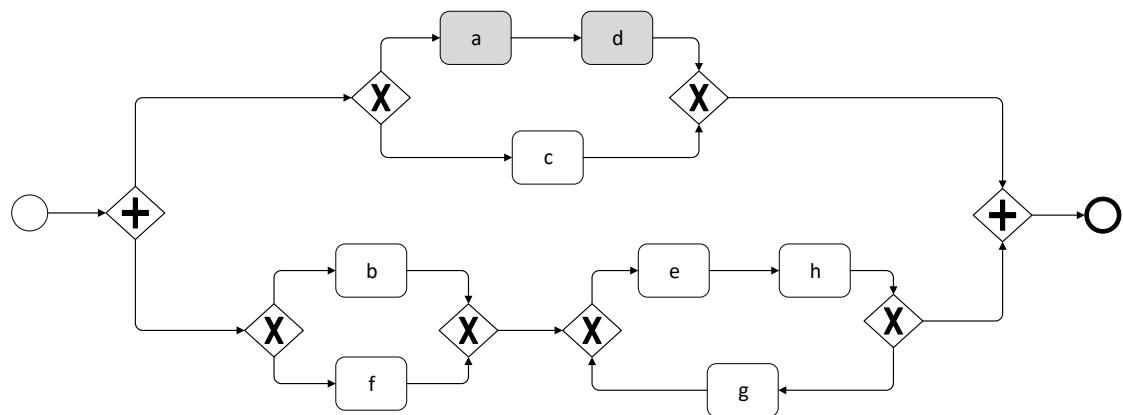
به عنوان برهان خلف، فرض کنید که  $b_1$  و  $b_2$  صحیح باشند، در حالی که  $b_3$  صحیح نباشد. در نتیجه، بر اساس قضیه ۴-۲-۲۰، دو شبکه‌ی پتری جریان کار  $PN_1$  و  $PN_2$  صحیح نخواهند بود و در نتیجه، بر طبق قضیه‌ی صحت، شبکه‌ی پتری  $PN_3$  نیز صحیح نخواهد بود و در نتیجه، مدل فرآیندی  $BPMN 2.0$  معادل با آن، یعنی مدل فرآیندی  $b_3$  نیز صحیح نخواهد بود که با فرض خلف، در تناقض خواهد بود. از سوی دیگر، فرض کنید که مدل فرآیندی  $b_3$  صحیح باشد و دو مدل فرآیندی  $b_1$  و  $b_2$  صحیح نباشند؛ در نتیجه، شبکه‌ی پتری جریان کار  $PN_3$  صحیح نخواهد بود، در حالی که دو مدل فرآیندی  $PN_1$  و  $PN_2$  صحیح خواهند بود و این امر، با قضیه‌ی صحت (قضیه‌ی ۴-۲-۲-۲۰) در تناقض است. در نتیجه فرض خلف باطل و حکم برقرار خواهد بود.  $\square$

**نتیجه ۴-۲-۴.** بر اساس این قضیه، می‌توان زیرفرآیندهایی صحیح را با یک فعالیت جایگزین کرد، بی‌آن که صحت مدل فرآیندی  $BPMN 2.0$  به مخاطره بیفت.

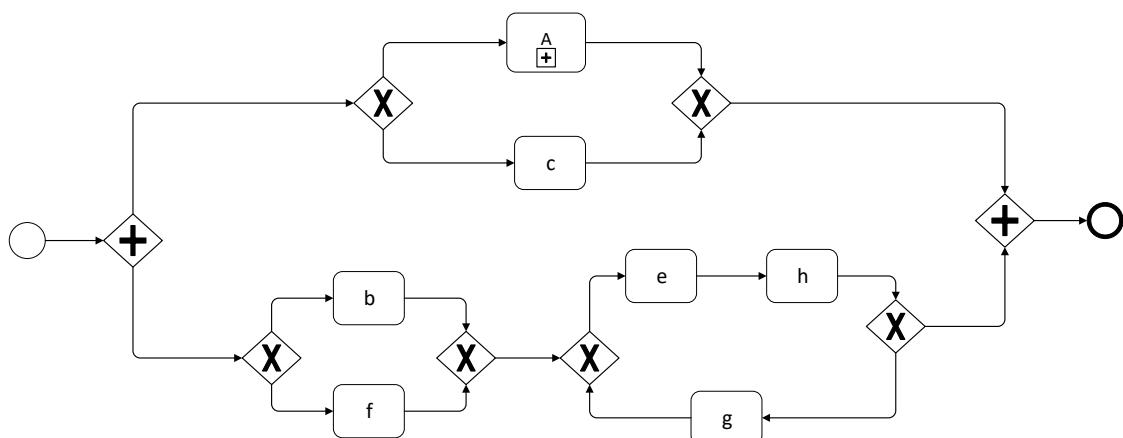
**نمونه ۴-۲-۵.** فرض کنید مدل فرآیندی  $BPMN 2.0$  در شکل ۴-۴۷ داده شده‌است و قرار است، تحلیل شود. در ابتدا، مشاهده می‌شود که فعالیت‌های  $a$  و  $d$  از ساختار بلوکی دنباله‌ای تبعیت می‌کنند. با توجه به این که این ساختار بلوکی صحیح است، بر طبق قضیه‌ی ۴-۲-۲-۳ می‌توان این زیرفرآیند را از مدل فرآیندی حذف کرد و به جای آن، یک فعالیت درنظر گرفت. نظر به این که این ساختار بلوکی، یک زیرفرآیند است می‌توان آن را با یک زیرفرآیند جانشین کرد که این کار بر اساس [۴] بلامانع است و بر اساس این پژوهش، در تحلیل ساختاری، زیرفرآیندها و فعالیت‌ها هم‌ارز هستند. بنابراین، این زیرفرآیند را حذف کرده و با زیرفرآیند  $A$  جانشین می‌کنیم و شکل ۴-۴۸ بدست می‌آید.



شکل ۴-۴: یک مدل فرآیندی BPMN 2.0 به منظور اجرای تحلیل ایستا

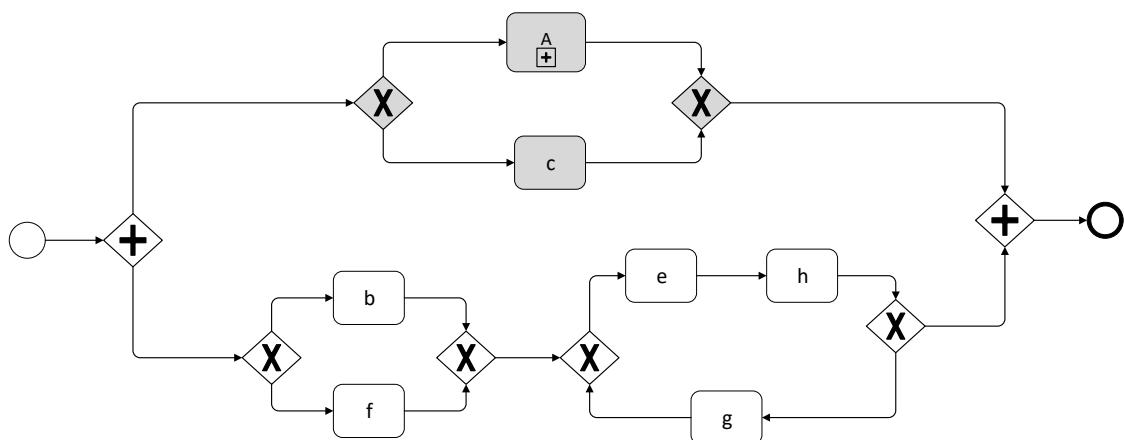


شکل ۴-۵: کشف ساختار بلوکی دنباله‌ای

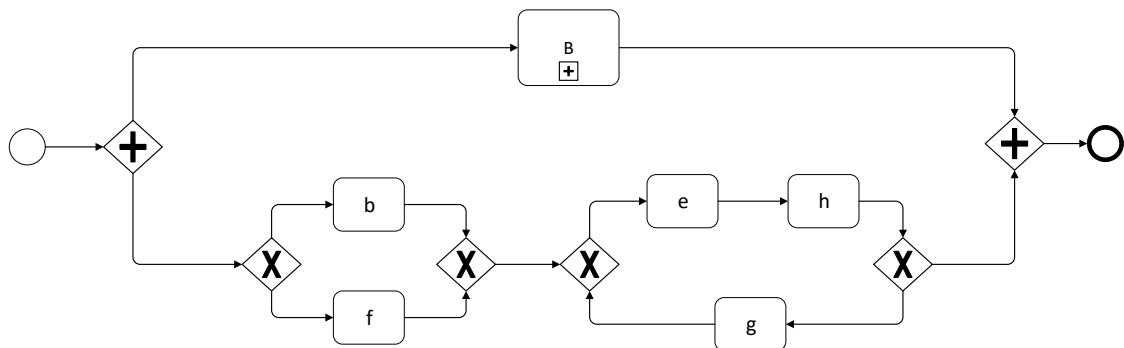


شکل ۴-۶: یک مدل فرآیندی BPMN 2.0 به منظور اجرای تحلیل ایستا

در ادامه مشاهده می‌شود که فعالیت  $c$  و زیرفرآیند  $A$ ، به همراه دروازه‌های انحصاری آن‌ها، از یکی دیگر از سازه‌های بلوکی صحت، تبعیت می‌کنند و بر آن منطبق هستند (شکل ۴-۴۹) بنابراین بر طبق قضیه‌ی ۲۳-۲-۴ می‌توان آن را با یک فعالیت جانشین کرد؛ بنابراین، کل این زیرفرآیند، در قالب یک زیرفرآیند، به نام زیرفرآیند  $B$  بازنمایی می‌شود و در نتیجه، شکل ۴-۵۰ بدست می‌آید.

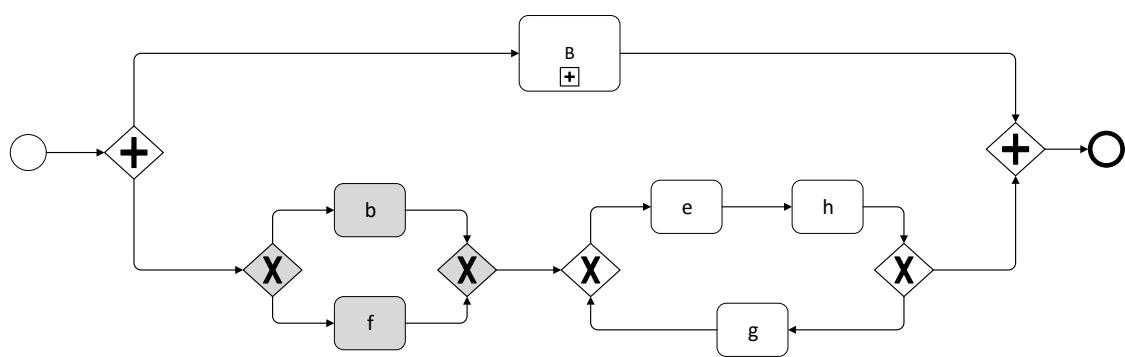


شکل ۴-۴۹: کشف سازه‌ی بلوکی انشعاب انحصاری

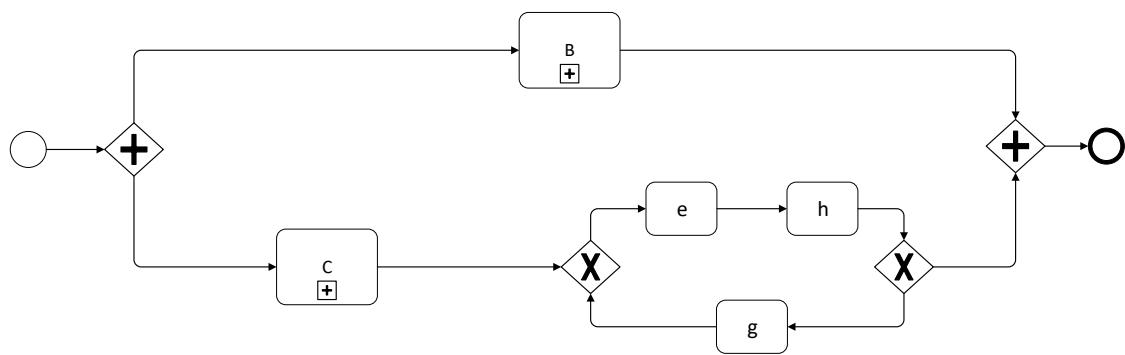


شکل ۴-۵۰: بازنمایی الگوی کشیده در شکل ۴-۴۹ در قالب یک زیرفرآیند

در ادامه تحلیل، مشاهده می‌شود که دو فعالیت  $b$  و  $f$ ، به همراه دروازه‌های انحصاری آن‌ها، تشکیل یک ساختار بلوکی انشعاب انحصاری را می‌دهند (شکل ۴-۵۱)؛ با توجه به این‌که این ساختار بلوکی، ساختاری صحیح است، بر طبق قضیه‌ی ۲۳-۲-۴ می‌توان آن را با یک فعالیت جانشین کرد؛ بی‌آن‌که صحت مدل فرآیندی اصلی، تغییر کند. بنابراین با جایگزین کردن آن با زیرفرآیند  $C$ ، به شکل ۴-۵۲ می‌رسیم.

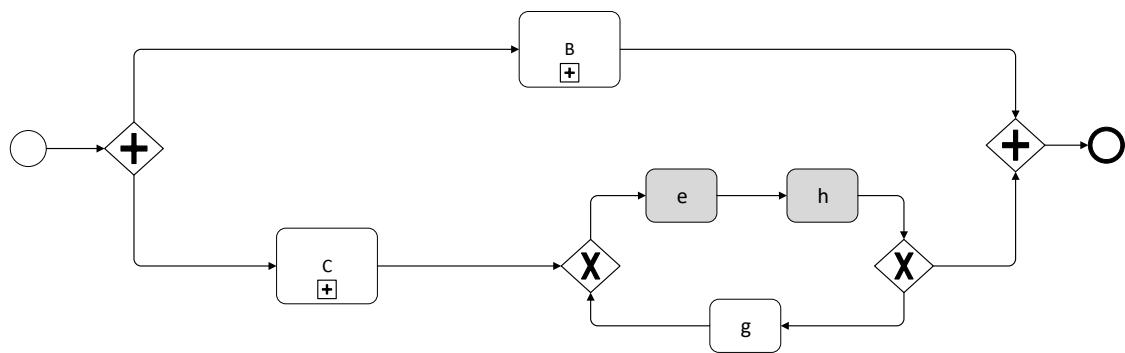


شکل ۴-۵۱: کشف سازه‌ی بلوکی صحت انشعبانه در شکل ۴-۵۰

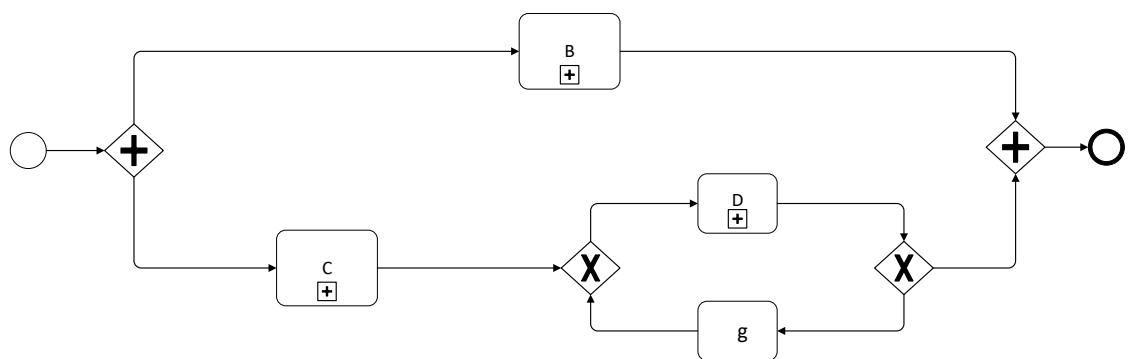


شکل ۴-۵۲: بازنمایی الگوی کشف شده در شکل ۴-۵۱ در قالب یک زیرفرآیند

در ادامه‌ی تحلیل، می‌توان مشاهده نمود که دو فعالیت  $e$  و  $h$  با یکدیگر ساختار بلوکی دنباله‌ای را تشکیل می‌دهند (شکل ۵۳-۴)؛ بنابراین می‌توان بر طبق قضیه‌ی ۴-۲-۲۳ آن‌ها را با یک فعالیت جانشین کرد؛ بنابراین، با زیرفرآیند  $D$  جانشین می‌شود و تصویر (۴-۵۴) بدست می‌آید.

