

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشکده مهندسی کامپیوتر و فناوری اطلاعات  
رشته مهندسی کامپیوتر - گرایش هوش مصنوعی

رساله دکتری

## یک مدل فرمال کنترل سوپروایزری برای تخصیص منابع در سیستم‌های پیچیده صنعتی

نگارنده: فاطمه جعفری نژاد

استاد راهنما

دکتر علی اکبر پویان

استاد مشاور

دکتر مرتضی زاهدی

دی ۱۳۹۵



مدیریت تحصیلات تکمیلی

باسمه تعالی

شماره:

تاریخ:

ویرایش:

فرم شماره ۱۲: صورت جلسه دفاع از رساله دکتری (Ph.D)

بدینوسیله گواهی می شود آقای خانم فاطمه دانشجو رشته روانشناسی به شماره دانشجویی ۹۱۲۵۱۹۵ در تاریخ ۱۳۹۹/۰۳/۰۳ از رساله خود با عنوان: تاثیر آموزش هیپنوتیزم بر کاهش اضطراب در بیماران مبتلا به سرطان دفاع و با اخذ نمره ۱۹ به درجه عالی نائل گردید.

الف) درجه عالی: نمره ۲۰-۱۹  (ب) درجه بسیار خوب: نمره ۱۸/۹۹ - ۱۷   
 ج) درجه خوب: نمره ۱۶/۹۹ - ۱۵  (د) غیر قابل قبول و نیاز به دفاع مجدد دارد   
 ه) رساله نیاز به اصلاحات دارد

ردیف	هیئت داوران	نام و نام خانوادگی	مرتبه علمی	امضاء
۱	دکتر علی اکبر پویان	استاد/ اساتید راهنما	استاد	
۲	دکتر مرضی زاهدی	مشاور / مشاورین	استاد	
۳	دکتر عباس دیدبان	استاد مدعو داخلی / خارجی	استاد	
۴	دکتر امیدرضا معروضی	استاد مدعو داخلی / خارجی	استاد	
۵	دکتر هدی مشایخی	استاد مدعو داخلی / خارجی	استاد	
۶	دکتر <u>سید مرتضی</u>	سرپرست ( نماینده ) تحصیلات تکمیلی دانشکده	استاد	

مدیر محترم تحصیلات تکمیلی دانشگاه:

ضمن تأیید مراتب فوق مقرر فرمائید اقدامات لازم بعمل آید.

رئیس دانشکده و رئیس هیأت داوران:

تاریخ و امضاء:



مدیریت تحصیلات تکمیلی

باسمه تعالی

شماره:

تاریخ: ۱۳۹۵، ۱، ۲۲

ویرایش:

پیوست شماره ۲

دانشکده: کامپیوتر و فن آوری اطلاعات

گروه: مهندسی مخابرات

پایان نامه کارشناسی ارشد / رساله دکتری آقای / خانم .....  
تحت عنوان: .....  
در تاریخ ..... توسط کمیته تخصصی زیر جهت اخذ مدرک کارشناسی ارشد / رساله دکتری ارزیابی گردید و با درجه ..... مورد پذیرش قرار گرفت.

امضاء	اساتید مشاور	امضاء	اساتید راهنما
	نام و نام خانوادگی: مریم زاهدی		نام و نام خانوادگی: علی زاهدی
	نام و نام خانوادگی:		نام و نام خانوادگی: مریم زاهدی

امضاء	نماینده تحصیلات تکمیلی	امضاء	اساتید داور
	نام و نام خانوادگی:		نام و نام خانوادگی: مریم زاهدی
	مفسور زاهدی		نام و نام خانوادگی: دکتر زاهدی
			نام و نام خانوادگی: دکتر زاهدی

تلاش ناچیزم را به یگانه دردانه هستی تقدیم می‌کنم، تا متبرک شود به نامش.

تقدیم به بال‌های پروازم؛ پدر و مادرم

تقدیم به همسرم؛ یکتا همراهم

تقدیم به برادر عزیزم و خواهران نازنینم

## تقدیر و تشکر

سپاس و ستایش خدای منان را که به عنایتش رخصتی داد تا گام کوچکی در زدودن جهالت از غبار اندیشه‌ام بردارم.

ابتدا از زحمات و پشتیبانی بی‌دریغ استاد ارجمندم جناب آقای دکتر علی اکبر پویان کمال تشکر را دارم که با راهنمایی‌های ارزنده خود این پژوهش را پربار ساختند. همچنین، مراتب تشکر و قدردانی خود را به استاد محترم، جناب آقای دکتر مرتضی زاهدی بیان می‌دارم که از هیچ زمانی برای مشورت در این رساله و بلکه در سراسر دوره دکتری بنده دریغ نکردند.

همچنین بر خود لازم می‌دانم از تمامی اساتید و معلمانی که در محضر آنها علم، ادب و اخلاق آموختم تشکر و قدردانی نموده و از خداوند متعال برای تمامی آن بزرگواران موفقیت و بهروزی مسألت دارم.

در نهایت خالص‌ترین و ویژه‌ترین سپاس‌هایم را به خانواده عزیزم نثار می‌کنم که در کلیه مراحل زندگی پشتیبان من بوده‌اند و همدلی‌هایشان امیدبخش انجام این رساله بود. از پدر نازنینم که علم را مهمتر از نان، نه حتی ثروت، می‌داند، از مادرم و خواهران و برادرم که همراه و پشتیبان من بودند، و از همسرم که همگام با من در مسیر این رساله پیش آمد متشکرم.

## تعهد نامه

اینجانب فاطمه جعفری نژاد دانشجوی دوره دکتری رشته مهندسی کامپیوتر-گرایش هوش مصنوعی دانشکده مهندسی کامپیوتر و فناوری اطلاعات دانشگاه شاهرود نویسنده رساله دکتری "یک مدل فرمال کنترل سوپروایزری برای تخصیص منابع در سیستم‌های پیچیده صنعتی" با راهنمایی دکتر علی‌اکبر پویان متعهد می‌شوم.

- تحقیقات در این رساله توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهش‌های محققان دیگر به مرجع مورد استفاده استناد شده است.
- مطالب مندرج در پایان‌نامه تاکنون توسط خود یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارائه نشده است.
- کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی شاهرود می‌باشد و مقالات مستخرج با نام «دانشگاه صنعتی شاهرود» و یا «Shahrood University of Technology» به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به دست آمدن نتایج اصلی رساله تأثیرگذار بوده‌اند در مقالات مستخرج از پایان‌نامه رعایت می‌گردد.
- در کلیه مراحل انجام این رساله، در مواردی که از موجود زنده (یا بافت‌های آن‌ها) استفاده شده است ضوابط و اصول اخلاقی رعایت شده است.
- در کلیه مراحل انجام این رساله، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته یا استفاده شده است اصل رازداری، ضوابط و اصول اخلاق انسانی رعایت شده است.

### تاریخ

امضای دانشجو

### مالکیت نتایج و حق نشر

- کلیه حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه‌های رایانه‌ای، نرم‌افزارها و تجهیزات ساخته شده) متعلق به دانشگاه صنعتی شاهرود می‌باشد. این مطلب باید به نحو مقتضی در تولیدات علمی مربوطه ذکر شود.
- استفاده از اطلاعات و نتایج موجود در رساله بدون ذکر مرجع مجاز نمی‌باشد.

## چکیده

بسیاری از سیستم‌های اطراف ما، مستقیماً و یا بطور انتزاعی، به کلاس سیستم‌های گسسته رخداد تعلق دارند. وجود فعالیت‌های موازی و عدم قطعیت ناشی از استقلال زمانی وقوع رخدادها در این سیستم‌ها، آنها را به سیستم‌هایی با پیچیدگی بالا تبدیل کرده است. این رساله، به مسئله تخصیص امن منابع در سیستم‌های گسسته رخداد می‌پردازد. در این سیستم‌ها، تخصیص امن منابع اشتراکی به فرآیندهای همروند بسیار اساسی است تا از بروز پدیده‌های نامطلوبی همچون بن‌بست، سرریز بافر یا صف و عدم بازگشت به حالت اولیه پیشگیری شود. به همین سبب، مدل‌های فرمال مانند شبکه‌های پتری برای کنترل و نظارت بر تخصیص منابع در این سیستم‌ها استفاده شده‌اند. این مدل‌ها علاوه بر قابلیت طراحی و شبیه‌سازی یک سیستم، امکان اعتبارسنجی آن را نیز فراهم می‌نمایند.

سه معیار مهم در ارزیابی یک سیاست کنترل نظارتی شامل بهینگی رفتاری، بهینگی ساختاری و بهینگی محاسباتی هستند. تحقیقات بسیاری در جهت بهبود برخی و یا تمامی این سه معیار به انجام رسیده است. استفاده از رویکردهای زبانی، مبتنی بر گراف دسترسی، رویکردهای مبتنی بر سایفون و رویکردهای مبتنی بر شباهت از جمله راهکارهای ارائه یک الگوریتم کنترل نظارتی بوده‌اند. این الگوریتم‌ها در سیستم‌های بزرگ مقیاس با پیچیدگی زمانی و مکانی بالایی روبرو می‌شوند. استفاده از روش‌های کاهش سایفون‌های مورد کنترل، معرفی مفهوم حالت مافوق و مفهوم پوشش برداری از جمله کارهای انجام شده در زمینه مواجهه با این مشکل هستند. البته مسئله سرعت و حافظه همچنان از مسائل باز در حوزه کنترل نظارتی هستند.

این رساله، با دو رویکرد مرتبط به هم، به ارائه کنترل نظارتی مبتنی بر شبکه‌های پتری در سیستم‌های بزرگ مقیاس می‌پردازد. در رویکرد اول، با استفاده از عملگرهای سنتز تعریف شده در شبکه‌های پتری به کاهش بار محاسباتی کنترل نظارتی بیشینه مجازیت خواهیم پرداخت. این روش



منجر به محاسبات پیمانهای کنترل نظارتی خواهد شد. همچنین استفاده از عملگر ادغام مکانها منجر به کاهش تعداد سایفونهای کنترل نشده مورد بررسی می شود. بعلاوه به اثبات حفظ ویژگی زنده بودن یک عملگر ادغام گذرها، در صورت برقراری شروط معین شده، می پردازیم. این نوع رویکرد مستقلا می تواند برای اثبات زنده بودن یک شبکه تحت یک عملگر سنتز نیز مورد توجه قرار گیرد. رویکرد دوم نیز با معرفی یک روش برنامه نویسی خطی، به حل مسئله کنترل نظارتی می پردازد. در این روش استفاده از مفاهیم بردار تخصیص و حالات مافوق، منجر به داشتن یک سوپروایزر پیشینه مجازیت، با ساختار کمینه و هزینه محاسباتی پایین خواهد شد. نهایتا با استفاده از رویکرد دوم، در طراحی پیمانهای یک کنترل نظارتی، به سوپروایزری خواهیم رسید که به بهبود سه معیار کنترل نظارتی می انجامد.

**کلمات کلیدی:** سیستمهای گسسته رخداد، سیستمهای تخصیص منابع، کنترل

**نظارتی، مدلسازی فرمال، شبکههای پتری.**

## مقالات استخراج شده از رساله

### مقالات پذیرفته شده در مجلات

- 1- F. Jafarinejad, A. A. Pouyan, Distributed Computation of Liveness-Enforcing Supervisory Control of Flexible Manufacturing Systems, Journal of advanced Manufacturing Systems, Vol. 16, No. 1, pp. 1-15, 2017.
- 2- F. Jafarinejad, A. A. Pouyan, A Modular Supervisory Control Design of Discrete Event Systems, Journal of advanced Manufacturing Systems, (in press).

### مقالات تحت داوری در مجلات

- 3- Modular Supervisory Control Design Method of Flexible Manufacturing Systems, Elsevier, Manufacturing Letters.

۴- مدلسازی و کنترل سوپروایزری تخصیص منابع در شبکه‌های ریلی با استفاده از شبکه‌های پتری،

مجله مدلسازی در مهندسی، دانشگاه سمنان

- 5- A Modular Petri Net based Supervisory Control Design Method for Flexible Manufacturing Systems, Springer, Japan Journal of Industrial and Applied Mathematics.
- 6- A Property Preserving Petri Net Synthesis Method for Supervisory Control System Design, The Computer Journal, Oxford University.
- 7- Collaborative Automation Using Petri net Entity Composition, Springer, Information Systems and e-Business Management.
- 8- A Modular Monitor-Based Supervisor to Enforce Well-Behavedness in Discrete Event Systems, Journal of Manufacturing Science and Engineering, Transactions of the ASME.
- 9- Modular Modelling and Verification of Manufacturing Cells, Springer, International Journal of Industrial Engineering: Theory, Applications and Practice.
- 10- Reducing Computational Complexity of Petri Net-Based Supervisory Control Using Synthesis Methods, Journal of Manufacturing Science and Engineering, Transactions of the ASME.

## مقالات چاپ شده در کنفرانس‌ها

- 1- A. A. Pouyan, F. Jafarinejad, "Collision Avoidance in Dynamic Multi-Agent Systems using Petri Net-based Supervision," International conference on Artificial Intelligence Energy and Manufacturing Engineering, Dubai (UAE), Jan. 7-8 2015.
- 2- F. Jafarinejad, A. A. Pouyan, "Decentralized Supervision of Mobile Sensor Networks using Petri Nets," International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol. 5, No.4, 23 Jul. 2015.
- 3- F. Jafarinejad, A. A. Pouyan, "A Decentralized Web Service Composition Framework using Petri Net Entity," 2th International Congress of Electrical Engineering, Computer Science and Information Technology(CSSE2015), Shahid Beheshti University, Tehran, pp. 363- 371, 23 Jul. 2015.
- 4- F. Jafarinejad, A. A. Pouyan, "A Modular Synthesis Approach for Intelligent Manufacturing System Design: a Petri net based Transformation Method," SPIS2015, Amirkabir University of Technology, Tehran, 16-17 Dec. 2015.
- 5- F. Jafarinejad, A. A. Pouyan, "A Divide and Conquer approach for Process Discovery," Conference on Technology, Energy and Data on Electrical and Computer Engineering, Kermanshah, 30-31 May 2015.

## فهرست مطالب

فصل ۱ : مقدمه .....	۱
۱-۱ سیستم‌های تخصیص منابع .....	۲
۲-۱ چالش‌ها و راه حل آنها .....	۵
۳-۱ فرضیات مسئله .....	۹
۴-۱ محدودیت‌های ناشی از فرضیات محدود کننده .....	۱۰
۵-۱ سیستم نهایی مورد انتظار .....	۱۱
۶-۱ معماری پیشنهادی .....	۱۲
۷-۱ نوآوری .....	۱۶
۸-۱ ساختار رساله .....	۱۸
۹-۱ جمع بندی .....	۱۹
فصل ۲ : چارچوب نظری مسئله .....	۲۱
۱-۲ مقدمه .....	۲۲
۲-۲ معرفی سیستم‌های گسسته رخداد .....	۲۳
۳-۲ روش‌های مدل‌سازی سیستم‌های گسسته رخداد .....	۲۵
۱-۳-۲ مدل‌سازی غیرفرمال .....	۲۶
۲-۳-۲ مدل‌سازی نیمه فرمال .....	۲۷
۳-۳-۲ مدل‌سازی فرمال .....	۲۷
۴-۳-۲ رویکرد ترکیبی .....	۳۱
۴-۲ معرفی شبکه‌های پتری .....	۳۱

۳۶	۱-۴-۲ قدرت شبکه‌های پتری در بیان ویژگی‌های مختلف سیستم‌ها
۳۸	۲-۴-۲ تعریف برخی از ویژگی‌ها در شبکه‌های پتری
۴۴	۳-۴-۲ انواع شبکه‌های پتری
۵۰	۴-۴-۲ عملگرهای سنتز در شبکه‌های پتری
۵۳	۵-۴-۲ عملگرهای ادغام
۵۸	۵-۲ کنترل مدل توسط کنترل نظارتی
۵۹	۱-۵-۲ معیارهای ارزیابی روش‌های کنترل نظارتی
۶۰	۶-۲ جمع‌بندی
۶۱	فصل ۳ : مروری تحلیلی بر منابع
۶۲	۱-۳ مقدمه
۶۵	۲-۲ معادلات محدودیت و رویکردهای کنترل نظارتی متمرکز
۶۸	۱-۲-۳ رویکرد زبانی
۶۹	۲-۲-۳ رویکرد تحلیل گراف دسترسی
۷۷	۳-۲-۳ رویکرد ساختاری
۹۶	۴-۲-۳ روش‌های مبتنی بر شباهت
۹۸	۳-۳ رویکردهای کنترل نظارتی غیرمتمرکز
۱۰۴	۴-۳ جمع‌بندی
۱۰۵	فصل ۴ : نمونه‌هایی از کاربرد شبکه‌های پتری در مدلسازی سیستم‌های گسسته رخداد
۱۰۶	۱-۴ مقدمه
۱۰۶	۲-۴ فرآیندکاوی پیمان‌های با استفاده از شبکه‌های پتری
۱۰۹	۳-۴ ترکیب وب سرویس‌ها

۱۱۴	۴-۴ شبکه‌های ریلی
۱۱۴	۱-۴-۴ مدلسازی شبکه
۱۱۷	۲-۴-۴ تحلیل شبکه
۱۱۸	۵-۴ جمع‌بندی
۱۱۹	فصل ۵: محاسبات پیمان‌های کنترل نظارتی
۱۲۰	۱-۵ مقدمه
۱۲۰	۲-۵ محاسبات پیمان‌های با استفاده از عملگر انتخاب
۱۲۳	۳-۵ محاسبات پیمان‌های کنترل نظارتی توسط عملگر ترتیب
۱۲۶	۴-۵ اثبات محاسبات پیمان‌های کنترل نظارتی در برهم‌کنشی که امکان اجرای همزمان ندارد
۱۲۷	۱-۴-۵ مثال کاربردی
۱۳۱	۵-۵ محاسبات پیمان‌های کنترل نظارتی با استفاده از عملگر برگ‌برگ‌سازی
۱۳۴	۱-۵-۵ مثال کاربردی
۱۳۵	۶-۵ استفاده از عملگر ادغام مکان‌ها در کاهش محاسبات کنترل نظارتی
۱۴۰	۱-۶-۵ مثال کاربردی
۱۴۲	۷-۵ جمع‌بندی
۱۴۳	فصل ۶: معرفی قضایای مربوط به حفظ زنده بودن توسط عملگر ادغام گذرها
۱۴۴	۱-۶ مقدمه
۱۴۵	۲-۶ معرفی و اثبات قوانینی برای حفظ خوش‌رفتاری پس از اعمال ادغام
۱۵۵	۱-۲-۶ مثال کاربردی برای حفظ زنده بودن توسط ادغام گذرها
۱۵۸	۳-۶ استفاده از عملگر ادغام در ترکیب همروند دو پیمان
۱۶۱	۱-۳-۶ مثال کاربردی برای حفظ زنده بودن توسط ادغام گذرهای غیر مجاور

۱۶۷	۴-۶ جمع‌بندی
۱۶۹	فصل ۷ : بهبود سرعت در طراحی سوپروایزر بیشینه مجازیت با ساختار کمینه
۱۷۰	۱-۷ مقدمه
۱۷۰	۲-۷ بردار تخصیص و بردار تعمیم
۱۷۱	۳-۷ محاسبه مجموعه کمینه بردارهای تخصیص مجاز
۱۷۷	۴-۷ محاسبه کنترل نظارتی کمینه ساختاری
۱۸۰	۵-۷ نتایج تجربی
۱۸۴	۶-۷ جمع‌بندی
۱۸۵	فصل ۸ : نتیجه‌گیری و پیشنهاد کارهای آتی
۱۸۶	۱-۸ نتیجه‌گیری
۱۸۹	۲-۸ پیشنهاد کارهای آتی

## فهرست شکل‌ها

- شکل ۱-۱ شمای کلی طراحی یک مدل خوش‌رفتار در سیستم‌های کوچک. .... ۱۳
- شکل ۲-۱ روند کلی اعمال پیمان‌های کنترل نظارتی در یک سیستم گسسته. رخداد ..... ۱۵
- شکل ۱-۲ مثالی از انتقال وضعیت در یک سیستم گسسته رخداد. .... ۲۴
- شکل ۲-۲ نحوه تغییر حالت شبکه پتری با وقوع یک گذر فعال. اجرای گذر فعال  $t_0$  در شکل الف منجر به تغییر وضعیت سیستم به شکل ب می‌شود. در این شکل گذرهای فعال با رنگ طوسی و گذرهای غیرفعال با رنگ مشکی نشان داده شده‌اند. .... ۳۵
- شکل ۳-۲ نمونه‌هایی از شبکه‌های پتری. الف) نمونه‌ای از شبکه پتری دارای تصادم. ب) نمونه‌ای از شبکه پتری دارای تعارض. .... ۳۶
- شکل ۴-۲ مدل شدن انحصار متقابل توسط شبکه‌های پتری. شبکه پتری مسئله خواننده-نویسنده. .... ۳۷
- شکل ۵-۲ یک فرآیند شبکه پتری و فرآیند همبسته متناظر آن [۴۰]. .... ۴۵
- شکل ۶-۲ شمای کلی ترکیب پیمان‌های  $B_1$  و  $B_2$  با استفاده از عملگر انتخاب [۵۵]. .... ۵۱
- شکل ۷-۲ شمای کلی ترکیب پیمان‌های  $B_1$  و  $B_2$  با استفاده از عملگر ترتیب [۵۵]. .... ۵۱
- شکل ۸-۲ شمای کلی ترکیب پیمان‌های  $B_1$  و  $B_2$  با استفاده از عملگر برگ‌برگ‌سازی [۵۵]. .... ۵۳
- شکل ۹-۲ شمای کلی عملگر ادغام گذرها [۵۷]. الف) شبکه پتری اولیه. ب) شبکه پتری پس از ادغام گذرها. دوایر پررنگ نشان‌دهنده مجموعه‌ای از مکان‌ها و مستطیل‌های سفیدرنگ نشان‌دهنده مجموعه گذرها هستند. .... ۵۵
- شکل ۱-۴ زیر مدل‌های شبکه پتری بدست آمده پس از فرآیند کاوی برای زیرلاگ فایل‌های  $\{1,2\}$ ،  $\{3,4\}$  و  $\{5,6\}$ . .... ۱۰۸
- شکل ۲-۴ - مدل شبکه پتری بدست آمده پس از ترکیب زیرمدل‌های شکل ۱-۴ با استفاده از روش ترکیب مبتنی بر قانون. .... ۱۰۹



