

## A method for modeling and analysis of fault propagation in hybrid systems using stochastic activity networks

Arman Sanahmadi<sup>1</sup>, Mohammad Abdollahi Azgomi<sup>2\*</sup>

1- School of Computer Engineering, Iran University of Science and Technology, Tehran, Iran.

2\*- School of Computer Engineering, Iran University of Science and Technology, Tehran, Iran.

<sup>1</sup>arman\_sanahmadi@comp.iust.ac.ir, <sup>2\*</sup>azgomi@iust.ac.ir

Corresponding author address: Mohammad Abdollahi Azgomi, Room 308, School of Computer Engineering, Iran University of Science and Technology, Hengam St., Resalat Sq., Tehran, Iran, Postal Code: 16846-13114.

**Abstract-** Hybrid systems consist of both continuous and discrete parts. These systems include several different components. A fault in one of these components can be activated and propagate to other components. Due to the advancement of technology and intelligent systems such as driverless car, health control devices and automated factories, the occurrence of a fault in one component and its propagation to other components can lead to financial and human-life losses. It is necessary to design a fault propagation model before construction of a system. With such a model, we can observe the propagation of the effects of a fault in a component to other components, before the construction of the system. It is also possible to identify critical components of the system. In this paper, a method for modeling fault propagation based on stochastic activity networks is presented. Based on this model, it is possible to identify the critical points of the system, the effect of different components on each other and the component failure behavior. The model has been applied to an aircraft fuel system, and its simulation and quantitative results are presented.

**Keywords-** Modeling, fault propagation, hybrid systems, stochastic activity networks (SANs), quantitative evaluation.

## روشی برای مدل سازی انتشار خطا در سیستم های هیبرید با استفاده از شبکه های فعالیت تصادفی

آرمان سان احمدی<sup>۱</sup>، محمد عبداللهی ازگمی<sup>۲\*</sup>

۱- دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، ایران.

۲- دانشکده مهندسی کامپیوتر، دانشگاه علم و صنعت ایران، تهران، ایران.

<sup>1</sup>arman\_sanahmadi@comp.iust.ac.ir, <sup>2\*</sup>azgomi@iust.ac.ir

\* نشانی نویسنده مسئول: محمد عبداللهی ازگمی، تهران، میدان رسالت، خیابان هنگام، دانشگاه علم و صنعت ایران، دانشکده مهندسی کامپیوتر، اتاق

۳۰۴

چکیده- سیستم های هیبرید از دو بخش پیوسته و گسسته تشکیل شده اند. این سیستم ها شامل چندین مؤلفه متفاوت هستند. وجود خطا در یکی از این مؤلفه ها و فعال شدن آن، می تواند به سایر مؤلفه ها انتشار پیدا کند. با توجه به پیشرفت تکنولوژی و به وجود آمدن سیستم های هوشمند هم چون ماشین خودران، دستگاه های کنترل سلامت و کارخانه های خودکار، وقوع خطا در یک مؤلفه و انتشار آن به سایر مؤلفه ها می تواند منجر به وقوع فاجعه و خسارات مالی و جانی فراوانی شود. بنابراین به منظور شناخت نقاط حساس سیستم و نحوه انتشار خطا بین مؤلفه ها، لازم است قبل از طراحی و بهره برداری از این سیستم ها، به مدل سازی انتشار خطا در آن ها پرداخت. در این مقاله روشی برای مدل سازی انتشار خطا بر اساس شبکه های فعالیت تصادفی ارائه شده است. بر اساس این مدل می توان به شناسایی نقاط حساس سیستم، تأثیر مؤلفه های مختلف بر روی همدیگر و رفتار خرابی مؤلفه ها پرداخت. مدل ارائه شده بر روی یک سیستم سوخت رسان هواپیما اعمال شده است و نتایج شبیه سازی و ارزیابی کمی آن آورده شده است.

واژه های کلیدی: مدل سازی، انتشار خطا، سیستم های هیبرید، شبکه های فعالیت تصادفی، ارزیابی کمی

### ۱- مقدمه

یک از این مؤلفه ها می تواند به دلیل نقص فنی، فرسودگی مؤلفه ها،

عمدی یا غیرعمدی باشد [۳].

چگونگی انتشار خطا در بین مؤلفه ها و بررسی پارامترهای مختلف اتکاپذیری، در محیط آزمایشگاهی و واقعی، هزینه بر و غیرممکن است [۴،۵]. از طریق مدل انتشار خطا می توان به مشاهده پارامترهای مختلفی از جمله رفتار خرابی مؤلفه ها در حضور و عدم حضور خطا در یک مؤلفه دیگر، قابلیت اطمینان<sup>۴</sup> کل سیستم و شناسایی مؤلفه های حساس سیستم پرداخت. در اینجا منظور از مؤلفه های حساس سیستم، مؤلفه هایی هستند که در صورت خرابی

سیستم های هیبرید<sup>۱</sup> از چندین مؤلفه فیزیکی و رایانشی تشکیل شده اند. این مؤلفه ها به منظور انجام هدف سیستم با همدیگر همکاری می کنند [۱،۲]. امروزه سیستم های هیبرید در حوزه های مختلفی مانند کنترل سلامت، حمل و نقل خودکار و کارخانه های صنعتی کاربرد فراوانی دارد. با توجه به کاربرد حساس این سیستم ها، وقوع خطا<sup>۲</sup> در یک بخش سیستم می تواند به سایر بخش ها انتشار<sup>۳</sup> پیدا کند و منجر به وقوع فاجعه گردد. در هر یک از این مؤلفه ها می تواند یک یا چندین خطا وجود داشته باشد. وجود خطا در هر

عوامل مختلفی از جمله: مشخصه‌سازی نادرست سیستم، طراحی نادرست، استفاده از ابزار نامناسب و عامل‌های بیرونی داشته باشد. اشکال به خطایی گفته می‌شود که فعال شده باشد. در واقع علت یک اشکال، همان خطا است. اشکال بخشی از وضعیت کل سیستم است که می‌تواند منجر به خرابی کل سیستم شود. حالت سیستم مجموعه‌ای از حالت‌های تک تک مؤلفه‌ها است. هنگامی که یک اشکال درون یک مؤلفه وجود دارد تا زمانی که آن اشکال به یک حالت خارجی آن مؤلفه انتقال پیدا نکند نمی‌تواند باعث خرابی کل سیستم شود [۷].

یک سیستم دچار خرابی شده است هرگاه خدمات مربوط به این سیستم از حالت صحیح آن انحراف داشته باشد. در واقع یک اشکال در صورت انتشار به محیط خارج از مؤلفه باعث ایجاد خرابی می‌شود. آسیب‌شناسی<sup>۹</sup> خرابی که نحوه ارتباط خطا، اشکال و خرابی را مشخص می‌کند در شکل ۱ آورده شده است [۷].



شکل ۱: آسیب‌شناسی خرابی [۷].

## ۲-۲- شبکه‌های فعالیت تصادفی

شبکه فعالیت تصادفی یک توسعه از شبکه‌های پتری است. شبکه‌های فعالیت تصادفی به منظور ارزیابی کارایی و اتکاپذیری سیستم‌ها در دهه ۸۰ گسترش داده شده است [۸]. هر شبکه فعالیت تصادفی یک ۱۲-تایی به صورت  $SAN = (P, A, I, O, \gamma, \tau, i, o, \mu_0, C, F, G)$  است که تعریف هر یک از این نمادها عبارتند از:

- **P**: مجموعه متناهی از مکان‌ها.
- **A**: مجموعه‌ای متناهی از فعالیت‌ها.
- **I**: مجموعه متناهی از دروازه‌های ورودی.
- **O**: مجموعه متناهی از دروازه‌های خروجی.
- **$\gamma$** : نشان‌دهنده تعداد اقدام‌های احتمالی متفاوت برای هر فعالیت است.
- **$\tau$** : مشخص‌کننده نوع هر فعالیت است (فوری یا زمانی).
- **i**: نداشت‌کننده دروازه‌های ورودی به فعالیت‌ها است.
- **o**: نداشت‌کننده دروازه‌های خروجی به فعالیت‌ها است.
- **$\mu_0$** : نشان‌گذاری اولیه است که باید پایدار باشد.
- **C**: تابع تخصیص توزیع‌های احتمالی مربوط به اقدام‌های احتمالی فعالیت‌ها است.
- **F**: تابع تخصیص نرخ‌های احتمالی فعالیت‌ها است.
- **G**: تابع تخصیص فعال‌سازی مجدد فعالیت‌ها است.

منجر به توقف کامل سیستم می‌شوند. می‌توان قبل از طراحی سیستم‌ها با مدل‌سازی انتشار خطا به بررسی مناسب بودن پارامترهای قابلیت اطمینان، ایمنی و امنیت قطعات بخش‌های مختلف سیستم پرداخت. با استفاده از نتایج مدل‌سازی می‌توان مشخص کرد چه نوع مؤلفه‌هایی برای ساخت این سیستم مناسب هستند [۶].

در کارهای انجام‌گرفته معمولاً معیارهای کافی جهت مدل‌سازی انتشار خطا در نظر گرفته نشده است. همچنین منطق انتشار خطا به درستی پیاده نشده است، که با توجه به پیچیدگی سیستم‌ها این مدل‌ها نمی‌توانند دقیق باشند. مدل ارائه شده باید دارای ویژگی‌هایی همچون مقیاس‌پذیری<sup>۵</sup> باشد که بتوان علاوه بر مدل‌سازی انتشار خطای یک سیستم به صورت جداگانه، رفتار انتشار خطای سیستم‌های مختلف هنگامی که با همدیگر همکاری می‌کنند را مشاهده کرد. همچنین مدل ارائه‌شده باید پویا باشد. مشکل مدل‌های ایستا شناسایی مسیرهای انتشار خطا در سیستم است که در بسیاری از مواقع هزینه‌بر و غیرممکن است. همچنین مدل باید معیارهای کافی همچون وابستگی‌های مستقیم و غیرمستقیم موجود در بین مؤلفه‌های سیستم را در نظر بگیرد.

در روش پیشنهادی این مقاله ابتدا بر اساس همبندی<sup>۶</sup> و مشخصات سیستم، گراف داده‌ای، گراف غیرداده‌ای و گراف ترتیب اجرایی مؤلفه‌های سیستم استخراج می‌شود. همچنین عبارت‌های احتمالی مربوط به رفتار هر یک از مؤلفه‌ها استخراج می‌گردد. سپس بر اساس این سه گراف و عبارت‌های احتمالی، مدل شبکه‌های فعالیت تصادفی<sup>۷</sup> هر یک از مؤلفه‌ها با استفاده از قوانین تبدیلی که ارائه خواهد شد استخراج می‌گردد. در نهایت با استفاده از صورت‌بندی تکرار-الحاق<sup>۸</sup>، مؤلفه‌های مختلف سیستم با همدیگر الحاق می‌شوند و مدل نهایی سیستم به دست خواهد آمد. به منظور ارزیابی سیستم، با تزریق خطا در بخش‌های مختلف سیستم و فعال‌سازی آن می‌توان به بررسی و مشاهده رفتار سایر مؤلفه‌ها در حضور و عدم حضور خطا و به دست آوردن قابلیت اطمینان کل سیستم پرداخت.

## ۲- مفاهیم پایه

در این بخش مفاهیم پایه‌ای که در حوزه این مقاله لازم است، توضیح داده شده است.

### ۲-۱- خطا، اشکال و خرابی

خطا به نقصی گفته می‌شود که در داخل سیستم وجود دارد. هر خطا می‌تواند فعال گردد و به یک اشکال تبدیل شود. خطا می‌تواند

## ۳- کارهای مرتبط

در این بخش به بررسی کارهایی که تاکنون در این زمینه انجام شده است پرداخته می‌شود.