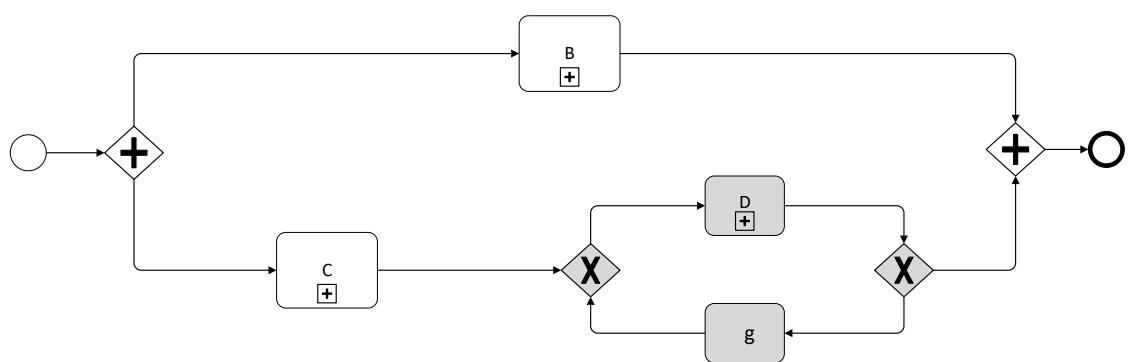


شکل ۴-۵۳: کشف بلوک ساختاری دنباله‌ای

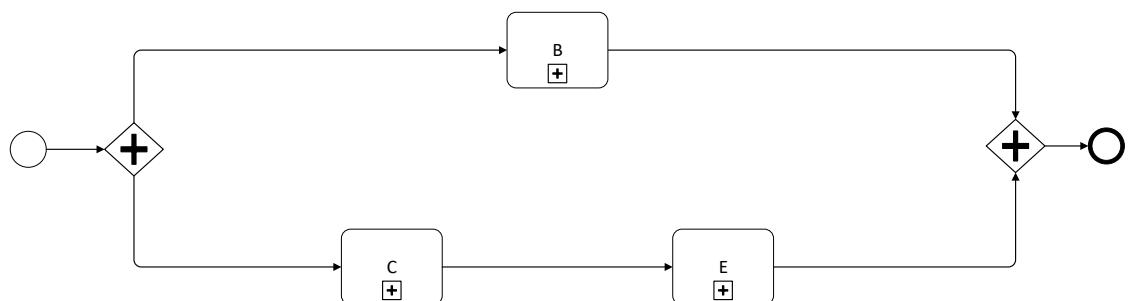


شکل ۴-۵۴: بازنمایی الگوی کشف شده در شکل ۴-۵۳ در قالب یک زیرفرآیند

در ادامه تحلیل، مشاهده می‌شود طبق شکل ۴-۵۵ که زیرفرآیند  $D$  و فعالیت  $g$ ، به همراه دروازه‌های انحصاری پیرامونشان، از الگوی تکراری تبعیت می‌کنند؛ بنابراین می‌توان آن را با یک فعالیت، جایگزین نمود؛ بنابراین، با زیرفرآیند  $E$  جایگزین می‌شود و شکل ۴-۵۶ بدست می‌آید.



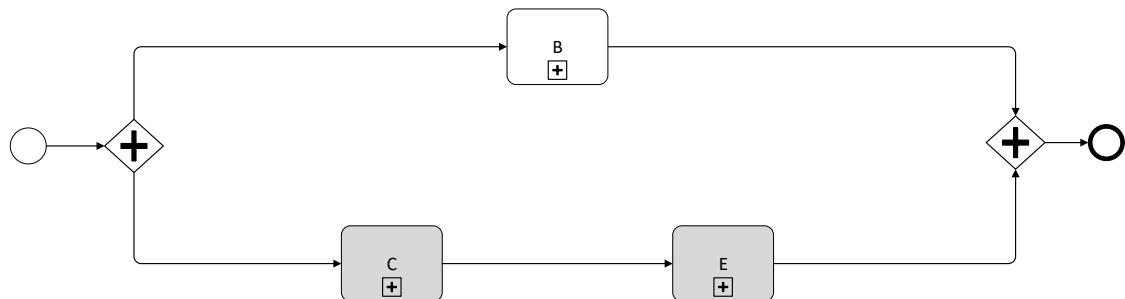
شکل ۴-۵۵: کشف الگوی تکراری



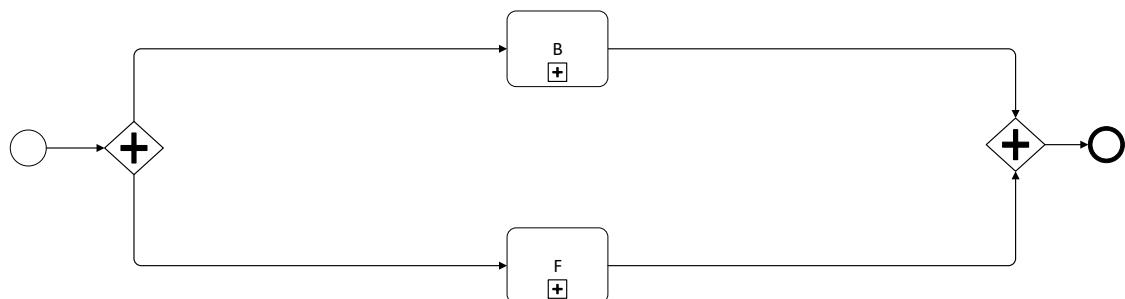
شکل ۴-۵۶: بازنمایی الگوی کشف شده در شکل ۴-۵۵ در قالب یک زیرفرآیند

در ادامه تحلیل، مشاهده می‌شود که دو زیرفرآیند  $E$  و  $C$ ، از الگوی صحت دنباله‌ای تبعیت

می‌کنند (شکل ۵۷-۴). بنابراین می‌توان بر طبق قضیه‌ی ۲۳-۲-۴ آن‌ها را با یک فعالیت جایگزین کرد؛ در نتیجه، آن‌ها را با زیرفرآیند  $F$  جایگزین می‌کنیم و به شکل ۵۸-۴ می‌رسیم.

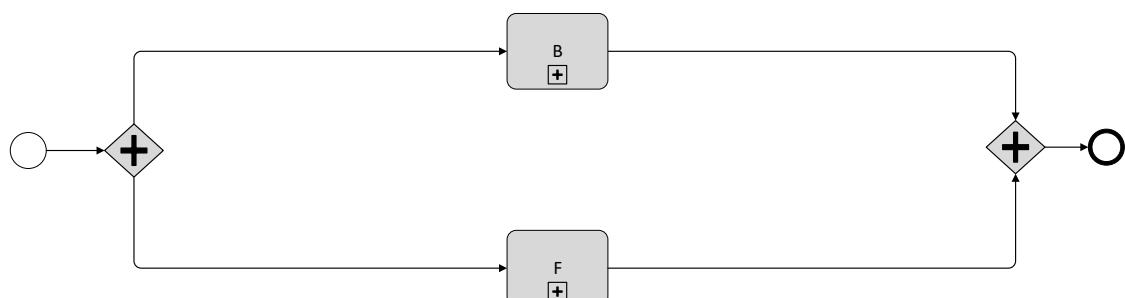


شکل ۵۷-۴: کشف الگوی دنباله‌ای در فرآیند



شکل ۵۸-۴: بازنمایی الگوی کشف شده در شکل ۵۶-۴ در قالب یک زیرفرآیند

در نهایت، مشاهده می‌شود که دو زیرفرآیند  $B$  و  $F$  از الگوی صحت انشعاب عطفی پیروی می‌کنند (شکل ۵۹-۴). بنابراین می‌توان طبق قضیه‌ی ۲۳-۲-۴ آن را با یک فعالیت، جایگزین نمود. در نتیجه، آن را با زیرفرآیند  $G$  جانشین می‌کنیم و شکل ۶۰-۴ بدست می‌آید.



شکل ۵۹-۴: کشف الگوی صحت انشعاب انحصاری در مدل فرآیندی



شکل ۴-۶: بازنمایی الگوی کشف شده در شکل ۵۹-۴ در قالب یک زیرفرآیند

در نهایت، شکل ۶۰-۴ نشان‌دهنده‌ی یک الگوی صحت پایه‌ای می‌باشد و فرآیند نشان داده شده در این شکل نیز صحیح است؛ درنتیجه، در این مثال موفق شدیم که یک فرآیند را با مجموعه‌ای از ساختارهای بلوکی صحت، بازنمایی کنیم. با توجه به این‌که این فرآیند با مجموعه‌ای از ساختارهای بلوکی صحت ساخته شده‌است، نتیجه حاصل می‌شود که بر طبق قضیه‌ی ۲۳-۲-۴ این فرآیند، فرآیندی صحیح است و بر طبق تعریف صحت در مدل‌های فرآیندی BPMN 2.0 (تعریف ۱۹-۲-۴) این مدل فرآیندی BPMN 2.0 فاقد بن‌بست و بن‌بست حلقوی است.

### محدودیت روش و ضرورت کشف پادالگوهای صحت به عنوان راهکار

در الگوریتم اصلی (در شبکه‌های پتری جریان کار)، عدم تطابق بخش‌هایی از فرآیند با الگوهای صحت (ساختارهای بلوکی صحت)، به معنای عدم صحت آن فرآیند نمی‌باشد؛ در نتیجه در الگوریتم تعمیم یافته‌ی آن نیز، این محدودیت برقرار است؛ این بدان معنا خواهد بود که عدم تطابق بخش‌هایی از مدل فرآیندی BPMN 2.0 با الگوهای صحت، به معنای عدم صحت مدل فرآیندی نمی‌باشد؛ اما در صورتی که تمام مدل فرآیندی با این الگوها مطابق باشد، صحت مدل فرآیندی قطعی خواهد بود. در غیر این صورت، صحت فرآیند اصلی، منوط به صحت بخش‌هایی از فرآیند خواهد بود که با این الگوها مطابقت ندارند. برای آن‌که بتوان یک پیش‌بینی مناسب از صحت این بخش‌ها ارائه داد، می‌توانیم در این بخش‌ها، به دنبال پادالگوهای صحت باشیم و بر مبنای آن، پیش‌بینی‌هایی مناسب در خصوص صحت این بخش‌ها داشته باشیم.

## ۳-۴ داده‌ساختار و الگوریتم پیشنهادی

در بخش‌های قبل، به الگوها و پادالگوهای صحت در شبکه‌های پتری جریان کار پرداخته شد و به مدل‌های فرآیندی BPMN 2.0 تعمیم داده شد. از روشی که به عنوان روش جایگذاری در بخش ۶-۲-۴ مورد بحث قرار گرفت، در کتاب [۳] به عنوان روش صحتسنجی، بدون نیاز به رایانه یاد شده‌است. این روش، جزو روش‌هایی است که می‌تواند فرآیند را فارغ از مفاهیم موجود در آن و بدون نیاز به اجرای آن، مورد

تحلیل قرار دهد و در بخش ۳-۲-۴ به مدل‌های فرآیندی BPMN2.0 تعمیم داده شد. در ادامه، قصد داریم الگوریتمی را به منظور اجرای این روش تعمیم داده شده، ارائه کنیم. حال سوالاتی در ادامه و در این بخش باید پاسخ داده شود:

۱. برای تحلیل ایستای یک مدل فرآیندی 2.0 BPMN چه اطلاعاتی مورد نیاز است؟
۲. اگر قرار باشد این روش، توسط رایانه اجرا شود، چه داده‌ساختاری برای بازنمایی اطلاعات مورد نیاز، مناسب است؟
۳. آیا می‌توان الگوریتمی را به منظور تحلیل ایستای یک مدل فرآیندی 2.0 BPMN و با استفاده از داده‌ساختار ارائه شده، ارائه کرد؟

در ادامه، در جهت پاسخ‌گویی به سوالات فوق، داده‌ساختاری به منظور اجرای الگوریتم صحت‌سنجدی ارائه شده در قضیه‌ی ۴-۲-۲۳ ارائه می‌شود و سپس، روشی را به منظور کشف الگوهای پادالگوهای صحت و ارائه شده در بخش‌های ۴-۲-۴ و ۵-۲-۴، پیشنهاد خواهد شد.

### ۱-۳-۴ داده‌ساختار گراف دروازه‌ای

در ابتداء، لازم به یادآوری است که در پژوهش حاضر، به دنبال تحلیل صحت ساختاری<sup>۱۶</sup> هستیم و قصد بر آن است که بر مبنای ساختار کنترلی یک مدل فرآیندی BPMN 2.0 در خصوص صحت ساختاری آن تصمیم‌گیری شود. یادآوری می‌شود که بر طبق تعریف ۲-۸-۱ یک مدل فرآیندی، فارغ از زبان مدل‌سازی مورد استفاده در ترسیم آن، یک گراف است که حاوی تعدادی گره و تعدادی یال است. در این تعریف مشاهده شد که گره‌های هر مدل فرآیندی را می‌توان به سه زیرمجموعه‌ی مجزا، تقسیم‌بندی نمود:

۱. مجموعه‌ی مدل‌های فعالیتی

۲. مجموعه‌ی مدل‌های دروازه‌ای

۳. مجموعه‌ی مدل‌های رخدادی

با توجه به الگوهای صحت ارائه شده در بخش ۴-۲-۴ می‌توان متوجه شد که آن‌چه که بر روی صحت ساختاری مدل‌های فرآیندی موثر است، انشعاب‌ها و اتصال‌ها می‌باشد و با توجه به پادالگوهای ارائه شده

<sup>16</sup>Structural Soundness

در بخش ۴-۲-۵ می‌توان دریافت که استفاده‌ی ناصحیح و نابجا از مدل‌های دروازه‌ای، می‌تواند صحت فرآیندهای کاری را با مشکل مواجه سازد.

**نتیجه ۴-۳-۱.** در یک مدل فرآیندی، مدل‌های دروازه‌ای بر روی صحت و یا عدم صحت ساختاری یک فرآیند موثر هستند و مدل‌های رخدادی و فعالیتی، بر روی صحت ساختاری یک فرآیند، موثر نیستند.

پژوهش [۲۸] بر این ادعا صحه می‌گذارد. در این پژوهش، با رویکرد احتمالاتی به صحت‌سنجی مدل‌های فرآیندی BPMN 2.0 پرداخته شده‌است. در این پژوهش، با نظر به این امر که احتمال ورود به یک فعالیت (ویا یک رخداد) برابر است با احتمال خارج شدن جریان کنترلی از آن فعالیت (و یا یک رخداد) و با تکیه بر احتمال ضربی، وجود و یا عدم وجود فعالیت‌ها را در هر کدام از شاخه‌های یک مدل فرآیندی، بر صحت ساختاری فرآیند موثر نیست و صحت فرآیند، بر ترکیب مدل‌های انشعابی در فرآیندها، متکی است.

بر اساس نتیجه ۴-۳-۱ بهترین داده ساختاری که می‌تواند در تحلیل ایستای یک مدل فرآیندی BPMN 2.0 موثر باشد، داده ساختاری است که ارتباطات بین مدل‌های دروازه‌ای یک مدل فرآیندی را بازنمایی کند. داده ساختار پیشنهادی این پژوهش، گرافی است که اتصال بین دروازه‌های یک مدل فرآیندی BPMN 2.0 را نشان می‌دهد. برای استخراج این گراف از مدل فرآیندی BPMN 2.0 مراحل زیر طی گردد:

- در ابتدا، مدل فرآیندی BPMN 2.0 مورد پیش‌پردازش قرار گیرد تا هم خوش‌تعریف شده و هم ساده‌سازی گردد (ساده‌سازی بر اساس بخش ۱-۲-۴ انجام شود).

- تمامی فعالیت‌ها و رخدادها (بجز رخداد شروع و رخداد پایانی) از مدل فرآیندی حذف شود.