- شکل ۳-۴- وبسرویس‌های eBay و TCP و نحوه برهم کنش آنها با سرویس دیگر [۱۳۶].
- شکل سمت چپ، نمودار عملکرد eBay و شکل سمت راست، نمودار TCP را نشان می‌دهد. .... ۱۱۱
- شکل ۴-۴- مدل موجودیت شبکه پتری برای وبسرویس‌های eBay و TCP. .... ۱۱۱
- شکل ۴-۵- مدل موجودیت شبکه پتری ترکیبی برای وبسرویس‌های eBay و TCT. .... ۱۱۴
- شکل ۴-۶- شمای یک ارتباط بین ایستگاهی ایجاد شده با  $n$  خط ریلی [۱۳۴]. .... ۱۱۵
- شکل ۴-۷- مدل شبکه پتری یک خط ریلی واقع در یک ایستگاه. .... ۱۱۵
- شکل ۴-۸- مدل تجریدی یک ایستگاه راه آهن با  $n$  خط ریلی. .... ۱۱۵
- شکل ۴-۹- مدل شبکه پتری ایستگاه راه آهن با  $n$  خط ریلی پس از پالایش. .... ۱۱۶
- شکل ۴-۱۰- مدل شبکه پتری یک ایستگاه راه آهن  $n$  خطی پس از اعمال کنترل نظارتی. .... ۱۱۸
- شکل ۵-۱- مدل شبکه پتری پیمان‌های  $B_1$ ،  $B_2$ ، و  $B_3$  در سیستم مثال بخش ۵-۴-۱. .... ۱۲۹
- شکل ۵-۲- مدل خوش‌رفتار شبکه پتری برای سیستم کلی پس از اعمال کنترل نظارتی روی پیمان‌ها و سنتز آنها. .... ۱۳۰
- شکل ۵-۳- مدل شبکه پتری پیمان‌های پایه‌ای  $B_1$ ،  $B_2$ ، و  $B_3$  در مثال بخش ۵-۵-۳. .... ۱۳۴
- شکل ۵-۴- مدل شبکه پتری سیستم کلی پس از اعمال کنترل نظارتی روی پیمان‌های مجزا و سنتز آنها. .... ۱۳۵
- شکل ۵-۵- یک مدل شبکه پتری  $S^3PR$  (زیر کلاسی از شبکه‌های پتری  $S^3PGR$ ). .... ۱۳۸
- شکل ۵-۶- مدل تجزیه شده شبکه پتری شکل ۵-۵ به دو پیمان. .... ۱۳۸
- شکل ۵-۷- شمای کلی یک سیستم تولیدی خودکار. .... ۱۴۰
- شکل ۵-۸- مدل شبکه پتری کنترل شده شکل ۵-۵. .... ۱۴۲
- شکل ۶-۱- مثالی برای روشن سازی قضیه ۶-۲. الف) شبکه اولیه که  $SM$ ،  $FC$  و  $AC$  است. ب) شبکه ادغام شده ( $T_2$  و  $T_3$  به گذر واحد  $T_{23}$  ادغام شده‌اند) که  $SM$ ،  $FC$  و  $AC$  نیست. .... ۱۵۰
- شکل ۶-۲- مثالی برای روشن سازی قضیه ۶-۵. الف) شبکه اولیه که همبند نیست. ب) شبکه ادغام شده ( $T_1$ ،  $T_4$  به  $T_{14}$  و گذرهای  $T_2$ ،  $T_3$  به  $T_{23}$  ادغام شده‌اند) که قویا همبند است. .... ۱۵۲

- شکل ۳-۶ مثال روشن کننده قضیه ۶-۷. الف) شبکه اولیه، یک شبکه پتری معمولی و خالص است. ب) شبکه پتری ادغام شده ( $T_1, T_2, T_3$  به گذر واحد  $T_{123}$  ادغام می‌شود) معمولی یا خالص نیست. ۱۵۴.....
- شکل ۴-۶ شمای کلی سیستم تولیدی مثال ۶-۲-۱. ۱۵۶.....
- شکل ۵-۶- مدل شبکه پتری پیمان‌های پایه در سیستم تولیدی شکل ۶-۴. الف) مدل شبکه پتری ماشین M1. ب) مدل شبکه پتری ماشین M2. ج) مدل شبکه پتری سیستم نقاله. د) مدل شبکه پتری بازوی رباطیکی R1. ه) مدل شبکه پتری ماشین M3. ۱۵۷.....
- شکل ۶-۶- شبکه پتری سیستم کلی شکل ۶-۴ پس از اعمال عملگر ادغام گذر روی پیمان‌های پایه در شکل ۶-۵. ۱۵۸.....
- شکل ۷-۶ شمای یک کارخانه تولیدی. شکل برگرفته از مرجع [۵۴] با اعمال تغییرات. ۱۶۱.....
- شکل ۸-۶- مدل شبکه پتری پیمان‌های یک سیستم تولیدی خودکار. الف) مدل شبکه پتری برای ماشین CNC-mill. ب) مدل شبکه پتری نقاله. ج) مدل شبکه پتری برای CNC-lathe. د) مدل شبکه پتری برای CMM. ه) مدل شبکه پتری VQC. و) مدل شبکه پتری ASS. ز) مدل شبکه پتری ربات ۳. ح) مدل شبکه پتری ربات ۴. ط) مدل شبکه پتری ربات ۵. ی) مدل شبکه پتری ربات ۲. ۱۶۳.....
- شکل ۹-۶- سیستم کلی حاصل از ادغام گذرها در پیمان‌های نشان داده شده در شکل ۶-۸. ۱۶۷.....
- شکل ۱-۷- یک مدل شبکه پتری. ۱۸۱.....
- شکل ۲-۷- یک مدل شبکه پتری برگرفته از مرجع [۴۳]. ۱۸۱.....
- شکل ۳-۷- یک مدل شبکه پتری برگرفته از مرجع [۱۱۱]. ۱۸۲.....
- شکل ۴-۷- یک مدل شبکه پتری برگرفته از مرجع [۴۲]. ۱۸۳.....

## فهرست جداول

- جدول ۱-۴-۱- مثالی از یک لاگ فایل ساده، شامل ۶ اجرای مختلف از یک فرآیند..... ۱۰۷
- جدول ۱-۵- نشان‌گذاری اولیه و ماتریس برخورد مکان‌های کنترلی محاسبه شده در مثال  
بخش ۱-۴-۵..... ۱۲۹
- جدول ۲-۵- مقایسه پیچیدگی زمانی روش‌های کنترل نظارتی پیمان‌های و متمرکز..... ۱۳۳
- جدول ۳-۵- جدول نشان‌دهنده بدون نشانه شدن هر یک از سایفون‌های کمینه پایه‌ای شکل  
۶-۵ در یک نشان‌گذاری غیرقانونی..... ۱۴۱
- جدول ۱-۶- تفسیر مکان‌های پیمان‌های نشان داده شده در شکل ۶-۸..... ۱۶۴
- جدول ۲-۶- گروه‌های گذرهای کاندید برای ادغام و برجسب ادغام متناظر آنها..... ۱۶۵
- جدول ۱-۷- مقایسه سرعت الگوریتم کنترل نظارتی مبتنی بر حالت مافوق با استفاده از روش  
تکرار شونده [۱۴۰] و روش پیشنهادی مبتنی بر ILP..... ۱۸۳

## فهرست علائم اختصاری

<b>AC</b>	<b>Asymmetric Choice</b>
<b>AMS</b>	Automated Manufacturing System
<b>DEM</b>	Discrete Event Models
<b>DES</b>	Discrete Event Systems
<b>DFZ</b>	Deadlock-Free Zone
<b>DZ</b>	Deadlock Zone
<b>FBM</b>	First-Met Bad Markings
<b>FC</b>	Free Choice
<b>GMEC</b>	Generalized Mutual Exclusion Constraints
<b>GS<sup>3</sup>PR</b>	Generalized System of Simple Sequential Processes with Resources
<b>ILP</b>	Integer Linear Programming
<b>IMNM</b>	Implementation of Maximal Number of MTSIs
<b>LESC</b>	Liveness Enforcing Supervisory Control
<b>LZ</b>	Live Zone
<b>MCP</b>	Minimal number of Control Places Problem
<b>MFPP</b>	Maximal Number of Forbidden Markings
<b>MG</b>	Marked Graph
<b>MPP</b>	Minimal Number of P-Semiflows Problem
<b>MST</b>	Marked Siphon Trap
<b>MTSI</b>	Marking/Transition Separation Instances
<b>OSDTR</b>	Optimal Supervisor Design by the Theory of Regions
<b>PC<sup>2</sup>R</b>	Processes Competing for Conservative Resources
<b>PI</b>	P-Invariant
<b>PLC</b>	Programmable Logic Controller
<b>PN</b>	Petri Nets
<b>PTM</b>	Propositional Transitions Model
<b>RAS</b>	Resource Allocation Systems
<b>S<sup>3</sup>PR</b>	System of Simple Sequential Processes with Resources
<b>SCT</b>	Supervisory Control Theory
<b>SFC</b>	Sequential Function Chart
<b>SM</b>	State Machine
<b>SPQR</b>	Systems of Processes Quarrelling over Resources
<b>ST</b>	Siphon Trap
<b>TI</b>	T-Invariant
<b>UML</b>	Unified Modelling language
<b>WS<sup>3</sup>PR</b>	Weighted System of Simple Sequential Processes with Resources

## فهرست علائم

<b>N</b>	یک شبکه پتری
<b>P</b>	مجموعه محدودی از مکان‌ها
<b>T</b>	مجموعه محدودی از گذرها
<b>F</b>	مجموعه کمان‌ها
<b>W</b>	ماتریس عددی نشان‌دهنده وزن اختصاص داده شده به هر یک از کمان‌ها
<b>M<sub>0</sub></b>	بردار عددی نشان‌دهنده نشانه‌گذاری اولیه
<b>M</b>	بردار عددی نشان‌دهنده یک نشانه‌گذاری
<b>M [t]</b>	فعال بودن گذر t در نشانه‌گذاری M
<b>C</b>	ماتریس برخورد یک شبکه پتری
<b>In</b>	ماتریس ورودی یک شبکه پتری
<b>Out</b>	ماتریس خروجی یک شبکه پتری
<b>R(N, M<sub>0</sub>)</b>	گراف دسترسی شبکه پتری N با نشانه‌گذاری اولیه M.
<b>R(-N, M<sub>0</sub>)</b>	گراف دسترسی معکوس شبکه پتری N با نشانه‌گذاری اولیه M.
<b>P<sub>u</sub></b>	زیرمجموعه‌ای از مجموعه مکان‌های یک شبکه پتری با نام مکان‌های بلااستفاده
<b>P<sup>0</sup></b>	زیرمجموعه‌ای از مکان‌های یک شبکه پتری مبتنی بر سیستم‌های تولیدی با نام مکان‌های بیکار
<b>P<sub>A</sub></b>	زیرمجموعه‌ای از مکان‌های یک شبکه پتری مبتنی بر سیستم‌های تولیدی با نام مکان‌های عملیاتی
<b>P<sub>R</sub></b>	زیرمجموعه‌ای از مکان‌های یک شبکه پتری مبتنی بر سیستم‌های تولیدی با نام مکان‌های منبع
<b>B</b>	یک فرآیند شبکه پتری
<b>P<sub>e</sub></b>	مکان ورودی یک فرآیند شبکه پتری
<b>P<sub>x</sub></b>	مکان خروجی یک فرآیند شبکه پتری
<b>S</b>	زیرمجموعه‌ای از مجموعه مکان‌های یک شبکه پتری
<b>I</b>	بردار عددی نشان‌دهنده یک P-Invariant
<b>P<sub>c</sub></b>	مکان کنترلی
<b>V</b>	مجموعه تمام مکان‌های کنترلی
<b>C<sub>c</sub></b>	ماتریس برخورد سوپروایزر
<b>M<sub>0c</sub></b>	نشانه‌گذاری اولیه مکان‌های کنترلی یک سوپروایزر
<b>n<sub>l</sub></b>	تعداد محدودیت‌های یک مجموعه GMEC

<b>L</b>	ماتریس عددی نشان دهنده ضرایب و وزن هر یک از مکان‌های یک شبکه پتری در نشانه‌گذاری‌های قانونی مشخص شده توسط یک مجموعه GMEC
<b><math>\beta</math></b>	بردار عددی نشان دهنده بالاترین حد جمع وزندار مکان‌های یک شبکه پتری در نشانه‌گذاری‌های قانونی مشخص شده توسط یک مجموعه GMEC
<b><math>\mu(L, \beta)</math></b>	مجموعه بردارهای قانونی مشخص شده با استفاده از یک GMEC با ضرایب L و $\beta$
<b><math>M_L</math></b>	مجموعه نشانه‌گذاری‌های قانونی
<b><math>M_{FBM}</math></b>	مجموعه نشانه‌گذاری‌های غیرقانونی
<b><math>M_L^*</math></b>	مجموعه کمینه نشانه‌گذاری‌های قانونی پوشش دهنده
<b><math>M_{FBM}^*</math></b>	مجموعه کمینه نشانه‌گذاری‌های غیرقانونی تحت پوشش
<b><math>M_{FI}</math></b>	مجموعه تمام حالات مافوق برای مجموعه نشانه‌گذاری‌های غیرقانونی
<b>D</b>	مجموعه بردارهای تخصیص یک مجموعه نشانه‌گذاری
<b>G</b>	مجموعه بردارهای تعمیم یک مجموعه نشانه‌گذاری
<b><math>t_q</math></b>	یک گذر خطرناک در شبکه‌های پتری که اجرای آن می‌تواند منجر به خروج از منطقه زنده شود.
<b><math>\Omega</math></b>	مجموعه موارد جداکننده نشانه‌گذاری خطرناک/ گذر
<b><math>\Omega^*</math></b>	مجموعه جداکننده‌های بهینه
<b><math>\Gamma_M</math></b>	هر مسیر در منطقه زنده از $M_0$ به $M_L$
<b><math>\mathcal{C}_L</math></b>	مجموعه تمام مدارهای موجود در منطقه زنده یک شبکه پتری
<b><math>\gamma</math></b>	یک مدار در منطقه زنده یک شبکه پتری
<b>f</b>	یک متغیر دارای مقدار ۰ یا ۱
<b>h</b>	یک متغیر دارای مقدار ۰ یا ۱
<b>O</b>	یک عدد خیلی بزرگ
<b>Q</b>	یک حالت در اتوماتا
<b>Z</b>	مجموعه اعداد صحیح
<b>N</b>	مجموعه اعداد طبیعی
<b> x </b>	تعداد عناصر مجموعه x
<b><math>N_x</math></b>	مجموعه اندیس‌ها از ۱ تا  x

## فصل ۱: مقدمه

## ۱-۱ سیستم‌های تخصیص منابع<sup>۱</sup>

در سیستم‌های صنعتی، استفاده اشتراکی از منابع، بدلیل صرفه‌جویی اقتصادی در منابع و بالا بردن نرخ بهره‌وری مناسب از آنها، کاربرد دارند. در نتیجه، مجموعه دو یا چند فرآیند همروند<sup>۲</sup> به رقابت برای دسترسی به منابع اشتراکی می‌پردازند. این سیستم‌ها در یک دسته بندی کلی از سیستم‌ها با نام سیستم‌های تخصیص منابع (RAS) قرار می‌گیرند. تعریف دقیق این سیستم‌ها در تعریف ۱-۱ مشخص شده است.

تعریف ۱-۱- سیستم‌های تخصیص منابع: سیستم‌های گسسته رخدادی<sup>۳</sup> (DES) [۱] هستند که در آنها مجموعه متناهی<sup>۴</sup> از فرآیندهای همروند، در یک محیط رقابتی برای دسترسی به مجموعه محدود منابع تلاش می‌کنند [۲].

رخدادها<sup>۵</sup> و منابع اشتراکی با توجه به سیستم مورد مطالعه تعریف می‌شوند. بعنوان مثال در یک شبکه کامپیوتری، پهنای باند شبکه بعنوان یک منبع اشتراکی بین فرآیندهای دخیل در ارتباطات شبکه‌ای مطرح است و وقایعی همچون ورود (یا خروج) یک بسته<sup>۶</sup> به (یا از) یک سوئیچ می‌توانند بعنوان رخدادهای سیستمی در نظر گرفته شوند. در یک سیستم عامل، انواع حافظه و واحدهای پردازشی و محاسباتی می‌توانند بعنوان منابع مشترک بین فرآیندها در نظر گرفته شوند. ورود یا خروج فرآیندها از صف پردازش، درخواست دسترسی به هر یک از منابع اشتراکی از سوی فرآیندها، تخصیص منابع به آنها و یا آزادسازی منابع از سوی فرآیندها نیز متناظر با وقوع یک رخداد در یک سیستم عامل هستند. در یک سیستم کنترل ترافیک هواپیمایی، فضاهاى مابین خطوط فرضی هوایی یا

<sup>1</sup> Resource Allocation Systems

<sup>2</sup> Concurrent Processes

<sup>3</sup> Discrete Event Systems

<sup>4</sup> Finite

<sup>5</sup> Events

<sup>6</sup> Packet



باند‌های فرود، منابع اشتراکی بین هواپیماها هستند و ورود یا خروج هواپیما به فضا‌های اشتراکی می‌تواند متناظر با وقوع یک رخداد در نظر گرفته شوند. همچنین در یک سیستم تولیدی خودکار<sup>1</sup> (AMS)، دستگاه‌های پردازشگر، بافرهای مورد استفاده در دستگاه‌ها، بازوهای رباتیکی و ربات‌های حامل قطعات می‌توانند بعنوان منابع اشتراکی در نظر گرفته شوند.

مهم‌ترین مسئله در این سیستم‌ها تخصیص امن منابع است. منظور از تخصیص امن، مرتب-سازی دنباله اجرای رخداد‌های مرتبط با تخصیص (رخداد‌های شروع و پایان هر مرحله از یک فرآیند، درخواست تخصیص و آزادسازی منابع) بگونه‌ای است که یک رفتار حافظ شرط زنده بودن<sup>2</sup>، کراندار بودن<sup>3</sup> و بازگشت‌پذیری<sup>4</sup> را از سیستم به نمایش گذارد. بعنوان مثال، از مسائل جالب توجه در این حوزه پیشگیری از بن‌بست<sup>5</sup> و اجتناب از بن‌بست<sup>6</sup> هستند. در یک حالت بن‌بست، رخدادی برای تغییر حالت سیستم تعریف نشده است و با قرار گرفتن سیستم در این حالت، در آن به دام خواهد افتاد. در نتیجه، وقوع بن‌بست موجب پایین آمدن بازده سیستم و کاهش بهره‌وری از منابع سیستمی خواهد شد. بنابراین می‌توان گفت وقوع بن‌بست به گونه‌ای نشان‌دهنده وجود یک تخصیص ناامن و یا دنباله-ای از تخصیص‌های ناامن در سیستم است.

اعمال کنترل نظارتی<sup>7</sup> بر سیستم‌های گسسته رخداد، منجر به تشکیل یک سیستم کنترل حلقه بسته<sup>8</sup> خواهد شد که مجموعه حالات سیستمی را محدود و محصور به حالات امن و عاری از بن‌بست می‌نماید.

---

<sup>1</sup> Automated Manufacturing System

<sup>2</sup> Liveness

<sup>3</sup> Boundedness

<sup>4</sup> Reversibility

<sup>5</sup> Deadlock Prevention

<sup>6</sup> Deadlock Avoidance

<sup>7</sup> Supervisory Control

<sup>8</sup> Closed-Loop Control System

این سیستم‌ها بدلیل داشتن فرآیندهای همروند و غیر وابسته به زمان بودن تکامل سیستم، در کلاس سیستم‌های با پیچیدگی بالا تقسیم‌بندی می‌شوند. تعریف ۱-۲- بطور دقیق به معرفی ویژگی- های سیستم‌های پیچیده<sup>۱</sup> مورد مطالعه می‌پردازد.

تعریف ۱-۲- سیستم‌های پیچیده: سیستم‌هایی هستند که:

- تکامل حالات<sup>۲</sup> این سیستم‌ها به رخدادهایی وابسته است که زمان وقوع و فواصل وقوع آنها کاملا مستقل از زمان است و دارای عدم قطعیت در زمان وقوع رخدادها هستند.
- در این سیستم‌ها فرآیندهای همروند دسترسی اشتراکی به منابع محدود سیستمی دارند و باید از پدیده‌های نامطلوبی مانند وقوع بن‌بست یا سرریز<sup>۳</sup> ظرفیت‌های سیستمی جلوگیری شود.
- این سیستم‌ها در مقیاس بزرگ مطرح هستند که تحلیل<sup>۴</sup> کامل آنها روندی بسیار هزینه‌بر است. بنابراین استفاده از مکانیزمی که بتواند یک طراحی پیمانه‌ای<sup>۵</sup> از سیستم را ارائه دهد، در جلوگیری از وقوع مشکلاتی مانند انفجار فضای حالت<sup>۶</sup> بسیار ضروری است.

منظور از سیستم بزرگ مقیاس، سیستمی است که از نظر تحلیلی و مدلسازی دارای اجزای بسیاری است؛ این اجزا اعم از رخدادهای اثرگذار بر سیستم و پیش‌شرط‌های فیزیکی و منطقی برای اجرای یک رخداد است. در این رساله، با بهره‌گیری از عملگرهای سنتز<sup>۷</sup> معرفی شده برای شبکه‌های پتری<sup>۸</sup> (PN)، به ارائه کنترل نظارتی پیمانه‌ای<sup>۹</sup> خواهیم پرداخت. بهره‌مندی از این روش پیمانه‌ای، علاوه بر کاهش بار محاسباتی، به ایجاد سوپروایزر توزیع شده<sup>۱۰</sup> می‌انجامد که کاهش هزینه ارتباطی و

---

<sup>1</sup> Complex Systems

<sup>2</sup> State Evolution

<sup>3</sup> Overflow

<sup>4</sup> Analysis

<sup>5</sup> Modular

<sup>6</sup> State Space Explosion

<sup>7</sup> Synthesis Operators

<sup>8</sup> Petri Nets

<sup>9</sup> Modular Supervisory Control

<sup>10</sup> Distributed Supervisory Control

افزایش امنیت سیستم کنترلی را نتیجه می‌دهد. همچنین، روشی را برای کاهش ساختار<sup>۱</sup> سوپروایزر<sup>۲</sup> با استفاده از کاهش تعداد مانیتورهای<sup>۳</sup> ( مکان‌های کنترلی<sup>۴</sup> ) مورد نیاز یک سوپروایزر ارائه خواهیم داد. کاهش ساختار سوپروایزر منجر به کاهش هزینه سخت‌افزاری و نرم‌افزاری و هزینه ارتباطی بین سوپروایزر و سیستم تحت کنترل خواهد شد.