## ۳-۱- مدل‌های با دید گرافی یا شبکه‌ای به سیستم

در کار شماره [۹] به مدل‌سازی انتشار خطا بین زیرساخت‌های حیاتی و وابسته به هم پرداخته شده است. بسیاری از زیرساخت‌های حیاتی به یکدیگر وابسته هستند. این وابستگی‌ها بین زیرساخت‌ها اجتناب‌ناپذیر است و برای عملکرد صحیح کل شبکه نیاز است. با توجه به این وابستگی، وقوع خطا در یک زیرساخت می‌تواند انتشار پیدا کند و سایر زیرساخت‌های مرتبط را تحت تأثیر قرار دهد. این انتشار می‌تواند منجر به وقوع فاجعه‌های بزرگ‌تری شود. به‌طور مثال در سال ۲۰۰۳ وقوع یک خطا در شبکه توزیع برق ایتالیا منجر به خاموشی گسترده و قطعی زیرساخت اینترنت ایتالیا شد. در این مقاله یک چارچوب زنجیره مارکوف<sup>۱۰</sup> با وابستگی متقابل ارائه شده است. این چارچوب، یک چارچوب احتمالی برای بررسی و ارزیابی تأثیر وجود خطا و رخ دادن خرابی بین زیرساخت‌های حیاتی است. هدف آن مدل‌سازی نحوه انتشار خرابی و بررسی تأثیر آن بر روی قابلیت اطمینان کل سیستم است. زنجیره مارکوف با وابستگی متقابل به این صورت است که ابتدا زنجیره مارکوف هر زیرساخت کشیده می‌شود سپس به روش خاصی این زنجیره مارکوف‌ها در کنار یکدیگر قرار می‌گیرند. زنجیره مارکوف حاصل‌شده از کنار قرار گرفتن این دو زنجیره مارکوف نیز ارگوریک است. این چارچوب بیشتر برای شبکه‌های زیرساختی مثل شبکه‌های توزیع برق، گاز و آب مناسب است و در آن تمامی گره‌های موجود در زیرساخت یکسان فرض می‌شود و تفاوتی از لحاظ ساختاری بین مؤلفه‌های رایانشی و فیزیکی در نظر گرفته نشده است. برای ارزیابی مدل ارائه شده به بررسی تعداد خطوط خراب‌شده در شبکه توزیع برق و گستردگی قطعی در شرایط مختلف پرداخته شده است. همین افراد در مقاله [۱۰] به بررسی انتشار خطا همراه با در نظر گرفتن تأثیر خطای انسانی پرداخته‌اند.

در مقاله [۱۱] همانند مقاله قبلی به بررسی انتشار خطا در سیستم‌های شبکه‌ای توزیعی پرداخته شده است. از آن جهت که سیستم‌های هیبرید دارای دو زیرشبکه شامل مؤلفه‌های رایانشی و فیزیکی است، برای مدل‌سازی با این روش به دو زیرشبکه با نام A و B نیاز است. ابتدا بر اساس همبندی سیستم ارتباط بین گره‌های A و B کشیده می‌شود. سپس هنگامی که یک گره از یکی از زیرشبکه‌ها خراب می‌شود تمامی یال‌هایی که از آن گره به سایر گره‌های زیرشبکه داخلی یا خارجی وصل شده است، حذف می‌گردد.

پس از حذف یال‌های مربوطه، هر یک از زیر شبکه‌ها را می‌توان به چندین خوشه تبدیل کرد. در اینجا منظور از خوشه مجموعه گره‌هایی است که فقط با خود ارتباط دارند و با سایر گره‌های موجود در زیرشبکه خود ارتباطی ندارند. این فرایند آن‌قدر ادامه پیدا خواهد کرد تا شبکه به یک حالت پایدار برسد و دیگر نتوان گره یا یالی را حذف کرد. مشکلی که این روش دارد این است که حذف گره‌ها و ارتباط آن با سایر گره‌ها بر اساس یک منطق ساده است و ممکن است یک گره در سیستم‌های پیچیده هیبرید صرفاً به خاطر از دست دادن ارتباط خود با یک گره خاص کارایی خود را از دست ندهد و هنوز بتواند در شبکه سیستم فعالیت کند. البته این نحوه‌ی مدل‌سازی بیشتر برای سیستم‌های هیبرید تورین مثل شبکه‌های توزیع برق و انرژی کاربرد دارد

در مقاله [۱۲] روشی برای ارزیابی انتشار خطا در سیستم‌های سایبر-فیزیکی ارائه شده است. در این روش مانند روش قبلی روش زیر شبکه در نظر گرفته شده است. تمامی مؤلفه‌های هر شبکه یکسان فرض شده است. در این روش هنگامی که یک مؤلفه دچار خرابی می‌شود گره مربوط به آن حذف می‌شود. بعد از حذف یال‌های مربوط به گره حذف‌شده در صورتی که یک گره‌ای ایجاد شود که به هیچ مؤلفه‌ای وصل نباشد آن گره نیز حذف خواهد شد. این کار ادامه پیدا خواهد کرد تا سیستم به یک حالت پایدار برسد. روش ارزیابی مدل به صورت قابلیت اطمینان مرتبه k است. قابلیت اطمینان مرتبه k به این معنا است که در صورتی که یکی از شبکه‌های سایبری یا فیزیکی بیشتر از k گره سالم داشته باشد آن شبکه می‌تواند به عملکرد خود ادامه دهد. یکی از مشکلات این روش این است که پیچیدگی سیستم‌های سایبر-فیزیکی<sup>۱۱</sup>، تفاوت مؤلفه‌های آن‌ها و ارتباط پیچیده آن‌ها مخصوصاً ارتباط بین مؤلفه‌های سایبری و فیزیکی درست دیده نشده و مدل بسیار سطح بالا و دور از واقعیت است.

در مقاله [۱۳] به ارزیابی انتشار خرابی در سیستم‌های سایبر-فیزیکی پرداخته شده است. این مقاله برخلاف مقاله قبلی بیشتر بر روی نحوه‌ی ارتباط گره‌های سایبری و فیزیکی کار کرده است و ارتباط آن‌ها را صرفاً وجود یال در بین آن‌ها را در نظر نگرفته است. در این روش یک قانونی در نظر گرفته شده است که هر گره محاسباتی باید توسط یک گره فیزیکی پشتیبانی شود، یعنی یک گره محاسباتی در صورتی خراب فرض می‌شود که ارتباط خود را با بخش فیزیکی از دست بدهد. اما هر گره محاسباتی می‌تواند توسط چندین گره فیزیکی پشتیبانی شود و در صورتی که ارتباط خود را با بخش محاسباتی از دست بدهد عملکرد خود را از دست می‌دهد. نحوه‌ی انتشار خرابی نیز در این مدل با روش‌های دیگر متفاوت

هستند. در این مقاله نشان داده شده است برای اینکه بتوانیم یک مدل صحیح انتشار اشکال در این سیستم‌ها داشته باشیم باید جریان داده و جریان کنترل به صورت جداگانه دیده شود. ایده اصلی این روش بر روی دو گراف کنترل داده و جریان داده قرار گرفته است. این دو گراف در هر فازی از طراحی سیستم می‌توانند تولید شوند و انتشار اشکال ارزیابی شود.

در مقاله [۱۸] به ارزیابی پویای ایمنی سیستم‌های پیچیده پرداخته شده است. مؤلفه‌های موجود در سیستم‌های پیچیده پویا دارای وابستگی‌های وابسته به زمان و عملکردی هستند. بنابراین برای مشاهده رفتار شکست یک سیستم باید علاوه بر مشاهده چگونگی ترکیب شکست مؤلفه‌ها، ترتیب زمانی آن‌ها نیز مشاهده شود. در این مقاله یک روش ارزیابی مبتنی بر مدل درخت خطاهای زمانی با استفاده از ترکیب Hip-hops با Pandora ارائه شده است. این دو روش در مقالات [۱۹، ۲۰] توضیح داده شده است. برای مدل‌سازی با این روش سه فاز مختلف باید به ترتیب انجام شود. در فاز اول مدل سیستم و نحوه‌ی خرابی آن باید مشخص سپس ارتباط مؤلفه‌ها و چگونگی قرار گرفتن آن‌ها در کنار یکدیگر تعیین گردد. در نهایت وابستگی‌های ترتیبی باید در مدل اضافه شود. بعد از مشخص کردن معماری مؤلفه‌ها و عبارت‌های خرابی، با استفاده از Hip-Hops درخت خطای زمانی استخراج می‌شود. سپس این درخت زمانی خطا به مدل شبکه پتری یا شبکه بیزی تبدیل می‌شود و بر روی آن تحلیل‌های لازم انجام می‌گردد. این کار یکی از کارهای نوین و خوبی است که در این زمینه انجام شده است. مشکلی که این مدل دارد این است که همانند درخت خطا تمامی رخدادهای خطا را باید مشخص کرد و حتی ترتیب آن‌ها را هم باید در نظر گرفت. تمامی رخدادها در این مدل مربوط به ورودی و خروجی است و نرخ آن‌ها ثابت است. این روش در مقاله [۲۱] ارزیابی شده است.

در مقاله [۲۲] روشی برای تولید خودکار مدل انتشار خطا ارائه شده است. در این مقاله ابتدا با مشخص کردن نحوه‌ی خرابی مؤلفه‌ها در برابر خطاهای گوناگون، حاشیه‌نویسی خرابی را برای هر مؤلفه ایجاد می‌کند. سپس با استفاده از این حاشیه‌نویسی‌ها درخت خطای مؤلفه‌ها تولید می‌شوند که می‌تواند برای تحلیل انتشار خطا استفاده گردد. مشکلی که این مقاله دارد این است که ساختار آن ایستا است و این مدل برای ارزیابی کمی مناسب نیست.

در مقاله [۲۳] به ارزیابی انتشار خطاهای گذرا در سیستم‌های کنترل شبکه پرداخته شده است. از مشکلات این مدل سختی پیاده‌سازی و وارد کردن جزئیات هر مؤلفه است.

در مقاله [۲۴] روشی برای ارزیابی ریسک با توجه به انتشار خطا در سیستم‌های سایبر-فیزیکی پرداخته شده است. در این روش

در این مدل یک گره به صورت تصادفی حذف خواهد شد و تمامی لینک‌های خارجی و داخلی آن حذف می‌شود سپس بزرگ‌ترین زیرشبکه را در هر یک از شبکه‌های سایبری و فیزیکی در نظر گرفته می‌شود و سایر زیرشبکه‌ها نیز حذف می‌گردد. این زیرشبکه‌ها گره‌های در معرض خطر نامیده می‌شود. در کار [۱۴] به بررسی وابستگی و انتشار خرابی در سیستم‌های تورین بر اساس نظریه شبکه‌های پیچیده پرداخته شده است.

### ۳-۲- مدل‌های با سطح انتزاع مؤلفه

در مقاله [۶] روشی جدید برای ارزیابی هم‌زمان ایمنی و امنیت سیستم‌های سایبر-فیزیکی در قالب درخت خطا حالت/رخداد ارائه شده است. از این مدل برای مشاهده چگونگی انتشار خطا در سیستم‌های پیچیده استفاده شده است. برخلاف روش‌های دیگر که همه‌ی مؤلفه‌ها را یکسان فرض می‌کردند، این روش هر مؤلفه را به صورت جداگانه و همراه با ویژگی‌های خاص آن طراحی نموده است. ویژگی‌های مختلف هر مؤلفه مانند نرخ خرابی و نرخ تعمیر در این مدل آورده شده است. ارتباط مؤلفه‌ها در این روش از طریق رخدادهای متفاوت بین آن‌ها است. این رخدادها از طریق درگاه‌های موجود بر روی مؤلفه‌ها انتقال پیدا می‌کنند. مشکلی که این روش دارد این است که همانند درخت خطا باید رخداد نهایی مشخص باشد و فقط مشاهده کرد که چه رخدادهایی پشت سر هم اتفاق خواهند افتاد که منجر به وقوع آن خواهند شد و ساختار مدل یک ساختار کاملاً ایستا است.

در مقاله [۱۵] روشی برای مدل‌سازی انتشار خطا در سیستم‌های هیبرید ارائه شده است. در این روش از آتاماتای هیبرید برای مدل‌سازی فرایند انتشار خطا استفاده شده است. هر مؤلفه به صورت مجزا همراه با تمام جزئیات مدل‌سازی می‌شود. در این روش چهار نوع وابستگی شامل وابستگی فیزیکی، سایبری، جغرافیایی و منطقی در نظر گرفته شده است.

مدل هر یک از مؤلفه‌ها طراحی می‌شود. سپس با خراب کردن یک مؤلفه نشان داده می‌شود که این خرابی چگونه در سطح سیستم انتشار پیدا می‌کند. با استفاده از این روش می‌توان به مطالعه انتشار خطا به دلیل وابستگی‌های مختلف در سیستم، ارزیابی آسیب‌پذیری سیستم و توسعه راهبردهای<sup>۱۲</sup> نگه‌داری

مشکل اصلی این روش جزئیات فراوان مؤلفه‌ها است. بنابراین طراحی و ساخت این مدل پیچیده می‌شود. عملاً باید کل سیستم مورد مطالعه از نظر ساختاری و رفتاری مدل‌سازی کرد.