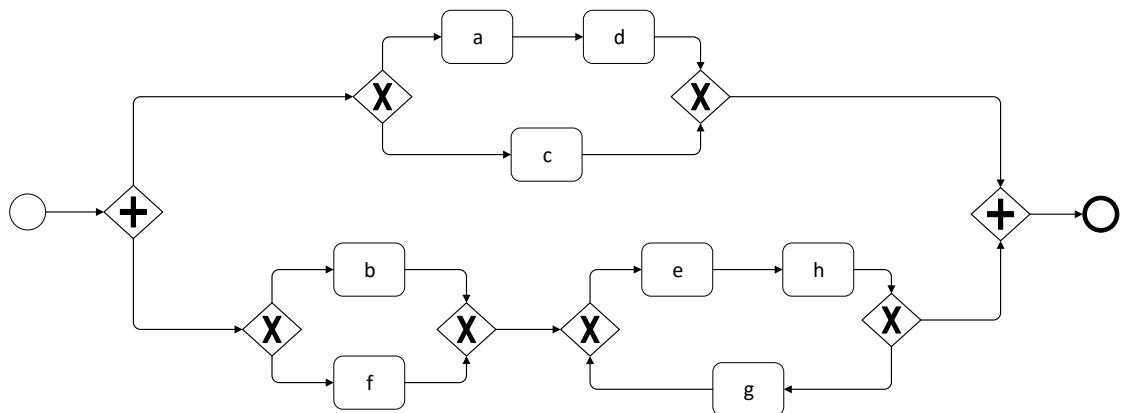
با طی دو مرحله‌ی بالا، گرافی از مدل فرآیندی BPMN 2.0 به وجود می‌آید که صرفاً حاوی رخداد شروع و پایان و نیز دروازه‌ها می‌باشد. بنابراین، یک گراف دروازه‌ای به صورت زیر تعریف می‌شود:

**تعريف ۴-۳-۲.** یک گراف دروازه‌ای مانند  $G = (V_p, E_p)$  برای یک مدل فرآیندی BPMN 2.0 مثل  $p$ ، گرافی چندگانه و جهت‌دار است، به گونه‌ای که داشته باشیم:

$$V_p = B_G \cup \{i, o\} \subset B \quad (۵۹-۴)$$

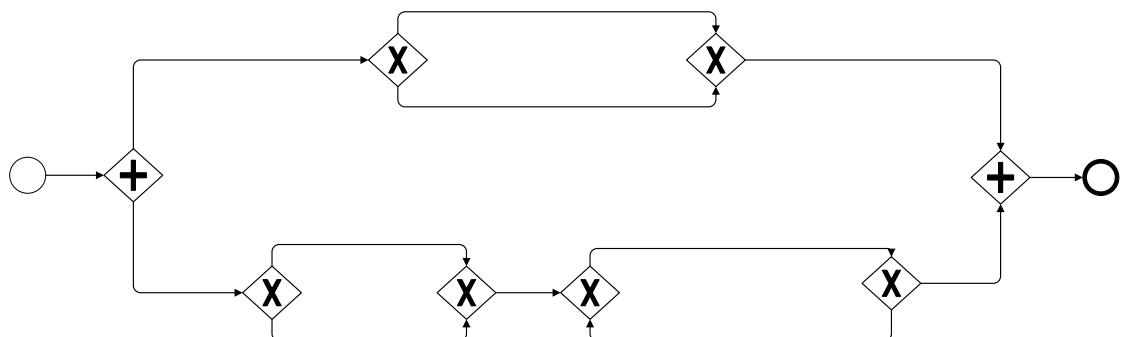
در رابطه‌ی قبل،  $B_G$  مجموعه‌ی گره‌های مدل‌های دروازه‌ای و  $B_E$  مجموعه گره‌های رخدادی است و  $i \in B_E$  رخداد شروع و  $o \in B_E$  رخداد پایانی مدل فرآیندی  $p$  می‌باشد.

نمونه ۴-۳-۳. به منظور درک بهتر تعریف، گراف دروازه‌ای از فرآیند مطرح شده در مثال ۲۵-۲-۴ در ادامه بدست می‌آید. مدل فرآیندی ارائه شده در تصویر ۶۱-۴ یک مدل فرآیندی خوش‌تعریف بوده و به



شکل ۴-۶۱: یک مدل فرآیندی BPMN 2.0 به منظور استخراج گراف دروازه‌ای از آن

ساده‌سازی نیاز ندارد. بنابراین، با حذف فعالیت‌های موجود در آن، گراف دروازه‌ای ترسیم شده در شکل ۶۲-۴ بدست می‌آید.



شکل ۶۲-۴: گراف دروازه‌ای مستخرج از مدل فرآیندی ترسیم شده در شکل ۶۱-۴

### الگوهای صحت از گراف دروازه‌ای

داده‌ساختار گراف دروازه‌ای کمک می‌کند تا صرفا شاکله‌ی جریان کاری یک مدل فرآیندی BPMN 2.0 بدست آید و تحلیل ایستا را از اجزایی که بر جریان فرآیند بی‌تأثیر هستند، مستقل می‌سازد. در ادامه، قصد داریم که الگوهای پاد الگوهای صحت را با استفاده از این داده ساختار و بر مبنای قضیه‌ی ۲۳-۲-۴

پیدا کنیم. پیش از این کار، لازم است گراف دروازه‌ای معادل با ساختار الگوها و پادالگوهای صحت را بیابیم و در نهایت، راهکاری را به منظور کشف این الگوها از گراف دروازه‌ای، ارائه کنیم.

در ابتدا، لازم است الگوهای صحت را در داده‌ساختار گراف دروازه‌ای، بازنمایی کنیم. لازم است بدانیم که با استفاده از این الگوها، نیازی به کشف الگوهای پایه‌ای و دنباله‌ای خواهد بود. با توجه به الگوهای صحت، در هر کدام از الگوها، دو دروازه از دروازه‌های مدل‌های فرآیندی BPMN 2.0 مشارکت داشتند و در گراف دروازه‌ای معادل با آن‌ها، زیرفرآیندهای موجود در آن‌ها حذف می‌شود.

به منظور کشف این الگوها و پادالگوهای گراف دروازه‌ای، لازم است در ابتدا، این الگوها در قالب گراف‌های دروازه‌ای ترسیم شوند و سپس به صورت مرحله به مرحله، از گراف دروازه‌ای فرآیند کشف شده و از آن حذف شوند و در نهایت در گراف باقی مانده، پادالگوها جستجو شوند و بر مبنای این موارد، صحت مدل فرآیندی BPMN 2.0 بیان شود.

در رویکرد تحلیل دروازه‌ای، دو گام اساسی پیشرو خواهیم داشت:

۱. گام اساسی اول : کشف الگوهای صحت از گراف دروازه‌ای و کنار گذاشتن بخش‌های صحیح. چرا که بر اساس قضیه‌ی ۴-۲۳ با این کار، صحت باقی اجزای فرآیند، منوط به صحت اجزای باقی مانده خواهد بود. این کار تا جایی انجام می‌شود که الگوی صحت دیگری، در گراف دروازه‌ای یافت نشود.

۲. گام اساسی دوم : جستجو در گراف دروازه‌ای باقی مانده در پی یافتن پادالگوهای صحت.

در گام اساسی اول، یعنی یافتن و حذف الگوهای صحت از گراف دروازه‌ای، به دنبال سه الگوی کلی زیر می‌گردیم:

۱. الگوی صحت فصلی

۲. الگوی صحت عطفی

۳. الگوی صحت تکرار

و در گام اساسی دوم، به دنبال پادالگوهای زیر می‌گردیم:

۱. پادالگوی بن‌بست در راستای کشف موارد بن‌بست قطعی.

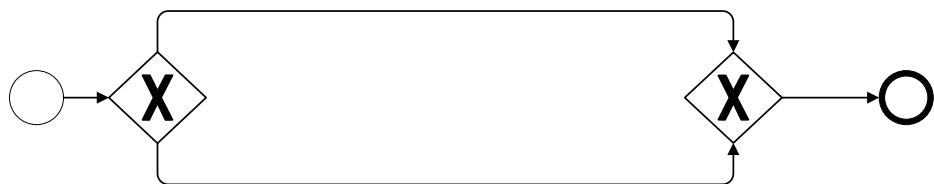
۲. پادالگوی بن‌بست در راستای کشف اجزایی که تحت حالاتی خاص ممکن است ایجاد بن‌بست کنند.

۳. پادالگوی بن‌بست حلقوی در راستای کشف مواردی که ممکن است به طور قطعی ایجاد بن‌بست حلقوی نمایند.

در ادامه، در خصوص هرکدام از الگوهای صحت در گراف‌های دروازه‌ای، بحث خواهد شد:

**الگوهای صحت فصلی** در گراف دروازه‌ای معادل با انواع الگوهای فصلی (اعم از الگوهای صريح و غير صريح) و نيز الگوي صحت انحصاری، يك يا چند شاخه‌ی منشعب شده از اين دروازه‌ها فعال شده و درنهایت، شاخه‌های فعال شده در يك دروازه‌ی اتصالی، با يك ديگر ادغام می‌شوند. در الگوهای فصلی، امكان فعال شدن بيش از يك شاخه وجود دارد و در الگوی انحصاری، صرفاً يكی از اين شاخه‌ها فعال شده و متصل می‌شود.

**نتیجه ۴-۳-۴.** زوج دروازه‌ی انشعابی/اتصالی که هرکدام از آن‌ها، دروازه‌های فصلی و يا انحصاری باشد، تشکیل يك الگوی صحت فصلی را می‌دهند؛ چرا که الگوی جريان کنترلي انحصاری، يك حالت خيلي خاصی از الگوی جريان کنترلي فصلی بوده که در آن، يكی از شاخه‌ها اجرا می‌شوند. تصویر ۴-۶۳ يك الگوی صحت فصلی را با دروازه‌های انشعابی و شکل ۴-۶۴ يك الگوی صحت فصلی را با دروازه‌های فصلی نشان می‌دهد. همچنان در صورتی که يكی از دروازه‌های انشعابی و يا اتصالی از نوع انحصاری بوده و ديگری از نوع فصلی باشد، بازهم اين الگوی صحت محقق خواهد شد.



شکل ۴-۶۳: يك الگوی صحت فصلی با دروازه‌های انحصاری

**الگوی صحت عطفی** وجود اين الگو در يك مدل فرآيندي BPMN 2.0 تضمین می‌کند که هم‌گام سازی شاخه‌های مدل‌های فرآيندي به درستی صورت می‌گيرد. در صورتی که سازه‌ی بلوکی صحت اين الگو را با استفاده از گراف دروازه‌ای نمايش دهيم، به شکل ۴-۶۵ می‌رسيم.



شکل ۴-۴: یک الگوی صحت فصلی با دروازه‌های فصلی



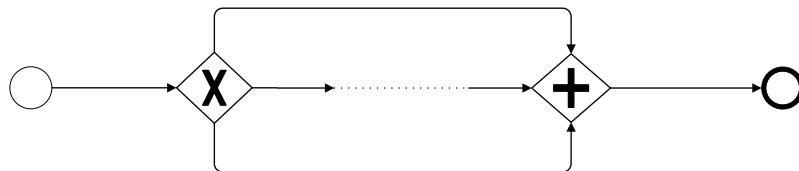
شکل ۴-۵: یک الگوی صحت عطفی با دروازه‌های عطفی

**الگوی صحت تکرار** با استخراج گراف دروازه‌ای از الگوی صحت تکراری در شکل ۲۴-۴، شکل ۴-۶ به وجود می‌آید.



شکل ۴-۶: یک الگوی صحت عطفی با دروازه‌های عطفی

**پادالگوی بنبست** همان‌طور که گفته شد، در دومین گام اساسی تحلیل ایستا و پس از کشف الگوهای صحت، در ادامه‌ی کار، پادالگوهای صحت کشف می‌شوند. یکی از اساسی‌ترین پادالگوهای صحت که در بخش ۴-۵ معرفی گشته، پادالگوی بنبست است که در شکل ۴-۷ نشان داده شده است. برای وقوع بنبست، دو وضعیت را می‌توان متصور شد:



شکل ۶۷-۴: پادالگوی بن‌بست در گراف دروازه‌ای

۱. در صورتی که حداقل دو ورودی یک اتصال عطفی، به دو خروجی از یک انشعاب انحصاری متصل باشد، وقوع بن‌بست در آن گرهی اتصال عطفی، امری قطعی و مسلم است؛ چرا که بر اساس الگوی جریان کنترلی انحصاری، حداقل شاخه‌ای وجود خواهد داشت که توسط انشعاب انحصاری، فعال نشود و در نتیجه، عمل هم‌گام‌سازی در آن اتصال عطفی، صورت نگرفته و بن‌بست رخ خواهد داد.

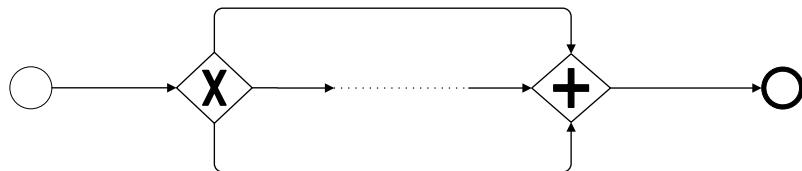
۲. در صورتی که حداقل یکی از ورودی‌های یک اتصال عطفی به یکی از خروجی‌های یک انشعاب فصلی و یا انشعاب‌های انحصاری (از هر انشعاب انحصاری حداکثر یک یال به اتصال عطفی متصل باشد)، احتمال وقوع بن‌بست در آن اتصال عطفی وجود خواهد داشت اما قطعی نیست؛ چرا که ممکن است شاخه‌های ورودی به آن، فعال شوند.

**پادالگوی بن‌بست حلقوی** پادالگوی دیگری که ممکن است صحت یک مدل فرآیندی BPMN 2.0 را تهدید کند، بن‌بست حلقوی است<sup>۱۷</sup>. با این پادالگو در بخش ۴-۲-۵ آشنا شدیم و معادل آن در زبان مدل‌سازی BPMN 2.0 در آن بخش ارائه شد. با استخراج گراف صحت دروازه‌ای از آن پادالگو، تصویر ۶۸-۴ بدست می‌آید.

#### ۴-۳-۲ صحت‌سنجی با روش تحلیل دروازه‌ای

به منظور صحت‌سنجی مدل فرآیندی BPMN 2.0 با استفاده از داده‌ساختار گراف دروازه‌ای، کافی است که روش جایگزینی تعمیم داده شده در قضیه‌ی ۴-۲-۲۳ را بر روی گراف دروازه‌ای اجرا نماییم و بعد از یافتن هر الگو و جانشینی یک زیرفرآیند به جای آن، بر طبق روش ساخت گراف دروازه‌ای، آن زیرفرآیند را حذف کنیم؛ درنهایت، آن قدر تحلیل را ادامه دهیم تا الگویی برای استخراج باقی نماند. در صورتی

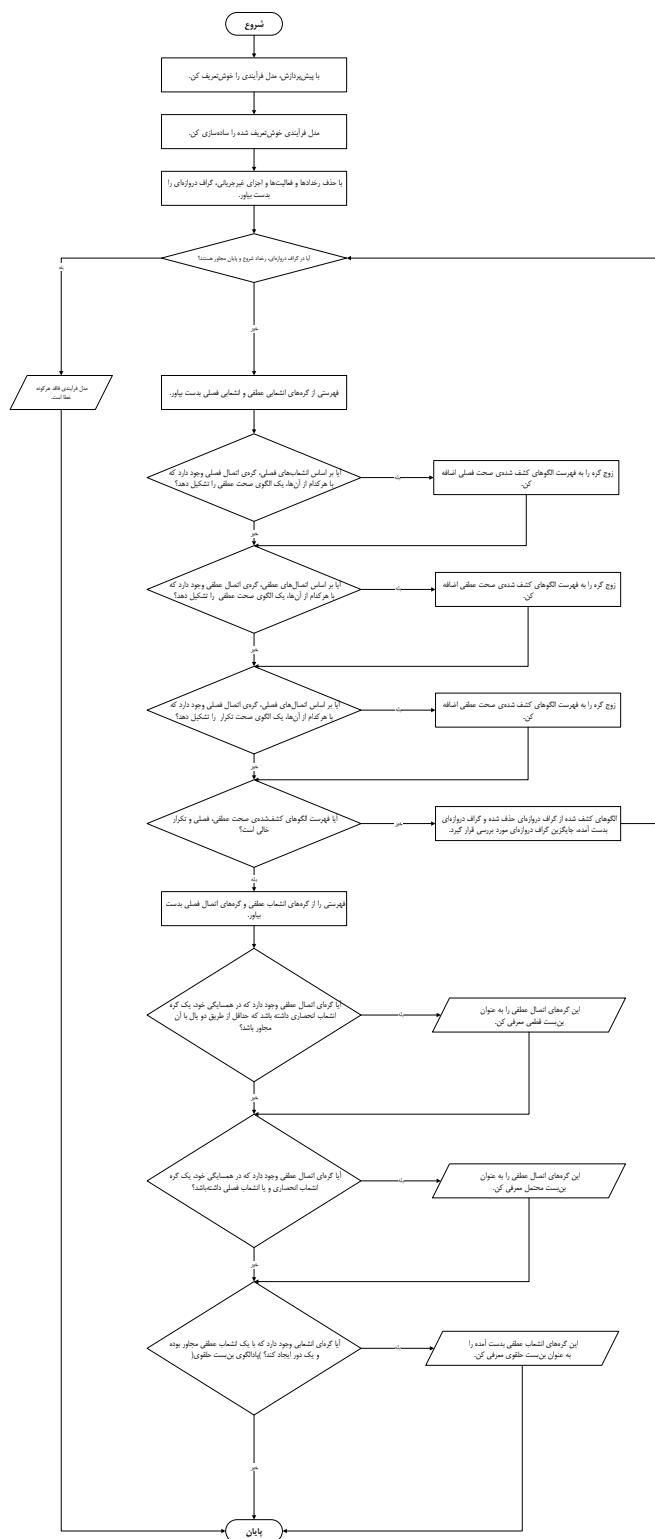
<sup>17</sup>Livelock



شکل ۴-۶۸: پادالگوی بنبست حلقوی در گراف دروازه‌ای

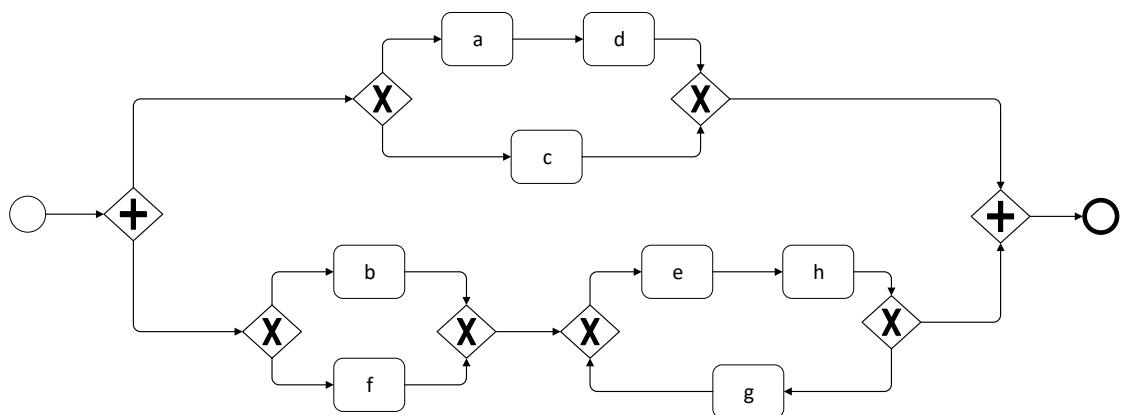
که تمام گراف دروازه‌ای، با استفاده از الگوهای صحت تجزیه شد، بر طبق قضیه‌ی ۲۳-۲-۴، فرآیند داده شده فرآیندی صحیح خواهد بود؛ در غیر این صورت، صحت آن منوط به صحت گراف باقی‌مانده‌ی آن می‌باشد. برای پیش‌بینی وضعیت صحت گراف باقی‌مانده، در آن به دنبال پادالگوهای صحت می‌گردیم و بر اساس آن، نظر نهایی در خصوص صحت مدل فرآیندی اعلام می‌شود.