## ۱-۲ چالش‌ها و راه حل آنها

پیچیدگی‌های ذاتی سیستم‌های تخصیص منابع و داشتن فضای حالت گسسته<sup>۵</sup>، کنترل خودکار<sup>۶</sup> این سیستم‌ها با استفاده از روش‌های کنترل کلاسیک را ناممکن می‌سازد [۳]. به همین سبب، استفاده از روش‌های مدلسازی فرمال<sup>۷</sup>، برای مدلسازی و کنترل این سیستم‌ها بسیار رایج است. انتخاب یک مدل فرمال با ابزارهای تحلیلی قوی و دارای قدرت بالا در شرح و بازنمایی ویژگی-های سیستم بسیار حائز اهمیت است. وجود قابلیت نمایش گرافیکی در یک مدل، کمک شایانی به فهم دقیقتر و سریعتر ساختار سیستم می‌نماید و عیب‌یابی‌های زمان طراحی را سرعت می‌بخشد. در این رساله، فرمالیزم شبکه‌های پتری، بعنوان ابزار گرافیکی و ریاضیاتی سطح بالا برای بیان و اعتبارسنجی<sup>۸</sup> ضوابط<sup>۹</sup> خاص مسئله، مورد استفاده قرار می‌گیرد. از دیگر ویژگی‌های حائز اهمیت این فرمالیزم، امکان کامپایل شدن یک مدل مبتنی بر شبکه پتری به کد سخت‌افزاری [۴] و عیب‌یابی آنها [۵، ۶] است.

---

<sup>1</sup> Reduction of Structure

<sup>2</sup> Supervisor

<sup>3</sup> Monitors

<sup>4</sup> Control Places

<sup>5</sup> Discrete State Space

<sup>6</sup> Automatic Control

<sup>7</sup> Formal Methodes

<sup>8</sup> Verification

<sup>9</sup> Specifications

از طرفی، در یک سناریوی تخصیص منابع امکان وقوع پدیده‌های ناخوشایندی همچون بن-بست، سرریز صف یا بافر موجود در سیستم و عدم دستیابی به حالت اولیه وجود دارد. به همین سبب، یک مدل از سیستم‌های تخصیص منابع نیاز به اعتبارسنجی ویژگی‌های مرتبط با خوش‌رفتاری<sup>۱</sup> سیستم یعنی زنده بودن، کرانداري و بازگشت‌پذیری دارد. پایه و اساس ریاضیاتی شبکه‌های پتری یک معادله حالت گسسته را برای سیستم‌های گسسته رخداد فراهم می‌نماید. این معادله، یک تحلیل از مدل مربوطه، از نظر دارا بودن خواص رفتاری<sup>۲</sup> بیان شده، ارائه می‌دهد.

استفاده از روش‌های مبتنی بر معادله حالت، در سیستم‌هایی با مقیاس بزرگ و فضای حالت بزرگ، ناکارآمد است. قضایای بسیاری در مورد ارتباط خواص رفتاری سیستم با خواص ساختاری<sup>۳</sup> شبکه‌های پتری بیان شده‌اند. استفاده از خواص ساختاری جهت اعتبارسنجی خواص رفتاری، هزینه اعتبارسنجی را کاهش می‌دهد. از طرفی، اگر روش‌های تحلیلی (مبتنی بر رفتار یا مبتنی بر ساختار)، کراندار نبودن یک سیستم را تأیید کنند، اعمال کنترل بر سیستم در جهت کراندار شدن ساده است. اما اگر سیستم مورد بررسی ویژگی زنده بودن را تضمین نکند، انتظار نامحدود فرآیندها به دلیل درخواست‌های چرخه‌ای، منجر به حالت ناخوشایند بن‌بست خواهد شد. نظریه کنترل نظارتی<sup>۴</sup> (SCT) برای تحدید حالات سیستم در یک مجموعه حالات غیر بن‌بست و اعمال ویژگی زنده بودن در یک سیستم ارائه شده است.

مقیاس بزرگ سیستم‌های صنعتی، طراحی و تحلیل این سیستم‌ها را با چالش انفجار فضای حالت مواجه می‌سازد و محاسبات کنترل نظارتی را با چالش پیچیدگی محاسباتی<sup>۵</sup> بالا روبرو می‌سازد. یکی از رویکردهای قابل استفاده برای حل این مسئله، استفاده از رویکرد کلی تقسیم و حل<sup>۱</sup> است.

---

<sup>1</sup> Well-Behavedness

<sup>2</sup> Behavioral Properties

<sup>3</sup> Structural Properties

<sup>4</sup> Supervisory Control Theory

<sup>5</sup> Computational Complexity

از طرفی، یکی از معایب شبکه‌های پتری، نداشتن مکانیزم ذاتی ترکیب‌پذیری برای ترکیب<sup>۲</sup> دو یا چند زیرمدل و بدست آوردن مدل کلی است. یکی از رویکردهایی که محققان برای جبران این ضعف اتخاذ نموده‌اند، معرفی روش‌هایی برای تبدیل یک کلاس خاص از شبکه‌های پتری به کلاسی از مدل‌های فرمال ترکیب‌پذیر دیگر و بلعکس است. بدین وسیله می‌توان زیر مدل‌های شبکه‌های پتری را با استفاده از فرمالیزم ترکیب‌پذیر ترکیب نموده، مدل کلی سیستم را بدست آورد. رویکرد دیگری که توجه محققان را جلب نموده است، معرفی روش‌هایی برای اضافه کردن قابلیت ترکیب‌پذیری به شبکه‌های پتری بوده است. در نتیجه اتخاذ این رویکرد، عملگرهای سنتز مانند عملگرهای ادغام<sup>۳</sup>، ترکیب<sup>۴</sup> و پالایش<sup>۵</sup> معرفی شدند.

در این رساله، از عملگرهای ترکیب شبکه‌های پتری در جهت اعمال رویکرد تقسیم و حل استفاده می‌کنیم. برای عملگرهای ترتیب<sup>۶</sup>، انتخاب<sup>۷</sup> و برگ‌برگ‌سازی<sup>۸</sup> با یک رویکرد نوین اثبات می‌کنیم که کنترل نظارتی بصورت پیمان‌های قابل محاسبه و اعمال است. این رویکرد می‌تواند برای اثبات امکان انجام کنترل سورپروایزری پیمان‌های و حفظ زنده بودن برای عملگرهای جدیدی که تعریف می‌شوند مورد استفاده قرار گیرد. از مزایای استفاده از مدل‌های ترکیب‌پذیر می‌توان به موارد ذیل اشاره نمود [۷]:

- فهم بهتر یک سیستم بزرگ-مقیاس و تحلیل آسانتر آن.
- قابلیت استفاده مجدد زیر سیستم‌ها در زیر سیستم‌های تقریباً مشابه.

---

<sup>1</sup> Divide and Conquer

<sup>2</sup> Composition

<sup>3</sup> Merge Operators

<sup>4</sup> Composition Operators

<sup>5</sup> Refinement Operators

<sup>6</sup> Enable Operator

<sup>7</sup> Choice Operator

<sup>8</sup> Interleave Operator

- سهولت نگهداری سیستم بدلیل جایگزینی زیرسیستم درگیر تغییرات با زیرسیستم جدید.

- تمرکز بر روی یک زیرسیستم خاص در هنگام طراحی آن، به جای سردرگمی در زیرسیستم‌های دیگر

از جمله دیگر عملگرهای رایج متصور در سیستم‌های تخصیص منابع، عملگر ادغام مکان‌ها<sup>۱</sup> است. روش‌های پیمانه‌ای در شرایطی که عملگر ادغام حافظ ویژگی زنده بودن باشد، قابل استفاده است. از مهمترین این شرایط، ایجاد نشدن سایفون<sup>۲</sup> جدید است. البته، ایجاد سایفون جدید در یک شبکه پتری، پس از اعمال عملگر ادغام مکان‌ها بسیار محتمل است. در این رساله، اثبات می‌نماییم در شرایطی که ادغام مکان‌ها حافظ خوش رفتاری نباشد نیز می‌توان با استفاده از اطلاعات این عملگر تعداد سایفون‌های مورد بررسی در کنترل نظارتی را کاهش داد. در نتیجه، بار محاسباتی کنترل نظارتی برای سیستم‌های تخصیص منابع صنعتی کم خواهد شد.

همچنین در این رساله به ارائه و اثبات قضایایی در مورد حفظ خوش رفتاری سیستم، پس از اعمال عملگر ادغام گذرها<sup>۳</sup> بر روی پیمانه‌های<sup>۴</sup> تشکیل دهنده سیستم خواهیم پرداخت. این نظریه‌ها با استفاده از خواص ساختاری یک سیستم مبتنی بر شبکه‌های پتری، به تحلیل خواص رفتاری مرتبط با خوش رفتاری خواهند پرداخت.

از جمله چالش‌های دیگری که در طراحی یک سیاست کنترل نظارتی مطرح است، طراحی یک سوپروایزر بیشینه مجازیت<sup>۵</sup>، با کمینه ساختار<sup>۶</sup> و پیچیدگی محاسباتی پایین است. در تحقیقات پیشین این سه معیار مهم در یک سیاست کنترل نظارتی قابل جمع نبوده‌اند. در این رساله با استفاده از

---

<sup>1</sup> Place-Merging Operator

<sup>2</sup> Siphon

<sup>3</sup> Transition Merging Operator

<sup>4</sup> Modules

<sup>5</sup> Maximally Permissive

<sup>6</sup> Minimal Structure

مفاهیم بردار تخصیص و بردار تعمیم به طراحی یک ILP برای محاسبه کنترل نظارتی مبتنی بر شبکه‌های پتری می‌پردازیم. این رویکرد منجر به داشتن یک سوپروایزر پیشینه مجازیت و کمینه ساختار می‌شود که نسبت به رویکردهای موجود بار محاسباتی بسیار کمتری دارد. با استفاده از این رویکرد در طراحی پیمان‌های یک سوپروایزر می‌توان به کاهش بیشتری در محاسبات کنترل نظارتی دست یافت.

## ۱-۳ فرضیات مسئله

روش‌های اعمال کنترل نظارتی پیمان‌های که در این رساله مطرح خواهند شد، بر روی سیستم‌های خاصی قابل اعمال هستند. فرضیات در نظر گرفته شده در مورد این سیستم‌ها عبارتند از:

۱- سیستم مورد مطالعه، گسسته رخداد است و رخدادها در زمان‌ها و فواصل زمانی غیر قابل پیش‌بینی امکان وقوع دارند.

۲- سیستم مورد مطالعه، دارای فرآیندهایی است که بصورت هم‌روند و در محیط رقابتی برای دسترسی به منابع اشتراکی تلاش می‌کنند.

۳- رخداد‌های مهم و اثر گذار بر سیستم، از قبل شناخته شده هستند.

۴- از نظر کنترل نظارتی، تمامی رخداد‌های سیستمی قابل مشاهده<sup>۱</sup> و قابل کنترل<sup>۲</sup> هستند.

۵- سیستم مورد مطالعه، یک سیستم فاقد اختلال<sup>۳</sup> در نظر گرفته می‌شود.

۶- هر یک از منابع در هنگام تخصیص، اختصاص کاملی به یک کار خواهند داشت. بعنوان مثال منابع انسانی که در آن واحد می‌توانند مأمور چندین کار باشند را در نظر نمی‌گیریم.

---

<sup>1</sup> Observable

<sup>2</sup> Controllable

<sup>3</sup>Noise

۷- منابع همیشگی هستند و پس از آزادسازی توسط یک فرآیند، بدون از دست رفتن و یا از دست دادن بخشی از کیفیت خود می‌توانند به وظیفه دیگری اختصاص یابند.

۸- تخصیص منابع به فرآیندها بصورت انحصار متقابل<sup>۱</sup> بوده و هر منبع در آن واحد حداکثر به یک فرآیند تخصیص می‌یابد.

۹- تخصیص منابع به یک فرآیند بدون وجود حق شفعه<sup>۲</sup> صورت می‌پذیرد. یعنی تا زمانی که یک فرآیند یک منبع را آزاد نکند، فرآیند دیگری به آن منبع دسترسی نخواهد داشت.

۱۰- فرآیندی که یک منبع را در اختیار دارد می‌تواند قبل از آزادسازی آن منبع، درخواست استفاده از منابع دیگر را داشته باشد. این فرضیه و دو فرضیه قبل ویژگی‌های ساختار یک سیستم که در آن وقوع بن‌بست محتمل است را نشان می‌دهند.

۱۱- سیستم مورد مطالعه، فاقد هرگونه همگام‌سازی<sup>۳</sup> زمانی بین رخدادها است و سیستم‌هایی که زمان طول کشیدن انجام فعالیت به صورت دقیق، نمایی، یا فازی بیان می‌شوند در نظر گرفته نشده است.

۱۲- فضای حالات سیستم مورد تحلیل، می‌تواند بسیار بزرگ باشد. در نتیجه، روش‌هایی که قابلیت تجزیه<sup>۴</sup> به زیرسیستم‌ها و در نهایت، ترکیب آنها را دارند ضروری هستند.

## ۴-۱ محدودیت‌های ناشی از فرضیات محدود کننده

با توجه به فرضیات مطرح شده، محدودیت‌هایی برای استفاده از روش‌های کنترل نظارتی که در

ادامه رساله بیان خواهد شد وجود دارد. این محدودت‌ها عبارتند از:

---

<sup>1</sup> Mutual Exclusion

<sup>2</sup> No Preemption

<sup>3</sup> Synchronization

<sup>4</sup> Decomposition



- ۱- این روش‌ها منجر به تخصیص امن منابع سیستمی به فرآیندها و تضمین خوش‌رفتاری سیستم می‌شوند. اما، در این روش‌ها به بالا بردن کارایی<sup>۱</sup> و تخصیص منصفانه منابع<sup>۲</sup> پرداخته نمی‌شود.
- ۲- این روش‌ها در هنگام وقوع رخداد خطا<sup>۳</sup> در منابع، کارخانه<sup>۴</sup> و یا سوپروایزر و شکست<sup>۵</sup> هر یک از آنها کارایی ندارند.
- ۳- روش‌هایی که مطرح خواهند شد پایداری یا ناپایداری سیستم در مقابل بروز رخدادهای ناشناخته<sup>۶</sup> را تضمین نمی‌نمایند.
- ۴- در عمل، منابع سیستمی و بخش‌های کنترلی سیستم می‌توانند از بین رفته و یا کارایی قبلی خود را از دست بدهند. سیستم کنترل خودکار مورد بررسی در این رساله، مدیریت خودکار این شکست را پوشش نخواهد داد.

## ۱-۵ سیستم نهایی مورد انتظار

- در بخش‌های قبل، فرضیات و محدودیت‌های سیستم مورد بررسی را بیان نمودیم. انتظار داریم مدل نهایی ارائه شونده از سیستم‌های تخصیص منابع دارای قابلیت‌های زیر باشد:
- قابلیت بیان ویژگی‌ها و پیچیدگی‌های فرآیندهای همروند موجود در سیستم و تحلیل آنها.

---

<sup>1</sup> Performance

<sup>2</sup> Fair Resource Allocation

<sup>3</sup> Fault

<sup>4</sup> Plant

<sup>5</sup> Failure Event

<sup>6</sup> Unrecognized Events

- قابلیت ارائه نمایش گرافیکی از حالت و تغییر حالت سیستم با توجه به وقوع رخدادها و در نتیجه سهولت پیگیری وقایع سیستم و عیب یابی سیستمی.
  - قابلیت مدیریت رخدادها همزمان در یک سیستم تخصیص منابع، جهت پیشگیری از وقوع بن‌بست.
  - قابلیت ارائه ابزارهایی برای تحلیل ویژگی‌های زنده بودن، کراندار بودن و بازگشت‌پذیری.
  - قابلیت ترکیب مدل‌های زیر سیستم‌ها برای ساختن سیستم نهایی، با توجه به بزرگ بودن مقیاس سیستم‌های مدل شده.
  - مدل نهایی باید با استفاده از محاسبات کارآ به یک مدل حافظ زنده بودن تبدیل شود.
  - سیاست کنترل نظارتی مورد استفاده باید از نظر رفتاری بیشینه مجازیت بوده و از نظر ساختاری کمینه باشد.
  - قابلیت استفاده از روش‌های کنترل نظارتی پیمانه‌ای برای اعمال محدودیت‌هایی<sup>۱</sup> جهت قانونی کردن رفتار سیستم در چارچوب تعیین شده.
- همچنین انتظار می‌رود مدل نهایی محدودیت زیادی برای نوع سیستم‌های دریافتی و ویژگی‌های خاص آنها نداشته باشد و بتواند در عمل برای گروه‌های متفاوتی از سیستم‌های تخصیص منابع مورد استفاده قرار گیرد.

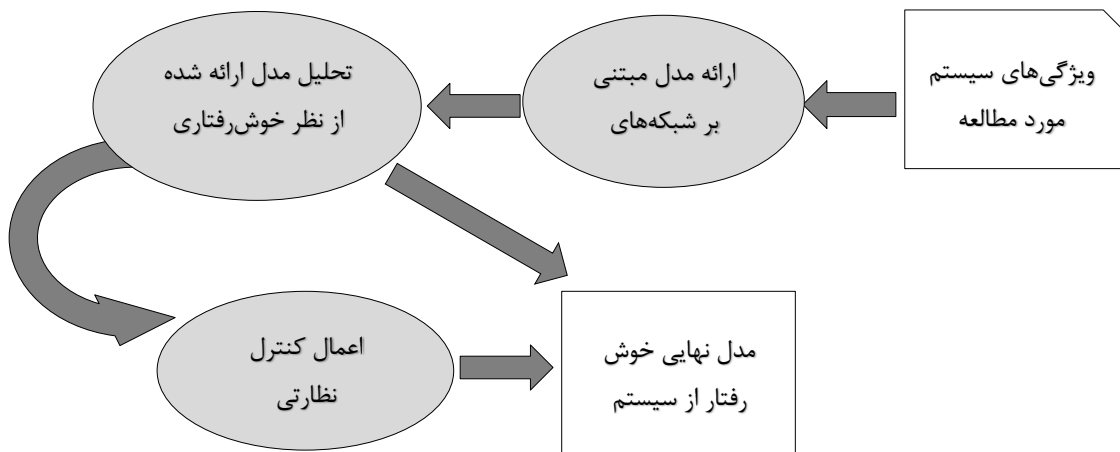
## ۱-۶ معماری پیشنهادی

وجود فرآیندهای همروند در سیستم‌های تخصیص منابع منجر به بالا رفتن پیچیدگی این سیستم‌ها شده و کار کنترل خودکار این سیستم‌ها را با مشکل مواجه می‌سازد. در این رساله، از روش‌های فرمال برای کنترل فرآیندها و تخصیص منابع به آنها استفاده می‌شود. برای این منظور، با استفاده

---

<sup>۱</sup>Constraints

از روش‌های کنترل نظارتی مبتنی بر شبکه‌های پتری و ارائه روش‌های جدید در این حوزه، به پیشگیری از وقوع پدیده نامطلوب بن‌بست خواهیم پرداخت. همچنین، این سوپروایزر را برای کنترل سیستم در جهت برآورده ساختن ویژگی‌های کرانداری و بازگشت‌پذیری نیز به کار خواهیم برد.



شکل ۱-۱ شمای کلی طراحی یک مدل خوش رفتار در سیستم‌های کوچک.

یک مدل شبکه پتری برای یک سیستم صنعتی، از مسئله انفجار فضای حالت رنج می‌برد. بنابراین محاسبه گراف دسترسی<sup>۱</sup> شبکه پتری، بعنوان گلوگاه<sup>۲</sup> محاسباتی روش‌های کنترل نظارتی متداول مطرح است. در این رساله، با استفاده از روش‌های سنتز معرفی شده در شبکه‌های پتری، به ارائه کنترل نظارتی پیمان‌های خواهیم پرداخت. معرفی روش‌های سنتز جدید نیز در ادامه مطرح خواهند شد. اگر شروط لازم برای حفظ زنده بودن یک عملگر سنتز برقرار باشد، کنترل نظارتی پیمان‌های قابل اعمال است. بدین ترتیب، ابتدا به طراحی، مدلسازی و کنترل پیمان‌های یک سیستم می‌پردازیم. این روند در شکل ۱-۱ نشان داده شده است.

همانطور که در شکل ۱-۱ مشاهده می‌شود، روند مدلسازی یک سیستم کوچک خوش رفتار (و یا یک پیمان خوش رفتار از یک سیستم) دارای سه مرحله است.

<sup>۱</sup> Reachability Graph

<sup>۲</sup> Bottleneck

در مرحله اول، با استفاده از توصیفات سیستمی، مدل مبتنی بر شبکه‌های پتری متناظر بدست می‌آید.

در مرحله دوم، به تحلیل مدل بدست آمده از نظر دارابودن ویژگی‌های زنده بودن، کرانداری و بازگشت‌پذیری پرداخته می‌شود. در صورتیکه این سه ویژگی تأمین شده باشد، مدل به دست آمده از سیستم، بعنوان یک مدل خوش‌رفتار، ارائه می‌شود.

نهایتاً در مرحله آخر، در صورتیکه یک یا برخی از ویژگی‌های مورد نظر توسط مدل تأمین نشود، به اعمال کنترل نظارتی جهت تأمین ویژگی‌های مورد نظر می‌پردازیم. در این مرحله، از سیاست‌های کنترل نظارتی<sup>۱</sup> مطرح شده در تحقیقات و یا از سیاست کنترل نظارتی که در فصل ۷ این رساله بیان خواهد شد، می‌توان استفاده نمود.