در مقاله‌های [۱۶، ۱۷] روشی برای مدل‌سازی انتشار اشکال در سیستم‌های مکترونیک ارائه شده است. سیستم‌های مکترونیک معمولاً غیرهمگن و دارای مؤلفه‌های متفاوت نرم‌افزاری و فیزیکی



#### ۲-۴- گراف وابستگی داده‌ای

اولین ارتباطی که مؤلفه‌های یک سیستم باهمدیگر دارند، ارتباط ورودی و خروجی است. بنابراین وجود گرافی که نحوه ارتباط مؤلفه‌ها را مشخص کند لازم و ضروری است. گراف وابستگی داده‌ای را می‌توان از همبندی شبکه به دست آورد این گراف مشخص می‌کند که هر یک از مؤلفه‌های سیستم به کدام یک از مؤلفه‌های دیگر داده می‌فرستد و از کدام مؤلفه‌ها ورودی می‌گیرد. در واقع ساده‌ترین وابستگی که بین مؤلفه‌ها وجود دارد این نوع وابستگی است و جزء بدیهی‌ترین پارامترهایی است که باید در مسئله انتشار خطا در نظر گرفته شود. این گراف در واقع همان ساختار همبندی سیستم است. هر مؤلفه می‌تواند چندین ورودی و خروجی داشته باشد. در صورتی که یک مؤلفه چندین ورودی برای یک مؤلفه دیگر فراهم کند باید به تعداد ورودی‌ها میان آن‌ها یال کشیده شود. به طوری که هر یال نشان‌دهنده یک ارتباط است. شکل ۳ یک نمونه از گراف وابستگی داده‌ای را نشان می‌دهد. تعریف گراف وابستگی در فرمول ۱ آورده شده است.

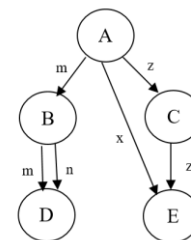
$$G^{DF} = [N, D^{DF}] \quad (1)$$

$$N = \{n_1, \dots, n_m\}$$

$$D^{DF} = \{(n_i, n_j, v_{i,j}), \dots, (n_k, n_l, v_{k,l})\}$$

$N$  مجموعه‌ای است که نشان‌دهنده رأس‌های گراف است. هر یک از این رأس‌ها، یک مؤلفه سیستم است. یال‌های بین آن‌ها که با مجموعه  $D^{DF}$  مشخص شده است ارتباط ورودی و خروجی مؤلفه‌ها است.

با توجه به شکل ۳ می‌توان نتیجه گرفت که ورودی مؤلفه  $E$ ،  $C$ ،  $B$  توسط مؤلفه  $A$  تأمین می‌شود و مؤلفه  $B$  دو ورودی برای  $D$  ارسال می‌کند همچنین ورودی  $E$  از طریق مؤلفه  $C$  تأمین می‌شود.



شکل ۳: یک نمونه گراف وابستگی داده‌ای.

#### ۳-۴- گراف وابستگی غیرداده‌ای

مؤلفه‌های یک سیستم ممکن است به صورت غیرمستقیم بر روی همدیگر تأثیر بگذارند. و این وابستگی بر روی مدل‌سازی انتشار خطا تأثیرگذار خواهد بود. گراف وابستگی غیرداده‌ای مشخص می‌کند که چه مؤلفه‌هایی به یکدیگر وابستگی دارند. این وابستگی از نوع وابستگی داده‌ای بخش قبل نیست. در اینجا منظور از وابستگی

تأثیری است که حضور یا عدم حضور یک مؤلفه بر نرخ خرابی یک مؤلفه دیگر می‌گذارد. به‌طور مثال خرابی مؤلفه خنک‌کننده باعث می‌شود نرخ خرابی یک مؤلفه دیگر بالاتر رود. در این گراف هر گره در صورت وجود وابستگی تنها یک یال به گره دیگر دارد. یک نمونه از گراف وابستگی غیرداده‌ای در شکل ۴ نشان داده شده است. این نوع وابستگی پارامتری است که در کارهای پیشین دیده نشده است. در حالی که پارامتری مهمی بوده و در بسیاری از سیستم‌ها این وابستگی غیرمستقیم بین مؤلفه‌ها وجود دارد. هر مؤلفه‌ای داخل سیستم دارای یک نرخ خرابی است. عددی که بر روی یال‌های گراف وابستگی غیرداده‌ای وجود دارد، اگر بدون علامت ریاضی باشد به این معنا است که در صورت خرابی مؤلفه پدر، نرخ خرابی گره فرزند با مقدار روی یال جمع می‌شود.

همچنین در صورتی که قبل از عدد روی یال گراف، علامت مساوی «=» قرار داده شده باشد. به این معنا است که نرخ خرابی مؤلفه برابر با آن مقدار می‌شود. اگر بر روی یال گراف علامت بی‌نهایت قرار داده شده باشد به این معنا است که در صورت خرابی مؤلفه پدر، مؤلفه فرزند آن فوراً خراب خواهد شد. همچنین در صورتی که علامت «\*» وجود داشته باشد به این معنا است که نرخ خرابی در عدد نوشته شده ضرب خواهد شد.

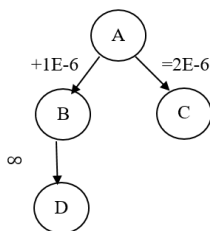
تعریف گراف وابستگی غیرداده‌ای در فرمول ۲ نشان داده شده است.

$$G^{NDF} = [N, D^{NDF}] \quad (2)$$

$$N = \{n_1, \dots, n_m\}$$

$$D^{NDF} = \{(n_i, n_j, v_{i,j}), \dots, (n_k, n_l, v_{k,l})\}$$

با توجه به شکل ۴ هنگامی که مؤلفه  $A$  خراب شود، نرخ خرابی مؤلفه  $B$  برابر با مقدار  $2E-6$  و نرخ خرابی مؤلفه  $C$  به اندازه  $1E-6$  افزایش پیدا خواهد کرد. همچنین در صورت خرابی مؤلفه  $B$  مؤلفه  $D$  فوراً دچار خرابی می‌گردد.



شکل ۴: یک نمونه گراف وابستگی غیرداده‌ای.

#### ۴-۴- گراف ترتیب اجرایی

گراف ترتیب اجرایی مشخص می‌کند که بعد از خرابی یک مؤلفه چه مؤلفه‌هایی باید کار خود را شروع می‌کنند. در سیستم‌های هیبرید معمولاً افزونگی مؤلفه وجود دارد تا در صورت خرابی یک مؤلفه، مؤلفه دیگری کار خود را شروع کند و جایگزین آن شود این

باشد. در واقع هر مؤلفه با توجه به این مقادیر تصمیم می گیرد که هم اکنون در حالت مشغول به کار باشد یا خیر.

#### ۴-۶- عبارتهای احتمالی

مؤلفه هایی که ورودی دارند طبق روش های احتمالی FPTA که قبلاً در مقاله [۲۵] ارائه شده است. می توانند رفتار احتمالی داشته باشند. در این مقاله نیز از عبارتهای احتمالی FPTA با تغییراتی جزئی استفاده شده است. در روش پیشنهادی ما، فرض شده است که هر ورودی یا خروجی دو حالت صحیح یا ناصحیح دارد. هر مؤلفه می تواند چندین ورودی داشته باشد. هریک از این ورودی ها می تواند صحیح یا ناصحیح باشد. رفتار مؤلفه با توجه به حالت های مختلفی که این ورودی ها دارند، می تواند احتمالی باشد. در واقع این عبارتهای نحوه ی انتشار خرابی از یک مؤلفه به مؤلفه های دیگر را به صورت احتمالی مشخص می کند. تمامی عبارتهای احتمالی نیازی نیست که نوشته شوند و اگر فقط برای حالت هایی که خروجی صحیح است نوشته شود سایر عبارتهای احتمالی قابل استنتاج است. بنابراین در صورتی که یک مؤلفه n ورودی و m خروجی مختلف داشته باشد به تعداد  $2^m * 2^n$  عبارت احتمالی متفاوت نیاز داریم که  $2^m$  حالت آن را می توان به صورت خودکار استخراج نمود. عبارتهای احتمالی به صورت مجموعه معادله ۴ نشان داده می شود.

$$\begin{aligned} 1) & \text{Input1. } T, \text{Input2. } T \rightarrow \text{Output1. } T, 1.0 \\ 2) & \text{Input1. } F, \text{Input2. } T \rightarrow \text{Output1. } T, 0.6 \\ 3) & \text{Input1. } T, \text{Input2. } F \rightarrow \text{Output1. } T, 0.7 \\ 4) & \text{Input1. } F, \text{Input2. } F \rightarrow \text{Output1. } T, 0.02 \end{aligned} \quad (4)$$

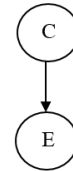
معادله های ۴ صحیح یا ناصحیح بودن خروجی مؤلفه در حالت های مختلفی که ورودی ها می توانند داشته باشند را نشان می دهد. به طور مثال با توجه به عبارت دوم هنگامی که ورودی اول ناصحیح باشد و ورودی دوم صحیح باشد این مؤلفه به احتمال ۰.۶ خروجی صحیح تولید می کند و به احتمال ۰.۴ مقدار خروجی این مؤلفه ناصحیح است.

#### ۴-۷- تبدیل گراف وابستگی داده ای به مدل SAN

برای تبدیل این گراف به مدل SAN هرگاه از مؤلفه نوعی A به مؤلفه نوعی B یالی وجود داشته باشد، به معنای این است که A باید برای B ورودی ایجاد کند و آن را در اختیار B قرار دهد. این وابستگی از طریق یک مکان بسط یافته<sup>۱۴</sup> در مدل SAN ایجاد می گردد. ورودی ها و خروجی ها در مدل پیشنهادی می تواند دو حالت مختلف داشته باشد، حالتی که ورودی صحیح است و حالتی که ورودی ناصحیح است. بنابراین نشانه موجود در این مکان می تواند یک عدد صحیح باشد که عدد یک به معنای داده صحیح و عدد دو به معنای داده ناصحیح است.

نوع ترتیب اجرایی با یک گراف نشان داده خواهد شد. در شکل ۵ یک گراف ترتیب اجرایی کوچک نشان داده شده است. شکل ۵ نشان می دهد که کار مؤلفه E در صورتی شروع می شود که مؤلفه C خراب شود. در فرمول ۳ تعریف گراف ترتیب اجرایی نشان داده شده است.

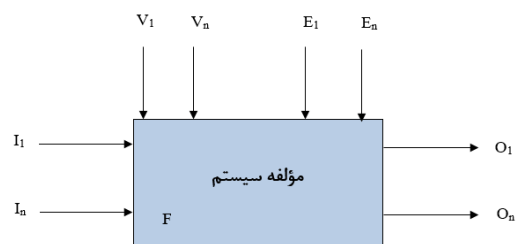
$$\begin{aligned} G^{CF} &= [N, D^{CF}] \\ N &= \{n_1, \dots, n_n\} \\ D^{CF} &= \{(n_i, n_j), \dots, (n_k, n_l)\} \end{aligned} \quad (3)$$



شکل ۵: یک نمونه گراف ترتیب اجرایی.

#### ۴-۵- رابطهای مؤلفه های سیستم

شکل ۶ رابطهایی که هر یک از مؤلفه های سیستم می توانند داشته باشند را نشان می دهد. این رابطها برای مؤلفه های رایانشی و فیزیکی صدق می کند. در هر مؤلفه علاوه بر رابطها، یک حالت به نام F وجود دارد که نشان دهنده سالمی یا خرابی مؤلفه است. هرگاه یک مؤلفه دچار خرابی شود و نتواند فعالیت عادی خود را دنبال کند مقدار این حالت تنظیم می شود.

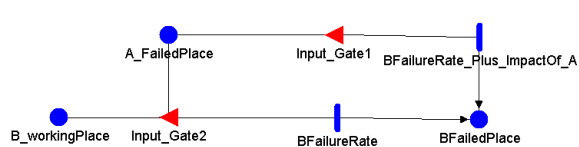


شکل ۶: رابطهای هر مؤلفه سیستم.