روش تحلیل دروازه‌ای می‌تواند به کمک انسان انجام شود؛ اما به منظور اجرای آن با استفاده از رایانه، الگوریتمی مبتنی بر زیرگراف کاوی ارائه می‌شود. این الگوریتم در قالب یک فلوچارت در تصویر ۶۹-۴ ارائه شده‌است. برای درک بهتر الگوریتم، آن را در حالت کلی و در قالب یک مثال اجرا می‌کنیم. البته باید توجه داشت که در الگوریتم ارائه شده در تصویر ۶۹-۴ مجموعه‌ای از الگوهای شناسایی شده، همزمان حذف می‌شوند و سپس، دور بعدی الگوریتم اجرا می‌شود؛ اما در مثال، برای درک بهتر مکانیزم الگوریتم، در هر مرحله یک الگو را شناسایی کرده و حذف می‌کنیم.



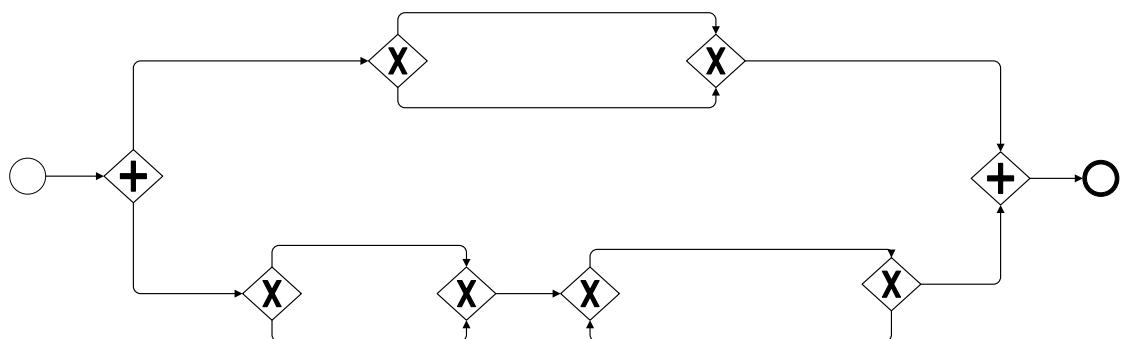
شکل ۴-۶۹: فلوچارت الگوریتم تحلیل دروازه‌ای یک مدل فرآیندی

نمونه ۴-۳-۵. مدل فرآیندی BPMN 2.0 ترسیم شده در تصویر ۴-۷۰ مفروض است. می‌خواهیم ضمن استخراج گراف دروازه‌ای از این مدل فرآیندی، الگوهای صحت را از روی آن گراف دروازه‌ای، استخراج نماییم.



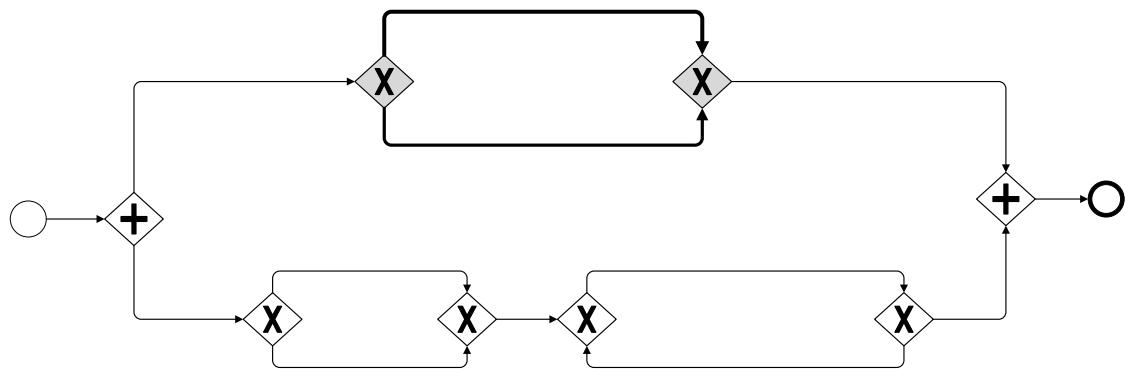
شکل ۴-۷۰: یک مدل فرآیندی که قرار است مورد تحلیل ایستا قرار گیرد.

در ابتدا، با حذف فعالیت‌ها، گراف دروازه‌ای را ترسیم می‌کنیم. سپس، مشاهده می‌شود که مطابق با

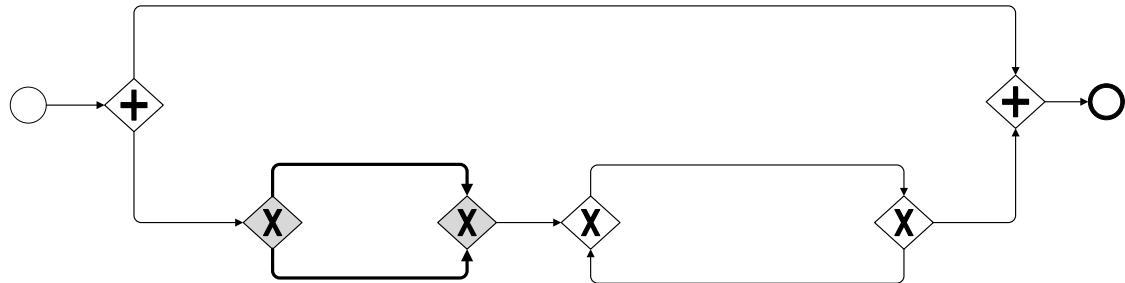


شکل ۴-۷۱: گراف دروازه‌ای معادل با مدل فرآیندی ترسیم شده در تصویر ۴-۷۰

الگوی صحت فصلی، یک الگوی صحت فصلی در زیرگراف بالایی این فرآیند، وجود دارد. این زیرگراف در تصویر ۴-۷۲ با رنگ‌هایی متفاوت نشان داده شده است. با حذف این زیرگراف از گراف دروازه‌ای، تصویر ۴-۷۳ بدست می‌آید. در شکل ۴-۷۳ مشاهده می‌شود دو دروازه‌ی انحصاری، تشکیل الگوی فصلی را می‌دهند. این دروازه‌ها بهمراه یال‌های زیرگراف مربوطه، با رنگی متفاوت نشان داده شده‌اند. بنابراین، این الگوی صحت نیز از گراف دروازه‌ای حذف شده و شکل ۴-۷۴ بدست می‌آید. در تصویر ۴-۷۴ مشاهده

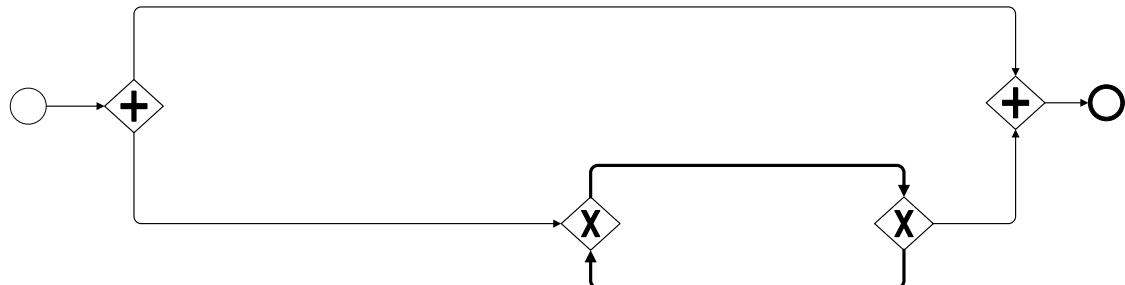


شکل ۷۲-۴: کشف الگوی صحت فصلی در گراف دروازه‌ای



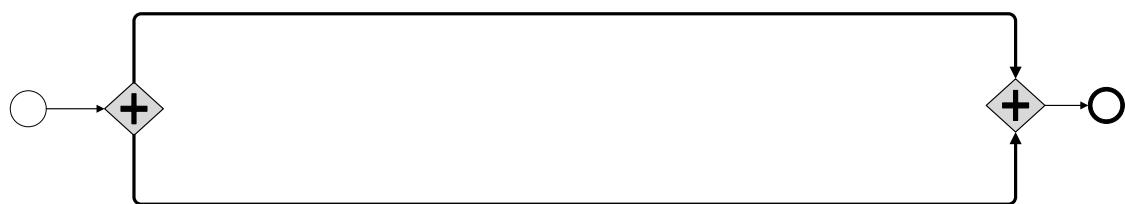
شکل ۷۳-۴: حذف زیرگراف مطابق با الگوی صحت فصلی و در ادامه، کشف الگوی صحت فصلی دیگر

می‌شود که دو دروازه‌ی فصلی باقی‌مانده، الگوی صحت تکراری را تشکیل می‌دهند. بنابراین می‌توان آن‌ها را حذف نمود. با حذف این الگوی صحت، تصویر ۷۵-۴ بدست می‌آید. در گراف دروازه‌ای ترسیم شده



شکل ۷۴-۴: حذف زیرگراف مطابق با الگوی صحت فصلی و در ادامه، کشف الگوی صحت تکراری

در شکل ۴-۷۵ مشاهده می‌شود که دو دروازه‌ی عطفی، با یک دیگر تشکیل یک الگوی صحت عطفی را می‌دهند. بنابراین، می‌توانند حذف شوند. در نتیجه‌ی آن، تصویر ۴-۷۶ بدست می‌آید. همان‌طور که مشاهده می‌شود، در این شکل، رخداد شروع و پایان در مجاورت یک‌دیگر قرار گرفته‌اند و این بدان معنا خواهد بود که گراف دروازه‌ای، به طور کامل با الگوهای صحت مورد تجزیه و تحلیل قرار گرفته‌است. در نتیجه، فرآیند معادل با این گراف دروازه‌ای، فرآیندی صحیح خواهد بود.



شکل ۴-۷۵: حذف زیرگراف مطابق با الگوی صحت تکراری و کشف الگوی صحت عطفی



شکل ۴-۷۶: حذف زیرگراف مطابق با الگوی صحت عطفی

## فصل پنجم

### پیاده‌سازی روش پیشنهادی و آزمایشات

## ۱-۵ ملزومات پیاده‌سازی

### ۱-۱-۵ زبان برنامه‌نویسی و کتابخانه‌های مورد استفاده

به منظور پیاده‌سازی روش ارائه شده در فصل قبل، تحت عنوان تحلیل دروازه‌ای، از زبان برنامه‌نویسی پایتون استفاده شده است. همچنین، به منظور تحلیل هرچه بهتر گراف‌ها، پیاده‌سازی روش پیشنهادی با استفاده از کتابخانه Network X صورت گرفته است [۳۳]. مهم‌ترین مزایای این کتابخانه که به منظور پیاده‌سازی الگوریتم به کار گرفته شده است، به شرح زیر است:

۱. امکان اشراف بر ویژگی‌های هر کدام از گره‌های شبکه (همانند دسترسی به همسایه‌های هر گره، اجداد و فرزندان آن).

۲. امکان استخراج زیرگراف با نودها و یا یال‌های دلخواه و تحلیل جداگانه‌ی آن‌ها.

همچنین، برنامه به صورت تابعی<sup>۱</sup> و با استفاده از محیط برنامه‌نویسی Jupyter Notebook پیاده‌سازی شده است. البته در انتهای، ساختاری را به منظور پیاده‌سازی شی‌گرای این نرم‌افزار، ارائه خواهیم داد.

### ۲-۱-۵ پیش‌پردازش و ابزار به کار رفته در آن

همان‌طور که در بخش‌های قبل ملاحظه شد، پیش از تحلیل ایستا لازم است که گراف دروازه‌ای معادل با مدل فرآیندی استخراج شده و مورد پردازش قرار گیرد. پیش‌پردازش یک مدل فرآیندی به منظور تحلیل ایستای آن، شامل سه مرحله‌ی اساسی است:

۱. خوش‌تعريف کردن مدل فرآیندی بر طبق روش ارائه شده توسط [۴]. برای این منظور، پیش‌تر ابزارهایی مثل ابزار BPMN2BPEL ابداع شده است.

۲. ساده‌سازی مدل فرآیندی بر اساس دستورالعمل ارائه شده در بخش ۱-۲-۴ که در نتیجه‌ی آن مدل فرآیندی، خاصیت یک شبکه‌ی جریان کار را پیدا می‌کند.

۳. استخراج گراف دروازه‌ای با حذف رخدادها و فعالیت‌ها، از مدل فرآیندی.

موارد بالا، به صورت سیستماتیک قابل انجام است؛ با این حال، در پیاده‌سازی اولیه و به منظور مشاهده‌ی بهتر مدل فرآیندی و نتایج آن، از نرم‌افزار بیزاجی مدلر<sup>۲</sup> استفاده کرده‌ایم [۳۴]. با استفاده از این

<sup>1</sup>functional

<sup>2</sup>Bizagi Modeler

نرم افزار، ضمن اعمال پیش پردازش های لازم، گراف دروازه ای نیز استخراج شده است و به صورت یک فایل با پسوند bpmn. به برنامه‌ی اصلی داده می‌شود و در برنامه‌ی اصلی، این فایل دریافت شده و از روی آن و با استفاده از کتابخانه‌ی Network X گرافی ساخته می‌شود و نمونه فرآیند مورد نظر برای تحلیل ایستا، آماده خواهد شد.

### ۳-۱-۵ پایگاه داده‌ی مورد استفاده

برای آزمودن روش با مثال‌هایی عملی، از پایگاه داده‌ی کاموندا<sup>۳</sup> استفاده شده است. این شرکت، تولیدکننده‌ی یکی از موتورهای جریان کار است که وظیفه‌ی این موتور نرم افزاری، اجرای مدل‌های فرآیندی BPMN 2.0 است. پایگاه داده‌ی ارائه شده توسط این شرکت، از گیت‌هاب قابل دریافت است<sup>۴</sup>. این پایگاه داده، مشتمل بر حدود ۳۷۰۰ نمودار ترسیم شده با زبان مدل‌سازی BPMN 2.0 است و در آخرین تحقیقات صورت گرفته در خصوص صحتسنجی مدل‌های فرآیندی BPMN 2.0 از این پایگاه داده استفاده شده است [۳۵]. در این پایگاه داده، ۵ تمرین طرح شده است و برای هر تمرین، یک نمودار به عنوان پاسخ صحیح قرار داده شده است و به ازای هر تمرین، یک پوشه که مشتمل بر دهها پاسخ مختلف به آن تمرین است، گردآوری شده است.

## ۲-۵ آزمایش روش پیشنهادی

### ۱-۲-۵ مشخصات آزمایشات طراحی شده

نظر به این که روش مورد استفاده در این پژوهش، دارای پشتونه‌ی نظری بوده و نیز از روش‌های مبتنی بر یادگیری ماشین استفاده نمی‌کند، آزمایش بر روی تعداد کمی از دادگان صورت گرفته است و هدف آن، آزمودن عملکرد کلی نرم افزار می‌باشد. برای این منظور، ۱۰ مدل فرآیندی از پایگاه داده‌ی کاموندا مورد استفاده قرار گرفته‌اند. این ۱۰ نمونه‌ی آزمایشی، شامل ۵ نمونه از تمرین اول و ۵ نمونه از تمرین دوم بودند. با توجه به این که روش پیشنهادی، ملاحظات بین‌سازمانی را در نظر نمی‌گیرد و فرآیندهای داخل یک سازمان را بررسی می‌کند، لازم بود تغییراتی در فرآیندهای انتخاب شده، صورت بگیرد؛ بنابراین، ۱۰ گراف دروازه‌ای بدست آمد که با الهام از ۱۰ نمونه‌ی انتخابی از پایگاه داده، ساخته شده است. عملیات پیش‌پردازش و استخراج گراف دروازه‌ای، با استفاده از نرم افزار بیزاری مدلر بر روی دادگان مصرفی، اجرا

<sup>3</sup>Camunda

<sup>4</sup><https://github.com/camunda/bpmn-for-research>

شد. گراف‌های دروازه‌ای مورد آزمایش، در ضمیمه‌ی این پژوهش درج شده‌است. در نتیجه‌ی اجرای برنامه، تمامی الگوهای مورد استفاده کشف شدند و نتایج بدست آمده از اجرای برنامه بر روی داده‌های آزمایشی در مقایسه با صحت واقعی فرآیندها، حاکی از آن است که این روش بر روی نمونه‌های آزمایشی به خوبی عمل کرده‌است.