در صورتی که برهم کنش پیمان‌های سیستم، محاسبه کنترل نظارتی پیمان‌های را ممکن سازد، این روش می‌تواند جایگزین اعمال کنترل نظارتی بر کل سیستم شود. در اینصورت، پس از مدلسازی پیمان‌های تشکیل دهنده سیستم و تضمین خوش‌رفتار بودن هر یک از آنها (همانند روند مشخص شده در شکل ۱-۱) به سنتز آنها و ایجاد مدل خوش‌رفتار از سیستم کلی پرداخته می‌شود. شمای کلی این روش، در شکل ۱-۲ نشان داده شده است. این روند از سه مرحله کلی تشکیل شده است:

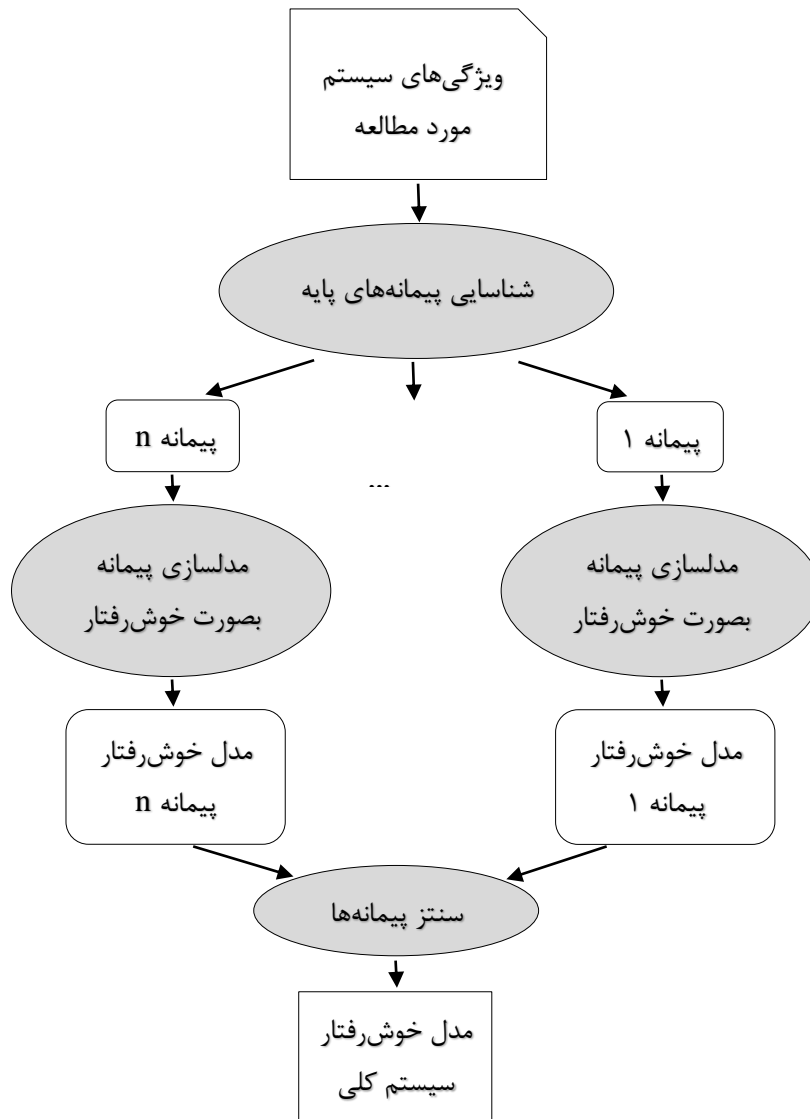
۱- تشخیص پیمان‌های سازنده اولیه: با استفاده از ضوابط سیستمی، پیمان‌های مختلف شناسایی می‌شوند. خروجی این مرحله، ضوابط و ویژگی‌های هر یک از پیمان‌ها را نشان می‌دهد.

۲- مدلسازی هر یک از پیمان‌ها بصورت خوش‌رفتار: پس از شناسایی پیمان‌های تشکیل دهنده، مدل شبکه پتری خوش‌رفتار آنها، با استفاده از روند مشخص شده در شکل ۱-۱ بدست می‌-

---

<sup>۱</sup> Supervisory Control Policies

آید. اگر یک پیمانۀ خوش رفتار نباشد، با استفاده از اعمال کنترل نظارتی، رفتار این پیمانۀ در مجموعه‌ای از نشانه‌گذاری‌های قانونی<sup>۱</sup> (ML) محدود خواهد شد.



شکل ۱-۲ روند کلی اعمال پیمانۀ‌ای کنترل نظارتی در یک سیستم گسسته رخداد

۳- سنتز پیمانۀ‌ها بر اساس برهم‌کنش مربوطه: پس از مدلسازی پیمانۀ‌های خوش رفتار به سنتز این مدل‌ها با توجه به نوع برهم‌کنش آنها می‌پردازیم. بدین ترتیب، به یک مدل

<sup>۱</sup> Legal Markings

شبکه پتری برای سیستم کلی دست می‌یابیم. در صورتی که برهم‌کنش مربوطه قابل اعمال بصورت توزیع شده باشد، مدل کلی نیز خوش‌رفتار خواهد بود.

در صورتیکه برهم‌کنش پیمان‌های سازنده یک سیستم، شرایط حفظ خوش‌رفتاری را تأمین نکند، گاهی می‌توان با استفاده از اطلاعات مربوط به برهم‌کنش، محاسبات کنترل نظارتی را کاهش داد. نمونه‌ای از این برهم‌کنش‌ها، برهم‌کنش اشتراک منابع سیستمی است. اگر در مدل شبکه پتری، منابع با استفاده از مکان<sup>۱</sup> نشان داده شوند، ادغام مکان‌ها در شبکه‌های پتری می‌تواند برهم‌کنش اشتراک منابع را شبیه‌سازی نماید. در فصل ۵ خواهیم دید که این عملگر چگونه می‌تواند موجب به کاهش سایفون‌های مورد بررسی سیستمی و در نتیجه کاهش محاسبات کنترل نظارتی شود.

## ۱-۷ نوآوری

با توجه به مقیاس بزرگ سیستم‌های تخصیص منابع صنعتی، یکی از مهمترین مسائل مطرح در کنترل نظارتی این سیستم‌ها کاهش هزینه محاسباتی یک سوپروایزر اعمال کننده زنده بودن است. در این رساله، برای نیل به این هدف روش‌هایی پیشنهاد خواهد شد.

در این رساله، قضایایی در مورد امکان انجام محاسبات کنترل نظارتی به صورت پیمان‌های و تحت برخی از عملگرهای ترکیب مطرح و اثبات شده است. این عملگرها شامل عملگرهایی که امکان اجرای همزمان<sup>۲</sup> پیمان‌های سیستمی را نمی‌دهند (همانند عملگرهای گذرگاه فرعی<sup>۳</sup>، عمل-پیشوند<sup>۴</sup>، انتخاب و ترتیب) و عملگر برگ‌برگ‌سازی می‌باشند. نتایج استفاده از این طراحی پیمان‌های، نشان

---

<sup>1</sup> Place

<sup>2</sup> Simultaneous Execution

<sup>3</sup> Bypass Operator

<sup>4</sup> Action-prefix Operator

دهنده کاهش پیچیدگی محاسباتی این رویکرد نسبت به محاسبه کنترل نظارتی در سیستم کلی است.

از طرفی، روش‌هایی که در تحقیقات پیشین در زمینه کنترل نظارتی توزیع شده<sup>۱</sup> اعلام شده است، نیاز به محاسبه کامل گراف دسترسی و محاسبات کنترل نظارتی متمرکز<sup>۲</sup> قبل از پیاده‌سازی کنترل نظارتی توزیع شده دارند. استفاده از این روش‌ها منجر به کاهش هزینه ارتباطی کنترلی میان سوپروایزر و کارخانه، در سیستم‌هایی با بسط جغرافیایی گسترده و افزایش ایمنی مخابره اطلاعات میان سوپروایزر و کارخانه خواهد شد. البته، نیاز به محاسبه کامل گراف دسترسی می‌تواند منجر به مسئله نامطلوب انفجار فضای حالت گردد. از آنجا که در تحقیقات پیشین استفاده از محاسبات توزیع شده گراف دسترسی در طراحی کنترل نظارتی پیمان‌های مطرح نشده است، نوآوری در این بخش مطرح است.

از جمله نوآوری‌های دیگر مطرح در رساله، استفاده از کنترل نظارتی پیمان‌های مبتنی بر ادغام مکان‌ها است. این سوپروایزر در شرایطی که شروط لازم برای حفظ زنده بودن توسط عملگر ادغام مکان‌ها تأمین نشود نیز می‌تواند منجر به کاهش کاهش بار محاسباتی سیاست‌های سوپروایزری کنترل در کلاس خاصی از شبکه‌های پتری شود.

همچنین، در ادامه روند معرفی سیاست کنترل نظارتی پیمان‌های، قضایایی در مورد شروط حفظ ویژگی زنده بودن با استفاده از یک عملگر ادغام گذرها معرفی و اثبات خواهد شد. برقراری این شروط در پیمان‌های پایه‌ای، زنده بودن سیستم کلی پس از اعمال عملگر ادغام گذرها را تضمین می‌کنند و

---

<sup>1</sup> Distributed Supervisory Control

<sup>2</sup> Centralized Supervisory Control

ما را از تحلیل پسین<sup>۱</sup> بی‌نیاز می‌نماید. در تحقیقات پیشین، شرایط محدودتری برای حافظ زنده بودن عملگر ادغام گذرها بیان شده بود.

نوآوری دیگری که در زمینه محاسبه کنترل نظارتی مطرح خواهد شد، استفاده از مفاهیم بردار تخصیص<sup>۲</sup> و حالت مافوق<sup>۳</sup> یک مجموعه نشانه‌گذاری<sup>۴</sup> است. معرفی یک معادله برنامه‌نویسی خطی عددی<sup>۵</sup> (ILP) برای محاسبه حالات مافوق در ضمن محاسبه کنترل نظارتی منجر به کاهش بار محاسباتی یک ILP خواهد شد.

## ۸-۱ ساختار رساله

رساله پیش رو در هشت فصل تنظیم شده است. ساختار این رساله در فصل‌های بعد بدین ترتیب است:

در فصل دوم، به ارائه مباحث نظری مرتبط با موضوع خواهیم پرداخت. ابتدا به معرفی سیستم‌های گسسته رخداد و روش‌های مدل‌سازی فرمال در این سیستم‌ها می‌پردازیم. سپس، شبکه‌های پتری بعنوان یکی از مهمترین ابزارهای فرمال مورد استفاده در مطالعه سیستم‌های گسسته رخداد به اجمال معرفی خواهد شد. در این قسمت، ویژگی‌های شبکه‌های پتری، عملگرهای سنتز معرفی شده در آنها و کلاس‌های متفاوت آنها مطرح خواهند شد. این فصل با معرفی نظریه کنترل نظارتی به اتمام می‌رسد.

فصل سوم، به مرور کارهای انجام شده در زمینه کنترل نظارتی مبتنی بر شبکه‌های پتری می‌پردازد. در این قسمت روش‌های پیشنهاد شده در زمینه کنترل نظارتی متمرکز و توزیع شده بیان خواهند شد.

---

<sup>1</sup> Posterior Analysis

<sup>2</sup> Specification Vector

<sup>3</sup> Generalization Vector

<sup>4</sup> Marking

<sup>5</sup> Integer Linear Programming



در فصل چهارم به نمونه‌هایی از کاربردهای شبکه‌های پتری در مدلسازی سیستم‌های گسسته رخداده اشاره خواهیم نمود.

فصل پنجم نیز به تشریح کارهای انجام شده در میسر انجام رساله در زمینه طراحی پیمانهای کنترل نظارتی خواهیم پرداخت. عملگرهای ترتیب، انتخاب و برگ‌برگ‌سازی برای طراحی پیمانهای کنترل نظارتی در این فصل استفاده خواهند شد. همچنین، اثبات خواهد شد که با استفاده از اطلاعات عملگر ادغام مکان‌ها می‌توان محاسبات کنترل نظارتی را کاهش داد.

فصل ششم به معرفی قضایای حفظ زنده بودن در مورد عملگر ادغام گذرها خواهد پرداخت. نتایج تجربی بیان شده در این فصل نشان دهنده آن است که این قضیه‌ها قادر به اعتبارسنجی برخی از سیستم‌ها از نظر حفظ زنده بودن پس از ادغام گذرها است که بر طبق قضایایی که در تحقیقات پیشین برای این عملگر انجام شده است قابل بررسی نبودند.

فصل هفتم به ارائه یک سیاست کنترل نظارتی، با استفاده از این مفهوم حالت مافوق خواهد پرداخت. نتایج تجربی نشان می‌دهد از نظر محاسباتی محاسبه حالات مافوق در ضمن معادلات ILP کنترل نظارتی زمان پردازش را کاهش می‌دهد. همچنین کنترل نظارتی بدست آمده از این روش، از نظر ساختاری کمینه و از نظر رفتاری دارای بیشینه مجازیت است.

در فصل هشتم نیز به جمع‌بندی و نتیجه‌گیری کلی از رساله و پیشنهادهایی برای کارهای آتی خواهیم پرداخت.

## ۱-۹ جمع بندی

در این فصل، ابتدا به معرفی و بیان اهمیت سیستم‌های تخصیص منابع پرداختیم. پس از ارائه تعریف دقیق از این سیستم‌ها و طرح مسئله، چالش‌های این حوزه و راهکاری برای حل هر یک از این

چالش‌ها مطرح گردید. سپس، فرضیات مسئله، محدودیت‌های ناشی از فرضیات محدود کننده و خروجی مورد انتظار این تحقیق مطرح گشتند. در ادامه در این فصل، به بیان معماری پیشنهادی در کنترل نظارتی پیمان‌های سیستم‌های تخصیص منابع پرداختیم. این فصل با بیان نوآوری‌هایی که در مسیر انجام رساله داشته‌ایم و معرفی ساختار رساله در فصل‌های بعد به پایان رسید.

## فصل ۲: چارچوب نظری مسئله

## ۲-۱ مقدمه

هدف مطالعه نظریه کنترل سنتی، سیستم‌هایی با متغیرهای پیوسته یا گسسته زمان است که تغییر حالات آن‌ها با استفاده از معادلات دیفرانسیل یا دیفرنس بیان می‌شوند. امروزه با گسترش حوزه نظریه کنترل به حوزه‌هایی نظیر تولید، رباتیک، کامپیوتر و شبکه‌های ارتباطی، نیاز رو به افزایش برای مدل‌های کردن سیستم‌هایی با فضای حالت منطقی<sup>۱</sup> و نمادین<sup>۲</sup> و نه مقادیر عددی [۸] وجود دارد. این سیستم‌ها تغییر حالات با بروز ناگهانی و در فواصل احتمالا ناشناخته و نامنظم رویدادهای فیزیکی، به وقوع می‌پیوندد.

سیستم‌های تخصیص منابع، خانواده مهمی از سیستم‌های گسسته رخداد را تشکیل می‌دهند. این سیستم‌ها متشکل از چندین فرآیند همروند هستند که برای بدست آوردن منابع محدود سیستمی به رقابت می‌پردازند. به همین دلیل، این سیستم‌ها بسیار حساس به خطا هستند. از طرفی، تعداد حالات سیستمی، بطور نمایی نسبت به اجزای سیستمی (شامل روندهای هر فرآیند و تعداد منابع) افزایش خواهد یافت. در چنین شرایطی اعتبارسنجی این سیستم‌های پیچیده با استفاده از تکنیک‌هایی مانند شبیه‌سازی تصادفی<sup>۳</sup> و آزمون‌های هدفمند<sup>۴</sup> معمولا امکان‌پذیر نیست. در این مواقع از روش‌های اعتبارسنجی فرمال<sup>۵</sup> استفاده می‌شود که روش‌هایی مبتنی بر ریاضیات برای اثبات درستی یا نادرستی یک ویژگی در سیستم هستند. از مهمترین ابزارهایی که برای تحلیل این سیستم‌ها وجود دارد می‌توان به شبکه‌های پتری اشاره کرد. نقش سوپروایزر بعنوان یک عامل کنترلی<sup>۶</sup> که سیستم را از قرار گرفتن در حالاتی که بعنوان حالات نامطلوب گزارش شده‌اند حفظ می‌کند نیز اساسی است.

---

<sup>1</sup> Logical States

<sup>2</sup> Symbolic States

<sup>3</sup> Random Simulation

<sup>4</sup> Directed Tests

<sup>5</sup> Formal verifications

<sup>6</sup> Control Agent

در این فصل به معرفی اجمالی سیستم‌های گسسته رخداد، روش‌های مدلسازی، پایه‌های نظری شبکه‌های پتری، روش‌های تحلیل آنها و همچنین رویکردهای مختلف استفاده از کنترل نظارتی در این فرمالیزم خواهیم پرداخت.

## ۲-۲ معرفی سیستم‌های گسسته رخداد

سیستم‌های گسسته رخداد، شاخه جدیدی از نظریه کنترل را نمایش می‌دهد. این سیستم‌ها در حوزه‌های مختلفی از جمله کنترل خط تولید [۹]، رباتیک [۱۰] و تحلیل پروتکل‌های ارتباطی [۱۱] ورود داشته و پیشرفت‌های بسیاری به همراه داشته‌اند. این بخش به معرفی اجمالی این سیستم‌ها می‌پردازد.

تعریف ۲-۱- سیستم‌های گسسته رخداد: سیستم‌هایی که مطابق با بروز ناگهانی و در فواصل احتمالاً ناشناخته و نامنظم، از رویدادهای فیزیکی عمل می‌کنند و در واقع تغییر حالات آنها به وقوع یک رخداد آسنکرون<sup>۱</sup> در زمان وابسته است را سیستم‌های گسسته رخداد [۱۲] و مدل مربوط به آنها را مدل گسسته رخداد<sup>۲</sup> (DEM) می‌گویند.

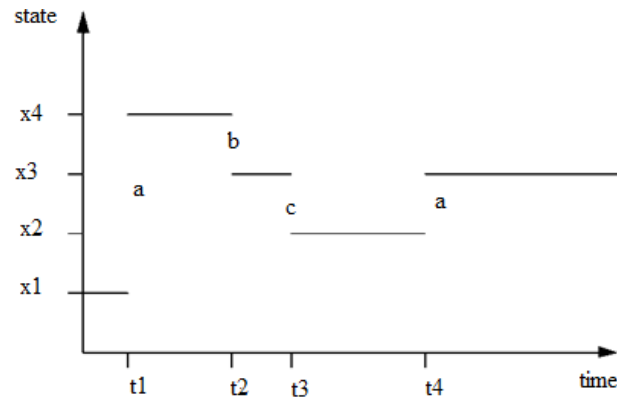
به تغییر حالت در این سیستم، انتقال وضعیت<sup>۳</sup> گفته می‌شود که معمولاً با برچسب‌هایی که به نوعی مشخص‌کننده و متناظر با رویدادی واقعی از یک پدیده فیزیکی هستند مشخص می‌شوند. بعنوان مثال، در یک سیستم تولیدی می‌توان از تخصیص یا آزاد شدن یک منبع توسط یک فرآیند [۹]، و در یک سیستم کنترل ترافیک از ورود یا خروج یک وسیله نقلیه به یک بخش [۱۳] بعنوان برچسب‌هایی برای تغییر حالت سیستم استفاده نمود. نمودار انتقال وضعیت یک DES می‌تواند مشابه شکل ۲-۱ باشد. در این نمودار با وقوع رخداد  $a$  در زمان  $t_1$ ، حالت سیستم از  $x_1$  به  $x_4$  تغییر کرده

<sup>1</sup> Asynchronous

<sup>2</sup> Discrete Event Model

<sup>3</sup> State Transition

است. به همین ترتیب پس از گذشت زمان  $|t_2 - t_1|$  با وقوع رخداد  $b$ ، سیستم از حالت  $x_4$  به حالت  $x_3$  انتقال وضعیت می‌دهد.



شکل ۱-۲ مثالی از انتقال وضعیت در یک سیستم گسسته رخداد.

مدل‌های گسسته رخداد، بر اساس حوزه‌های مختلفی که سیستم گسسته رخداد در آنها تعریف شده است و بر اساس جنبه‌های مختلف رفتاری مورد مطالعه، به دو دسته کلی قابل تقسیم هستند [۱۲]:

۱- مدل‌های منطقی<sup>۱</sup>: فرض ساده‌ساز این مدل‌ها، چشم‌پوشی از زمان وقوع رخداد و در نظر گرفتن ترتیب وقوع آنها به تنهایی است. این فرض وقتی استفاده می‌شود که از مدل برای مطالعه ویژگی‌های پویای رخدادهای سیستمی استفاده می‌شود که مستقل از زمان وقوع رخدادها باشند. این مدل‌ها در مطالعه ویژگی‌های کیفی DES در کاربردهای مختلف مانند: کنترل همروندی در سیستم عامل، اعتبارسنجی پروتکل‌های ارتباطی، محاسبات توزیع شده تحمل‌پذیر خطا، تحلیل منطقی مدارهای دیجیتال و کنترل نظارتی یک سیستم استفاده شده‌اند.

<sup>۱</sup>Logical DEM

۲- مدل‌های زماندار<sup>۱</sup> یا مدل‌های کارآیی<sup>۲</sup>: هنگامی که زمان وقوع رخداد، در مطالعه ویژگی‌های سیستمی مهم باشد، از این نوع مدل‌ها استفاده می‌شود. این مدل‌ها بر اساس اینکه زمان به عنوان یک پیش‌فرض در نظر گرفته شده است و یا بعنوان یک فرض آماری مناسب مطرح است به دو دسته جزئی‌تر تقسیم می‌شوند:

الف) مدل‌های زماندار غیرآماری<sup>۳</sup> مانند: جبر بیشینه<sup>۴</sup> و شبکه‌های پتری زماندار<sup>۵</sup>.

ب) مدل‌های زماندار آماری<sup>۶</sup> مانند: زنجیره مارکف، شبکه‌های صف، شبه زنجیره مارکف تعمیم یافته<sup>۷</sup>.

در سیستم‌های RAS، تخصیص و آزادسازی منابع می‌توانند بعنوان رخداد‌های ناوابسته به زمان و با فواصل زمانی ناشناخته در نظر گرفته شوند. بنابراین می‌توان از رویکرد DES برای تحلیل وضعیت این سیستم‌ها و بررسی برخی از ویژگی‌های آنها استفاده نمود.

## ۲-۳ روش‌های مدل‌سازی سیستم‌های گسسته رخداد

یک مدل از یک سیستم، جزء کوچک بازسازی شده‌ای از یک پدیده واقعی است که با ساده-سازی سیستم در جهت حفظ ویژگی‌های مورد علاقه، به بررسی و تحلیل آن می‌پردازد. روش‌های مدل‌سازی یک سیستم به دو روش عمده تقسیم می‌شوند: روش‌هایی که قبل از پیاده‌سازی سیستم به مدل‌سازی آن می‌پردازند و روش‌هایی که یک سیستم موجود را مدل می‌کنند. هدف روش‌های دسته

---

<sup>1</sup>Timed DEM

<sup>2</sup> Performance DEM

<sup>3</sup>Non Stochastic

<sup>4</sup> Max Algebra

<sup>5</sup> Timed Petri nets

<sup>6</sup>Stochastic

<sup>7</sup>Generalized Semi Markov Chain

اول، عیب‌یابی سیستم پیش از پیاده سازی و کاهش هزینه‌های تغییر سیستم ایجاد شده است. هدف روش‌های دسته دوم، کشف خواص یک سیستم و کنترل سیستم در جهت برقرار شدن ویژگی‌های مطلوب است.

پس از طراحی یک مدل با توجه به هر یک از این اهداف، نیاز به تحلیل مدل داریم. تحلیل مدل به دو روش قابل انجام است. تحلیل پسین و تحلیل پیشین<sup>۱</sup> یا مبتنی بر سنتز. در روش اول، پس از طراحی یک سیستم بطور کامل، به بررسی ویژگی‌های مورد توجه پرداخته می‌شود. این روش در سیستم‌های بزرگ مقیاس می‌تواند بسیار هزینه‌بر و حتی ناممکن باشد. روش‌های تحلیل مبتنی بر سنتز، پس از مدلسازی اجزای یک سیستم، به کنترل برقراری ویژگی‌های مهم در این اجزا می‌پردازد. بدین ترتیب، پس از ساختن سیستم کلی از روی مدل‌های زیرسیستم‌ها، مدل بدست آمده دارای ویژگی‌های مورد انتظار بوده و نیاز به تحلیل بیشتر ندارد. روش‌های تحلیل پیشین می‌تواند کاهش قابل توجهی در هزینه محاسباتی بررسی یک ویژگی ایجاد نماید.