با توجه به شکل ۶ هر مؤلفه سه دسته ورودی و یک دسته خروجی مختلف دارد که عبارتند از:

- **ورودی های داده ای:**  $I_i$  ها در شکل ۶ ورودی های داده ای را مشخص می کند که از طریق مؤلفه های دیگر تأمین می شود. این مقادیر از گراف وابستگی داده ای استخراج می شود
- **ورودی های غیر داده ای:** ورودی های  $E_i$  تأثیری است که هر مؤلفه از سایر مؤلفه ها می پذیرد. به طور مثال خرابی یک مؤلفه باعث افزایش نرخ خرابی مؤلفه ی دیگری می شود این مقادیر از گراف وابستگی غیر داده ای استخراج می شود
- **خروجی های داده ای:**  $O_i$  ها در شکل ۶ خروجی های داده ای را مشخص می کند که برای مؤلفه های دیگر تأمین می شود. این مقادیر از گراف وابستگی داده ای استخراج می شود
- **ورودی های ترتیب اجرایی:** ورودی های  $V_i$  مشخص می کند که هم اکنون این مؤلفه باید کار خود را انجام دهد یا در حالت آماده به کار





شکل ۸: مدل مؤلفه B در صورت داشتن وابستگی غیرداده‌ای به مؤلفه A

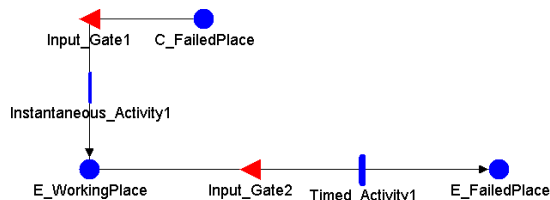
همان‌طور که در شکل ۸ نشان داده شده است به دلیل وابستگی مؤلفه B به مؤلفه A باید مکان مربوط به خرابی مؤلفه A در مدل مؤلفه B قرار داده شود. این مکان به اسم A\_FailedPlace در شکل ۸ نشان داده شده است. تا زمانی که مؤلفه A سالم باشد نرخ خرابی مؤلفه B از فعالیت BFailureRate پیروی می‌کند و هنگامی که مؤلفه A خراب شد، نرخ خرابی مؤلفه B از فعالیت BFailureRate\_Plus\_ImpactOf\_A پیروی می‌کند. هرکدام از دروازه‌های ورودی Input\_Gate1 و Input\_Gate2 نظارت می‌کنند که فعالیت خرابی مؤلفه کدام یک از فعالیت‌ها باشد.

#### ۴-۹- تبدیل گراف ترتیب اجرایی به مدل SAN

در صورتی که کار مؤلفه E بعد از خرابی مؤلفه C شروع شود برای تبدیل گراف ترتیب اجرایی به مدل SAN باید به صورت زیر عمل کرد:

- مکان مربوط به حالت خرابی مؤلفه C باید در داخل مؤلفه E قرار گیرد و این مکان بین این دو مؤلفه به اشتراک گذاشته شود.
- یک فعالیت باید به مؤلفه E اضافه شود تا در صورتی که نشانه‌ای داخل مکان خرابی مؤلفه C قرار گرفت این فعالیت فعال شود و یک نشانه داخل مکان مربوط به در حال کار بودن این مؤلفه قرار دهد.

همان‌طور که در شکل ۹ نشان داده شد هنگامی که نشانه‌ای در مکان C\_FailedPlace قرار بگیرد، فعالیت فوری Activity1 یک نشانه در مکان E\_WorkingPlace قرار می‌دهد و این به معنای آغاز به کار کردن این مؤلفه است. فعالیت Activity1 زمانی شلیک می‌کند که در داخل C\_FailedPlace نشانه‌ای وجود داشته باشد و در مکان‌های E\_WorkingPlace و E\_FailedPlace نشانه‌ای وجود نداشته باشد.



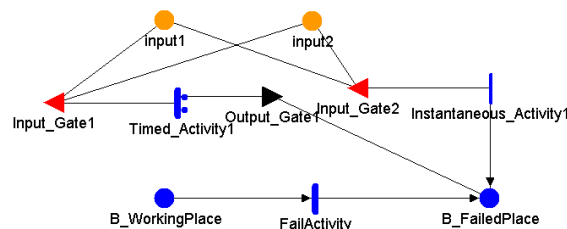
شکل ۹: مدل مؤلفه E در صورت وجود ترتیب اجرایی بین مؤلفه E و C

$$O_i = \begin{cases} 1, & \text{مقدار خروجی ناصحیح باشد} \\ 2, & \text{مقدار خروجی صحیح باشد} \end{cases} \quad (۴)$$

$$I_i = \begin{cases} 1, & \text{مقدار ورودی ناصحیح باشد} \\ 2, & \text{مقدار ورودی صحیح باشد} \end{cases} \quad (۵)$$

همچنین می‌توان از مکان بسط‌یافته برای نشان دادن مقدار یک پارامتر استفاده کرد. به‌طور مثال برای نشان دادن تعداد کار انجام شده توسط یک پردازنده می‌توان از یک مکان بسط‌یافته برای شمارش این کارها استفاده کرد.

پس از ساخت مؤلفه‌های سیستم، هنگام الحاق مؤلفه‌ها به یکدیگر به منظور تشکیل مدل نهایی، مکان‌های بسط‌یافته مؤلفه‌هایی که به همدیگر وابستگی داده‌ای دارند باید به اشتراک گذاشته شود. یکی از مؤلفه‌ها با یک نرخ مشخص مقادیر خروجی را در آن مکان قرار می‌دهد و مؤلفه دیگر نیز با یک نرخ مشخص این مقادیر را به‌عنوان ورودی برمی‌دارد. در شکل ۷ مدل یک مؤلفه با ۲ ورودی مختلف نشان داده شده است.



شکل ۷: مؤلفه A با دو ورودی از سایر مؤلفه‌ها.

#### ۴-۸- تبدیل گراف وابستگی غیرداده‌ای به مدل SAN

برای تزیق وابستگی غیرداده‌ای به مدل SAN باید به صورت زیر عمل کرد:

- در صورتی که مؤلفه B به مؤلفه A وابستگی غیرداده‌ای داشته باشد باید حالت خرابی F مؤلفه A در مدل مؤلفه B قرار گیرد و این مکان بین این دو مؤلفه به اشتراک گذاشته شود. با استفاده از این مکان خرابی زمانی که مؤلفه A خراب شد مؤلفه B از طریق این مکان متوجه خرابی مؤلفه A شود.
- به ازای هر وابستگی غیرداده‌ای یک فعالیت جدید با نرخ جدیدی که از گراف وابستگی غیرداده‌ای محاسبه می‌شود در مدل مؤلفه قرار داده می‌شود.
- به ازای هر فعالیت باید یک دروازه ورودی قرار داده شود که تابع شرطی آن با توجه به حالت خرابی مؤلفه A فعالیت مربوطه را فعال یا غیرفعال کند. به‌صورتی که قبل از خرابی مؤلفه A نرخ خرابی مؤلفه B از فعالیت اول پیروی کند و در صورتی که نشانه‌ای در مکان خرابی مؤلفه A قرار داده شد باید از فعالیت جدید پیروی کند. در آن واحد باید یکی از تابع‌های شرطی مربوط به دروازه‌های ورودی درست باشد و در هیچ حالتی دو تابع شرطی نباید هم‌زمان صحیح باشد.

#### ۴-۱۰- تزییق عبارت‌های احتمالی به مدل SAN

در عدد  $n$  متوقت شود. میانگین زمان تا خرابی مؤلفه برابر می‌شود با نقطه زمانی که در آن نقطه،  $n \times$  از آزمایش‌های مورد نظر در حالت خرابی باشد.

به منظور محاسبه احتمال خرابی، اگر احتمال خرابی در ثانیه  $S$  مد نظر باشد مقدار این احتمال از معادله زیر بدست می‌آید:

$$p(s) = \frac{\text{تعداد آزمایش‌هایی که در نقطه } S \text{ زمانی در حالت خرابی هستند}}{\text{تعداد کل آزمایش‌ها}}$$

لازم به ذکر است تعداد کل آزمایش‌ها در نقطه‌ای متوقت شده است که اجرای بیشتر آزمایش‌ها تغییری در احتمالات به وجود نیارود و در واقع نتایج همگرا شده است.

در مواقعی که نیاز به محاسبه میزان زمان بیکاری را داریم. یک سطح آستانه به طور مثال  $0 \leq S \leq 1$  را مشخص می‌کنیم. این سطح آستانه مشخص می‌کند، در صورتی که  $S$  درصد از مؤلفه‌ها در حالت فعال باشند ما آن مؤلفه را دیگر فعال در نظر می‌گیریم. نقطه‌ای در طول زمان انتخاب می‌شود که در آن نقطه، مؤلفه مورد نظر در  $nS$  از آزمایش‌های انجام شده در حالت فعال باشد یا به بیانی دیگر فقط در  $1 - nS$  آزمایش مؤلفه در حالت بیکار باشد.

#### ۵- مطالعه موردی

در بخش قبل روش پیشنهادی مدل‌سازی انتشار خطا در سیستم‌های هیبرید شرح داده شد. در این فصل به مدل‌سازی سیستم سوخت‌رسان هواپیما پرداخته شده است. ابتدا مشخصات این سیستم شرح داده می‌شود. سپس روش ساخت مدل هر یک از مؤلفه‌های سیستم نشان داده می‌شود. در نهایت برای ساخت مدل نهایی، مؤلفه‌ها با روش گفته‌شده در بخش قبل به یکدیگر الحاق می‌شوند. این هواپیما دارای دو موتور است که هر یک از این موتورها دارای یک مخزن سوخت داخلی و یک مخزن سوخت خارجی است. سوخت‌رسانی به هریک از موتورها از طریق مخزن داخلی انجام می‌شود و در صورت بروز مشکل برای مخزن داخلی از مخزن خارجی استفاده می‌شود. یک مخزن رزرو شده نیز برای سیستم وجود دارد که در مواقع اضطراری می‌تواند فرایند سوخت‌رسانی به هر دو موتور را انجام می‌دهد. این سیستم دارای یک خنک‌کننده و چندین حسگر است که در ادامه جزئیات این سیستم و مشخصات کامل‌تر آن آورده شده است.

#### ۵-۱- مشخصات سیستم سوخت‌رسان

مثال بررسی‌شده در این قسمت یک سیستم سوخت‌رسان است که در مقاله [۱۹] ارائه شده است. همبندی مؤلفه‌های این سیستم در

برای تزییق عبارت‌های احتمالی مربوط به یک مؤلفه در داخل مدل آن مؤلفه، می‌توان از فعالیت‌های احتمالی با چندین خروجی استفاده کرد. فعالیت‌های احتمالی به این صورت است که می‌توان مشخص کرد در صورت فعال بودن فعالیت، در  $p$  درصد مواقع نشانه را از طریق خروجی اول بفرستد و در  $1-p$  درصد مواقع نشانه را از طریق خروجی دوم ارسال کند. هر فعالیت می‌تواند  $n$  خروجی مختلف داشته باشد ولی مجموع احتمال آن‌ها باید یک باشد. به ازای هر عبارت احتمالی و برعکس آن یک فعالیت چندتایی نیاز داریم و هر یک از این فعالیت‌ها به یک دروازه ورودی متصل شده است که با توجه به ورودی‌های مؤلفه مشخص می‌کند این فعالیت هم‌اکنون باید فعال باشد یا خیر. در هر لحظه یکی از چندین فعالیت احتمالی باید فعال باشد. به‌طور مثال برای دو عبارت  $e$  یک فعالیت احتمالی با ۲ خروجی نیاز داریم.

$$\begin{aligned} 1) \text{ Input1. } F, \text{ Input2. } T \rightarrow \text{ Output1. } T, 0.6 \\ 2) \text{ Input1. } F, \text{ Input2. } T \rightarrow \text{ Output1. } F, 0.4 \end{aligned} \quad (۶)$$

در شکل ۱۰ یک فعالیت احتمالی با ۲ خروجی نشان داده شده است. در تعریف این فعالیت می‌توان مشخص کرد که هنگام فعال بودن در ۶۰ درصد مواقع نشانه از خروجی اول انتقال پیدا کند و در ۴۰ درصد مواقع از خروجی دوم منتقل شود. مجموع این احتمالات برای هر فعالیت باید برابر با یک شود.



شکل ۱۰: یک نمونه فعالیت احتمالی با دو خروجی

#### ۴-۱۱- الحاق مؤلفه‌ها از طریق صورت‌بندی تکرار-الحاق

برای الحاق مؤلفه‌ها از قوانین زیر استفاده می‌شود. قوانین اتصال مؤلفه‌ها به یکدیگر به صورت زیر دسته‌بندی می‌شود:

- اتصال مؤلفه‌هایی که به همدیگر وابستگی داده‌ای دارند.
- اتصال مؤلفه‌هایی که به همدیگر وابستگی غیرداده‌ای دارند از طریق اشتراک مکان خرابی هر یک از مؤلفه‌ها.
- اتصال مؤلفه‌هایی که به‌منظور افزونگی در سیستم استفاده شده است از طریق اشتراک مکان خرابی مؤلفه‌های پیش‌نیاز.