## ۲-۲-۵ نتایج آزمایش

نتایج آزمایش‌های صورت گرفته بر روی ۱۰ گراف دروازه‌ای، در جدول ۱-۵ نمایش داده شده‌است. در هر گراف دروازه‌ای، تعداد موارد وقوع الگوها و پادالگوها، از قبل مشخص شده؛ سپس با استفاده از برنامه رایانه‌ای، گراف‌های دروازه‌ای مورد بررسی قرار گرفتند و نتایج حاصل از اجرای برنامه، در سطرهای مربوطه و بر مبنای تعداد الگوهای تشخیص داده شده، درج شده‌است.

## ۳-۲-۵ ارزیابی نتایج

نتایج بدست آمده بر روی ۱۰ گراف دروازه‌ای، در جدول ۱-۵ مشخص شده‌است و در ادامه قصد داریم به تحلیل این نتایج بپردازیم؛ گراف‌های دروازه‌ای مورد بررسی، در ضمیمه آورده شده‌اند. اگرچه نتایج نشان می‌دهند که این روش به طور کامل و به درستی توانسته است صحت مدل‌های فرآیندی را تایید و یا رد کند، اما لازم است که معیارهایی به منظور ارزیابی ارائه شوند تا در صورت توسعه‌ی آزمایش در پژوهش‌های آتی، قابل استفاده باشند. ارزیابی نتایج، در سه محور کلی صورت می‌گیرد:

۱. میزان دقت روش پیشنهادی در کشف هرکدام از الگوها و پادالگوهای پنجگانه.

۲. میزان دقت روش پیشنهادی در هر یک از نمونه‌های آزمایشی و مشاهده‌ی میزان پایداری آن‌ها.

۳. میزان دقت روش پیشنهادی در اعلام صحت و یا عدم صحت مدل‌های فرآیندی.

برای هرکدام از این موارد، معیارهای صحت زیر، در نظر گرفته شده‌است:

**معیار Precision** این معیار بیان می‌کند که: «چه نسبتی از الگوهای کشف شده، به درستی تشخیص داده شده‌اند؟». این نسبت، بر مبنای تعداد الگوهایی که به درستی کشف شده‌اند (ثبت صحیح یا  $TP$ ) و تعداد الگوهایی که به اشتباه کشف شده‌اند (ثبت ناصحیح یا  $FP$ )، بدست می‌آید. این

الگوی صحت										شماره گراف										الگوی صحت									
۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	عطفی	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	فصی	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	نکرار	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	بن بست احتمالی	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	بن بست واقعی	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	بن بست حلقوی	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست	تعداد واقعی	تشخیص درست	تشخیص نادرست
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	صحت فرآیند	صحت واقعی فرآیند	صحت کامل با الگوهای صحت	دارد	دارد	دارد	دارد	دارد	دارد	دارد
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	صحت اعلامی توسط برنامه	دارد	دارد	دارد	دارد	دارد	دارد	دارد	دارد	دارد
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	نتیجه آزمایش	منطبق	منطبق	منطبق	منطبق	منطبق	منطبق	منطبق	منطبق	منطبق

جدول ۵-۱: نتیجه‌های آزمایشات صورت گرفته بر روی دادگان آزمایشی

نسبت، از طریق رابطه‌ی ۱-۵ بدست می‌آید.

$$Precision = \frac{TP}{TP + FP} \quad (1-5)$$

**معیار Recall** این معیار بیان می‌کند که: «چه نسبتی از الگوهای موجود در مدل فرآیندی، به درستی کشف شده‌اند؟». این نسبت، بر مبنای تعداد الگوهایی که به درستی کشف شده‌اند (مثبت صحیح یا  $TP$ ) و تعداد الگوهایی که در مدل فرآیندی وجود داشتند، اما توسط برنامه‌ی پیشنهادی کشف نشده‌اند (منفی ناصحیح یا  $FN$ )، بدست می‌آید. این نسبت، از طریق رابطه‌ی ۲-۵ ۲-۵ بدست می‌آید.

$$Recall = \frac{TP}{TP + FN} \quad (2-5)$$

**معیار نمره‌ی F1** این معیار، میانگین هارمونیک دو معیار قبلی است. این نسبت، از طریق رابطه‌ی ۳-۵ بدست می‌آید.

$$F1Score = \frac{2TP}{2TP + FN + FP} \quad (3-5)$$

### نتایج آزمایش بر روی هر کدام از الگوها

در آزمایش اول، بررسی می‌کنیم که هر کدام از الگوها با چه دقیقی کشف شده‌اند. برای سنجش این مقدار، لازم است که در خصوص هر الگو، معیارهای صحت را بدست آوریم. این کار در جدول ۲-۵ صورت گرفته‌است و مشاهده می‌شود که در مجموع، تمامی الگوها و پادالگوها به درستی کشف شده‌اند. با توجه به این که در نمونه‌های مورد آزمایش، هیچ کدام از پادالگوهای بن‌بست غیرقطعی و نیز بن‌بست حلقوی وجود نداشتند، این معیارها نیز تعریف نمی‌شوند. اما در آزمایشات صورت گرفته، عدم وجود آن‌ها توسط مدل، تشخیص داده شده‌است.

### نتایج آزمایش بر روی هر کدام از نمونه‌های آزمایشی

در آزمایش دوم، بررسی می‌شود که در هر کدام از نمونه‌های آزمایشی، الگوها و پادالگوها با چه میزانی از دقیقت کشف شده‌اند. برای سنجش این مقدار، لازم است که در خصوص هر نمونه‌ی آزمایشی، معیارهای صحت را بدست آوریم. این کار در جدول ۳-۵ صورت گرفته‌است و مشاهده می‌شود که در مجموع، تمامی

جدول ۲-۵: ارزیابی روش پیشنهادی بر روی هر کدام از الگوهای و پادالگوهای صحت

<i>F1 – Score</i>	<i>Recall</i>	<i>Precision</i>	<i>FN</i>	<i>FP</i>	<i>TP</i>	الگوی صحت عطفی
۱	۱	۱	۰	۰	۸	الگوی صحت فصلی
۱	۱	۱	۰	۰	۶	الگوی صحت تکراری
۱	۱	۱	۰	۰	۱	پادالگوی بنبست (غیرقطعی)
نامعین	نامعین	نامعین	۰	۰	۰	پادالگوی بنبست (قطعی)
۱	۱	۱	۰	۰	۱	بنبست حلقوی
نامعین	نامعین	نامعین	۰	۰	۰	

الگوهای و پادالگوهای به درستی کشف شده‌اند. در نمونه‌های ۲، ۷، ۸ و ۱۰، هیچ کدام از الگوهای و پادالگوهای کشف نشده‌اند. یعنی اگرچه دلیلی برای صحت آن‌ها یافته نشده، اما پادالگویی در آن مشاهده نشده‌است. وقوع مشکلات ساختاری، در گره‌هایی رخ می‌دهد که در آن‌ها، عمل هم‌گام‌سازی و یا موازی‌سازی انجام می‌شود. مورد اول ممکن است منجر به وقوع بنبست و مورد دوم، ممکن است منجر به وقوع بنبست حلقوی شود؛ روش پیشنهادی، بر روی مشاهده‌ی این دو نوع گره، حساس است و در صورت مشاهده‌ی آن‌ها، با بررسی گره‌های مجاورشان، سعی می‌کند که وضعیت آن‌ها را تعیین کند؛ بنابراین در صورت عدم وجود پادالگوها، فرآیند، یک فرآیندی صحیح درنظر گرفته شده‌است؛ با توجه به عدم وجود الگوهای و یا پادالگوهای در آن چهار فرآیند، معیارهای صحت، نامعین خواهند شد و در عین حال، فرآیندهای مذکور صحیح درنظر گرفته خواهند شد.

جدول ۳-۵: ارزیابی روش پیشنهادی بر روی هر کدام از نمونه‌های آزمایشی

<i>F1 – Score</i>	<i>Recall</i>	<i>Precision</i>	<i>FN</i>	<i>FP</i>	<i>TP</i>	شماره‌ی گراف مورد آزمایش
۱	۱	۱	۰	۰	۴	۱
نامعین	نامعین	نامعین	۰	۰	۰	۲
۱	۱	۱	۰	۰	۴	۳
۱	۱	۱	۰	۰	۴	۴
۱	۱	۱	۰	۰	۲	۵
۱	۱	۱	۰	۰	۱	۶
نامعین	نامعین	نامعین	۰	۰	۰	۷
نامعین	نامعین	نامعین	۰	۰	۰	۸
۱	۱	۱	۰	۰	۱	۹
نامعین	نامعین	نامعین	۰	۰	۰	۱۰

### ارزیابی نتایج بر روی نتیجه‌ی نهایی

در آزمایش سوم و نهایی، به بررسی میزان دقیقت کلی روش پیشنهادی در اعلام نتیجه‌ی نهایی صحت‌سنگی پرداخته می‌شود. در این بخش، اعلام صحیح بودن فرآیند به منزله‌ی جواب مثبت و منفی بودن آن به

منزله‌ی جواب منفی در نظر گرفته شده است. با این اوصاف، بر طبق جدول ۱-۵ مقادیر مورد استفاده برای آزمایش روش به شرح زیر هستند:

$$TP = 8 \quad (4-5)$$

$$TN = 2 \quad (5-5)$$

$$FN = 0 \quad (6-5)$$

در این آزمایش، منفی کاذب نیز می‌تواند مورد بررسی قرار گیرد؛ مثبت کاذب به حالتی اطلاق می‌شود که در آن، وضعیت فرآیند داده شده، صحیح اعلام می‌شود، در حالی که فرآیند، ناصحیح است. در آزمایش صورت گرفته چنین وضعیتی رخ نداده.

$$FP = 0 \quad (7-5)$$

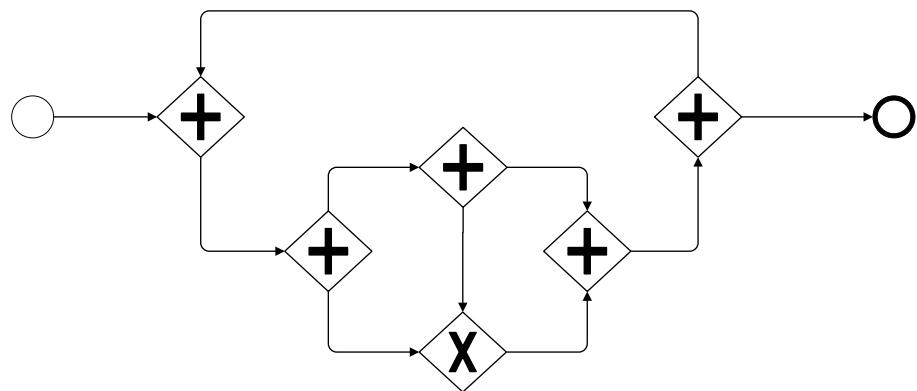
در نتیجه، نمرات *F1Score*، *Precision* و *Recall* همگی برابر با ۱ خواهند بود.

#### ۴-۲-۵ موارد شکست روش

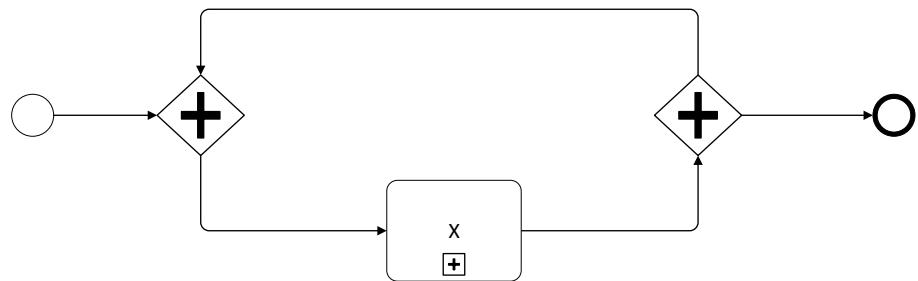
الگوریتم متناظر با فلوچارت ترسیم شده در شکل ۶۹-۴ یکی از روش‌های ممکن را برای تحلیل گراف دروازه‌ای، بر مبنای قضیه‌ی ۲۳-۲-۴ بدست می‌دهد. روش ارائه شده در قضیه‌ی ۲۳-۲-۴ روشی صحیح است و درستی آن در بخش ۳-۲-۴ به صورت تفصیلی مورد بررسی و اثبات قرار گرفت؛ با این حال، الگوریتم تحلیل دروازه‌ای که فلوچارت آن در تصویر ۶۹-۴ ترسیم شده است، ممکن است در به کارگیری قضیه‌ی ۲۳-۲-۴ دچار خطأ شود؛ مواردی که این الگوریتم در یافتن الگوهای پادالگوهای صحت با مشکل مواجه می‌شود، بدین شرح می‌باشد: «در صورتی که بخش‌های متغیر یک الگو و یا پادالگوی صحت، حاوی زیرگرافی باشد که با هیچ کدام از الگوهای پادالگوهای ارائه شده مطابقت نداشته باشد، این الگوریتم نمی‌تواند الگو و یا پادالگوی خارجی را تشخیص دهد.» برای روشن‌تر شدن این مشکل، مثالی را در ادامه ارائه می‌کنیم.

نمونه ۱-۵. در شکل ۱-۵ گراف دروازه‌ای معادل با یک مدل فرآیندی BPMN2.0 نشان داده شده است که حاوی یک بن‌بست حلقوی است؛ با این حال، با اجرای الگوریتم بر روی آن (با استفاده از برنامه‌ی نوشته شده) هیچ پادالگویی از آن استخراج نشده است. این گراف دروازه‌ای را می‌توان به یک فرآیند کلی و یک زیرفرآیند کوچک تقسیم نمود. این فرآیند کلی، در تصویر ۲-۵ نشان داده شده است. این فرآیند کلی، حاوی یک زیرفرآیند به نام  $X$  می‌باشد که گراف دروازه‌ای معادل با جزئیات آن، در تصویر ۳-۵ ترسیم شده است. مشاهده می‌شود که این زیرفرآیند، از هیچ کدام از الگوهای پادالگوهای صحت پیروی نمی‌کند، اما یک زیرفرآیند صحیح است و دچار بن‌بست و یا بن‌بست حلقوی نمی‌شود.

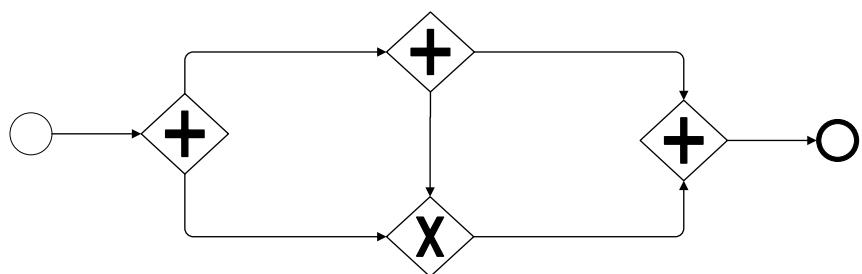
با توجه به این که زیر فرآیند (زیرگراف)  $X$  صحیح است، می‌توان آن را از تحلیل کنار گذاشت و صحت فرآیند، منوط به صحت مدل فرآیندی بزرگ‌تر، یعنی مدل فرآیندی ترسیم شده در شکل ۲-۵ است. با توجه به این که این مدل فرآیندی دارای یک بن‌بست حلقوی است، مدل فرآیندی ترسیم شده در تصویر ۱-۵ نیز صحیح نخواهد بود؛ اما با توجه به این که هیچ کدام از بخش‌های زیرگراف  $X$  با هیچ کدام از الگوهای پادالگوهای صحت مطابقت ندارد، این زیرگراف کنار گذاشته نمی‌شود و پادالگوی بن‌بست حلقوی دیده نمی‌شود و در نتیجه، فرآیند اشتباه‌ها یک فرآیند صحیح اعلام می‌شود.