از منظری دیگر، ابزارها و روش‌های مدلسازی را از نظر قدرت مدل برای تحلیل سیستم می‌توان به ابزارها و روش‌های مدلسازی غیر فرمال (مانند زبان مدلسازی یکپارچه<sup>۲</sup> (UML))، نیمه فرمال<sup>۳</sup> (مانند شبیه‌سازی) و ابزارهای فرمال (مانند شبکه‌های پتری) تقسیم نمود. در ادامه این بخش مرور اجمالی بر این روش‌ها و به کارگیری آنها در حوزه مدلسازی سیستم‌های گسسته رخداد خواهیم داشت.

## ۲-۳-۱ مدلسازی غیر فرمال

---

<sup>۱</sup> Prior Analysis

<sup>۲</sup> Unified Modelling Language

<sup>۳</sup> Semi-Formal Tools



روش‌های مدل‌سازی غیر فرمال، تنها قابلیت مدل‌سازی و نه تحلیل گرافیکی سیستم مورد مطالعه را دارا هستند. در این روش، عمدتاً یک شمای گرافیکی از جنبه‌ای از سیستم مورد مطالعه بدست خواهد آمد. رویکردهای مهندسی نرم‌افزار و روش‌های طراحی شیء‌گرا<sup>۱</sup> از جمله UML از جمله این روش‌ها هستند [۱۴]. بعنوان نمونه، کای و همکاران [۱۵] از روش‌های مدل‌سازی غیرفرمال در زمینه مدل‌سازی و نه تحلیل یک شبکه ریلی که نمونه‌ای از سیستم‌های تخصیص منابع است استفاده نموده‌اند.

## ۲-۳-۲ مدل‌سازی نیمه فرمال

روش‌های مدل‌سازی نیمه فرمال، روش‌هایی هستند که قابلیت مدل‌سازی یک سیستم و تحلیل غیرمستقیم آن را دارا هستند. مهمترین کلاس گنجانده شده در این دسته از روش‌ها، روش‌های مبتنی بر شبیه‌سازی هستند. این روش‌ها به خودی خود، فاقد پایه و اساس ریاضی برای تحلیل سیستم گسسته رخداد هستند. البته با استفاده از تکنیک‌های تحلیل آماری و تشکیل ثوابت حلقه<sup>۲</sup>، می‌توان به تحلیل سیستم مدل شده با استفاده از روش‌های نیمه فرمال پرداخت. نرم افزارهای بسیاری در زمینه مدل‌سازی و تحلیل آماری سیستم‌های گسسته رخداد نوشته شده‌اند. از جمله معروفترین این نرم‌افزارها می‌توان به نرم افزارهای NS3 [۱۶] و OMNET++ [۱۷] اشاره کرد.

## ۳-۳-۲ مدل‌سازی فرمال

این روش‌ها، بطور مستقیم به اعتبارسنجی سیستم مدل شده (اثبات مستقیم ریاضی ویژگی-های سیستم مدل شده)، علاوه بر تصدیق<sup>۳</sup> آن‌ها (تطبیق یافته‌های مدل با عملکرد واقعی سیستم مدل

---

<sup>۱</sup> Object Oriented

<sup>۲</sup> Loop-Invariants

<sup>۳</sup> Validation

شده) می‌پردازند. این روش‌ها مبتنی بر زبان‌های ضوابط فرمال<sup>۱</sup> هستند که به توصیف دقیق سیستم‌های گسسته رخداد می‌پردازند. این زبان‌ها می‌توانند مبتنی بر منطق [۱۸]، مبتنی بر روش‌های جبری [۱۹] و یا مبتنی بر نمایش‌های گرافیکی باشند [۳، ۲۰، ۲۱]. در ادامه، به نمونه‌هایی از کارهای انجام شده با استفاده از روش‌های مدلسازی فرمال برای مدلسازی و تحلیل سیستم‌های DES خواهیم پرداخت.

## ۱-۳-۳-۲ روش‌های مبتنی بر منطق

کارهای بسیاری با استفاده از منطق زمانی و شاخه‌های مختلف آن در زمینه تحلیل سیستم‌های گسسته رخداد و معرفی سوپروایزر در آنها انجام شده است. از جمله در [۲۲] برای کنترل واکنش‌های ربات‌ها از منطق زمانی استفاده شده است. [۱۰] علاوه بر ساختن یک مدل کارخانه با استفاده از منطق زمانی، به سنتز آن با استفاده از سوپروایزر و رویکرد زبانی پرداخته است. چن و کومار [۱۸] از سیستم‌های مبتنی بر منطق زمانی برای تشخیص خطا در سیستم‌های گسسته رخداد استفاده کرده‌اند.

## ۲-۳-۳-۲ روش‌های مبتنی بر مدل‌های جبری

روش‌های جبری، از جمله جبر فرآیندها<sup>۲</sup> از فرمالیزم‌های معروف مدلسازی و تحلیل در زمینه تحلیل سیستم‌های گسسته رخداد و سیستم‌های تخصیص منابع هستند. بعنوان مثال در [۲۳] نرم-افزاری را برای مدلسازی و تحلیل سیستم‌های گسسته رخداد با استفاده از فرمالیزم جبر فرآیندها ارائه شده است. در [۲۴] به معرفی فرمالیزم جبر ماکس پلاس<sup>۳</sup> پرداخته است. سپس این فرمالیزم را برای مدلسازی و تحلیل سیستم‌های گسسته رخداد بکار برده است. در این مرجع با استفاده از یک مدل

<sup>۱</sup> Formal Specification Language

<sup>۲</sup> Process Algebra

<sup>۳</sup>Max-Plus Algebra

خطی از فرمالیزم گفته شده، با نام جبر ماکس پلاس خطی<sup>۱</sup> به مدلسازی و تحلیل یک سیستم با ویژگی همگام‌سازی بین اعمال و البته بدون داشتن قابلیت همروندی در سیستم پرداخته است.

### ۲-۳-۳-۳ روش‌های مبتنی بر اتوماتا

روش‌های مبتنی بر نمایش گرافیکی، مشاهده حالات سیستم و نحوه تغییر حالت آن را ممکن می‌سازند. بهره‌گیری این روش‌ها از یک پایه و اساس ریاضیاتی، زمینه را برای تحلیل سیستم‌های مدل شده با این روش‌ها آماده می‌سازد. اتوماتا از جمله معروفترین فرمالیزم‌های مورد استفاده در سیستم‌های گسسته رخداد هستند. رمج و ونهام [۲۵] برای اولین بار با استفاده از این فرمالیزم به بیان نظریه کنترل نظارتی برای اعتبارسنجی سیستم‌های DES پرداختند. میرعمادی [۳] از اتوماتای تعمیم‌یافته برای مدلسازی و تحلیل سیستم گسسته رخداد استفاده نموده است. [۳] علت استفاده از اتوماتا نسبت به بقیه فرمالیزم‌ها را سادگی بکارگیری، امکان تحلیل، قابلیت استفاده از عملگرهای ترکیب و مطابقت آن با نظریه کنترل نظارتی دانست. البته، اتوماتا ساختار پیچیده‌ای از سیستم را به نمایش می‌گذارند.

### ۲-۳-۳-۴ روش‌های مبتنی بر دایگراف<sup>۲</sup>

دایگراف از دیگر فرمالیزم‌های گرافیکی برای مدلسازی و تحلیل سیستم‌های گسسته رخداد است. یال‌های این گراف، نمایش دهنده عملیات‌های سیستمی و رؤس گراف نشان‌دهنده منابع سیستمی هستند. از اولین تحقیقات انجام شده که از نظریه گراف در زمینه تشخیص بن‌بست در سیستم‌های تولیدی استفاده کرده است، مرجع [۲۶] می‌باشد. در [۲۷] نیز به مرور کارهای انجام شده با استفاده از دایگراف برای تحلیل بن‌بست سیستم‌های تولیدی پرداخته است.

<sup>۱</sup> Linear Max Plus Algebra

<sup>۲</sup> Digraph

## ۲-۳-۵ روش‌های مبتنی بر شبکه‌های پتری

شبکه‌های پتری یک فرمالیزم مدلسازی قدرتمند سطح بالا است که قابلیت نمایش گرافیکی حالات شبکه و قابلیت استفاده از روش‌های تحلیلی ریاضیاتی را در کنار هم قرار داده است. در [۲۸] لیستی از مزایای استفاده از شبکه‌های پتری در مسئله تخصیص امن منابع بیان شده است:

- ۱- اساس ریاضیاتی شبکه‌های پتری و تکنیک‌های مختلفی برای تحلیل آن. مانند: تحلیل فضای دسترسی، تحلیل Invariant ها.
- ۲- حفظ ویژگی‌های مطلوب در هنگام انتقال شبکه به شبکه دیگر مثلاً در هنگام کاهش شبکه.
- ۳- قابلیت استفاده از نظریه‌های ساختاری، نظریه زبان فرمال در شبکه‌های پتری.
- ۴- قابلیت تشخیص برابر بودن دو شبکه پتری.

از دیگر مزایای استفاده از شبکه‌های پتری می‌توان به موارد ذیل اشاره نمود:

- قابلیت نمایش همروندی. این قابلیت در فرمالیزم‌هایی مانند اتوماتا که در هر لحظه تنها یک مکان آن دارای نشانه<sup>۱</sup> است وجود ندارد. اما شبکه‌های پتری که در آنها وجود نشانه‌ها در مکان‌ها، بیانگر حالات محلی سیستم است می‌توانند به مدلسازی همروندی در سیستم پردازند.
- بهره‌مندی از نمایش گرافیکی سیستم. قابلیت نمایش گرافیکی منجر به ارائه یک بازنمایی شهودی از سیستم و نحوه عملکرد آن و در نتیجه سهولت عیب‌یابی آن خواهد شد.
- قابلیت نگاشته شدن شبکه پتری به فرمالیزم‌های دیگر.
- دارا بودن ساختار سلسله مراتبی.
- قابلیت شبکه‌های پتری در کامپایل شدن به کد قابل اجرا در سخت افزار.

---

<sup>۱</sup> Token

- قابلیت ایجاد سوپروایزر توزیع شده، که امنیت و صرفه اقتصادی سیستم را نیز به همراه دارد، یکی از ویژگی‌های استثنایی شبکه‌های پتری است که آن را از اتوماتا جدا می‌سازد. زیرا اتوماتای تعمیم یافته می‌توانند مدل‌های فشرده‌تری از سیستم ارائه دهند، اما قابلیت ایجاد سوپروایزر توزیع شده را ندارند.

## ۲-۳-۴ رویکرد ترکیبی

در برخی از تحقیقات با استفاده از یک رویکرد ترکیبی و با بهره‌گیری از مزایای دو یا چند از فرمالیزم به ساخت مدلی از سیستم پرداخته‌اند. [۲۹] مزیت ترکیب‌پذیری مدل‌های مبتنی بر جبر فرآیندها و مزیت نمایش گرافیکی شبکه پتری را تجمیع نموده است. در این رویکرد، ابتدا به طراحی پیمانها با استفاده از شبکه‌های پتری امن<sup>۱</sup> پرداخته و با تبدیل آن‌ها به جبر فرآیندها، به سنتز آنها و ارائه مدلی از سیستم کلی می‌پردازد. لئارتسون و همکاران [۳۰] یک مدل یکپارچه، با نام مدل گذرهای گزاره‌ای<sup>۲</sup> (PTM)، را پیشنهاد دادند. این مدل بعنوان یک مدل کلی برای بسیاری از فرمالیزم‌های گرافیکی مانند اتوماتا، اتوماتای تعمیم یافته و شبکه‌های پتری در نظر گرفته شده است.

## ۲-۴ معرفی شبکه‌های پتری

شبکه‌های پتری یک زبان توصیفی سطح بالای فرمال برای مدلسازی سیستم‌های پیچیده است که در همه زمینه‌هایی که پردازش فرآیندهای همروند در آنها انجام می‌شود کاربرد دارد [۳۱]. فرآیندکاوی [۳۲]، مهندسی نرم‌افزار [۳۳]، تحلیل پروتکل‌های ارتباطی [۱۱]، تحلیل کارایی و گلوگاه‌های یک سیستم [۳۴] و کنترل نظارتی یک سیستم تولیدی [۹] از جمله این زمینه‌ها هستند.

<sup>۱</sup> Safe Petri Nets

<sup>۲</sup> Propositional Transitions Model

بطور کلی شبکه‌های پتری ابزار مناسب برای شرح و مطالعه سیستم‌های پردازش اطلاعات است که ویژگی آنها همروندی، توازی<sup>۱</sup>، آسنکرون بودن، توزیع شده بودن<sup>۲</sup> و غیر قطعی بودن<sup>۳</sup> است [۱۲].

شبکه‌های پتری در سال ۱۹۶۲ توسط فیزیکدان کارل آدام پتری معرفی شد [۳۵] و یک نظریه ریاضیاتی خوش فرم<sup>۴</sup> را با یک نمایش گرافیکی از رفتار دینامیکی سیستم ترکیب می‌کند؛ از طرفی نمایش گرافیکی آن نیز اجازه مصورشدن تغییر حالات در سیستم را می‌دهد. این ترکیب باعث موفقیت این مدل در بسیاری از جنبه‌های سیستم‌های دینامیکی مبتنی بر رخداد شده است.

از نظر نمایشی، شبکه‌های پتری همانند یک گراف دوبخشی<sup>۵</sup> است که شامل عناصر سازنده مکان، گذر<sup>۶</sup> و کمان‌هایی<sup>۷</sup> است که مکان و گذر را برای پیاده سازی مفهومی "جریان کنترل و داده" به هم متصل می‌کند. مکان‌ها می‌توانند شامل نشانه‌ها باشند که از نظر مفهومی می‌توانند بیانگر در دسترس بودن یا نبودن تعدادی از منابع سیستمی، داده‌های پردازشی، مشتری در صف و ... و بطور کلی فراهم شدن پیش‌شرط‌های وقوع یک رخداد باشند. توزیع نشانه‌ها ( و نوع، اگر نشانه‌ها قابل تفکیک باشند) در هر یک از مکان‌ها در هر زمان، بیانگر حالت سیستم در آن زمان است. این توزیع بصورت برداری نشان داده می‌شود که به آن بردار حالت<sup>۸</sup> یا بردار نشانه‌گذاری<sup>۹</sup> ( $M$ ) گفته می‌شود. به حالت اولیه سیستم که سیستم از آن وضعیت شروع به کار می‌کند، نشانه‌گذاری اولیه<sup>۱۰</sup> ( $M_0$ ) گویند [۳۶]. بسته به کاربرد شبکه‌های پتری از قالب‌بندی‌های متفاوتی برای آنها استفاده شده است. یکی از قالب‌ها که در [۱۲] معرفی شده و در بسیاری از تحقیقات نیز مورد استفاده قرار گرفته است، قالبی

---

<sup>1</sup> Parallel

<sup>2</sup> Distributed

<sup>3</sup> Nondeterminism

<sup>4</sup> Well-formed Mathematical Theory

<sup>5</sup> Bipartite

<sup>6</sup> Transition

<sup>7</sup> Arcs

<sup>8</sup> State Vector

<sup>9</sup> Marking Vector

<sup>10</sup> Initial Marking

است که شبکه‌های پتری را بصورت یک پنج تایی تعریف می‌کند. در این رساله نیز از این قالب استفاده شده است. تعریف ۲-۲ با استفاده از این قالب به تعریف فرمال شبکه‌های پتری می‌پردازد.

تعریف ۲-۲ - شبکه‌های پتری: یک شبکه پتری بطور فرمال بصورت یک پنج تایی مرتب

$$N = (P, T, F, W, M_0) \text{ است که در آن:}$$

$P$ : مجموعه محدودی از مکان‌ها است که از نظر نمایشی با دایره نشان داده می‌شود.

$T$ : مجموعه محدودی از گذرها است که از نظر نمایشی با مستطیل نمایش داده می‌شود.

$F \subseteq (P \times T) \cup (T \times P)$  مجموعه کمان‌هاست که رابط میان گذرها و مکان‌هاست.

$W: F \rightarrow \{1, 2, 3, \dots\}$  یک تابع وزن است که به هر یک از کمان‌ها وزنی را اختصاص می‌دهد.

$M_0: P \rightarrow \{1, 2, 3, \dots\}$  نشانه‌گذاری اولیه را بیان می‌کند.

در هر شبکه مجموعه مکان‌ها متفاوت از مجموعه گذرها است و نمی‌توانند همزمان تهی باشند:

$$P \cap T = \emptyset \text{ and } P \cup T \neq \emptyset. \quad (2-1)$$

ماتریس برخورد<sup>۱</sup>  $C$  یک ماتریس  $|P| \times |T|$  است که مقادیر درایه‌های آن با استفاده از معادله (۲-۲)

(۲) مشخص می‌شود:

$$C_{ij} = W(t_j, p_i) - W(p_i, t_j) \quad (2-2)$$

می‌توان ماتریس برخورد را بصورت معادله (۲-۳) نیز مقداردهی کرد:

$$C = Out - In \quad (2-3)$$

---

<sup>۱</sup> Incidence Matrix

که  $Out$  ماتریس خروجی و  $In$  ماتریس ورودی گذرهای شبکه هستند.

گره  $x$  را گره ورودی به گره  $y$  گوئیم اگر  $W(x, y) \neq 0$  باشد. این مفهوم به صورت  $x = y^*$  نشان داده می‌شود. در این صورت، گره  $y$  را گره خروجی گره  $x$  گوئیم و با  $x^* = y$  نشان داده می‌شود. همچنین این مفهوم بصورت  $x = pre(y)$  و یا  $y = post(x)$  نشان داده می‌شود.

شبکه‌های پتری برای توضیح رفتار دینامیکی یک سیستم گسسته رخداد، تغییر حالت‌ها را در قالب قانون‌هایی بیان می‌کند که بدین ترتیب هستند:

۱- در یک نشانه‌گذاری  $M_k$  گذر  $t$  را فعال<sup>۱</sup> گوئید اگر هر یک از مکان‌های ورودی آن،  $p$ ، حداقل به تعداد وزن کمان اتصال دهنده آنها به گذر،  $W(p, t)$ ، نشانه داشته باشند. این مفهوم با استفاده از  $M_k[t]$  نشان داده می‌شود.

۲- گذر فعال قابلیت اجرا شدن<sup>۲</sup> دارد و وقوع آن بستگی به این دارد که در واقعیت در این زمان فعالیت مربوط به گذر اتفاق بیفتد یا خیر.

۳- اگر گذری اتفاق بیفتد از هر یک از مکان‌های ورودی (خروجی) آن، به تعداد وزن کمان مرتبط کننده،  $W(p, t)$ ،  $W(t, p)$  نشانه کم (افزوده) خواهد شد. در نتیجه سیستم از نشانه‌گذاری  $M_k$  به نشانه‌گذاری  $M_{k+1}$  انتقال وضعیت خواهد داد. این مفهوم با استفاده از  $M_{k+1}$  نشان داده می‌شود. معادله انتقال وضعیت در معادله (۲-۴) آمده است.

$$M_{k+1} = M_k + C.v \quad (2-4)$$

که  $v$  یک بردار غیر منفی عددی است که در آن یک عنصر 1 و بقیه 0 هستند. عنصر

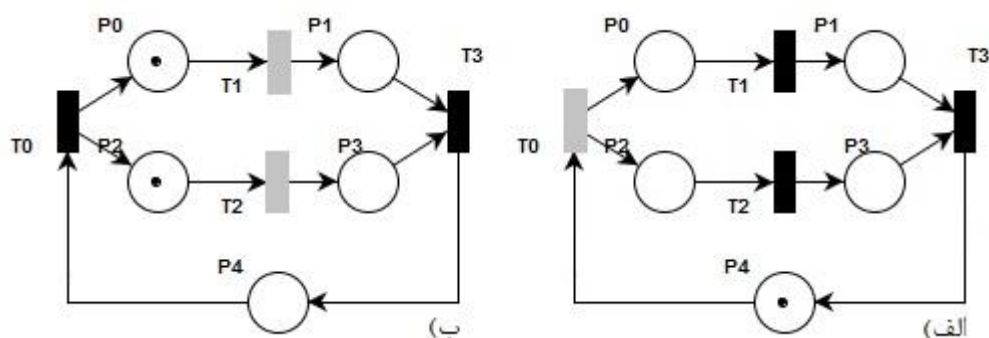
۱ مربوط به گذر فعال شبکه می‌باشد. مثالی از نحوه اتفاق افتادن یک گذر فعال در شکل ۲-

۲ آمده است:

<sup>1</sup> Enabled

<sup>2</sup> Fire





شکل ۲-۲ نحوه تغییر حالت شبکه پتری با وقوع یک گذر فعال. اجرای گذر فعال  $t_0$  در شکل الف منجر به تغییر وضعیت سیستم به شکل ب می‌شود. در این شکل گذرهای فعال با رنگ طوسی و گذرهای غیرفعال با رنگ مشکی نشان داده شده‌اند.

با داشتن نشانه‌گذاری اولیه و محاسبه تمامی نشانه‌گذاری‌های مابعد با معادله (۲-۴)، به نمایش غیرفشرده‌ای از فضای حالت شبکه پتری با نام گراف دسترسی می‌رسیم که با  $R(N, M_0)$  نشان داده می‌شود. گره‌های این گراف، نشان دهنده نشانه‌گذاری‌های مختلف و یال‌های بین دو گره دارای برچسب متناظر با گذر فعالی است که اجرای آن منجر به انتقال وضعیت از نشانه‌گذاری متناظر گره مبدأ به نشانه‌گذاری متناظر گره مقصد می‌شود.

به همین ترتیب، مفهوم گذر فعال معکوس<sup>۱</sup> در شبکه‌های پتری بیان شده است. گذر  $t$  را فعال معکوس گوئیم اگر تمامی مکان‌های خروجی آن حداقل به تعداد وزن کمان اتصال دهنده آنها به گذر،  $W(t, p)$ ، نشانه داشته باشند. با وقوع یک گذر فعال معکوس، به شکل معکوس، وضعیت سیستم، برطبق

$$M_{k+1} = M_k - C.v \quad (2-5)$$

بر اساس مفهوم فعال معکوس در گذرها، مفهوم گراف دسترسی معکوس<sup>۲</sup> قابل تعریف است. در ساخت گراف دسترسی معکوس، مابعدهای هر نشانه‌گذاری به جای اینکه از معادله (۲-۴) تعیین

<sup>۱</sup> Co-enabled

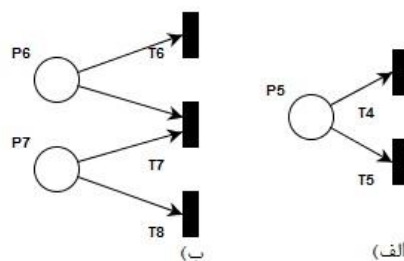
<sup>۲</sup> Co-reachability Graph

شوند، با استفاده از معادله (۲-۵) معین خواهند شد. گراف دسترسی معکوس با نماد  $R(-N, M_0)$  نمایش داده می‌شود. مفاهیم گراف دسترسی و گراف دسترسی معکوس در تحلیل شبکه‌های پتری و ایجاد یک شبکه زنده کاربرد دارند. در فصل‌های بعد به نمونه‌هایی از این کاربرد اشاره خواهیم داشت.