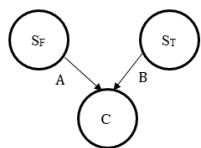
#### ۴-۱۲- روش به دست آوردن نتایج ارزیابی

برای محاسبه مقادیر میانگین زمان تا خرابی، احتمال خرابی و مدت زمان بیکار بودن هر یک از مؤلفه‌ها به صورت زیر عمل شده است.

روش استخراج میزان MTTF به این صورت است که اگر ما حد آستانه خرابی مؤلفه را  $0 \leq x \leq 1$  در نظر بگیریم و همچنین هنگام انجام آزمایش، تعداد دفعات اجرای مدل بعد از همگرایی نتایج

□ **گراف وابستگی داده‌ای سیستم سوخت‌رسان**

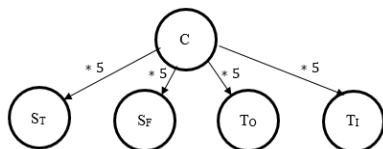
در زیرسیستم سمت چپ، حسگرهای فشار و دما اطلاعات اندازه‌گیری شده را به خنک‌کننده ارسال می‌کنند. گراف مربوط به آن در شکل ۱۲ نشان داده شده است.



شکل ۱۲: گراف وابستگی داده‌ای سیستم سوخت‌رسان

□ **گراف وابستگی غیرداده‌ای سیستم سوخت‌رسان**

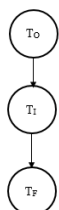
در صورتی که سیستم خنک‌کننده دچار خرابی شود، نرخ خرابی مؤلفه‌های حسگر فشار، حسگر دما، مخزن داخلی و مخزن خارجی افزایش پیدا می‌کند. گراف وابستگی غیرداده‌ای مربوط به این سیستم در شکل ۱۳ نشان داده شده است.



شکل ۱۳: گراف وابستگی غیرداده‌ای سیستم سوخت‌رسان

□ **گراف ترتیب اجرایی سیستم سوخت‌رسان**

گراف ترتیب اجرایی مربوط به این سیستم در شکل ۱۴ نشان داده شده است. همان‌طور که در توضیحات سیستم گفته شد؛ ابتدا از مخزن خارجی استفاده می‌شود در صورت خرابی این مخزن، پمپ مربوط به مخزن داخلی فعال می‌شود و بعد از خرابی این مؤلفه از مخزن رزرو شده استفاده می‌گردد.

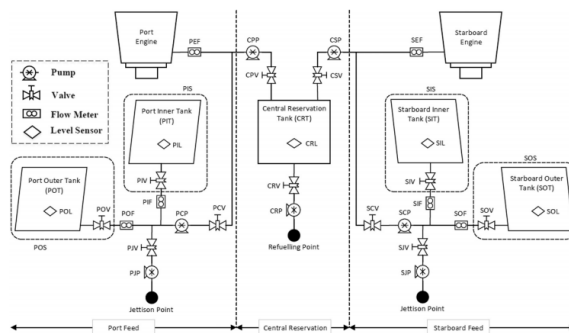


شکل ۱۴: گراف ترتیب اجرایی سیستم سوخت‌رسان

□ **عبارت‌های احتمالی مربوط به سیستم سوخت‌رسان**

هنگامی که حسگر فشار و حسگر دما درست کار نکنند خنک‌کننده نمی‌تواند عملکرد صحیح خود را انجام دهد. اما در صورتی که یکی از این دو حسگر درست کار نکنند در ۹۵ درصد مواقع خنک‌کننده

شکل ۱۱ نشان داده شده است. این سیستم دارای دو موتور و سه زیرسیستم مختلف است. در زیرسیستم بخش راست یک موتور با دو مخزن سوخت آن قرار داده شده است. همچنین در زیرسیستم سمت چپ موتور دیگر و مخازن مربوط به آن وجود دارد. در زیرسیستم وسط مخزن سوخت رزرو شده قرار دارد. نحوه سوخت‌رسانی به موتورها به این‌گونه است که هر موتور یک مخزن سوخت داخلی و یک مخزن سوخت خارجی دارد در حالت عادی سوخت هر یک از موتورها از طریق مخزن خارجی زیرسیستم مربوطه تأمین می‌شود. در صورت خرابی مخزن خارجی، موتور از مخزن داخلی استفاده می‌کند. هنگامی که هر دو مخزن سوخت دچار خرابی شدند، موتور از مخزن سوخت رزرو شده استفاده می‌کند. دو حسگر فشار و دما و همچنین یک سیستم خنک‌کننده برای هر زیرسیستم وجود دارد، که خنک‌کننده با توجه به فشار و دمای محیط، میزان خنک‌سازی محیط را تنظیم می‌کند.



شکل ۱۱: سیستم سوخت‌رسان

۲-۵- **گراف‌ها و عبارت‌های احتمالی مربوط به سیستم**

در این بخش تشریح هر یک از سه گراف مربوط به این سیستم قرار داده شده است. با توجه به اینکه ساختار هر دو زیرسیستم سمت چپ و راست دقیقاً همانند یکدیگر است، یکی از این زیرسیستم‌ها در نظر گرفته شده است. مخفف‌هایی که برای اشاره به مؤلفه‌های مختلف استفاده شده است در جدول ۲ نشان داده شده است.

جدول ۲: مخفف اسامی مؤلفه‌های سیستم سوخت‌رسان.

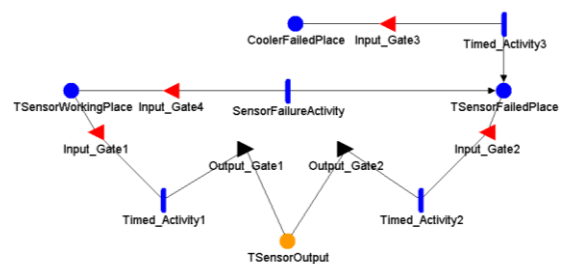
نام مؤلفه	مخفف
C	خنک‌کننده
Ti	مخزن داخلی
To	مخزن خارجی
TR	مخزن رزرو شده
SP	حسگر فشار
ST	حسگر دما

- درست کار می کند. عبارت های احتمالی مربوط به این مؤلفه در معادله های ۷ آورده شده است.
  - $S_T.False, S_F.False \rightarrow C.Fail, 1$
  - $S_T.True, S_F.False \rightarrow C.Fail, 0.05$  (۷)
  - $S_T.False, S_F.True \rightarrow C.Fail, 0.01$
- ۳-۵- طراحی مدل SAN مؤلفه های سیستم سوخت رسان
- در این بخش هر یک از مؤلفه های سیستم سوخت رسان طراحی شده است. در طراحی هر یک از مؤلفه ها، نحوه ی استفاده از گراف های تولید شده شرح داده است. تمامی متغیرهایی که برای نرخ خرابی یا تأثیر بر روی نرخ خرابی در نظر گرفته شده است در هنگام شبیه سازی مقداردهی می شود.

#### طراحی مؤلفه حسگر حرارتی

مدل SAN مربوط به حسگر حرارتی در شکل ۱۵ نشان داده شده است. مدل مؤلفه حسگر فشار نیز به همین صورت است. با توجه به شکل ۱۵، مکان های مدل مؤلفه SAN به صورت زیر است:

- **TCoolerFailedPlace**: به دلیل اینکه حسگر حرارتی به خنک کننده وابستگی غیرداده ای دارد بنابراین باید مکان خرابی خنک کننده با حسگر به اشتراک گذاشته شود.
- **TSensorFailedPlace**: این مکان نشان دهنده خرابی حسگر است.
- **TSensorWorkingPlace**: این مکان نشان دهنده سالم بودن حسگر است.
- **TSensorOutput**: در این مکان خروجی حسگر حرارتی قرار می گیرد. هنگامی که حسگر سالم باشد مقدار ۱ را در این مکان قرار می دهد و هنگامی که حسگر دچار خرابی شود مقدار ۲ را در این مکان قرار می دهد.



شکل ۱۵: مدل SAN حسگر حرارتی

- جداول مربوط به دروازه های ورودی در جدول ۳ آمده است همچنین مشخصات دروازه های ورودی این مدل به این صورت است:
- **Input\_Gate1**: این دروازه هنگامی اجازه شلیک به فعالیت  $Time\_Activity1$  را می دهد که حسگر خراب نشده باشد و مقدار داخل مکان  $TsensorOutput$  یک نباشد.
  - **Input\_Gate2**: این دروازه هنگامی اجازه شلیک به فعالیت  $Time\_Activity2$  را می دهد که حسگر خراب شده باشد

جدول ۳: جدول دروازه های ورودی مدل مؤلفه حسگر حرارتی

عبارت شرطی	
TSensorOutput->IOvalue->Mark()=1&&TSensorWorkingPlace->Mark()=1	Input_Gate1
عبارت شرطی	
TSensorOutput->IOvalue->Mark()=2 && TSensorFailedPlace->Mark()=1	Input_Gate2
عبارت شرطی	
CoolerFailedPlace->Mark()=1 && TSensorWorkingPlace->Mark()=1 && TSensorFailedPlace->Mark()=0	Input_Gate3
عبارت شرطی	
CoolerFailedPlace->Mark()=0 && TSensorWorkingPlace->Mark()=1 && TSensorFailedPlace->Mark()=0	Input_Gate4
تابع ورودی	
TSensorWorkingPlace->Mark()=0;	

جداول مربوط به دروازه های خروجی در جدول ۴ آمده است. همچنین نرخ فعالیت های این مؤلفه در جدول ۵ نشان داده شده است. تمامی نرخ ها از توزیع نمایی پیروی می کنند.

مشخصات دروازه های خروجی این مدل به صورت زیر است:

- **Output\_Gate1**: هنگامی که فعالیت  $Time\_Activity1$  شلیک کند این دروازه مقدار مکان  $TsensorOutput$  را برابر یک قرار می دهد.
- **Output\_Gate2**: هنگامی که فعالیت  $Time\_Activity2$  شلیک کند این دروازه مقدار مکان  $TsensorOutput$  را برابر دو قرار می دهد.

جدول ۴: جدول دروازه های خروجی مدل مؤلفه حسگر حرارتی

تابع خروجی	
TSensorOutput->IOvalue->Mark()=1;	Output_Gate1
تابع خروجی	
TSensorOutput->IOvalue->Mark()=2;	Output_Gate2

جدول ۵: نرخ فعالیت های مدل مؤلفه حسگر حرارتی

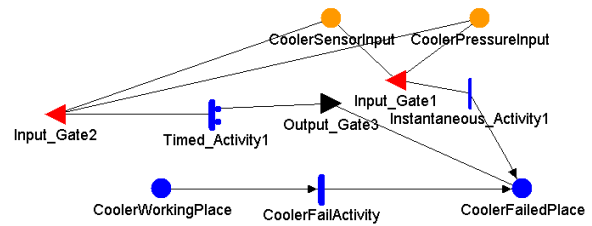
نرخ شلیک	نام فعالیت
10	Time_Activity1
10	Time_Activity2
$TSensorFailureRate + TSensorFailureRate*4$	Time_Activity3
$TSensorFailureRate$	SensorFailureActivity

طراحی مؤلفه خنک کننده

جدول ۷: جدول دروازه‌های ورودی مدل مؤلفه خنک کننده

عبارت شرطی	Input_Gate1
CoolerWorkingPlace->Mark()==1 && CoolerSensorInput->IOvalue->Mark()==2 && CoolerPressureInput->IOvalue->Mark()==2	
تابع ورودی	Input_Gate2
CoolerWorkingPlace->Mark()==0	
عبارت شرطی	Input_Gate2
CoolerWorkingPlace->Mark()==1 && ((CoolerSensorInput->IOvalue->Mark()==2 && CoolerPressureInput->IOvalue->Mark()==1)    (CoolerSensorInput->IOvalue->Mark()==1 && CoolerPressureInput->IOvalue->Mark()==2))	

خنک کننده در این سیستم به این صورت کار می کند که اطلاعات فشار و دما را از حسگرهای حرارتی و فشار می گیرد، سپس دمای محیط را با توجه به این مقادیر تنظیم می کند. در صورتی که هر دو مقدار ورودی به این مؤلفه نادرست باشد خنک کننده نمی تواند درست کار کند و به اصطلاح دچار خرابی می شود. در صورتی که یکی از این دو مقدار ناصحیح باشد در ۹۵ درصد مواقع خنک کننده درست کار می کند و در ۵ درصد مواقع اشتباه کار می کند. مدل SAN این مؤلفه در شکل ۱۶ نشان داده شده است.