شکل ۱-۵: یک نمونه مدل فرآیندی منجر به شکست روش تحلیل دروازه‌ای



شکل ۵-۲: پادالگوی بنبست حلقوی در فرآیند ترسیم شده در شکل ۱-۵



شکل ۵-۳: زیرگراف دروازه‌ای صحیح و فاقد الگوهای صحت

## فصل ششم

# یافته‌های پژوهش و پیشنهادهایی برای پژوهش‌های آتی

## ۱-۶ دستاوردها و نوآوری‌های پژوهش

در پژوهش جاری، به ارائه‌ی راهکاری برای تحلیل ایستای مدل‌های فرآیندی BPMN 2.0 پرداخته شد. بزرگ‌ترین مزیت راهکار ارائه شده در پژوهش جاری نسبت به بخش عمده‌ای از پژوهش‌های صورت گرفته در این حوزه، عدم نیاز تبدیل مدل فرآیندی BPMN 2.0 به مدل‌های صوری بوده است؛ این در حالی است که در بسیاری از پژوهش‌های پیشین، برای تحلیل ایستای یک مدل فرآیندی BPMN 2.0 در ابتدا لازم بود که در ابتدا، مدل فرآیندی مورد نظر به یک مدل صوری نگاشته شده و سپس بر مبنای قواعد و الگوریتم‌های صحت‌سنجدی آن مدل صوری، مدل فرآیندی BPMN 2.0 داده شده، صحت‌سنجدی شود.

در پژوهش جاری، با معرفی گراف دروازه‌ای و روشی برای تحلیل آن (روش جایگزینی تعمیم یافته) توانستیم راهکاری را ارائه کنیم که در طی آن، نیازی به انتقال مدل فرآیندی BPMN 2.0 به یک مدل صوری وجود نداشته باشد و با بخشی از همان مدل فرآیندی BPMN 2.0 و با استفاده‌ی مجدد از الگوها و پادالگوهای مدل‌های فرآیندی صحیح، بتوان بخش مهمی از تحلیل ایستا را انجام داد؛ در نتیجه روش ارائه شده در پژوهش جاری، جزو محدود روش‌های صحت‌سنجدی مدل‌های فرآیندی BPMN 2.0 است که می‌توان به وسیله‌ی آن، عملیات صحت‌سنجدی را به طور مستقیم بر روی یک مدل فرآیندی BPMN 2.0 انجام داد.

## ۲-۶ دستاوردهای ذیل پژوهش

هر کار پژوهشی، در کنار دستاوردهای اساسی‌اش، دارای دستاوردهایی است که جزو اجزای شکل‌دهنده‌ی راه حل نهایی است. در این پژوهش نیز دستاوردهایی در ذیل دستاوردهایی به وجود آمده است که در ادامه، مرور می‌شوند.

### ۱-۲-۶ نگاشت مدل‌های فرآیندی BPMN 2.0 شبکه‌های پتری جریان کار

همان‌طور که در فصل ۳ مشاهده شد، تحقیقات فراوانی در خصوص نگاشت مدل‌های فرآیندی BPMN 2.0 به مدل‌های صوری، صورت گرفته بود. اما در هیچ‌یک از این تحقیقات، عمل صحت‌سنجدی با استفاده از شبکه‌های پتری جریان کار صورت نگرفته بود؛ حال آن‌که این توسعه از شبکه‌های پتری، به واسطه‌ی ساده‌سازی‌های صورت گرفته بر روی آن، سازگاری بیشتری را با مدل‌های فرآیندی BPMN 2.0 دارد. بنابراین، یکی از دستاوردهای ذیل این پژوهش می‌تواند نگاشت ارائه شده بین مدل‌های فرآیندی BPMN

2.0 و شبکه‌های پتری جریان کار باشد و بدین‌وسیله، امکان انتقال یک مدل فرآیندی BPMN به یک شبکه‌ی پتری جریان کار و سپس، صحت‌سنجی آن به وسیله‌ی روش‌های صحت‌سنجی شبکه‌های پتری جریان کار، همچون روش‌های جانشینی و اتصال کوتاه [۲] [۳] وجود خواهد داشت.

## ۲-۲-۶ تعمیم مفهوم صحت به مدل‌های فرآیندی BPMN 2.0

همان‌طور که در مراجع [۳] و [۲] دیده شده است، مفهوم صحت بر مبنای اجزای شبکه‌های پتری صورت گرفته بود و در پژوهش‌های مختلف، تفاسیر متفاوتی از یک مدل فرآیندی BPMN 2.0 صحیح وجود داشت؛ این در حالی است که بر طبق نظر ماتیاس در کتاب مدیریت فرآیندهای کسب و کار، مفهوم صحت برای مدل‌های فرآیندی ایجاد شده با استفاده از سایر زبان‌های مدل‌سازی فرآیندهای کسب و کار نیز قابل تعمیم و تعریف است [۲]. در این پژوهش، با تعمیم مفهوم صحت به مدل‌های فرآیندی BPMN 2.0 توانستیم بر پایه‌ی یک مدل صوری (شبکه‌های پتری جریان کار)، تعریفی جامع را برای یک مدل فرآیندی BPMN 2.0 صحیح ارائه کنیم.

## ۳-۲-۶ صحت‌سنجی بر مبنای استفاده‌ی مجدد از الگوهای صحت

در ادامه‌ی پژوهش، بر مبنای نگاشت ارائه شده بین شبکه‌ی پتری جریان کار و مدل‌های فرآیندی BPMN 2.0 و بر مبنای تعریف تعمیم داده شده‌ی صحت در مدل‌های فرآیندی BPMN 2.0، موفق به ارائه‌ی یک رویکرد مبتنی بر الگو<sup>۱</sup> برای تحلیل ایستای مدل‌های فرآیندی BPMN 2.0 شدیم. این رویکرد می‌تواند به طرق مختلف، حتی بدون نیاز به رایانه، در صحت‌سنجی فرآیندهای کسب و کار و توسط توسعه‌دهندگان مورد استفاده قرار گیرد. با این حال، استفاده از داده‌ساختار گراف دروازه‌ای و نیز، اجرای الگوریتم تحلیل دروازه‌ای می‌توانند جزو راهکارهای مناسب به منظور اجرای این روش با استفاده از کامپیوترها باشند.

## ۴-۲-۶ ارائه‌ی داده‌ساختار گراف دروازه‌ای

داده‌ساختار گراف دروازه‌ای، دارای مزایای متعددی است؛ از جمله:

۱. گراف دروازه‌ای، بازتاب‌دهنده‌ی جریان کنترلی یک فرآیند می‌باشد و به عبارت دیگر، اسکلت اصلی و شاکله‌ی جریان کنترلی فرآیند می‌باشد و با تحلیل آن، می‌توان درک خوبی را از ساختار جریان کنترلی فرآیند بدست آورد.

<sup>1</sup>Pattern Oriented Approach

۲. مدلی انتزاعی است و بسیاری از پیچیدگی‌هایی که تحلیل ساختاری را با دشواری مواجه می‌سازد، در آن وجود ندارد.

### ۳-۶ پیشنهاداتی برای پژوهش‌های آتی

هر کار پژوهشی، در کنار تمام ارزش‌های افزوده‌ی خود، می‌تواند در آینده تکمیل شده و به نتایج بهتری دست پیدا کند. این پژوهش نیز از این قاعده مستثنی نیست. پیشنهادهای ارائه شده، در دو بخش تنظیم شده‌اند؛ بخش اول، ناظر بر پیشنهاداتی است که به منظور بهبود نتایج پژوهش ارائه شده‌اند و بخش بعدی، ناظر بر پیشنهاداتی هستند که در جهت به کارگیری روش پیشنهادی، در سیستم‌هایی مثل سیستم‌های مدیریت فرآیندهای کسب و کار و پلتفرم‌های توسعه‌ی کمک و مبتنی بر زبان BPMN2.0 ارائه شده‌اند.

#### پیشنهاداتی به منظور بهبود نتایج پژوهش

در ادامه، قصد داریم پیشنهاداتی را ارائه کنیم تا با انجام آن در پژوهش‌های آتی، نتایج بهتری در خصوص تحلیل ایستای مدل‌های فرآیندی BPMN2.0 بدست آید؛ برای پژوهش‌های بعدی، پیشنهاد می‌شود که بررسی موارد زیر در اولویت قرار گیرد:

ارائه‌ی یک روش تحلیل پیشرو در پژوهش جاری، رویکرد کشف الگوهای صحت از مدل‌های فرآیندی، به صورت پسرو انجام می‌شود. اگرچه این روش می‌تواند صحت بسیاری از فرآیندهای صحیح را تصدیق کرده و با بررسی پادالگوهای مشکلات ساختاری احتمالی و یا قطعی فرآیندها را اعلام کند، اما ممکن است در مواردی، نتواند برخی از الگوهای صحت را کشف کند. ممکن است در نمونه‌هایی، یک الگوی صحت شامل زیرفرآیندهایی باشد که از هیچ یک از الگوهای صحت پیروی نکند؛ در چنین وضعیتی، الگوی صحت کلی و دربرگیرنده‌ی زیرفرآیندهای ناشناس، کشف نخواهد شد. برای رفع این مشکل، پیشنهاد می‌شود که الگوریتمی با رویکرد پیشرو<sup>۲</sup> ارائه شود تا بتواند الگوهای خارجی را نیز مورد بررسی قرار دهد. چنین الگوریتمی، می‌تواند همانند الگوریتم پسرو، زوج دروازه‌های تشکیل‌دهنده‌ی الگوها را مورد بررسی قرار داده و زیرفرآیندهای داخل ساختار آنها را استخراج کرده و در قاب یک زیرگراف جداگانه و به صورت بازگشتی، مورد تحلیل قرار دهد. در شکل ۱-۶، روندnamای یک پیشنهاد اولیه به منظور ارائه‌ی یک الگوریتم تحلیل پیشرو برای تحلیل دروازه‌ای، ارائه شده است که بررسی و ارزیابی آن، به پژوهش‌های آتی موكول خواهد

<sup>2</sup>Forward Analysis

شد. رویکرد این الگوریتم، پیشمایش گره‌های گراف دروازه‌ای با ترتیب جستجوی اول-سطح و یا بر مبنای مرتبسازی توپولوژیکی است و در طی هر پیمایش، هر گرهی شروع‌کننده‌ی یک الگوی صحت، با دورترین گرهی ممکن تطبیق پیدا کند.

**گردآوری و تدوین یک دادگان در خصوص صحت مدل‌های فرآیندی BPMN2.0** اگرچه پایگاه‌های داده‌ای همچون کاماندو، در حوزه‌ی مدل‌های فرآیندی BPMN2.0 تدوین شده‌اند، اما در حوزه‌ی صحت‌سنجی مدل‌های فرآیندی BPMN2.0 هیچگونه دادگانی تدوین نشده‌است. چنین دادگانی، باید حاوی مدل‌های فرآیندی BPMN2.0 در ساختارهای XML و یا JSON باشد و برای هر کدام از آن‌ها، وضعیت صحت مدل فرآیندی و علت آن، مشخص باشد. تهیه‌ی چنین دادگانی، نیازمند زحمت زیادی خواهد بود.

استفاده از روش‌های یادگیری ماشین در پژوهش جاری، از روش مبتنی بر یادگیری ماشین استفاده نشده‌است. در حل مسائل حوزه‌ی علوم کامپیوتر، بهتر است که در گام نخست، از الگوریتم‌های یادگیری ماشین استفاده نشود و در صورتی از الگوریتم‌های یادگیری ماشین استفاده شود که یکی از شرایط زیر برقرار باشد:

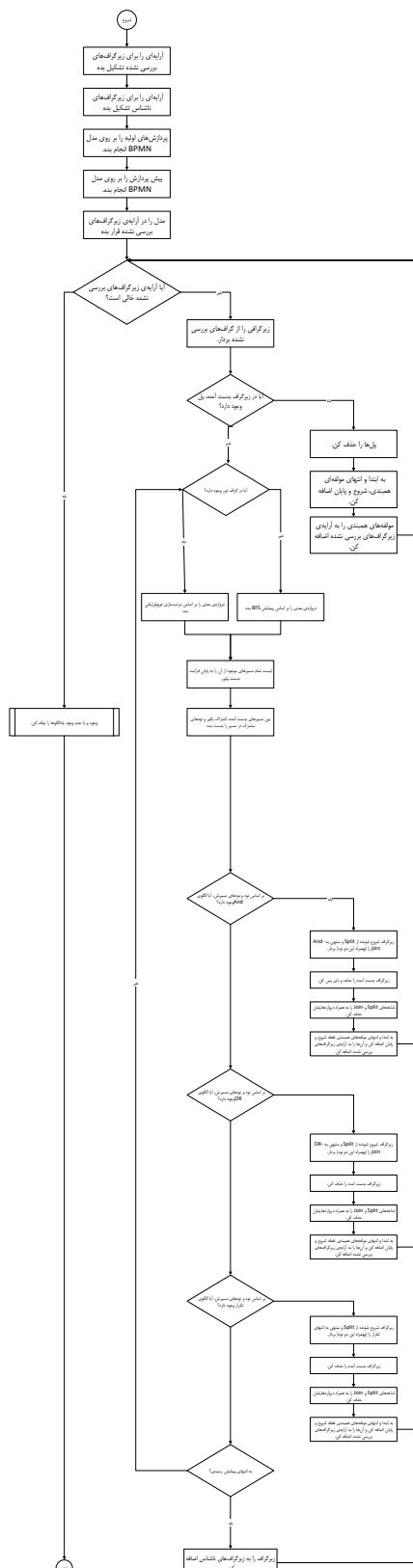
۱. هیچگونه رابطه‌ی منطقی و صریح بین ورودی‌های مسئله و خروجی‌های آن، وجود نداشته باشد.
۲. راهکاری صریح برای حل مسئله وجود داشته باشد، اما هزینه‌های زیادی را تحمل کند.

بنابراین، پیشنهاد می‌شود که در پژوهش‌های آتی، سعی شود که با استفاده از روش‌های مبتنی بر یادگیری ماشین، این مسئله حل شده و نتایج اجرای آن بر روی داده‌های آزمونی گردآوری شده، با نتایج اجرای روش این پژوهش بر روی همان داده‌ها، مقایسه شود.

### پیشنهاداتی به منظور به کارگیری عملی روش پیشنهادی

هدف نهایی پژوهش‌های حوزه‌ی علوم کامپیوتر، راههایی راهکارهایی به منظور ساده‌سازی کارها به وسیله‌ی کامپیوترها می‌باشد و بخش مهمی از آن‌ها، در راستای حل مشکلی از مشکلات صنعت، مطرح شده و حل می‌شوند. این پژوهش نیز در راستای حل یکی از کاستی‌های موجود در صنعت مهندسی نرم‌افزار، تعریف شده‌است.

روش‌های تحلیل ایستای مدل‌های فرآیندی BPMN2.0 می‌توانند در سیستم‌های مدیریت فرآیندهای کسب و کار و پلتفرم‌های توسعه‌ی کم‌کد و مبتنی بر زبان مدل‌سازی BPMN2.0 به کار گرفته شوند.



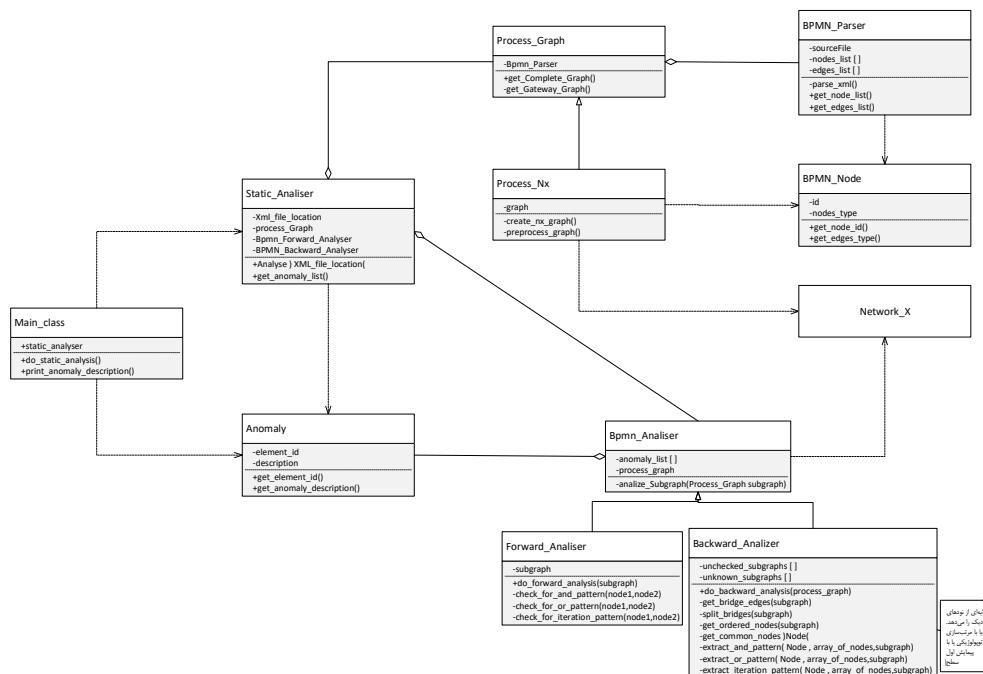
شکل ۶: روند نمای الگوریتم پیشنهادی برای تحلیل پیشرو

بنابراین دو موضوع مهم در خصوص آن‌ها مطرح می‌شود:

۱. چگونگی استقرار آن‌ها در سطح سیستم‌های نرم‌افزاری.

۲. تاثیر آن‌ها بر اصول طراحی فرآیندهای کسب و کار و نرم‌افزارهای پشتیبان آن‌ها.