## ۲-۴-۱ قدرت شبکه‌های پتری در بیان ویژگی‌های مختلف سیستم‌ها

شبکه‌های پتری ابزاری قدرتمند در توصیف ویژگی‌های خاصی از سیستم‌ها هستند که گاه با استفاده از مدل‌های دیگر همه یا برخی از آنها قابل بیان نیست. این ویژگی‌ها عبارتند از: همروندی<sup>۱</sup>، تصادم<sup>۲</sup>، تعارض<sup>۳</sup>، اولویت<sup>۴</sup> و انحصار متقابل.

فعالیت‌های موازی<sup>۵</sup> یا همروندی: هنگامی که دو گذر یا فعالیت از نظر علی کاملاً مستقل از هم باشند و یکی از آنها بتواند قبل، بعد، یا هم‌زمان با دیگری اتفاق افتد، به آنها فعالیت‌های موازی گویند. گذرهای  $t_1$  و  $t_2$  در شکل ۲-۳. الف نمونه‌ای از این گذرها را نشان می‌دهد.



شکل ۲-۳ نمونه‌هایی از شبکه‌های پتری. الف) نمونه‌ای از شبکه پتری دارای تصادم. ب) نمونه‌ای از شبکه پتری دارای تعارض.

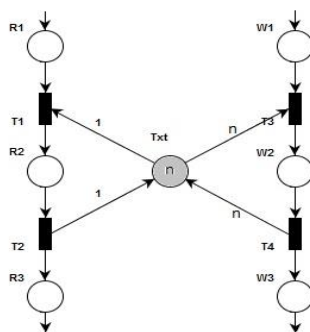
تصادم: یکی از ویژگی‌های دیگری که در سیستم‌های عدم قطعیت می‌تواند وجود داشته باشد تصادم است. این ویژگی توسط شبکه‌های پتری قابل بازنمایی است. این ویژگی در شرایطی ایجاد می‌-

<sup>1</sup> Concurrency  
<sup>2</sup> Conflict  
<sup>3</sup> Confusion  
<sup>4</sup> Priority  
<sup>5</sup> Parallel Activities

شود که دو گذر می‌توانند بطور هم‌زمان شرایط وقوع را داشته باشند اما اگر هر یک از گذرها اتفاق بیفتد گذر دیگر شانس وقوع خود در آن لحظه را از دست خواهد داد. به این ویژگی، تصمیم<sup>۱</sup> یا انتخاب<sup>۲</sup> نیز گفته می‌شود. مثالی از نمایش گرافیکی این ویژگی در شبکه‌های پتری در شکل ۳-۳. الف آمده است.

ویژگی سوم تعارض است که در مواقعی رخ می‌دهد که سیستم دارای رخدادهای موازی و رخدادهای متعارض باشد. نمونه‌ای از نمایش این ویژگی با شبکه‌های پتری در شکل ۳-۳. ب آمده است.

ویژگی چهارم مدل کردن انحصار متقابل با شبکه‌های پتری است. بعنوان مثال یک شبکه برای مسئله خواندن نوشتن یک فایل در شکل ۲-۴ آمده است. در این شکل انحصار متقابل میان دو فرآیند خواندن و فرآیند نوشتن برای دریافت منبع متنی نشان داده شده است. این چهار ویژگی در نمایش سیستم‌هایی که در آنها چند فرآیند به منابع اشتراکی دسترسی دارند مثلاً در سیستم عامل یا تحلیل گلوگاه‌های یک خط تولید استفاده می‌شود.



شکل ۲-۴ مدل شدن انحصار متقابل توسط شبکه‌های پتری. شبکه پتری مسئله خواننده-نویسنده.

<sup>۱</sup>Decision  
<sup>۲</sup>Choice

ویژگی آخر نیز مدل کردن اولویت است که توسط تعمیمی از شبکه‌های پتری که در آن به معرفی کمان‌های بازدارنده<sup>۱</sup> پرداخته شده، بدست می‌آید [۳۶]. گذر مقصد این کمان هنگامی فعال است که مکان ورودی مربوطه خالی از نشانه باشد.

## ۲-۴-۲ تعریف برخی از ویژگی‌ها در شبکه‌های پتری

تعریف ۲-۳- مکان‌های بی‌فایده<sup>۲</sup>: یک مکان  $p_u$  را بی‌فایده نامیم [۳۷] اگر و تنها اگر زیر مجموعه‌ای از مکان‌ها به نام مجموعه جانشینی،  $E$ ، و تابع وزنی  $X$  به شکل  $X: E \cup \{p_u\} \rightarrow N$  وجود داشته باشد که نشانه‌گذاری اولیه و ماتریس برخورد مکان‌های بی‌فایده با استفاده از ترکیب خطی نشانه‌گذاری‌های اولیه و ماتریس برخورد مکان‌های جانشین قابل محاسبه باشد. مکان‌های بی-فایده زیرمجموعه‌ای از مکان‌های زائد ساختاری<sup>۳</sup> هستند که در [۳۸] معرفی شده است. حذف این مکان‌ها هیچ تغییری در ویژگی‌های مرتبط با خوش رفتاری یک شبکه پتری ایجاد نمی‌کنند. تشخیص یک مکان بی‌فایده با استفاده از فرمول ۲-۶ انجام می‌شود:

$$X(p_u)M_0(p_u) = \sum_{p \in E} X(p)M_0(p).$$

$$\forall t \in T, X(p_u)C(p_u, t) = \sum_{p \in E} X(p)C(p, t). \quad (2-6)$$

تعریف ۲-۴- شبکه‌های پتری همبند<sup>۴</sup>، قویا همبند<sup>۵</sup>، معمولی<sup>۶</sup>، خالص<sup>۷</sup>:

۱- یک شبکه پتری را همبند<sup>۸</sup> (قویا همبند<sup>۹</sup>) گوییم اگر گراف متناظر آن همبند (قویا همبند) باشد.

<sup>1</sup>Inhibitor Arcs

<sup>2</sup>Useless Places

<sup>3</sup>Structurally Redundant Places

<sup>4</sup>Connected PNs

<sup>5</sup>Strongly connected PNs

<sup>6</sup>Ordinary Petri Nets

<sup>7</sup>Pure

<sup>8</sup>Connected

<sup>9</sup>Strongly Connected

۲- یک شبکه پتری  $N = (P, T, F, W)$  را معمولی گوییم اگر و تنها اگر وزن تمامی کمان‌های اتصال دهنده ۰ یا ۱ باشد. عبارتی  $W(p_i, t_j) \in \{0,1\}, W(t_j, p_i) \in \{0,1\}$  در  $\forall p_i, t_j$ . در شبکه‌های پتری معمولی  $F = W$  است. از این رو، می‌توانند بجای یکدیگر مورد استفاده قرار گیرند.

۳- یک شبکه  $N = (P, T, F, W)$  را خالص و یا بدون حلقه<sup>۱</sup> گوییم اگر و تنها اگر هیچ مکانی ورودی و خروجی یک گذر نباشد؛ یعنی:  $\forall x, y \in P \cup T, (x, y) \in F \Rightarrow (y, x) \notin F$ .

تعریف ۲-۵- شبکه‌های پتری ماشین وضعیت<sup>۲</sup>، گراف نشانه‌دار<sup>۳</sup>، بدون تصادم<sup>۴</sup>، تصادم نامتقارن<sup>۵</sup>:

۱- یک شبکه پتری  $N = (P, T, F, W)$  را ماشین وضعیت (SM) گوییم اگر و تنها اگر تمامی گذرهای آن دارای دقیقاً یک کمان ورودی و یک کمان خروجی باشند:  $\forall t \in T: |t^*| = 1$ .

۲- یک شبکه پتری  $N = (P, T, F, W)$  را گراف نشانه‌دار (MG) گوییم اگر و تنها اگر هر مکان آن دارای دقیقاً یک کمان ورودی و یک کمان خروجی باشد:  $\forall p \in P: |p^*| = 1$ .

۳- یک شبکه پتری  $N = (P, T, F, W)$  را بدون تصادم (FC) گوییم اگر و تنها اگر، هر کمان از یک مکان به گذر تنها کمان خروجی از آن مکان و یا تنها کمان ورودی به آن گذر باشد؛ یعنی  $\forall p \in P: |p^*| \leq 1 \vee {}^*(p^*) = \{p\}$ . در یک شبکه FC همروندی و تصادم بطور همزمان وجود ندارند.

<sup>1</sup> Self-Loop Free

<sup>2</sup> State Machine

<sup>3</sup> Marked Graph

<sup>4</sup> Free Choice

<sup>5</sup> Asymmetric Choice

۴- شبکه پتری  $N = (P, T, F, W)$  را تصادم نامتقارن (AC) گوییم اگر و تنها اگر دو مکان که دارای مابعد مشترک هستند، تمامی مابعدهای یکی از مکانها زیرمجموعه مابعدهای مکان دیگر باشند؛ یعنی  $(\forall p_i, p_j \in P: (p_i \cap p_j \neq \emptyset \Rightarrow p_i \subseteq p_j \text{ or } p_j \subseteq p_i))$  در یک شبکه AC همروندی و تصادم تنها بصورت نامتقارن می‌توانند وجود داشته باشند. بطور کلی نمی‌توان رابطه‌ای میان یک شبکه پتری FC و یک شبکه پتری AC برقرار دانست. یک شبکه ممکن است FC باشد و AC نباشد و بالعکس. رابطه FC بودن روی هر یک از مکانهای شبکه به تنهایی بررسی می‌شود. اما رابطه AC بودن میان زوج مکانهای یک شبکه بررسی می‌شود.

تعریف ۲-۶- بن‌بست: یک شبکه  $(N, M_0)$  را دچار بن‌بست گوییم اگر و تنها اگر با قرار گرفتن در یک حالت M هیچ گذری امکان فعال شدن را نداشته باشد.

تعریف ۲-۷- زنده بودن: برای زنده بودن یک گذر،  $t$ ، در یک شبکه پتری تعاریف متفاوتی ارائه شده است:

- زنده بودن سطح ۰ (گذر مرده<sup>۱</sup>): اگر  $t$  توسط هیچ دنباله‌ای از اجرا در  $M_0$  فعال نشود.
- زنده بودن سطح ۱: اگر  $t$  حداقل یک بار توسط برخی از دنباله‌های اجرا فعال شود.
- زنده بودن سطح ۲: اگر برای هر عدد مثبت  $k$ ، گذر  $t$  حداقل  $k$  بار توسط برخی از دنباله‌های اجرا فعال شود.
- زنده بودن سطح ۳: اگر  $t$  در برخی از دنباله‌های اجرا بصورت نامحدود فعال شود.

<sup>1</sup> LO-Live

<sup>2</sup> Dead Transition

- زنده بودن سطح ۴: اگر  $t$  برای تمامی دنباله‌های اجرا ویژگی زنده بودن سطح ۱ را دارا باشد. این تعریف به این معناست که در هر حالت  $M$  از سیستم، نشانه‌گذاری دسترس-پذیری مانند  $M'$  وجود دارد که در آن  $t$  فعال است؛ یعنی:

$$\forall M \in R(N, M_0) \rightarrow \exists M' \in R(N, M), M[t] \quad (2-7)$$

به همین ترتیب، یک شبکه پتری زنده سطح  $k$  است اگر تمامی گذرهای آن ویژگی زنده بودن سطح  $k$  را داشته باشند. در این رساله زنده بودن سطح ۴ بعنوان تعریف پیش‌فرض برای ویژگی زنده بودن در یک گذر و یا شبکه پتری در نظر گرفته می‌شود. با زنده بودن یک سیستم، این اطمینان وجود خواهد داشت که هیچ دنباله‌ای از اجرای گذرها منجر به توقف سیستم در حالت بن‌بست نخواهد شد.

تعریف ۲-۸- بازگشت‌پذیری: شبکه  $(N, M_0)$  را بازگشت‌پذیر گوئیم اگر و تنها اگر برای یک حالت دسترس‌پذیر  $M$  دنباله‌ای از اجرای گذرها وجود داشته باشد که سیستم را به حالت  $M_0$  برگرداند؛ یعنی:

$$\forall M \in R(N, M_0) \rightarrow M_0 \in R(N, M) \quad (2-8)$$

در یک سیستم بازگشت‌پذیر می‌توان اطمینان داشت اگر سیستم دچار شرایط ناگواری شود، قادر به بازسازی خودکار خود حداقل بصورت قرار گرفتن در حالت اولیه است. البته در بسیاری از سیستم‌ها بازگشت به یک حالت امن کافی است. این مسئله باعث ساده‌تر شدن مسئله بازگشت‌پذیری به مسئله‌ی دارا بودن حالت خانه<sup>۱</sup> است.

---

<sup>۱</sup>Home State

تعریف ۹-۲- شبکه‌های پتری کراندار، شبکه‌های پتری امن، شبکه‌های پتری کراندار ساختاری<sup>۱</sup>:

۱- یک مکان  $p \in P$  را در یک شبکه  $(N, M_0)$  کراندار گوئیم اگر و تنها بیشینه تعداد نشانه‌های  $p$  در نشانه‌گذاری‌های مختلف، یک عدد متناهی باشد؛ یعنی:

$$\exists k \in \mathbb{N} \cup \{0\}, \forall M \in R(N, M_0) \rightarrow M(p) \leq k \quad (2-9)$$

شبکه  $(N, M_0)$  را یک شبکه کراندار گوئیم اگر و تنها اگر تمامی مکان‌های آن کراندار باشند. اطمینان از کراندار بودن شبکه در سیستم‌هایی که دارای مفاهیمی مانند بافر و رجیستر برای نگه‌داری داده‌های میانی هستند، از عدم سرریز بافر و در نتیجه پایداری سیستم گزارش می‌دهد.

۲- اگر بیشینه تعداد نشانه‌ها برای تمامی مکان‌های یک شبکه پتری برابر عدد یک باشد، به آن یک شبکه پتری امن گفته می‌شود.

۳- یک شبکه پتری را شبکه پتری کراندار ساختاری گوئیم اگر و تنها اگر به ازای هر حالت اولیه  $M_0$  مفروض،  $(N, M_0)$  یک شبکه کراندار باشد. یک شبکه پتری کراندار ساختاری است اگر و تنها اگر یک بردار  $I$  با اندازه  $|P|$  وجود داشته باشد که  $I.C \leq 0$  باشد.

تعریف ۱۰-۲-  $P$ -Invariant،  $T$ -Invariant، شبکه‌های پتری پایا<sup>۲</sup>، شبکه‌های پتری سازگار<sup>۳</sup>:

۱- یک  $P$ -Invariant (PI) در یک شبکه پتری بصورت بردار  $I$  تعریف می‌شود که  $I.C = 0$  در مورد آن برقرار باشد.

۲- یک  $T$ -Invariant (TI) در یک شبکه پتری به صورت بردار  $I'$  تعریف می‌شود که  $C.I' = 0$  برقرار باشد.

<sup>1</sup> Structurally Bounded Petri Nets

<sup>2</sup> Conservative Petri Nets

<sup>3</sup> Consistent Petri Nets



۳- شبکه  $N$  را پایا گوئیم اگر و تنها اگر P-Invariant مثبتی با اندازه  $|P|$  داشته باشد.

۴- شبکه  $N$  را سازگار گوئیم اگر و تنها اگر دارای T-Invariant مثبتی با اندازه  $|T|$  باشد.

تعریف ۲-۱۱- سایفون، تله<sup>۱</sup>، سایفون کمینه<sup>۲</sup>، سایفون دارای تله<sup>۳</sup>، سایفون دارای تله نشانه دار<sup>۴</sup>:

۱- برای یک شبکه  $N$ ، زیرمجموعه‌ای از مکان‌ها مانند  $S$  را سایفون (تله) گوئیم اگر و تنها

اگر  $S^* \subseteq S^*$  ( $S^* \subseteq S^*$ ) باشد. بردار معرف یک سایفون؛ یک  $P$ -بردار است که عنصر  $1$  در یک

اندیس نشان‌دهنده وجود مکان متناظر در سایفون و عنصر  $0$  در یک اندیس نشان‌دهنده عدم

وجود مکان متناظر در سایفون است.  $T$ -بردار مشخصه<sup>۵</sup> یک سایفون از ضرب ماتریسی  $P$ -بردار

معرف سایفون در ماتریس برخورد بدست می‌آید. می‌توان نشان داد که هر گاه یک سایفون

خالی از نشانه شود، در تمامی نشانه‌گذاری‌های مابعد خالی از نشانه خواهد ماند. همچنین اگر

یک تله نشانه دار شود، در هیچیک از نشانه‌گذاری‌های مابعد بدون نشانه نخواهد شد.

۲- یک سایفون را سایفون کمینه گوئیم اگر شامل سایفون دیگری نباشد. بعبارت دیگر، برای هر

سایفون غیر کمینه حداقل یک سایفون کمینه وجود دارد که بردار معرف آن کوچکتر از بردار

معرف سایفون غیر کمینه است.

۳- یک شبکه پتری را دارای سایفون دارای تله (ST) گوئیم، اگر و تنها اگر برای هر زیر مجموعه از

مکان‌ها مانند  $S$  که سایفون است، بتوان حداقل یک زیرمجموعه  $S_1 \subseteq S$  یافت که تله باشد. یک

شبکه پتری را دارای سایفون دارای تله نشانه دار (MST) گوئیم، اگر و تنها اگر برای هر زیر

مجموعه از مکان‌ها مانند  $S$  که سایفون است، بتوان حداقل یک زیرمجموعه  $S_1 \subseteq S$  یافت که تله

بوده و حداقل یکی از مکان‌های  $S_1$  در نشانه‌گذاری اولیه دارای یک یا بیشتر نشانه باشد.

---

<sup>1</sup>Trap

<sup>2</sup> Minimal Siphon

<sup>3</sup> Siphon Trap

<sup>4</sup> Marked Siphon Trap

<sup>5</sup> Chacteristic T-Vector

## ۲-۴-۱ روش‌های مشخصه‌سازی<sup>۱</sup>

بطور معمول، روند بررسی یک ویژگی (مانند زنده بودن، کراندار بودن یا بازگشت‌پذیری) در یک سیستم تولیدی صنعتی با مقیاس بزرگ، بسیار زمانبر است. روش‌های مشخصه‌سازی، ویژگی‌های ساختاری همچون FC بودن، شبکه پتری معمولی و ویژگی MST را برای اعتبارسنجی برقرار بودن ویژگی مهم زنده بودن به کار می‌برند. در این بخش سه روش مشخصه‌سازی برای اعتبارسنجی زنده بودن یک شبکه پتری را که در ادامه رساله به کار می‌روند بیان می‌کنیم.

مشخصه‌سازی ۱-۲- قضیه کامونر<sup>۲</sup> - شرط کافی برای زنده بودن یک شبکه پتری FC: یک شبکه پتری FC زنده است اگر و تنها اگر دارای ویژگی MST باشد [۱].

مشخصه‌سازی ۲-۲- شرط کافی برای زنده بودن یک شبکه پتری AC: یک شبکه پتری AC زنده است اگر و تنها اگر معمولی بوده و دارای ویژگی MST باشد [۳۹].

مشخصه‌سازی ۳-۲- شرط کافی برای زنده بودن یک SM: یک شبکه پتری SM زنده است اگر و تنها اگر قویا همبند باشد [۱۲].

## ۲-۴-۳ انواع شبکه‌های پتری

در مقالات علمی محدودیت‌ها و تعمیم‌های مختلفی بر روی شبکه‌های پتری عمومی گزارش شده است که هر یک منجر به شکل‌گیری کلاس جدیدی از شبکه‌های پتری با قدرت بیان متفاوت

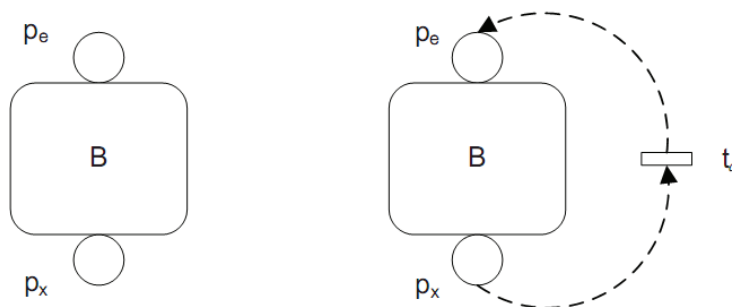
---

<sup>1</sup> Characterization Methods

<sup>2</sup> Commoner's Theorem

خواهد شد. در این بخش به برخی از این تحدیدها و تعمیم‌ها که می‌توانند مرتبط با بحث سیستم‌های تخصیص منابع باشند اشاره خواهیم کرد.

تعریف ۲-۱۲- فرآیند شبکه پتری<sup>۱</sup> [۴۰]: یک فرآیند شبکه پتری، کلاس خاصی از شبکه‌های پتری است و بصورت  $B = (N, p_e, p_x)$  تعریف می‌شود. این کلاس از شبکه‌های پتری یک شبکه معمولی و همبند است که دارای دو مکان خاص  $p_e$  (مکان ورودی) و  $p_x$  (مکان خروجی) علاوه بر مکان‌های گنجانده شده در تعریف شبکه پتری  $N$  است. مکان ورودی و خروجی بترتیب دارای هیچ گذر ورودی و گذر خروجی نیستند.



شکل ۲-۵- یک فرآیند شبکه پتری و فرآیند همبسته متناظر آن [۴۰].

اگر به یک فرآیند شبکه پتری، گذر جدید  $t_a$  افزوده شود و کمائی از  $p_e$  به گذر  $t_a$  و کمائی نیز از گذر  $t_a$  به مکان  $p_x$  وارد شود، فرآیند همبسته<sup>۲</sup> متناظر بدست می‌آید. بدین ترتیب فرآیند شبکه پتری به یک شبکه پتری قویا همبند تبدیل خواهد شد که می‌توان ویژگی‌های زنده بودن، کراندار و بازگشت‌پذیری را در این شبکه بررسی کرد. در صورت برقرار بودن هر یک از این ویژگی‌ها در سیستم تغییر یافته، سیستم اولیه را بترتیب تقریباً زنده<sup>۳</sup>، تقریباً کراندار<sup>۱</sup> و تقریباً بازگشت‌پذیر<sup>۲</sup> گوییم. شکل ۲-۵ شمای کلی یک فرآیند شبکه پتری و فرآیند همبسته متناظر آن را نشان می‌دهد.