شکل ۱۶: مدل SAN خنک کننده

جدول ۸: نرخ فعالیت های مدل مؤلفه خنک کننده

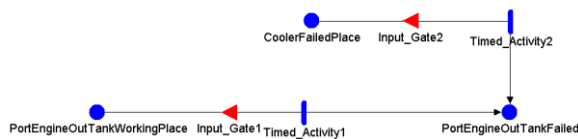
نرخ شلیک	نام فعالیت
10	Time_Activity1
CoolerFailureRate	CoolerFailureRate

مشخصات مکان های مدل این مؤلفه:

طراحی مؤلفه مخزن خارجی

مدل SAN این مؤلفه در شکل ۱۷ نشان داده شده است. مشخصات مکان ها این مدل به صورت زیر است:

- **CoolerFailedPlace**: این مکان به دلیل وابستگی داده ای خنک کننده با حسگر حرارتی، بین آن ها به اشتراک گذاشته است و مقدار دمایی که حسگر حرارتی گزارش می دهد از این مکان برمی دارد.
  - **CoolerWorkingPlace**: نشان دهنده سالمی خنک کننده است.
  - **CoolerFailedPlace**: نشان دهنده خرابی خنک کننده است.
  - **CoolerSensorInput**: این مکان به دلیل وابستگی داده ای خنک کننده با حسگر فشار، بین آن ها به اشتراک گذاشته است و فشاری که حسگر فشار گزارش می دهد از این مکان برمی دارد.
  - **CoolerPressureInput**: این مکان به دلیل وابستگی داده ای خنک کننده با حسگر فشار، بین آن ها به اشتراک گذاشته است و مقدار فشاری که حسگر فشار گزارش می دهد از این مکان برمی دارد.
- جدول مربوط به دروازه های خروجی و ورودی در جدول ۶ و ۷ نشان داده شده است. همچنین نرخ خرابی فعالیت های این مدل در جدول ۸ نشان داده شده است. تمامی نرخ ها از توزیع نمایی پیروی می کنند. جدول مربوط به دروازه های ورودی در جدول ۱۰ آورده شده است.



شکل ۱۷: مدل SAN مخزن خارجی

جدول ۹: نرخ فعالیت های مدل مؤلفه مخزن خارجی

نرخ شلیک	نام فعالیت
PortEngineTankFailureRate	Time_Activity1
PortEngineTankFailureRate + 4*PortEngineTankFailureRate	Time_Activity2

جدول ۶: جدول دروازه های خروجی مدل مؤلفه خنک کننده

تابع خروجی	Output_Gate1
CoolerFailedPlace->Mark()==1; CoolerWorkingPlace->Mark()==0;	

همچنین نرخ خرابی فعالیت های این مدل در ۱۱ نشان داده شده است. تمامی نرخ ها از توزیع نمایی پیروی می کند.

جدول ۱۱: نرخ فعالیت های مدل مؤلفه داخلی

نرخ شلیک	نام فعالیت
PInnerTankFailureRate	Time_Activity1
PInnerTankFailureRate+ 4* PInnerTankFailureRate	Time_Activity2

طراحی مؤلفه مخزن رزرو شده

مدل SAN این مؤلفه در شکل ۱۹ نشان داده شده است. مکان های استفاده شده در این مدل عبارت اند از:

- **CRWorking**: این مکان نشان دهنده انجام عمل سوخت رسانی توسط مخزن رزرو شده است. با توجه به اینکه مخزن رزرو شده وابستگی غیرداده ای به مخزن داخلی دارد، مکانی که نشان دهنده خرابی مخزن داخلی است باید در مدل قرار داده شود.
  - **PEIFailedPlace**: این مکان نشان دهنده خرابی مخزن داخلی است. به دلیل دارا بودن ترتیب اجرایی با مخزن داخلی، این مکان در مدل قرار داده شده است.
  - **CRFailed**: هنگامی که مخزن رزرو شده دچار خرابی شود یک نشانه داخل این مکان قرار داده می شود.
- جدول مربوط به دروازه ورودی در جدول ۱۳ نشان داده شده است. همچنین مشخصات دروازه ی ورودی این مدل به صورت زیر است:
- **Input\_Gate1**: این دروازه هنگامی اجازه شلیک به فعالیت Instantaneous\_Activity1 را می دهد که مخزن داخلی دچار خرابی شود.



شکل ۱۹: مدل SAN مخزن رزرو شده

نرخ خرابی فعالیت های این مدل در جدول ۱۴ نشان داده شده است. تمامی نرخ ها از توزیع نمایی پیروی می کند:

جدول ۱۳: جدول دروازه های ورودی مدل مؤلفه رزرو شده

عبارت شرطی	Input_Gate1
PEIFailedPlace->Mark()==1 && CRWorking->Mark()==0 && CRFailed->Mark()==	

جدول ۱۴: نرخ فعالیت های مدل مؤلفه رزرو شده

نرخ شلیک	نام فعالیت
CRFailureRate	Time_Activity1

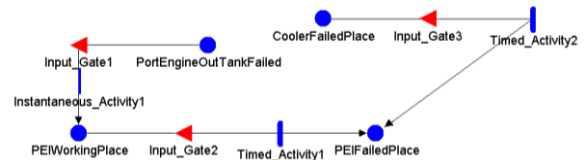
جدول ۱۰: جدول دروازه های ورودی مدل مؤلفه مخزن خارجی

عبارت شرطی	Input_Gate1
CoolerFailedPlace->Mark()==0 && PortEngineOutTankFailed->Mark()==0 && PortEngineOutTankWorkingPlace->Mark()==1	
عبارت شرطی	Input_Gate2
CoolerFailedPlace->Mark()==1 && PortEngineOutTankFailed->Mark()==0 && PortEngineOutTankWorkingPlace->Mark()==1	

طراحی مؤلفه مخزن داخلی

مدل SAN این مؤلفه در شکل ۱۸ نشان داده شده است. مکان های استفاده شده در این مدل عبارت اند از:

- **CoolerFailedPlace**: با توجه به اینکه مخزن داخلی به خنک کننده وابستگی غیرداده ای دارد، بنابراین مکانی که نشان دهنده خرابی خنک کننده است باید در این مدل قرار داده شود.
- **PortEngineOutTankFailed**: این مکان نشان دهنده خرابی مخزن خارجی است. با توجه به گراف ترتیب اجرایی، فعالیت مخزن داخلی بعد از خرابی مخزن خارجی شروع می شود، بنابراین مکانی که نشان دهنده خرابی مخزن خارجی است باید در مدل قرار داده شود.
- **PEIFailedPlace**: این مکان نشان دهنده خرابی مخزن داخلی است.
- **PEIWorkingPlace**: این مکان نشان دهنده آماده به کار بودن مخزن است.



شکل ۱۸: مدل SAN مخزن داخلی

جداول مربوط به دروازه های ورودی در جدول ۱۲ آورده شده است.

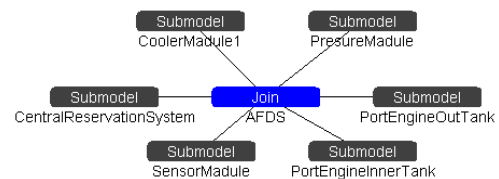
جدول ۱۲: جدول دروازه های ورودی مدل مؤلفه داخلی

عبارت شرطی	Input_Gate1
PortEngineOutTankFailed->Mark()==1 && PEIWorkingPlace->Mark()==0 && PEIFailedPlace->Mark()==0	
تابع ورودی	PortEngineOutTankFailed->Mark()=1;
عبارت شرطی	Input_Gate2
CoolerFailedPlace->Mark()==0 && PEIWorkingPlace->Mark()==1 && PEIFailedPlace->Mark()==0	
تابع ورودی	PEIWorkingPlace->Mark()=0;
عبارت شرطی	Input_Gate3
CoolerFailedPlace->Mark()==1 && PEIFailedPlace->Mark()==0 && PEIWorkingPlace->Mark()==1	

#### ۵-۴- الحاق مؤلفه‌های سیستم سوخت‌رسان

با توجه به گراف‌های سه‌گانه‌ی سیستم سوخت‌رسان که در بخش قبل توضیح داده شد، برای الحاق مؤلفه‌ها با یکدیگر باید مکان‌ها به‌صورت زیر با یکدیگر به اشتراک گذاشته شوند.

- مکان‌های CoolerFailedPlace در مدل مؤلفه‌های حسگر حرارتی، حسگر فشار، خنک‌کننده، مخزن خارجی و مخزن داخلی (این اشتراک به دلیل وجود وابستگی غیرداده‌ای است).
  - مکان‌های PEIFailedPlace در مدل مؤلفه‌های مخزن داخلی و مخزن رزرو شده (این اشتراک به خاطر حضور این دو مؤلفه در گراف ترتیب اجرایی است).
  - مکان‌های PortEngineOutTankFailed در مدل مؤلفه‌های مخزن داخلی و مخزن خارجی (این اشتراک به خاطر حضور این دو مؤلفه در گراف ترتیب اجرایی است).
  - مکان‌های CoolerPressureInput و PressureOutput در مدل مؤلفه‌های خنک‌کننده و حسگر فشار (این اشتراک به دلیل داشتن وابستگی داده‌ای بین این دو مؤلفه است).
  - مکان‌های CoolerSensorInput و TSensorOutput در مدل مؤلفه‌های خنک‌کننده و حسگر حرارتی (این اشتراک به دلیل داشتن وابستگی داده‌ای بین این دو مؤلفه است).
- تصویر مربوط به الحاق مؤلفه‌ها در شکل ۲۰ نشان داده شده است.



شکل ۲۰: مدل تکرار-الحاق سیستم سوخت‌رسان

#### ۵-۵- ارزیابی مطالعه موردی

برای مدل‌سازی و شبیه‌سازی از نرم افزار Mobius در محیط ویندوز ۱۰ استفاده شده است.

پس از الحاق مؤلفه‌های سیستم، به منظور اجرای شبیه‌سازی ابتدا سیستم را بدون حضور هیچ خطایی اجرا می‌کنیم، سپس با تزریق خطایی که منجر به خرابی حسگر حرارتی شود شبیه‌سازی را اجرا می‌کنیم و نتایج این دو حالت را باهم مقایسه می‌کنیم. مقادیری که برای متغیرهای این مدل در نظر گرفته شده است در جدول ۱۵ نشان داده شده است. در هر آزمایش، تعداد اجرای مدل تا زمان همگرا شدن نتایج به یک عدد ادامه پیدا می‌کند. در واقع از آن نقطه به بعد، افزایش تعداد دفعات اجرای مدل در خروجی مدل تغییری ایجاد نمی‌کند.

جدول ۱۵: مقادیر در نظر گرفته‌شده برای متغیرهای موجود در مدل

مقدار	نام متغیر
1.0E-3	CRFailureRate
1.0E-3	CoolerFailureRate
1.0E-3	PlnnerTankFailureRate
1.0E-3	PortEngineTankFailureRate
1.0E-4	PressureFailureRate
1.0E-4	TSensorFailureRate

#### تحلیل رفتار خنک‌کننده در حضور و عدم حضور خطا

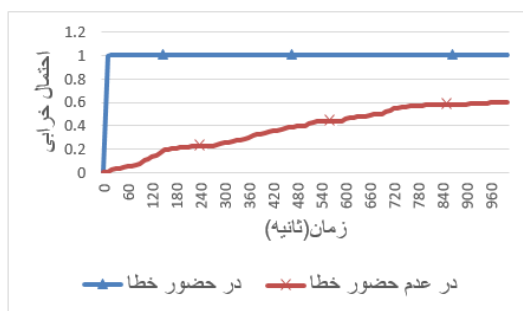
با توجه به تحلیل نمودار شکل ۲۱ اگر خنک‌کننده را با عبور از میانگین ۰.۷۵ خراب فرض کنیم. میانگین زمان لازم برای خرابی خنک‌کننده به‌صورت زیر است:

میانگین زمانی تا خرابی خنک‌کننده در حضور خطا:

$$MTTF = 10s$$

میانگین زمانی تا خرابی خنک‌کننده در عدم حضور خطا:

$$MTTF = 1330s$$



شکل ۲۱: رفتار خرابی خنک‌کننده در حضور و عدم حضور خطا

هنگامی که حسگر خراب باشد در ۷۵ درصد از آزمایش‌های انجام شده خنک‌کننده در ثانیه ۱۰ خراب شده است و هنگامی که حسگر خراب نباشد در ۷۵ درصد آزمایش از ثانیه ۱۳۳۰ به بعد خنک‌کننده دچار خرابی شده است.