استقرار روش پیشنهادی در سیستم‌های نرم‌افزاری به منظور استقرار روش پیشنهادی در یک سیستم نرم‌افزاری، همانند یک سیستم مدیریت فرآیندهای کسب و کار و یا یک پلتفرم توسعه‌ی کم‌کد و مبتنی بر BPMN2.0، یک نمودار طراحی ساختار کلاسی، پیشنهاد شده‌است. این نمودار در تصویر ۶-۲ آورده شده‌است.



شکل ۶-۲: نمودار ساختار کلاسی پیشنهادی به منظور پیاده‌سازی پیشنهادی در سیستم‌های نرم‌افزاری

به کارگیری روش پیشنهادی در متدولوژی‌های ایجاد نرم‌افزار همان‌طور که می‌دانیم، فعالیت‌تضمین کیفیت نرم‌افزارها، یکی از اساسی‌ترین فعالیت‌های چتری در مدل‌های فرآیندی ایجاد نرم‌افزارها هستند [۳۶] و این مدل‌ها، بخشی اساسی از بدنه‌ی متدولوژی‌های ایجاد نرم‌افزار هستند.

با ظهور و گسترش پلتفرم‌های توسعه‌ی کم‌کد و مبتنی بر زبان مدل‌سازی BPMN2.0 امکان تبدیل نمودارهای ترسیم شده به نرم‌افزارهایی قابل اجرا فراهم شده‌است. این امر به افرادی که

تخصصی در مهندسی نرم‌افزار ندارند، کمک می‌کند که بدون نیاز به دانش برنامه‌نویسی و صرفاً با ترسیم نمودارهای BPMN2.0 نرم‌افزارهای مورد نظر خود را ایجاد کنند؛ بنابراین، هزینه‌های ایجاد نرم‌افزار برای سازمان‌ها کاهش خواهد یافت. اگرچه این امر مزایای متعددی را دارد، اما ممکن است یک خوشبینی بیش از حد را برای سازمان‌ها ایجاد نماید. با توجه به عدم آشنایی توسعه‌دهندگان فرآیندهای کسب و کار و نیز حذف کدنویسی از فرآیند ایجاد نرم‌افزارها، ممکن است که ملاحظات لازم در مواردی مثل آزمون نرم‌افزار و اجرای سیاست‌های تضمین کیفیت نرم‌افزارها، به درستی اعمال نشود؛ درنتیجه با ایجاد فرآیندهایی که دارای خطاهایی معنایی هستند، نرم‌افزارهایی معیوب در سطح سازمان تولید شده و به کارگرفته می‌شود.

روش پیشنهادی می‌تواند به این دسته از توسعه‌دهندگان، در ایجاد و توسعه‌ی نرم‌افزارهایی صحیح و با استفاده از پلتفرم‌های توسعه‌ی کمک و مبتنی بر زبان مدل‌سازی BPMN2.0 کمک کند.

در خصوص چالش صحبت‌سنگی نرم‌افزارهای تولید شده با استفاده از پلتفرم‌های توسعه‌ی کمک به علت عدم وجود کد، تحقیقی توسط استفان برامز [۳۷] انجام شده‌است؛ او ضمن بیان این چالش، یک چارچوب کیفیت نرم‌افزار را برای تضمین کیفیت این دسته از نرم‌افزارها، ارائه کرده‌است. رویکرد ارائه شده در آن تحقیق، بر مبنای توسعه‌ی مبتنی بر رفتار<sup>۳</sup> است و سعی شده تا بر مبنای این رویکرد، چارچوبی برای تضمین کیفیت این نرم‌افزارها ارائه شود.

روش پیشنهاد شده در پژوهش جاری و یا تحقیقات آتی، می‌توانند منجر به برداشتن گامی موثر در توسعه‌ی اصول طراحی نرم‌افزارها با استفاده از پلتفرم‌های توسعه‌ی کمک و مبتنی بر زبان مدل‌سازی BPMN2.0 شوند و امکان توسعه‌ی نرم‌افزارهایی صحیح را با هزینه‌ی مناسب، برای سازمان‌ها به وجود آورند.

---

<sup>3</sup>Behaviour Driven Development

## منابع و مراجع

- [1] “Difference between verification and validation - testing diaries. (2020), retrieved 28 june 2020 from <http://www.testingdiaries.com/difference-between-verification-and-validation/>,”
- [2] Weske.M. *Business Process Management Concepts, Languages, Architectures*. Springer, 2004.
- [3] Van Hee.K. *Workflow Management: Models, Methods, and Systems*. MIT Press, 2004.
- [4] M. D. Dijkman.R.M. and Ouyang.C, “Semantics and analysis of business process models in bpmn,” *Information and Software Technology*, vol.50, no.12, pp.1281–1294, 2008.
- [5] S. N. Prinz.T.M and A. W, “A first step towards a compiler for business processes,” in *Compiler Construction* (A. Cohen, ed. ), (Berlin, Heidelberg), pp.238–243, Springer Berlin Heidelberg, 2014.
- [6] Takemura.T, “Formal semantics and verification of bpmn transaction and compensation,” in *2008 IEEE Asia-Pacific Services Computing Conference*, pp.284–290, 2008.
- [7] J. Ye, S. Sun, W. Song, and L. Wen, “Formal semantics of bpmn process models using yawl,” in *2008 Second International Symposium on Intelligent Information Technology Application*, vol.2, pp.70–74, 2008.
- [8] S. I. *Software Engineering*. Addison-Wesley, 2011.

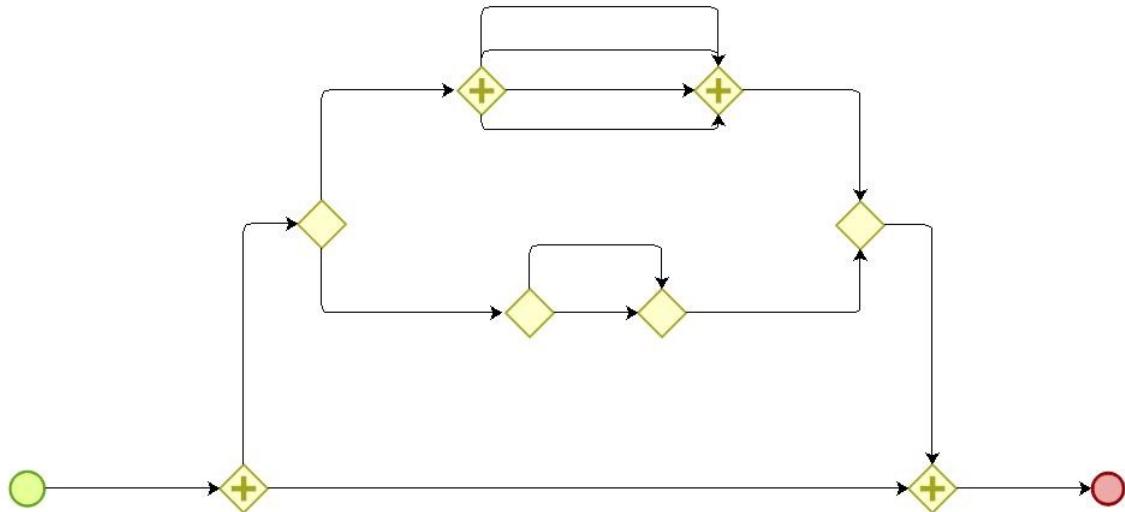
- [9] van der Aalst.W and Stahl.C. *Modeling Business Procecess A Petri Net-Oriented Approach.* MIT Press, 2011.
- [۱۰] سیدحسن حقیقی. مهندسی نرمافزار. مدرسان شریف، ۱۳۹۱.
- [۱۱] سیدعلی رضا هاشمی گلپایگانی. جزوه‌ی کلاسی درس مدیریت و بازمهندسی فرآیندهای کسب و کار. دانشگاه صنعتی امیرکبیر، ۱۳۹۹.
- [۱۲] فاطمه ذبیحی. آموزش مدل‌سازی فرآیندهای کسب و کار با زبان مدل‌سازی BPMN2.0. فرادرس، ۱۳۹۷.
- [13] M. Zurawski, R., “Petri nets and industrial applications: A tutorial,” *IEEE Transactions on Industrial Electronics*, vol.41, no.6, pp.567–583, 1994.
- [14] Kherbouche.O.M, Ahmad..A, and Basson.H, “Detecting structural errors in bpmn process models,” in *2012 15th International Multitopic Conference (INMIC)*, pp.425–431, 2012.
- [15] Kherbouche.O.M, Ahmad.A, and Basson.H, “Using model checking to control the structural errors in bpmn models,” in *IEEE 7th International Conference on Research Challenges in Information Science (RCIS)*, pp.1–12, 2013.
- [16] A. B. E. El Hichami.O, Naoum.M, “An algebraic method for analysing control flow of bpmn models,” *International Journal of Recent Contributions from Engineering, Science and IT (iJES)*, vol.3, no.3, pp.20–26, 2015.
- [17] Waszkowski.R, “Low-code platform for automating business processes in manufacturing,” *IFAC-PapersOnLine*, vol.52, no.10, pp.376–381, 2019. 13th IFAC Workshop on Intelligent Manufacturing Systems IMS 2019.
- [18] K. Rymer.J., “Low-code development platforms for ad and d professionals,” 2020.
- [19] “Vendor Landscape: The Fractured, Fertile Terrain Of Low-CodeApplication Platforms.,” 2016.

- [20] K. Waszkowski.R., “Comparative analysis of Business Process Management frameworks,” *Sustainable Economic Development, Innovation Management, and Global Growth, vols I-IX*, pp.3542–3553, Dec. 2017.
- [21] H. G. Poursoltani.Y, Shirali-Shahreza.M.H, “ using low-code development platforms for developing commercial software based on business processes,” *26th International Computer Conference, Computer Society of Iran Tehran, Iran - 3-4*, 2021.
- [22] “Flowable open source documentation. (2020). retrieved 19 december 2020, from <https://flowable.com/open-source/docs/bpmn/ch02-gettingstarted/>,”
- [23] “Bizotop wiki. (2020). retrieved 19 december 2020, from <https://wiki.bizotop.com/>,”
- [24] D. Lohmann.N, Verbeek.E, ”*Petri Net Transformations for Business Processes – A Survey*”, pp.”46–63”. ”Berlin, Heidelberg”: ”Springer Berlin Heidelberg”, ”2009”.
- [25] Watahiki.K, Ishikawa.K, and Hiraishi.K, “Formal verification of business processes with temporal and resource constraints,” in *2011 IEEE International Conference on Systems, Man, and Cybernetics*, pp.1173–1180, 2011.
- [26] D. Mallek.S., Chapurlat.V., and Vallespir.B, “Enabling model checking for collaborative process analysis: from bpmn to ‘network of timed automata’,” *Enterprise Information Systems*, vol.9, no.3, pp.279–299, 2015.
- [27] Falcioni.D, Polini.A, Polzonetti.A, and Re.B, “”livelock and deadlock detection for pa inter-organizational business processes”,” in ”*Advancing Democracy, Government and Governance*” (A. ”Kő, C. Leitner, H. Leitold, and A. Prosser, eds. ), (”Berlin, Heidelberg”), pp.”125–139”, ”Springer Berlin Heidelberg”, ”2012”.
- [28] Naoum.M, El Hichami.O, A. M, and E. B, “A probabilistic method for business process verification: Reachability, liveness and deadlock detection,” in *2016 4th IEEE International Colloquium on Information Science and Technology (CiSt)*, pp.128–132, 2016.

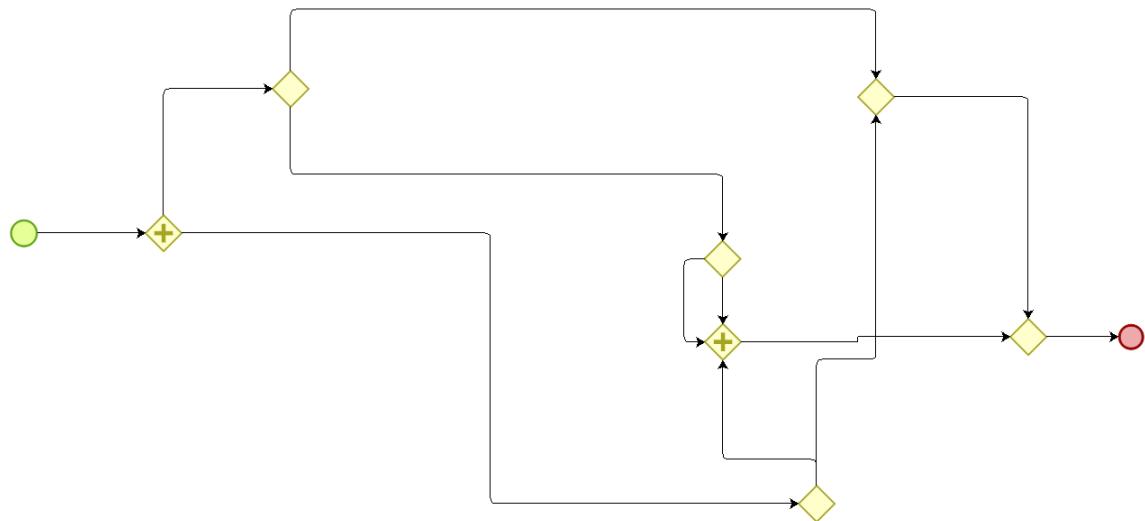
- [29] Li.L and Dai.F, “Transformation and visualization of BPMN models to petri nets,” *IOP Conference Series: Earth and Environmental Science*, vol.186, p.012047, oct 2018.
- [30] S. K. Meghzili.S, Chaoui.A, “An approach for the transformation and verification of bpmn models to colored petri nets models,” *International Journal of Software Innovation (IJSI)*, vol.8, p.33, 2020.
- [31] Ou-Yang.C and Lin.Y.D, “Bpmn-based business process model feasibility analysis: a petri net approach,” *International Journal of Production Research*, vol.46, no.14, pp.3763–3781, 2008.
- [32] Wong.P.Y.H. and Gibbons.J, “Property specifications for workflow modelling,” *Science of Computer Programming*, vol.76, no.10, pp.942–967, 2011. Integrated Formal Methods (iFM09).
- [۳۳] فرشید شیرافکن. آموزش گراف کاوی و تحلیل شبکه ها در پایتون با NetworkX. فرادرس، ۱۳۹۹.
- [۳۴] نازنین حبیبی. آموزش مدلسازی فرآیندهای کسب و کار با نرم افزار Bizagi. فرادرس، ۱۳۹۹.
- [35] Firdaus.S, Ding.C, and Sadeghian.A, “Retweet prediction based on topic, emotion and personality,” *Online Social Networks and Media*, vol.25, p.100165, 2021.
- [36] R. Pressman. *Software Engineering: A Practitioner's Approach*. McGraw-Hill, european ed. , 1994.
- [37] Braams.S, “Developing a software quality framework for low-code model driven development platforms based on behaviour driven development methodology.,” in *27th Twente Student Conference on IT*, 2017.

## پیوست: نمونه‌های آزمایشی

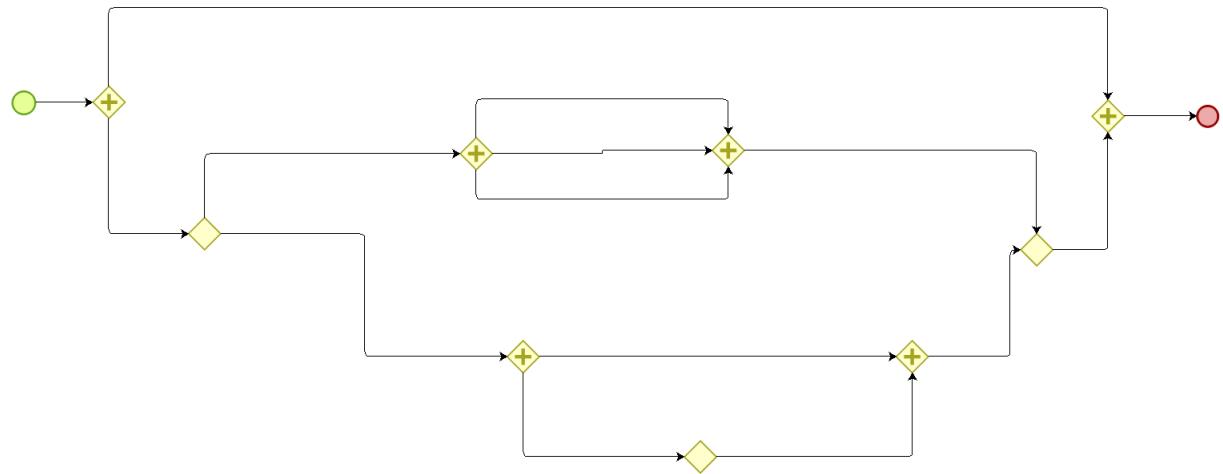
در ادامه، گراف‌های دروازه‌ای مورد آزمایش، آورده شده‌اند. این گراف‌ها، با نرم‌افزار بیزاجی مدل‌طراحی شده‌اند و در برخی موارد ممکن است دو کمان، روی یکدیگر افتاده باشند.