<sup>۱</sup> Petri net Process

<sup>۲</sup> Associated Process

<sup>۳</sup> Almost Live

تعریف ۲-۱۳- شبکه‌های پتری مبتنی بر سیستم‌های تولیدی<sup>۳</sup>: مدلی است که در آن مجموعه مکان‌ها به سه زیرمجموعه افزاری شود: مکان‌های بیکار<sup>۴</sup> ( $P^0$ )، مکان‌های عملیاتی<sup>۵</sup> ( $P_A$ ) و مکان‌های منبع<sup>۶</sup> ( $P_R$ ). نشانه‌گذاری اولیه مکان‌های بیکار نشان دهنده بیشینه تعداد فرآیندهای تولیدی همروند در سیستم است. مکان‌های عملیاتی در نشانه‌گذاری اولیه، نشانه‌ای ندارند. مکان‌های منبع، نشان دهنده منابع سیستمی بوده و نشانه‌گذاری اولیه آنها تعداد واحدهای در دسترس از هر یک از منابع است. برای جلوگیری از وقوع حالات ممنوع در یک مدل شبکه پتری مبتنی بر سیستم‌های تولیدی، تنها نشانه‌های مکان‌های عملیاتی تحت کنترل قرار می‌گیرند. انواع مختلفی از شبکه‌های پتری مبتنی بر سیستم‌های تولیدی در تحقیقات معرفی شده‌اند. شبکه‌های  $S^3PR$  [۴۱]،  $S^3PGR$  [۴۲] و  $S^3MPR$  [۴۳] نمونه‌هایی از این شبکه‌ها هستند. یک شبکه سیستم فرآیندهای ترتیبی ساده با منابع<sup>۷</sup> ( $S^3PR$ )، ( $S^3PR$ )، متشکل از تعدادی فرآیند به شکل ماشین وضعیت معمولی، و تعدادی منابع اشتراکی است. هر حالت پردازشی در هر فرآیند، حداکثر از یک واحد منبع استفاده می‌کند.

از جمله تعمیم‌های این کلاس از شبکه‌های پتری، سیستم فرآیندهای ترتیبی ساده تعمیم یافته با منابع<sup>۸</sup> ( $GS^3PR$ ) یا همان سیستم فرآیندهای ترتیبی ساده با منابع وزندار<sup>۹</sup> ( $WS^3PR$ ) هستند [۴۴] که در هر حالت عملیاتی<sup>۱۰</sup> در هر فرآیند، حداکثر از یک نوع منبع و به تعداد محدود را (نه فقط یک واحد) استفاده می‌کنند.

<sup>1</sup> Almost Bounded

<sup>2</sup> Almost Reversible

<sup>3</sup> AMS-Oriented Petri nets

<sup>4</sup> Idel Places

<sup>5</sup> Oprational Places

<sup>6</sup> Resource Places

<sup>7</sup> System of Simple Sequential Processes with Resources

<sup>8</sup> Generalized System of Simple Sequential Processes with Resources

<sup>9</sup> Weighted System of Simple Sequential Processes with Resources

<sup>10</sup> Operational State

تعمیمی از شبکه‌های پتری  $GS^3PR$ ، شبکه‌های  $S^4PR$  هستند که در مراجع از آنها با نام‌های  $S^4R$  و  $S^3PGR^2$  نیز یاد شده است [۴۵]. شبکه‌های  $S^4PR$  اجازه ذخیره‌سازی همزمان منابعی با نوع-های متفاوت را نیز می‌دهد. عمومی‌ترین نوع این شبکه‌ها، شبکه‌های  $S^nPR$  یا  $S^*PR$  ها هستند که در آنها فرآیندها، ماشین وضعیت معمولی هستند که می‌توانند دارای چرخه داخلی نیز باشند .

از دیگر تعمیم‌های  $S^nPR$ ها، فرآیندهای رقابتگر برای منابع پایا<sup>۱</sup> ( $PC^2R$ ) را می‌توان نام برد که آنها نیز برای پیشگیری از بن‌بست در برنامه‌های کامپیوتری چند ریسمانه استفاده می‌شوند [۴۶]. این شبکه‌ها، پایا و در نتیجه کراندار ساختاری هستند. کلاس دیگری از شبکه‌های پتری مبتنی بر سیستم‌های تولیدی، فرآیندهای رقابتگر روی منابع<sup>۲</sup> ( $SPQR$ ) هستند که از نظر قدرت بیان همسطح  $PC^2R$  ها هستند. از دیگر کلاس‌های شبکه‌های پتری می‌توان به شبکه‌های گادارا<sup>۳</sup> [۴۷] اشاره کرد که برای مدلسازی اجرای برنامه‌های کامپیوتری چند ریسمانه<sup>۴</sup> و تحلیل پیشگیری از بن‌بست استفاده می‌شوند.

کنترل نظارتی در شبکه‌های مبتنی بر سیستم‌های تولیدی، به نحو ساده‌تری قابل اعمال هستند. البته تعداد نشانه‌گذاری‌های قانونی و تعداد نشانه‌گذاری‌های ممنوع اولین بار دیده شده<sup>۵</sup> (FBMs) بر پیچیدگی محاسباتی روش‌های کنترل نظارتی تأثیرگذار هستند. در [۴۸] روش‌هایی به نام پوشش برداری برای کاهش تعداد نشانه‌گذاری‌های مورد بررسی در یک سیاست کنترل نظارتی بیان شده‌اند.

<sup>1</sup> Processes Competing for Conservative Resources

<sup>2</sup> Systems of Processes Quarrelling over Resources

<sup>3</sup> Gadara Nets

<sup>4</sup> Multi Thread

<sup>5</sup> First-Met Bad Markings

اگر  $P_A \subseteq P$  را مجموعه مکان‌های عملیاتی یک شبکه پتری مبتنی بر سیستم‌های تولیدی در نظر بگیریم، نشانه‌گذاری  $M_1$  نشانه‌گذاری  $M_2$  را  $A$ -پوشش<sup>۱</sup> می‌دهد ( $M_2$  توسط  $M_1$ ،  $A$ -پوشیده<sup>۲</sup> می‌شود) اگر و تنها اگر  $\forall p \in P_A \Rightarrow M_1(p) \geq M_2(p)$ ، که با نماد  $M_1 \geq_A M_2$  بیان می‌شود.

مجموعه کمینه نشانه‌گذاری‌های قانونی پوشش دهنده<sup>۳</sup> (مجموعه کمینه نشانه‌گذاری‌های بد تحت پوشش<sup>۴</sup>)، زیرمجموعه‌ای از نشانه‌گذاری‌های قانونی (بد<sup>۵</sup>) است که بقیه نشانه‌گذاری‌های قانونی را  $A$ -پوشش می‌دهد (توسط بقیه نشانه‌گذاری‌های قانونی  $A$ -پوشیده می‌شود) و با نماد  $(M_{FBM}^*) M_L^*$  نمایش داده می‌شود.

فرض کنید  $M_1, M_2, M \in R(N, M_0)$  باشند. نشانه‌گذاری  $M_1$  نشانه‌گذاری  $M_2$  را  $M$ -پوشش می‌دهد اگر  $\forall p \in P, M(p) \neq 0 \Rightarrow M_1(p) \geq M_2(p)$  که با استفاده از  $M_1 \geq_M M_2$  نمایش داده می‌شود.

مجموعه کمینه نشانه‌گذاری‌های قانونی  $M$  - مرتبط<sup>۶</sup>، زیر مجموعه‌ای از نشانه‌گذاری‌های قانونی است که بقیه نشانه‌گذاری‌های قانونی را  $M$ -پوشش می‌دهند.

علاوه بر محدودیت‌هایی که منجر به ساختن زیرکلاسی از شبکه‌های پتری عمومی می‌شوند، تعمیم ویژگی‌های اجزای مختلف شبکه‌های پتری، منجر به ایجاد تعمیم‌هایی از شبکه‌های پتری می‌شوند. در صورت عدم استفاده شبکه‌های پتری از کمان‌های بازدارنده، قدرت هیچیک از این شبکه‌های تعمیم‌یافته بیشتر از شبکه‌های پتری معمولی نیست. از مهمترین تعمیم‌هایی که در شبکه‌های پتری ایجاد شده افزودن کمان‌های بازدارنده بود که باعث شد قدرت زبان شبکه‌های پتری در حد قدرت

<sup>1</sup> A-Covers

<sup>2</sup> A-Covered By

<sup>3</sup> Minimal Covering Set of Legal Markings

<sup>4</sup> Minimal Covered Set of Bad Markings

<sup>5</sup> Bad Markings

<sup>6</sup> Minimal Covering Set of M-Related Markings

کاملترین خانواده از زبان‌ها یعنی ماشین تورینگ افزایش یابد. شبکه‌های پتری رنگی<sup>۱</sup>، تعمیمی از شبکه‌های پتری عمومی هستند. در این شبکه‌ها، نشانه‌ها علاوه بر اینکه دارای اطلاعات تعداد برای یک مکان هستند، نوع را نیز در بر می‌گیرند. گذرها می‌توانند شامل توابع دودویی<sup>۲</sup> بنام گارد<sup>۳</sup> باشند که نقش شرط فعال شدن را ایفا می‌کنند. در اجرای یک گذر تنها مقادیر متناسب با هر مکان خروجی، تولید می‌شوند.

شبکه‌های پتری بطور ذاتی فاقد زمان هستند. البته می‌توان مفهوم زمان را به گذرها و یا مکان‌های شبکه‌های پتری افزود بدین ترتیب شبکه‌های پتری با گذر زماندار<sup>۴</sup> و شبکه‌های پتری با مکان زماندار<sup>۵</sup> تعریف شده‌اند [۴۹]. اگر زمان مورد نظر بصورت تصادفی تعریف شده باشد مفهوم شبکه‌های پتری تصادفی<sup>۶</sup> بدست می‌آید که نقش مهمی در ارزیابی کارایی یک سیستم و یافتن گلوگاه‌های آن دارد.

در شبکه‌های پتری برچسب‌دار<sup>۷</sup> به هریک از گذرها برچسب‌هایی اختصاص داده می‌شود. برای تعریف این شبکه‌ها علاوه بر اجزای شبکه‌های پتری اولیه، به یک تابع برچسب نیاز داریم. از مهمترین کاربردهای شبکه‌های پتری برچسب‌دار، نگاشتن یک مدل شبکه پتری به یک زبان است. تعریف این نوع شبکه‌ها بصورت  $N' = (N, \sigma, M_0, M_f)$  است. که  $N$ ، ساختار شبکه پتری اولیه،  $\sigma$  تابع برچسب دهی که به هر گذر برچسبی را نسبت می‌دهد،  $M_0$  حالت شروع و  $M_f$  حالات پایانی یا حالات پذیرش هستند.

---

<sup>1</sup> Colored Petri Nets

<sup>2</sup> Boolean Expressions

<sup>3</sup> Guard

<sup>4</sup> T- Timed PNs

<sup>5</sup> P- Timed PNs

<sup>6</sup> Stochastic Petri Nets

<sup>7</sup> Labeled Petri Nets

## ۲-۴-۴ عملگرهای سنتز در شبکه‌های پتری

روش‌های دگرسازی<sup>۱</sup> منجر به تعریف یک شبکه پتری جدید می‌شوند که خواص شبکه‌های (های) پتری سازنده را حفظ می‌کنند. این روش‌ها به دو گروه کلی دسته‌بندی می‌شوند: روش‌های کاهش و روش‌های سنتز. قوانین کاهش [۵۰]، یک سیستم پیچیده را به سیستم ساده‌تری تبدیل می‌کنند و در عین حال برخی از ویژگی‌های تحلیلی سیستم ابتدایی را در سیستم نهایی حفظ خواهند کرد.

شبکه‌های پتری فاقد مکانیزم‌های سنتز بطور ذاتی هستند. مدلسازی و اعتبارسنجی سیستم بزرگ - مقیاس با استفاده از روش‌های سنتز، منجر به همبستگی اجزای سیستمی در کنار یکدیگر و تشکیل سیستم کلی می‌شود. روش‌های سنتز به چهار دسته جزئی‌تر تقسیم می‌شوند: روش‌های ترکیب [۵۱]، ادغام [۵۲]، پالایش [۵۳] و بافتن<sup>۲</sup> [۵۴].

## ۲-۴-۴-۱ عملگرهای ترکیب

از مهمترین روش‌های سنتز، روش‌های ترکیب هستند. این روش‌ها دارای رویکرد پایین به بالا<sup>۳</sup> هستند. عملگرهای تکرار، انتخاب، ترتیب، برگ‌برگ‌سازی و موازی [۵۵] از جمله این عملگرها هستند. در این قسمت به توضیح برخی از این عملگرها که در ادامه رساله از آنها استفاده خواهد شد می‌پردازیم.

تعریف ۲-۱۴ - عملگر انتخاب، به مدلسازی تصادم بین دو فرآیند می‌پردازد. دو فرآیند شبکه پتری نشانه‌دار  $(B_i(N_i, p_{ei}, p_{xi}, M_{0i}))$ ,  $i = 1, 2$  را در نظر بگیرید. ترکیب آنها با استفاده از عملگر انتخاب منجر به یک فرآیند پتری جدید  $(B(N, p_e, p_x, M_0))$  می‌شود که [۵۵]:

<sup>1</sup> Transformation Methods

<sup>2</sup> Knitting

<sup>3</sup> Bottom-Up



$$B = B_1 \square B_2$$

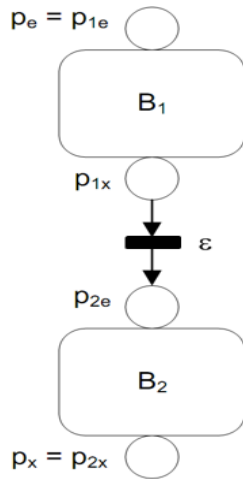
$$P = P_1 \cup P_2 \cup \{p_e, p_x\} - \{p_{1x}, p_{2x}\}$$

$$T = T_1 \cup T_2 \cup \{\varepsilon_1, \varepsilon_2\}$$

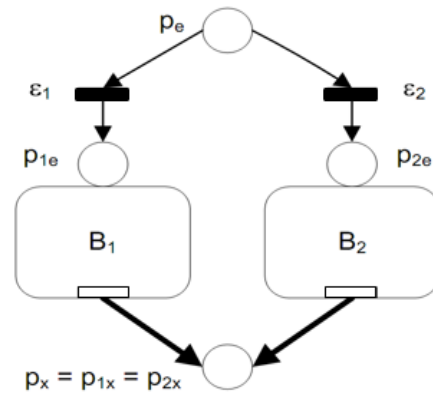
$$F = F'_1 \cup F'_2 \cup \{(p_e, \varepsilon_1), (p_e, \varepsilon_2), (\varepsilon_1, p_{1e}), (\varepsilon_2, p_{2e})\}, \text{ where}$$

$$F'_i = F_i - \{(t, p_{xi}): t \in \bullet p_{xi}\} \cup \{(t, p_x): t \in \bullet p_{xi}\}, i = 1, 2.$$

$$M_0 = (1, 0, M_{01}', 0, M_{02}', 0). \quad (2-11)$$



شکل ۷-۲- شمای کلی ترکیب پیمانه‌های  $B_1$  و  $B_2$  با استفاده از عملگر ترتیب [۵۵].



شکل ۶-۲- شمای کلی ترکیب پیمانه‌های  $B_1$  و  $B_2$  با استفاده از عملگر انتخاب [۵۵].

که  $M'_{0i}$  برداری از نشانه‌گذاری‌های تمامی مکان‌های  $B_i$  بجز  $p_e$  و  $p_x$  است. در اثر اعمال این عملگر، مکان ورودی جدیدی بنام  $p_e$  با استفاده از گذرهای جدید  $\varepsilon_1$  و  $\varepsilon_2$  بترتیب با مکان‌های ورودی فرآیند شبکه پتری  $B_1$  و  $B_2$  مرتبط می‌شود. گذرهای خروجی هر یک از شبکه‌ها نیز که به مکان خروجی  $p_{xi}$  متصل بودند، به مکان خروجی جدید  $p_x$  متصل خواهند شد و در نتیجه مکان‌های خروجی  $p_{xi}$  حذف خواهند شد. شمای گرافیکی این عملگر در شکل ۶-۲ آمده است.

تعریف ۲-۱۵- عملگر ترتیب در شبکه‌های پتری، به مدلسازی برهم‌کنش ترتیب بین دو فرآیند می‌پردازد. دو سیستم نشانه‌دار  $(B_i(N_i, p_{ei}, p_{xi}, M_{0i}))$ ,  $i = 1, 2$  را در نظر بگیرید. ترکیب آنها با استفاده از عملگر ترتیب منجر به یک فرآیند پتری جدید  $(B(N, p_e, p_x, M_0))$  می‌شود که  $[\Delta\delta]$ :

$$B = B_1 \gg B_2.$$

$$P = P_1 \cup P_2, p_e = p_{e1}, p_x = p_{x2}.$$

$$T = T_1 \cup T_2 \cup \{\varepsilon\}.$$

$$F = F'_1 \cup F'_2 \cup \{(p_{x1}, \varepsilon), (\varepsilon, p_{e2})\}, \text{ where}$$

$$F'_i = F_i - \{(t, p_{xi}): t \in \bullet p_{xi}\} \cup \{(t, p_x): t \in \bullet p_{xi}\}, i = 1, 2.$$

$$M_0 = (1, 0, M_{01}', 0, M_{02}', 0). \quad (2-12)$$

که  $M_{0i}'$  برداری از نشانه‌گذاری‌های تمامی مکان‌های  $B_i$  بجز  $p_x$  و  $p_e$  است. در اثر اعمال این عملگر، مکان ورودی فرآیند شبکه پتری  $B_1$  بعنوان مکان ورودی شبکه ترکیبی در نظر گرفته می‌شود. مکان خروجی این فرآیند نیز با استفاده از گذر جدید  $\varepsilon_1$  به مکان ورودی فرآیند شبکه پتری  $B_2$  مرتبط می‌شود. نهایتاً مکان خروجی جدید  $p_{x2}$  بعنوان مکان خروجی شبکه ترکیبی در نظر گرفته می‌شود. شمای گرافیکی این عملگر در شکل ۲-۷ آمده است.

تعریف ۲-۱۶- عملگر برگ‌برگ‌سازی [۸۳]، به مدلسازی اجرای همروند دو فرآیند بدون انتقال

پیام بین آنها می‌پردازد. دو فرآیند پتری نشانه‌دار  $(B_i(N_i, p_{ei}, p_{xi}, M_{0i}))$ ,  $i \in \{1, 2\}$  که  $M_{0i} = (1, M_{0ci}, 0)$  را در نظر بگیرید. ترکیب این دو فرآیند با استفاده از عملگر برگ‌برگ‌سازی منجر به ایجاد فرآیند پتری جدید  $(B(N, p_e, p_x, M_0))$  خواهد شد که:

$$B = B_1 \parallel B_2.$$

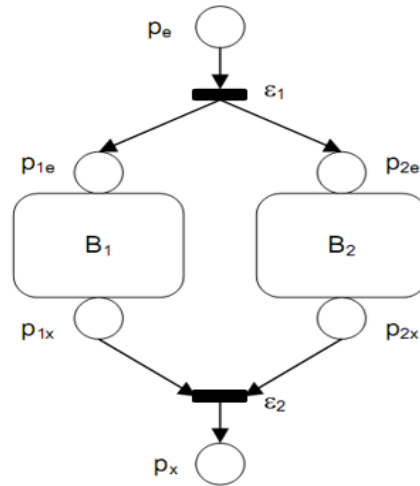
$$P = P_1 \cup P_2 \cup \{p_e, p_x\}.$$

$$T = T_1 \cup T_2 \cup \{\varepsilon_1, \varepsilon_2\}.$$

$$F = F_1 \cup F_2 \cup \{(p_e, \varepsilon_1), (\varepsilon_1, p_{e1}), (\varepsilon_1, p_{e2}), (p_{1x}, \varepsilon_2), (p_{2x}, \varepsilon_2), (\varepsilon_2, p_x)\}.$$

$$M_0 = (1, 0, M_{0c1}, 0, 0, M_{0c2}, 0, 0). \quad (2-13)$$

این عملگر مکان جدید  $p_e$  را بعنوان مکان ورودی فرآیند شبکه پتری ترکیبی در نظر می‌گیرد. گذر جدید  $\varepsilon_1$  نیز موجب ایجاد توازی میان اجرای فرآیندهای  $B_1$  و  $B_2$  می‌شود. گذر جدید  $\varepsilon_2$  نیز منجر به ایجاد همزمانی میان اتمام این دو فرآیند می‌شود که با فعال شدن خود، مکان خروجی جدید  $p_x$  را نشانه‌دار می‌کند. شمای کلی این عملگر در شکل ۸-۲ آمده است.



شکل ۸-۲ شمای کلی ترکیب پیمان‌های  $B_1$  و  $B_2$  با استفاده از عملگر برگ‌برگ‌سازی [۵۵].

تعریف ۲-۱۷- عملگر ترکیب همروند<sup>۱</sup>: این عملگر بر روی شبکه‌های پتری برچسب‌دار تعریف شده و در کنترل نظارتی کاربرد دارد. حاصل اعمال این عملگر بر دو شبکه پتری  $N_1$  و  $N_2$  یک شبکه پتری جدید است که مولد رشته‌هایی است که تصویر آنها نسبت به الفبای  $N_1$ ، برابر با رشته‌ای از  $N_1$  باشد و تصویر آن نسبت به الفبای  $N_2$  برابر رشته‌ای از  $N_2$  باشد.

## ۲-۴-۵ عملگرهای ادغام

<sup>۱</sup> Concurrent Composition

از دیگر روش‌های سنتز که با رویکرد پایین به بالا به دگرسازی در یک سیستم می‌پردازند، روش‌های ادغام هستند. بدلیل رویکرد پایین به بالای این روش‌ها، گاه در کنار روش‌های ترکیب و در یک دسته تقسیم‌بندی می‌شوند. تفاوت این دسته از روش‌ها با روش‌های ترکیب، در این است که روش‌های ترکیب به تجمیع تعداد دو یا بیشتر از شبکه‌های پتری در کنار هم می‌پردازند؛ ولی روش‌های ادغام اجزای یک شبکه را با هم تجمیع و ادغام ساخته و شبکه ساده‌تری را بدست می‌آورند. از جمله عملگرهای این روش می‌توان به عملگرهای ادغام گذرها [۳۷، ۵۶]، ادغام مکان‌ها [۵۲] و ادغام مسیر<sup>۱</sup> [۵۷] اشاره کرد.