#### تحلیل رفتار مخزن خارجی در حضور و عدم حضور خطا

اولین مخزنی که عملیات سوخت‌رسانی را انجام می‌دهد مخزن خارجی است. با توجه به تحلیل نمودار شکل ۲۲ این مخزن را اگر پس از گذراندن میانگین ۰.۷۵ خراب فرض کنیم میانگین زمان لازم تا اولین خرابی به‌صورت زیر است:

میانگین زمانی تا خرابی مخزن خارجی در حضور خطا:

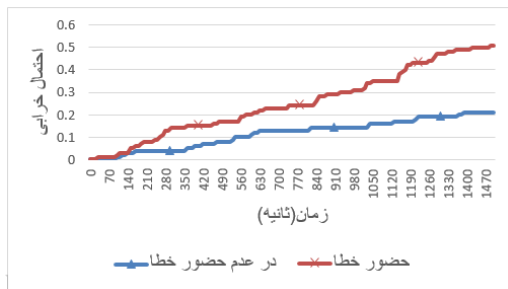
$$MTTF = 280s$$

میانگین زمانی تا خرابی مخزن خارجی در عدم حضور خطا:

$$MTTF = 960s$$

احتمال خرابی حسگر فشار در ثانیه ۱۵۰۰ در عدم حضور خطا:

$$P(F) = 0.21$$



شکل ۲۴: رفتار خرابی حسگر فشار در حضور و عدم حضور خطا

□ تحلیل رفتار مخزن رزرو شده در حضور و عدم حضور خطا

هنگامی مخزن رزرو شده فعالیت سوخت‌رسانی خود را آغاز می‌کند که مخزن داخلی دچار مشکل شده باشد. نمودار تحلیل نحوه خرابی این مؤلفه در شکل ۲۵ آمده است. با توجه به تحلیل این نمودار:

میانگین زمان تا شروع سوخت‌رسانی در حضور خطا:

$$T = 130s$$

میانگین زمان تا شروع سوخت‌رسانی در عدم حضور خطا:

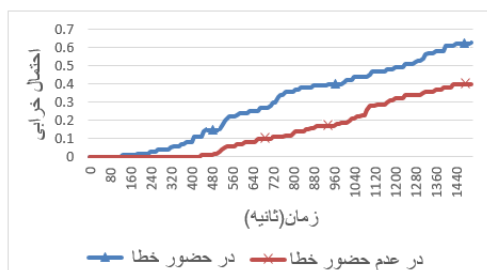
$$T = 440s$$

احتمال خرابی کامل سیستم در حضور خطا:

$$P(F) = 0.63$$

احتمال خرابی کامل سیستم در عدم حضور خطا:

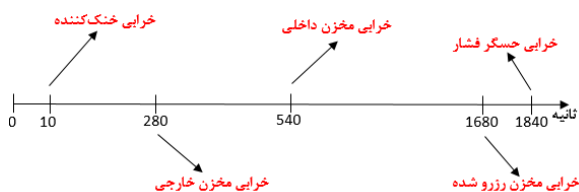
$$P(F) = 0.4$$



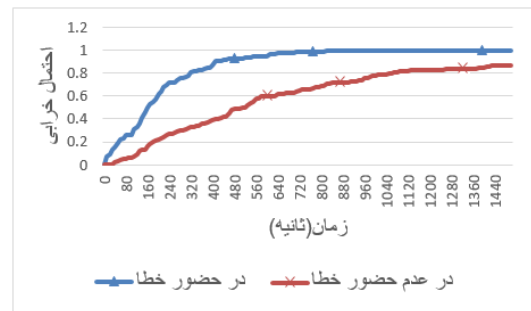
شکل ۲۵: رفتار مخزن رزرو شده در حضور و عدم حضور خطا

□ نحوه انتشار خطا از حسگر حرارتی به سایر مؤلفه‌ها

با توجه به مشاهده رفتار هر یک از مؤلفه‌ها که در بخش قبل توضیح داده شده، انتشار خطا از مؤلفه حسگر حرارتی به سایر مؤلفه‌ها در شکل ۲۶ نشان داده شده است.



شکل ۲۶: چگونگی انتشار خطای حسگر حرارتی در سایر مؤلفه‌ها



شکل ۲۲: رفتار خرابی مخزن خارجی در حضور و عدم حضور خطا

با وجود خطا در حسگر حرارتی، مخزن خارجی در ثانیه ۲۸۰ام دچار خرابی شده و موتور سیستم از این به بعد از مخزن داخلی برای سوخت‌رسانی استفاده می‌کند. اما در حالتی که خطایی وجود ندارد این اتفاق در ثانیه ۹۶۰ام اتفاق می‌افتد. بنابراین در صورتی که مدت زمان عملیات سیستم کمتر از ۹۶۰ باشد این مخزن در طول زمان عملیات دچار مشکلی نخواهد شد.

□ تحلیل رفتار مخزن داخلی در حضور و عدم حضور خطا

نمودار رفتار این مؤلفه در شکل ۲۳ نشان داده شده است. بنابراین این مؤلفه تا زمان خرابی مخزن خارجی بی‌کار است. میانگین مدت‌زمان بیکاری این مخزن در حضور و عدم حضور خطا به صورت زیر است:

میانگین زمانی بیکاری مخزن داخلی در حضور خطا:

$$F(T_1) = 50s$$

میانگین زمانی بیکاری مخزن داخلی در عدم حضور خطا:

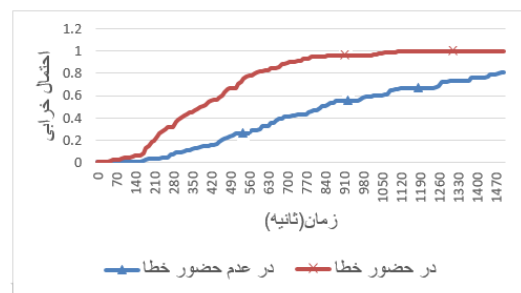
$$F(T_1) = 170s$$

میانگین زمانی تا خرابی مخزن داخلی در حضور خطا:

$$MTTF = 540s$$

میانگین زمانی تا خرابی مخزن داخلی در عدم حضور خطا:

$$MTTF = 1380s$$



شکل ۲۳: رفتار خرابی مخزن داخلی در حضور و عدم حضور خطا

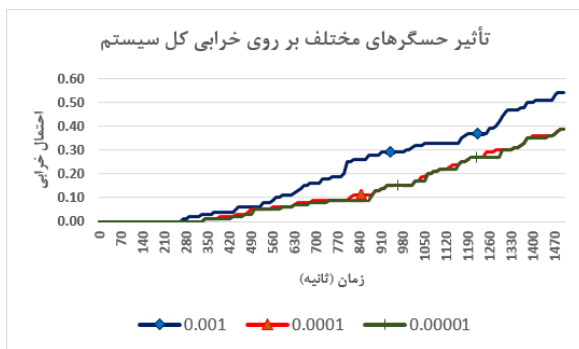
□ تحلیل رفتار حسگر فشار در حضور و عدم حضور خطا

با توجه به نمودار شکل ۲۴ احتمال خرابی حسگر در ثانیه ۱۵۰۰ام به صورت زیر است:

احتمال خرابی حسگر فشار در ثانیه ۱۵۰۰ در حضور خطا:

$$P(F) = 0.51$$





شکل ۲۸: تأثیر حسگرهای مختلف بر روی خرابی کل سیستم

### ۷-۵- مقایسه روش پیشنهادی و نتیجه‌گیری

در این بخش روش پیشنهادی این پژوهش با مقاله‌های [۱۵، ۱۸] مقایسه شده است. در جدول ۱۶ جزئیات این مقایسه را نشان می‌دهد. در ادامه مقایسه روش پیشنهادی از منظر پارامترهای مختلف با سایر روش‌ها آورده شده است.

جدول ۱۶: مقایسه روش پیشنهادی با دو روش دیگر

روش پیشنهادی	[۱۸]	[۱۵]
در نظر گرفتن تفاوت مؤلفه‌ها	✓	✓
در نظر گرفتن ترتیب اجرایی	✓	×
در نظر گرفتن وابستگی غیرداده‌ای	✓	×
نیاز به وارد کردن جزئیات سیستم	×	✓
نیاز به پیش‌بینی مسیرهای خطا	×	✓
سادگی مدل	✓	×
زمان بر بودن ارزیابی	×	✓
سختی به دست آوردن پارامترها	✓	✓

### ۷-۵-۱- سختی مدل‌سازی

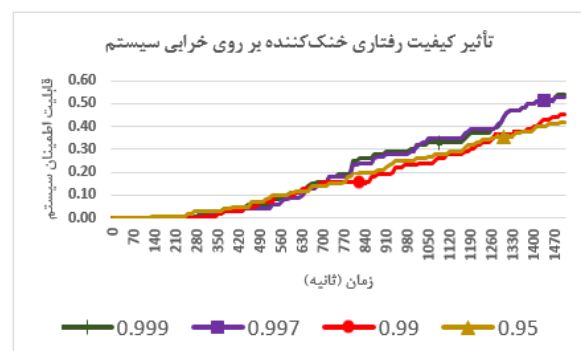
در این بخش سختی مدل‌سازی روش پیشنهادی با دو روش دیگر بررسی شده است. در اینجا منظور از سختی مدل‌سازی چگونگی ساخت مدل و آماده کردن آن جهت ارزیابی است. در مقاله [۱۸] که مثال اول از این مقاله گرفته شده بود، برای مدل‌سازی از شبکه پتری و شبکه بیزی استفاده کرده است. این مدل‌ها با توجه به ساده بودن و امکانات کم آن‌ها فرایند مدل‌سازی را سخت و پیچیده می‌کند. این امکانات شامل مکان‌های بسط‌یافته، فعالیت‌های دارای چند حالت احتمالی و غیره است، که در مدل SAN وجود دارد. بنابراین در صورتی که یک سیستم را با این روش مدل‌سازی شود به دلیل قابلیت‌های کم مدل، مدل بسیار بزرگ و فرایند مدل‌سازی زمان‌بر و پیچیده می‌شود. همچنین در این روش باید قبل از مدل‌سازی همه‌ی خطاها را پیش‌بینی کرد و نحوه‌ی

### ۶-۵- تحلیل حساسیت ورودی‌های مختلف بر نتایج مدل‌سازی

در این بخش به تحلیل حساسیت استفاده از خنک‌کننده‌ها و حسگرهای حرارتی مختلف در نتایج ارزیابی پرداخته شده است. در ادامه جزئیات این دو آزمایش ذکر شده است.

#### □ تأثیر استفاده از خنک‌کننده‌های با دقت‌های مختلف بر قابلیت اطمینان سیستم

در این آزمایش از خنک‌کننده‌های مختلف با دقت‌های متفاوت استفاده شده است. قابلیت اطمینان کل سیستم با خنک‌کننده‌های متفاوت در شکل ۲۷ نشان داده شده است. همانطور که در شکل ۲۷ نشان داده شده است. در صورتی که بتوانیم رفتار خنک‌کننده در صورت خراب بودن یکی از حسگرها از ۰.۹۵ به ۰.۹۹۷ افزایش دهیم، قابلیت اطمینان سیستم از ۰.۴۱ به ۰.۵۴ افزایش پیدا خواهد کرد. همین نمودار نشان می‌دهد که استفاده کردن از خنک‌کننده با دقت ۰.۹۹۹، تفاوت چندانی با خنک‌کننده با دقت ۰.۹۹۷ ندارد. از این نمودار می‌توان نتیجه گرفت که بهترین خنک‌کننده برای این سیستم خنک‌کننده‌ای با دقت ۰.۹۹۷ است.



شکل ۲۷: تأثیر کیفیت رفتاری خنک‌کننده بر روی خرابی سیستم

#### □ تأثیر استفاده از حسگرهای حرارتی متفاوت بر قابلیت اطمینان سیستم

در این آزمایش از حسگرهای با نرخ خرابی متفاوت استفاده شده است. این آزمایش مشخص می‌کند که با استفاده از حسگرهای حرارتی مختلف تا چه اندازه‌ای می‌توان قابلیت اطمینان سیستم را بالا برد. نمودار مربوط به این آزمایش در شکل ۲۸ نشان داده شده است. همانطور که در شکل ۲۸ نشان داده شده است. در صورت استفاده از حسگر با نرخ خرابی ۰.۰۰۰۱ می‌توان قابلیت اطمینان سیستم را تا حدود ۰.۶ بالا برد. استفاده کردن از حسگر بهتر مانند E-5 دیگر تأثیر چندانی بر قابلیت اطمینان کل سیستم نخواهد گذاشت.