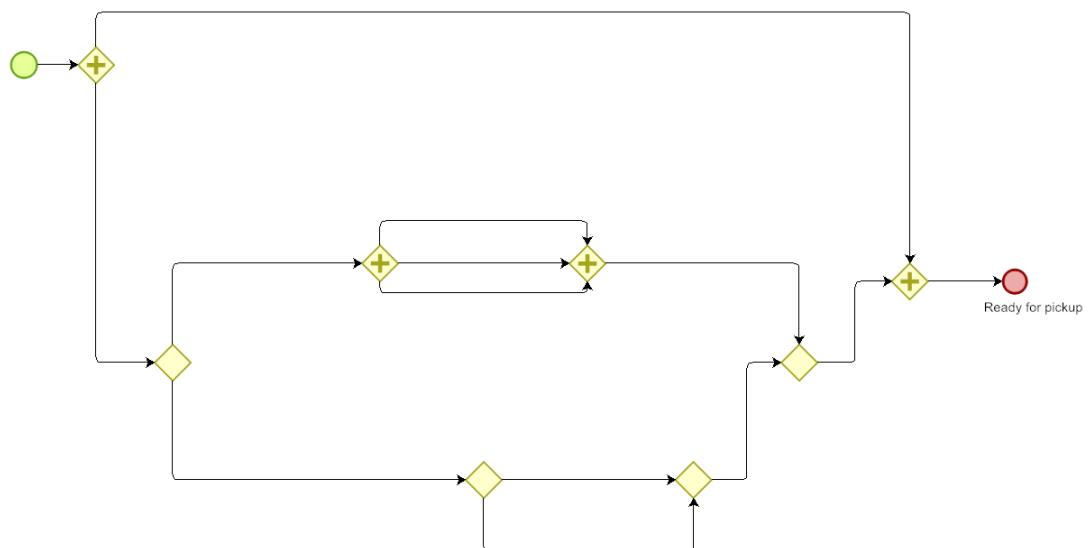
شکل ۳: نمونه‌ی آزمایشی شماره‌ی ۱



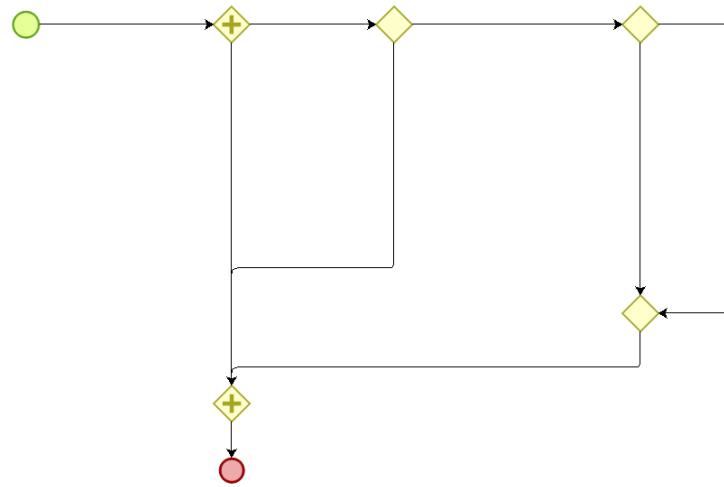
شکل ۴: نمونه‌ی آزمایشی شماره‌ی ۲



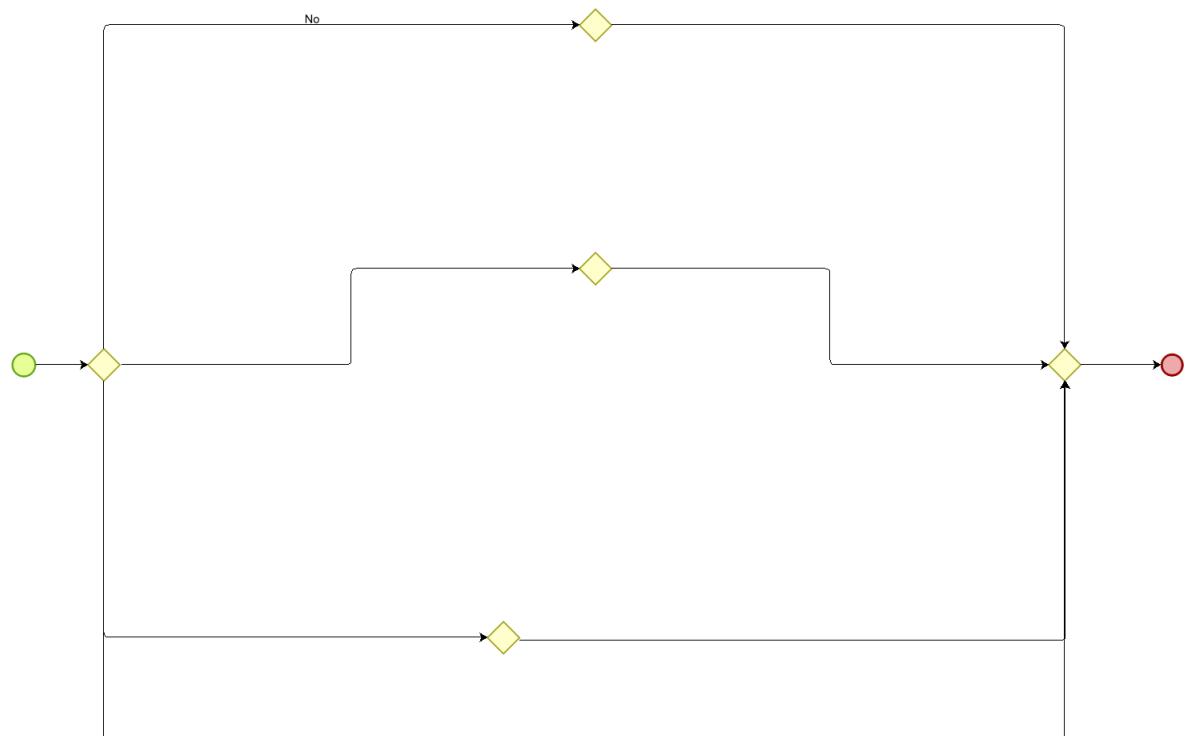
شکل ۵: نمونه‌ی آزمایشی شماره‌ی ۳



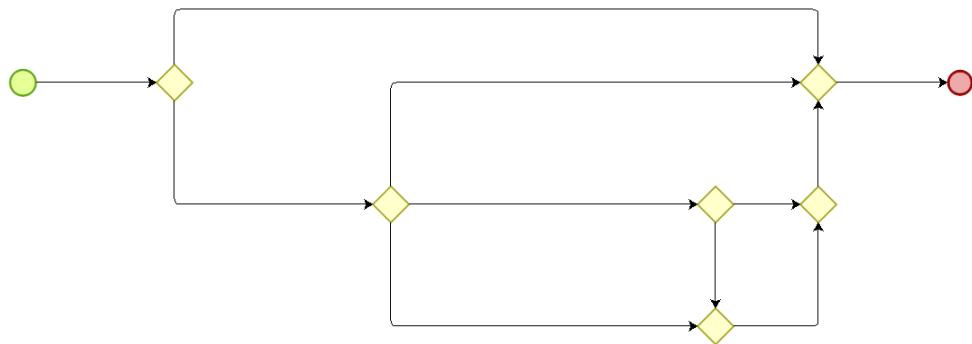
شکل ۶: نمونه‌ی آزمایشی شماره‌ی ۴



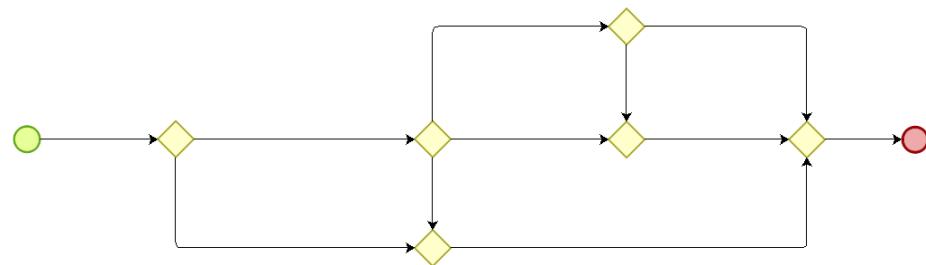
شکل ۷: نمونه‌ی آزمایشی شماره‌ی ۵



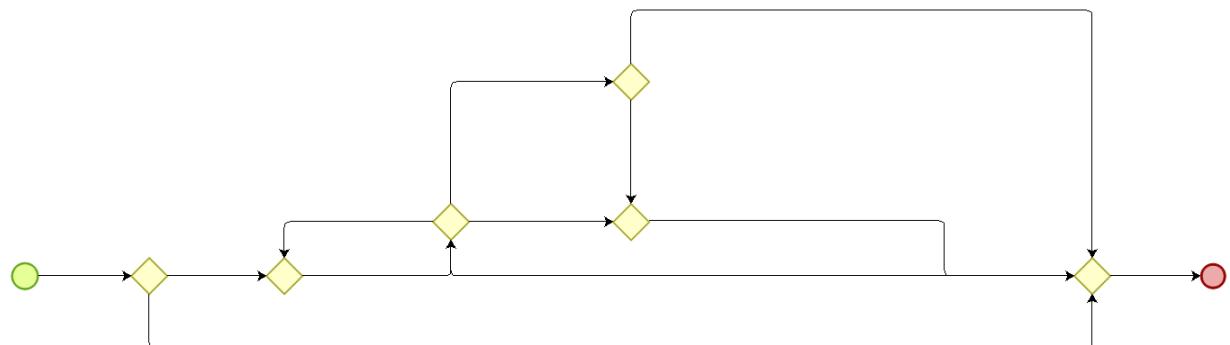
شکل ۸: نمونه‌ی آزمایشی شماره‌ی ۶



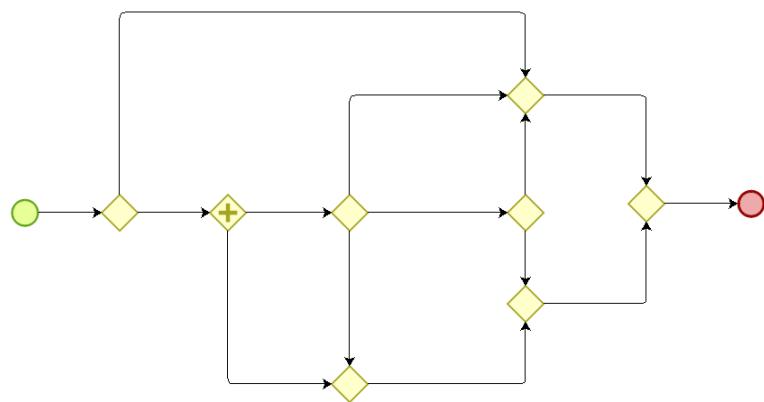
شکل ۹: نمونه‌ی آزمایشی شماره‌ی ۷



شکل ۱۰: نمونه‌ی آزمایشی شماره‌ی ۸



شکل ۱۱: نمونه‌ی آزمایشی شماره‌ی ۹



شکل ۱۲: نمونه‌ی آزمایشی شماره‌ی ۱۰

# واژه‌نامه‌ی فارسی به انگلیسی

Inspection .....	بازرسی (نرم‌افزار)	آ
Deadlock .....	بن‌بست	اتصال
Livelock .....	بن‌بست حلقوی	اتصال انحصاری
Low-Code .....	کم‌کد	اتصال عطفی
Pre process .....	پیش‌پردازش	اتصال فصلی
Static Analysis .....	تحلیل ایستا	استخراج
Transaction .....	تراکنش	اعتبارسنجی
Semantic Mistakes .....	خطاهای منطقی	انتقال
Lanes .....	خطوط شنا (خطوط شناوری)	اشعب
Gateway .....	دروازه	اشعب انحصاری
		انشعاب عطفی
		انشعاب فصلی
		بازبینی (نرم‌افزار)

Information Technology .	فناوری اطلاعات .	Rخداد . . . . .
	ک	ذ
Boundness .....	کراندار بودن (فرآیند) .	زنده بودن (فرآیند) . . . . .
Arc .....	کمان . . . . .	ش
Low code .....	کم کد . . . . .	شبکه‌های پتری . . . . .
Software Quality .....	کیفیت نرم افزار . . . . .	شبکه‌های پتری جریان کار Nets (WFP)
	گ	
Graph .....	گراف . . . . .	Firing . . . . .
Gateway Graph (GG) ...	گراف دروازه‌ای . . . . .	شلیک شدن . . . . .
	م	شی جریانی . . . . .
Process Model .....	مدل فرآیندی . . . . .	ص
Place .....	مکان . . . . .	صحت . . . . .
Case .....	مورد (مورد کاری) . . . . .	صحت‌سنجی . . . . .
	ن	صحیح . . . . .
Token .....	نشانه (توکن) . . . . .	صوری . . . . .
BPMN . . .	نمادگذاری فرآیندهای کسب و کار . . . . .	ف
Process Instance .....	نمونه‌ی فرآیندی . . . . .	فرآیند . . . . .
Semi formal .....	نیمه صوری . . . . .	فرآیندهای کسب و کار . . . . .
	و	فعال شدن . . . . .
Model Checking .....	وارسی مدل . . . . .	فعالیت . . . . .
Task .....	وظیفه . . . . .	

# واژه‌نامه‌ی انگلیسی به فارسی

<b>A</b>	Event .....	رخداد .....
Activity .....	فعالیت .....	
And Join .....	اتصال عطفی .....	Firing .....
And Split .....	انشعاب عطفی .....	شی جریانی .....
Arc .....	کمان .....	صوري .....
<b>B</b>		G
Boundness .....	کراندار بودن (فرآیند) .....	دروازه .....
Business Process ..	فرآیندهای کسب و کار ..	گراف دروازهای .....
BPMN ..	نمادگذاری فرآیندهای کسبوکار ..	گراف گرا .....
<b>C</b>		I
Case .....	مورد (مورد کاری) .....	فناوری اطلاعات .....
<b>D</b>		IT .....
Deadlock .....	بنبست .....	بازرسی (نرمافزار) .....
<b>E</b>		J
Enabling .....	فعال شدن .....	اتصال .....
<b>L</b>		

Lanes .....	خطوط شنا (خطوط شناوری) .....	خطاهای منطقی .....
Livelock .....	بنبست حلقوی .....	نیمه صوری .....
Liveness .....	زنده بودن (فرآیند) .....	کیفیت نرم افزار .....
Low-Code .....	کم کد .....	صحیح .....
<b>M</b>		صحت .....
Model Checking .....	وارسی مدل .....	انشعاب .....
<b>O</b>		Split .....
OR Join .....	اتصال فصلی .....	وظیفه .....
OR Split .....	انشعاب فصلی .....	نمایش (توکن) .....
<b>P</b>		تراکنش .....
Petri nets .....	شبکه‌های پتری .....	انتقال .....
Place .....	مکان .....	اعتبارسنجی .....
Pool .....	استخر .....	Validation .....
Pre process .....	پیش‌پردازش .....	Verification .....
Process .....	فرآیند .....	گردش کار - جریان کار .....
Process Instance .....	نمونه‌ی فرآیندی .....	Workflow .....
Process Model .....	مدل فرآیندی .....	گراف گردش کار .....
<b>R</b>		شبکه‌های پتری جریان کار .....
Review .....	بازبینی (نرم افزار) .....	Nets
<b>S</b>		<b>X</b>

Xor Join ..... اتصال انحصاری ..... Xor Split ..... انشعاب انحصاری .....

# **Abstract**

BPMN2.0 is a standard language for modeling of Business Processes. However Processes that are developed by the language are understandable, but because of its non-formal nature, they may throw semantic errors such deadlocks or live locks. Most of these process models can be executed by many business process management systems (BPMS) in different levels of an organization such as information systems. They also may be used to produce executable software by using a BPMN-based Low-Code Development Platform (LCDP). Therefor occurrence of the semantic errors can lead to mistakes in the organization's information systems, and consequently its operations. Therefor Static Analysis of the models is essential to increase the quality of the information systems before deploying them. In prior works, process models that were created by BPMN2.0 modelling language, have been mapped into formal models such as Petri Net and its extensions and were analyzed in that models.

In this research, based on general definition of Process Models and Control flow Patterns, a mapping between BPMN2.0 process models and Workflow Petri Nets have been proposed. Then based on that mapping, a verification method has been proposed using the concept of Soundness of Processes Patterns and Anti-patterns. Then, by introducing Gateway Graph Data structure and applying the method on it, BPMN2.0 process models have been verified. This method can be used as an algorithm for static analysis of BPMN models using soundness patterns and anti-patterns without need to map them to a formal model.

## **Key Words:**

BPMN2.0 Process Models, Structural Soundness, Workflow Petri nets, Model Driven Development(MDD)



**Amirkabir University of Technology  
(Tehran Polytechnic)**

**Department of Mathematics and Computer Science**

**M. Sc. Thesis**

# **Static Analysis of BPMN 2.0 Process Models**

**By**  
**Yahya Poursoltani**

**Supervisor**  
**Dr. Mohammad Hassan Shirali Shahreza**

**Advisor**  
**Dr. Seyyed Alireza Hashemi Golpayegani**

**October 2021**