#### ۵-۷-۴- سختی ارزیابی و به دست آوردن نتایج

در روش پیشنهادی مقاله [۱۵] برای هر بار اجرای مدل و مشاهده نحوه‌ی انتشار خطا، حدود ۲۲ ساعت زمان لازم است که برای مدل‌سازی و به دست آوردن نتایج زمان بسیار زیادی است. درحالی‌که روش ارائه شده در این مقاله پس از ساخت مدل، حدود ۲ دقیقه زمان بر روی سیستم یاد شده جهت اجرای مدل و به دست آوردن نتایج زمان لازم دارد.

#### ۵-۷-۵- جامعیت و مقیاس‌پذیری

با توجه به رابط‌های ارائه شده برای مؤلفه‌های سیستم، هر مؤلفه‌ای شامل تعدادی ورودی و تعدادی خروجی است و نحوه نگاشت ورودی‌ها به خروجی‌ها را نیز می‌توان با عبارت‌های احتمالی مشخص کرد. مشخص است که هر مؤلفه‌ای در این ساختار می‌تواند قرار بگیرد.

با توجه به اینکه ارتباط تمامی مؤلفه‌ها در مدل ارائه شده این مقاله بر اساس ورودی و خروجی و ارتباط غیرداده‌ای است، در صورتی‌که چندین سیستم وجود داشته باشند که باهمدیگر ارتباط داشته باشند و بتوان آن‌ها را کنار هم قرار داد، به راحتی می‌توان ارتباط بین آن‌ها را با توجه به قوانین گفته شده فراهم کرد و مدل سیستم ترکیبی، متشکل از چند زیرسیستم را ایجاد نمود. بنابراین می‌توان نشان داد که مدل ارائه شده مقیاس‌پذیر نیز است.

#### ۵-۷-۶- جمع‌بندی مقایسه روش پیشنهادی

در نهایت همانطور که در بخش‌های قبل توضیح داده شد، می‌توان گفت مدل پیشنهادی در این مقاله دارای روش مدل‌سازی ساده‌تری نسبت به دو مدل دیگر است. همچنین، پارامترهایی مانند وابستگی غیرداده‌ای در مدل پیشنهادی در نظر گرفته شده است که در مدل‌های قبلی دیده نشده بود. زمان اجرای مدل پیشنهادی در مقایسه با روش مقاله [۱۵] بسیار پایین‌تر است که در بخش قبلی مقایسه شدند. ویژگی مهم مدل پیشنهادی مقیاس‌پذیری آن است که می‌توان هر یک مؤلفه‌ها و حتی سیستم‌ها را به صورت جداگانه مدل‌سازی کرد و سپس مدل‌ها را به راحتی با استفاده از مدل تکرار-الحاق کنار هم قرارداد و به ارزیابی رفتاری آن‌ها در کنار یکدیگر پرداخت. همچنین همانطور که در بخش قبل توضیح داده شد با استفاده از این روش می‌توان نتایج متنوع‌تری را به دست آورد.

#### ۵-۸- کارهای آینده

به منظور بهبود در روش پیشنهادی، موضوعات و عناوین زیر برای تحقیق‌های آتی و کارهای آینده پیشنهاد می‌شود:

ترکیب آن‌ها برای انتشار در سایر مؤلفه‌ها را نیز در نظر گرفت. در روش مقاله [۱۵]، از آتاماتای هیبریدی استفاده شده بود. برای مدل‌سازی با این روش باید تمامی جزئیات مدل را در نظر گرفت. این جزئیات شامل فرمول‌های مربوط به هر یک از مؤلفه‌ها نیز می‌شود. به‌طور مثال برای پر شدن مخزن آب باید دقیقاً فرمول نحوه‌ی پر شدن مدل را قرار داد و عملاً باید سیستم به صورت واقعی مدل شود. که این روش فرایند مدل‌سازی را بسیار پیچیده می‌کند. در روش پیشنهادی این مقاله نیز به دست آوردن گراف وابستگی غیرداده‌ای می‌تواند بسیار زمان‌بر باشد. در واقع مدل‌ساز به منظور مدل‌سازی سیستم‌های بزرگ، باید زمان زیادی را صرف شناسایی ارتباطات غیرمستقیم بین مؤلفه‌های سیستم کند.

#### ۵-۷-۲- پارامترهای در نظر گرفته شده در مدل

در روش‌های مقاله [۱۵, ۱۸] به ارتباط مستقیم مؤلفه‌ها و نحوه‌ی ترکیب خطاها باهمدیگر پرداخته شده است. درحالی‌که پارامترهای دیگری در فرایند انتشار خطا می‌توانند مؤثر باشند. در روش پیشنهادی این مقاله علاوه بر در نظر گرفتن ارتباط مستقیم مؤلفه‌ها، پارامترهای دیگری در نظر گرفته شده‌اند که این پارامترها عبارت‌اند از ترکیب احتمالی خطاها و وابستگی غیرداده‌ای که بین مؤلفه‌ها وجود دارد. همچنین در این روش ترتیب اجرایی مؤلفه‌ها در نظر گرفته شده است. یک مؤلفه ممکن است به‌طور مداوم در حال کار نباشد بلکه بعد از به اتمام رسیدن فعالیت یک مؤلفه دیگر، کار آن شروع شود. البته در مقاله [۱۶] احتمال فعالیت مؤلفه‌ها بعد از اتمام کار هر مؤلفه در نظر گرفته شده است.

#### ۵-۷-۳- نتایج به دست آمده از مدل

در روش پیشنهادی این مقاله می‌توان نتایج متنوعی از اجرای مدل به دست آورد که این نتایج عبارت‌اند از:

- تأثیر فعال شدن یک خطا در یک مؤلفه خاص بر رفتار خرابی سایر مؤلفه‌ها
- ارزیابی قابلیت اطمینان کل سیستم
- احتمال خرابی هر مؤلفه در هر لحظه
- مشاهده مقادیر ورودی و خروجی مؤلفه‌ها در هر لحظه
- ترتیب خرابی مؤلفه‌ها و نحوه انتشار تأثیرات خطا بر سایر مؤلفه‌ها
- پیدا کردن مؤلفه‌های حیاتی سیستم
- مشخص کردن مؤلفه‌های مناسب به منظور طراحی سیستم

در سایر روش‌های پیشنهاد شده در این حوزه مشاهده نحوه تأثیر یک خطا بر ورودی و خروجی سایر مؤلفه‌ها در هر لحظه و همچنین مشاهده رفتار خرابی تمامی مؤلفه‌ها یا غیرممکن است یا باید مدل بسیار پیچیده‌تر شود تا این پارامترها را مشاهده کرد.

- [12] Z. Zuyuan, W. An and S. Fangming, "Cascading Failures on Reliability in Cyber-Physical System," *IEEE Reliability Society*, vol. 65, no. 4, pp. 1745 - 1754, 2016.
- [13] Z. Huang and C. Wang, "Characterization of Cascading Failures in Interdependent Cyber-Physical Systems," *IEEE Transactions on Computers*, vol. 64, no. 8, pp. 2158-2168, 2015.
- [14] H. Peng, Z. Kan, D. Zhao, J. Han, J. Lu and Z. Hu, "Reliability analysis in interdependent smart grid systems," *Physica A: Statistical Mechanics and its Applications*, vol. 500, pp. 50-59, 2018.
- [15] C. Heracleous, M. M. Polycarpou, G. Ellinas, C. G. Panayiotou and P. Kolios, "Hybrid systems modeling for critical infrastructures interdependency analysis," *Reliability Engineering & System Safety*, vol. 165, pp. 89-101, 2017.
- [16] A. Morozov and K. Janschek, "Probabilistic error propagation model for mechatronic systems," *Mechatronics*, vol. 24, no. 8, pp. 1189-1202, 2014.
- [17] A. Morozov and K. Janschek, "Dual Graph Error Propagation Model for Mechatronic System Analysis," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 9893-9898, 2011.
- [18] S. Kabir, M. Walker and Y. Papadopoulos, "Dynamic system safety analysis in HiP-HOPS with Petri Nets and Bayesian Networks," *Safety Science*, vol. 105, pp. 55-70, 2018.
- [19] S. Kabir, Y. Papadopoulos, M. Walker, D. Parker, J. Ignacio Aizpurua, J. Lampe and E. Rde, "A Model-Based Extension to HiP-HOPS for Dynamic Fault Propagation Studies," in *5th International Symposium on Model-Based Safety and Assessment*, Aug 2017.
- [20] M. Walker, L. Bottaci and Y. Papadopoulos, "Compositional Temporal Fault Tree Analysis," in *International Conference on Computer Safety, Reliability, and Security*, 2007.
- [21] E. Edifor, M. Walker, N. Gordon and Y. Papadopoulos, "Using simulation to evaluate dynamic systems with weibull or lognormal distributions," in *Proceedings of the Ninth International Conference on Dependability and Complex Systems*, Brunow, June 2014.
- [22] L. Grunske and B. Kaiser, "Automatic generation of analyzable failure propagation models from component-level failure annotations," in *Fifth International Conference on Quality Software*, Melbourne, Sep 2005.
- [23] C. Zhou, X. Huang, X. Naixue, Y. Qin and S. Huang, "A class of general transient faults propagation analysis for networked control systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 4, pp. 647 - 661, 2015.
- [24] Y. Liu, D. Lu, L. Deng, T. Bai, K. Hou and Y. Zeng, "Risk assessment for the cascading failure of electric cyber-physical system considering multiple information factors," *IET Cyber-Physical Systems: Theory & Applications*, vol. 2, no. 4, pp. 155 - 160, 2017.
- [25] X. Ge, R. F. Paige and J. A. McDermid, "Probabilistic Failure Propagation and Transformation Analysis," in *28th International Conference on Computer Safety, Reliability, and Security*, Berlin, 2009.
- [1] R. Alur, Principles of Cyber-Physical Systems, Massachusetts: MIT Press, 2015.
- [2] S. Seshia and E. Lee, Introduction to Embedded Systems - A Cyber-Physical Systems Approach, MIT Press, 2017.
- [3] M. Fan, Z. Zeng, E. Zio, R. Kang and Y. Chen, "A stochastic hybrid systems model of common-cause failures of degrading components," *Reliability Engineering & System Safety*, vol. 172, pp. 159-170, 2018.
- [4] R. Kang and Z. Li, "Strategy for reliability testing and evaluation of cyber physical systems," in *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, Singapore, Dec 2015.
- [5] G. Simko, T. Levendovszky, M. Maroti and J. Sztipanovits, "Towards a theory for cyber-physical systems modeling," in *Proceedings of the 4th ACM SIGBED International Workshop on Design, Modeling, and Evaluation of Cyber-Physical Systems*, Berlin, April 2014.
- [6] R. Michael and P. Liggesmeyer, "Modeling and analysis of safety-critical cyber physical systems using state/event fault trees," in *International Conference on Computer Safety, Reliability and Security*, Toulouse, Sep 2013.
- [7] A. Avizienis, J.-C. Laprie, B. Randell and C. Landwehr, "Basic Concepts and Taxonomy of dependable and secure computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 1, no. 1, pp. 11-33, 2004.
- [8] W. H. Sanders and J. F. Meyer, "Stochastic activity networks: formal definitions and concepts," in *Lectures on formal methods and performance analysis*, New York, Springer, 2001, pp. 315 - 343.
- [9] M. Rahnamay Naeini and M. M. Hayat, "Cascading Failures in Interdependent Infrastructures: An Interdependent Markov-Chain Approach," *IEEE Transactions on Smart Grid*, vol. 7, no. 4, pp. 1997-2006, 2016.
- [10] R. A. Shuvro, Z. Wangt , P. Das, M. R. Naeini and M. M. Hayat, "Modeling cascading-failures in power grids including communication and human operator impacts," in *IEEE Green Energy and Smart Systems Conference*, Long Beach, Nov 2017.
- [11] S. V. Buldyrev, R. Parshani, G. Paul, H. Stanley and S. Havlin, "Catastrophic cascade of failures in interdependent networks," *Nature*, vol. 464, pp. 1025-1028, 2010.

## مراجع

## زیر نویس ها:

- |  |                                |
|--|--------------------------------|
| 9 Pathology  | 1 Hybrid Systems               |
| 10 Marcov chain                                    | 2 Fault                        |
| 11 Cyber-Physical Systems                          | 3 Propagation                  |
| 12 Strategy  | 4 Reliability                  |
| 13 failure propagation and transformation analysis | 5 Scalable                     |
| 14 Extended Place                                  | 6 Topology                     |
| 15 Converge  | 7 Stochastic Activity Networks |
|  | 8 Replication/Join             |