تعریف ۲-۱۸- عملگر ادغام گذرها: این عملگر مجموعه‌ای از گذرها را به یک گذر واحد ادغام می‌کند. با استفاده از این عملگر، می‌توان چندین ادغام گذر را بصورت همزمان انجام داد. از آنجا که اغلب، گذرهای یک شبکه پتری نشان دهنده یک رخداد در سیستم متناظر هستند، ادغام گذرها می‌تواند همگام‌سازی بین رخدادهای سیستم را نتیجه دهد. فرض کنیم  $N = (P, T_u \cup T_{m1} \cup \dots \cup T_{mk}, F, W)$  یک شبکه پتری و  $T_{mi} = \{t_{i1}, \dots, t_{i|T_{mi}|}\}$  باشد. با دانستن اینکه گذرهای شبکه  $N$  به  $k + 1$  زیرمجموعه افراز شده‌اند، پس از ادغام هر یک از گذرهای درون مجموعه‌های  $T_1, \dots, T_k$  به یک گذر، سیستم جدیدی با نام  $(N', M_0')$  بدست خواهد آمد که  $N' = (P', T', F', W')$  و  $T' = T_u \cup T_m$  شامل تمام گذرهای نماینده برای ادغام‌ها است و  $t_{mi}$  از ادغام مجموعه گذرهای درون  $T_{mi}$  به یک گذر به دست آمده است. شمای گرافیکی این عملگر در شکل ۲-۹ آمده است. تعریف رسمی این عملگر بدین شرح است [۵۷]:

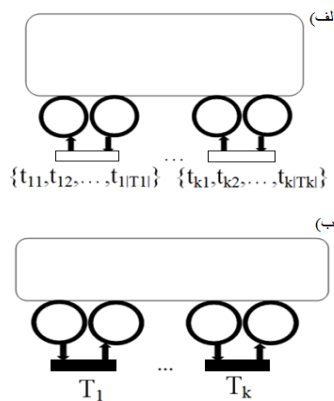
$$1- P' = P.$$

$$2- T' = \begin{cases} t & \text{if } t \in T_u \text{ in } N \\ t_{mi} & \text{if } t \in T_{mi} \text{ in } N \end{cases}$$

---

<sup>1</sup> Path merging

$$\begin{aligned}
3- F' &= \begin{cases} (t,p) \text{ if } t \in T_u \text{ and } (t,p) \in f \\ (p,t) \text{ if } t \in T_u \text{ and } (p,t) \in f \\ (p,t) \text{ if } t \in T_m \text{ and } \exists t_{ik} \in T_{mi}, (p,t_{ik}) \in F \\ (t,p) \text{ if } t \in T_m \text{ and } \exists t_{ik} \in T_{mi}, (t_{ik},p) \in F \end{cases} \\
4- W'(p,t) &= \begin{cases} (t,p) \text{ if } t \in T_u \\ \text{sum}(W(p,t_i)), t_i \in T_{mi}, t_i \text{ is merged into } t \end{cases} \\
W'(t,p) &= \begin{cases} (p,t) \text{ if } t \in T_u \\ \text{sum}(W(t_i,p)), t_i \in T_{mi}, t_i \text{ is merged into } t \end{cases} \\
5- M_0' &= M_0. \tag{2-14}
\end{aligned}$$



شکل ۹-۲ شمای کلی عملگر ادغام گذرها [۵۷]. (الف) شبکه پتری اولیه. (ب) شبکه پتری پس از ادغام گذرها. دواير پرنرنگ نشان دهنده مجموعه‌ای از مکان‌ها و مستطیل‌های سفیدرنگ نشان دهنده مجموعه گذرها هستند.

تعریف ۲-۱۹- عملگر ادغام مکان‌ها: عملگر ادغام مکان‌ها چندین مجموعه از مکان‌ها را بطور جداگانه به یک مکان ادغام می‌کند. از آنجاکه معمولا منابع اشتراکی یک سیستم، با استفاده از مکان‌های یک شبکه پتری مدل می‌شوند، ادغام مکان‌ها می‌تواند اشتراک‌گذاری منابع را شبیه‌سازی کند. فرض کنیم  $N = (P_u \cup P_{m1} \cup P_{m2} \cup \dots \cup P_{mk}, T, F, W)$  یک شبکه پتری و  $P_{mi} = \{p_{i1}, \dots, p_{i|T_i|}\}$  باشد. با دانستن اینکه مکان‌های شبکه  $N$  به  $k + 1$  زیرمجموعه افراز شده‌اند، پس از ادغام هر یک از مکان‌های درون مجموعه‌های  $P_{m1}, \dots, P_{mk}$  به یک مکان، سیستم جدیدی با نام  $(N', M_0')$  بدست خواهد آمد که  $N' = (P_u \cup P_m, T, F', W')$  شامل  $P_m = \{p_{m1}, p_{m2}, \dots, p_{mk}\}$

تمام مکان‌های بدست آمده از ادغام‌ها است و  $p_{mi}$  از ادغام مجموعه مکان‌های درون  $P_{mi}$  به یک مکان به دست آمده است. تعریف رسمی این عملگر بدین شرح است [۵۲]:

$$1- T' = T.$$

$$2- P' = \begin{cases} p & \text{if } p \in P_u \text{ in } N \\ p_{mi} & \text{if } p \in P_{mi} \text{ in } N \end{cases}$$

$$3- F' = \begin{cases} (t, p) & \text{if } p \in P_u \text{ and } (t, p) \in f \\ (p, t) & \text{if } p \in P_u \text{ and } (p, t) \in f \\ (t, p) & \text{if } p \in P_m \text{ and } \exists p_{ik} \in P_{mi}, (t, p_{ik}) \in F \\ (p, t) & \text{if } p \in P_m \text{ and } \exists p_{ik} \in P_{mi}, (p_{ik}, t) \in F \end{cases}$$

$$4- W'(p, t) = \begin{cases} W(p, t), & \text{if } p \in P_u \\ \text{Sum}(W(p_{ik}, t)), & \text{if } p_{ik} \in P_{mi} \text{ and } p_{ik} \text{ is merged into } p \end{cases}$$

$$W'(t, p) = \begin{cases} W(t, p), & \text{if } p \in P_u \\ \text{Sum}(W(t, p_{ik})), & \text{if } p_{ik} \in P_{mi} \text{ and } p_{ik} \text{ is merged into } p \end{cases}$$

$$5- M_0'(p) = \begin{cases} M_0(p), & \text{if } p \in P_u \\ \max_{p \in P_{mi}} \{M_0(p)\} \end{cases} \quad (2-15)$$

در [۵۲] اثبات می‌شود در صورت برقرار بودن شروط ذیل در شبکه پتری  $N$ ، عملگر ادغام

مکان‌ها یک عملگر حافظ ویژگی زنده بودن است:

۱- مکان‌های ادغام شونده غیر مجاور باشند؛ یعنی دارای مابعد مشترک نبوده و ماقبل مشترک نداشته باشند.

۲- بتوان شبکه پتری ادغام شده  $N'$  را به پیمانه‌هایی که هر یک از آنها SM هستند تجزیه نمود (افراز لازم نیست) بگونه‌ای که اجتماع گذر(مکان)های پیمانه‌ها، گذر(مکان)های شبکه کلی را تشکیل دهد.

۳- سیستم شبکه پتری  $(N, M_0)$  دارای ویژگی سایفون دارای تله نشانه‌دار با کاهش تکرار مجموعه<sup>۱</sup> باشد. این ویژگی زمانی برقرار است که برای هر سایفون  $S$ : اگر  $S \subseteq P_0$ ،  $S$  دارای

<sup>1</sup> Reduction-Set-Redundant Marked Siphon Trap

یک تله نشانه‌دار  $S'$  است. اگر  $P_r \subseteq S$  باشد، یک تله نشانه‌دار در  $S$  وجود دارد که یا  $S' \subseteq P^0$

است و یا  $P_r \subseteq S'$  است که  $P_r \subseteq P_R$  و  $P_0 = P_A \cup P^0$ .

تعریف ۲-۲۰- عملگر پالایش: عملگر پالایش  $B_1(t_r \rightarrow B_2)$ ، که در آن  $(B_1, M_r)$

شبکه پالایش شونده و  $(B_2, M_r)$  یک فرآیند شبکه پتری است، گذر  $t_r$  با مکان‌های ۱-

کراندار ورودی و خروجی  $r_i$  و  $r_o$  از شبکه پتری  $B_1$  را با شبکه پتری  $B_2$  جایگزین می‌کند و

شبکه جدید  $(B, M_o)$  را نتیجه می‌دهد. تعریف فرمال این عملگر بدین ترتیب است [۵۸]:

$$B = B_1(t_r \rightarrow B_2) = (N, p_i, p_o); N = (P, T, F)$$

$$P = P_1 \cup P_2 \cup \{p_i, p_o\} - \{r_i, r_o, p_e, p_x\}$$

$$T = T_1 \cup T_2 - \{t_r\}$$

$$F = F_1 \cup F_2 \cup (\{(p_i, x) | x \in p_x^*\} \cup \{(x, p_o) | x \in \bullet p_x\}) \cup \{(x, p_i) | x \in \bullet r_i\} \cup \\ \{(p_o, x) | x \in r_o^*\} - (\{(r_i, t_r), (t_r, r_o)\} \cup \{(x, r_i) | x \in \bullet r_i\} \cup \{(r_o, x) | x \in r_o^*\} \cup \\ \{p_e, x\} | x \in p_e^*\} \cup \{(x, p_x) | x \in \bullet p_x\})$$

$$M_o(p_o) = M_r(r_i); M_o(p_o) = M_r(r_o);$$

$$M_o(p) = M_r(p) \forall p \in P_1 - \{r_i, r_o\}$$

$$M_o(p) = M_s(p) \forall p \in P_2 - \{p_e, p_x\} \quad (2-16)$$

عملگر بافتن [۵۴، ۵۹] از دیگر عملگرها با رویکرد پایین به بالاست. این عملگر به ارائه مکانیزم

مبتنی بر قانون برای سنتز پیمان‌های مختلف سیستمی استفاده می‌کنند. روش‌های سنتز ترکیبی،

برای استفاده همزمان از مزایای رویکردهای پایین به بالا و بالا به پایین نیز معرفی شده‌اند [۶۰].

## ۲-۵ کنترل مدل توسط کنترل نظارتی

در سال ۱۹۸۷ رمج و ونهام [۲۵] یک چارچوب انتزاعی برای کنترل سیستم‌های گسسته رخداده با نام نظریه کنترل نظارتی را ارائه دادند. اساس این نظریه بر محدود کردن رفتار سیستم مورد مطالعه بر اساس رفتار مطلوب است. این محدودسازی با استفاده از غیر فعال نمودن گذرهای قابل کنترل صورت می‌پذیرد. حالات نامطلوب سیستمی اعم از حالات بن‌بست و حالاتی است که خصوصیت خاصی را نقض می‌کنند. رمج و ونهام [۲۵] نشان دادند با داشتن یک سیستم به نام کارخانه و داشتن خصوصیت مطلوب، یک تابع کنترلی یکتا با نام سوپروایزر وجود دارد که کمترین محدودکنندگی<sup>۱</sup> را برای رفتار سیستم داشته و تنها از حالاتی که خلاف خصوصیات مطلوب هستند پیشگیری می‌نماید. آنها همچنین روشی برای سنتز این سوپروایزر ارائه دادند که در آن از رویکرد زبانی مبتنی بر اتوماتا برای ایجاد زبان کارخانه و زبان ضوابط استفاده می‌کردند. البته در طول سال‌ها رویکردهای دیگری نیز بر اساس این نظریه برای محدود کردن رفتار سیستم ارائه شده است. همچنین تعمیم‌های مختلفی از مسئله کنترل نظارتی مانند کنترل با قابلیت مشاهده‌پذیری جزئی از رخدادها، و یا کنترل‌پذیری جزئی رخدادها، کنترل غیر متمرکز، کنترل سیستم‌های غیر قطعی، کنترل رفتار نامحدود و غیره ارائه شده است.

تعریف ۲-۲۱- سوپروایزر: سوپروایزر یک عامل کنترلی است که رفتار سیستم را در حوزه رفتار مشخص شده قانونی محدود می‌کند. سوپروایزر همانند یک تابع بازخورد<sup>۲</sup> است که یک مدل حلقه بسته از سیستم می‌سازد. تکنیک‌های تحلیلی پتری در محاسبه کنترل نظارتی به کار گرفته می‌شود.

---

<sup>۱</sup> Minimally Restrictive

<sup>۲</sup> Feedback Function



تفاوت کنترلر و سوپروایزر در این است که کنترلر (که مطابق رفتار کنترلرهای منطقی قابل برنامه‌نویسی<sup>۱</sup> (PLC) است) رخدادهای خاصی را در جهت کنترل فعال می‌کند، اما سوپروایزر رخدادهای خاصی را غیر فعال می‌کند. هدف سوپروایزر بیان این است که سیستم مجاز به انجام چه اعمالی است، اما هدف کنترلر بیان این است که سیستم چه اعمالی را باید انجام دهد.

کنترل نظارتی، با غیر فعال کردن برخی از گذرهای فعال، منجر به دسترس‌ناپذیر شدن حالات ممنوع می‌شود. بطور کلی ضوابط به دو دسته قابل تقسیم هستند: ضوابط ایستا<sup>۲</sup> و ضوابط رفتاری<sup>۳</sup> [۶۱]. ضوابط ایستا، در هر حالت به تنهایی مورد بررسی قرار می‌گیرند. این ضوابط نشان دهنده محدودیت‌های خاص مسئله مانند محدودیت‌هایی برای تعداد منابع مشترک مورد استفاده، یا تعداد بیشینه کارهای هم‌زمان مجاز هستند. این محدودیت‌ها با بررسی حالات یک سیستم قابل بررسی هستند و اگر با الگوریتمی حالات ممنوع متناظر یافته شوند، کاهش گراف دسترسی قانونی (پس از آزمایش دیگر محدودیت‌ها) این محدودیت‌ها را نقض نمی‌نماید.

از طرفی، ضوابط رفتاری، منعکس‌کننده خواص رفتاری سیستم هستند [۶۱]. از مهمترین ویژگی‌های یک سیستم مبتنی بر شبکه‌های پتری که نیازمند اعتبارسنجی است، زنده بودن شبکه پتری است. هدف از یک کنترل نظارتی اعمال‌کننده زنده بودن<sup>۴</sup> (LESC) مبتنی بر شبکه‌های پتری، الزام ضوابط خاص سیستمی است. این ضوابط، مشخص‌کننده مجموعه حالات ممنوع و حالات قانونی<sup>۵</sup> در گراف دسترسی شبکه‌های پتری هستند.

## ۲-۵-۱ معیارهای ارزیابی روش‌های کنترل نظارتی

---

<sup>1</sup> Programmable Logic Controller

<sup>2</sup> Static Specifications

<sup>3</sup> Behavioral Specifications

<sup>4</sup> Liveness Enforcing Supervisory Control

<sup>5</sup> Legal States

سه معیار مهم برای ارزیابی یک سیاست کنترل نظارتی اعمال کننده زنده بودن مطرح است:

پیچیدگی ساختاری، پیچیدگی محاسباتی و بیشینه مجازیت [۶۲].

معیار پیچیدگی ساختاری، به تعداد مکان‌های کنترلی و کمان‌های رابط آنها که به سیستم کنترل شونده اضافه شده‌اند بستگی دارد. کاهش پیچیدگی ساختاری، منجر به کاهش هزینه‌های سخت‌افزاری، نرم‌افزاری و ارتباطات کنترلی بین سوپروایزر سیستم کنترل شونده خواهد شد.

در یک الگوریتم سوپروایزری با پیچیدگی محاسباتی پایین، مسئله محاسبه کنترل نظارتی از نظر محاسباتی یک مسئله رام شدنی است و می‌تواند در سیستم‌های دنیای واقعی که بزرگ-مقیاس هستند مورد استفاده قرار گیرد. استفاده از سوپروایزر بیشینه مجازیت، موجب پیشگیری از وقوع نشانه-گذاری‌های ممنوع<sup>۱</sup> شده و تمامی نشانه‌گذاری‌های زنده را حفظ می‌نماید. بنابراین، استفاده از سوپروایزر بیشینه مجازیت منجر به بهره‌وری بالای منابع سیستمی خواهد شد.

## ۲-۶ جمع بندی

در این فصل به ارائه مباحث نظری مرتبط با موضوع رساله پرداختیم. در ابتدا یک معرفی کلی از سیستم‌های گسسته رخداد بیان شد. سپس با معرفی روش‌های مدل‌سازی، به اهمیت استفاده از روش‌های مدل‌سازی فرمال و اهمیت شبکه‌های پتری در مقایسه با سایر فرمالیزم‌ها پرداختیم. در ادامه، به معرفی رسمی شبکه‌های پتری، انواع آنها و روش‌های مختلف تحلیل مدل‌های مبتنی بر این شبکه‌ها پرداختیم. در نهایت نظریه کنترل نظارتی بعنوان یکی از روش‌های مهم در محدود کردن رفتار سیستم در دامنه رفتار قانونی مورد انتظار، به اجمال مطرح شد.

---

<sup>۱</sup> Forbidden Markings

## فصل ۳: مروری تحلیلی بر منابع

## ۳-۱ مقدمه

یکی از مهمترین ویژگی‌هایی که برقراری آن در سیستم‌های گسسته رخداد مورد تحلیل قرار می‌گیرد، زنده بودن است که نبودن بن‌بست جزئی یا کلی را نشان می‌دهد. کافمن و همکاران [۶۳] چهار شرط انحصار متقابل، قبضه انحصاری، نگه داشتن و انتظار<sup>۱</sup> و انتظار چرخشی<sup>۲</sup> را برای رویداد بن‌بست در یک سیستم تخصیص منابع معرفی نموده‌اند. سه شرط اول به ویژگی‌های فیزیکی یک سیستم و منابع آن بستگی دارد، در حالیکه شرط آخر توسط درخواست تخصیص و آزاد شدن منابع سیستم ایجاد می‌شود و قابل کنترل است. سه استراتژی تشخیص بن‌بست و بازیابی<sup>۳</sup> [۶۴]، اجتناب از بن‌بست [۶۵] [۶۶] و پیشگیری از بن‌بست [۶۷] برای نقض شرط آخر ارائه شده‌اند.

نظریه کنترل نظارتی که برای اولین بار توسط رمج و ونهام [۲۵] و بر اساس زبان اتوماتا بیان شد، یک روش جامع و ساختارمند برای مدل کردن و پیشگیری از بن‌بست در سیستم‌های وقایع گسسته است. این روش، چندی بعد در تحقیقاتی نظیر [۴۲، ۶۸] در شبکه‌های پتری به کار گرفته شد. نظریه کنترل نظارتی برای اجتناب از بن‌بست، بصورت برخط<sup>۴</sup>، و یا برای پیشگیری از بن‌بست، به صورت برون خط<sup>۵</sup>، قابل اعمال است. روش‌های اجتناب از بن‌بست مبتنی بر اتوماتا، با استفاده از یک پیش‌بین یک مرحله‌ای<sup>۶</sup> به ارائه سوپروایزر برخط می‌پردازند. بدلیل ساختار فشرده شبکه‌های پتری، در این فرمالیزم رویکردهای کنترل نظارتی اجتناب از بن‌بست، بسیار کم استفاده شده است. بعنوان مثال، در [۶۵] از کنترل نظارتی، برای اجتناب از بن‌بست در شبکه‌های پتری S\*PUR استفاده شده است. این شبکه‌ها تعمیمی از شبکه‌های پتری S\*PR هستند که در آنها امکان رخداد خطا در برخی از

<sup>1</sup> Hold and Wait

<sup>2</sup> Circular Wait

<sup>3</sup> Deadlock Detection and Recovery

<sup>4</sup> Online

<sup>5</sup> Offline

<sup>6</sup> One-Step Look-A-Head

نشانه‌های مکان‌های منبع غیر مطمئن وجود دارد. رولیوتیس [۶۹، ۷۰] به ارائه یک مرور جامع از روش‌های بلادرنگ مدیریت تخصیص منابع با استفاده از فرمالیزم‌های متفاوت و مسائل باز این حوزه پرداخته است.

چهار رویکرد کلی برای ارائه یک سوپروایزر جهت اعمال زنده بودن و پیشگیری از بن‌بست در شبکه‌های پتری مطرح شده است: رویکرد زبانی<sup>۱</sup> [۷۱]، رویکرد مبتنی بر تحلیل فضای حالت و گراف دسترسی [۷۲، ۷۳]، رویکرد مبتنی بر تحلیل‌های ساختاری<sup>۲</sup> [۷۴] و رویکرد مبتنی بر شباهت<sup>۳</sup> [۷۵].

در رویکرد زبانی [۷۱]، رخدادهای سیستم بعنوان الفبا در نظر گرفته می‌شود. توالی رخدادها در تناظر با یک رشته از زبان قرار دارند. با بررسی این توالی‌ها، از وقوع توالی‌هایی که منجر به تولید رشته‌هایی می‌شوند که ضوابط خاص مسئله را برآورده نمی‌سازند، جلوگیری می‌شود. این رویکرد بیشتر برای کنترل نظارتی در اتوماتا استفاده می‌شود. در شبکه‌های پتری، بدلیل ساختار فشرده‌تری که نسبت به اتوماتا دارند، استفاده از این روش، بسیار پیچیده بوده و محدود به برخی از تحقیقات است.

در رویکرد دوم [۷۲، ۷۶]، گراف دسترسی، رفتار سیستم را بطور کامل منعکس می‌کند. هر چند محاسبات این روش پرهزینه است؛ همیشه یک سوپروایزر اعمال کننده زنده بودن با بیشینه مجازیت می‌تواند بدست آید.

---

<sup>1</sup> Language Based

<sup>2</sup> Structural Analysis-Based Approach

<sup>3</sup> Similarity-Based Approach

در رویکرد ساختاری [۴۳، ۷۷]، رابطه خواص ساختاری و خواص رفتاری یک شبکه پتری کشف و مورد استفاده قرار می‌گیرد. این رویکرد، با استفاده از اشیای ساختاری خاصی از شبکه‌های پتری مانند سایفون‌ها و مدارهای منبع-گذر<sup>۱</sup> به یک سیاست پیشگیری از بن‌بست دست می‌یابد. این روش به یافتن محدودیت‌های انحصار متقابل تعمیم یافته<sup>۲</sup> (GMECs) می‌پردازد که منجر به اضافه شدن مکان‌های کنترلی جدید به سیستم خواهد شد. رویکرد ساختاری، نسبت به رویکرد مبتنی بر تحلیل فضای حالت، هزینه محاسباتی کمتری دارد اما الزاما یک سوپروایزر بیشینه مجازیت را نتیجه نمی‌دهد.

رویکرد مبتنی بر شباهت [۷۵]، به محاسبه کامل یک سوپروایزر نمی‌پردازد؛ بلکه از محاسبات انجام شده برای شبکه‌های پتری دیگر و یا محاسبات انجام گرفته برای شبکه پتری مورد بررسی و با ضوابط دیگر، برای محاسبه کنترل نظارتی یک شبکه پتری استفاده می‌کند. یافتن شباهت بین دو شبکه پتری و استفاده از محاسبات کنترل نظارتی یک شبکه پتری در یافتن مکان‌های کنترلی شبکه پتری مشابه از جمله این رویکردها است. از دیگر روش‌های مبتنی بر رویکرد شباهت محور، می‌توان به روش‌هایی اشاره کرد که به محاسبه تغییر جزئی کنترل نظارتی در هنگام تغییرات جزئی در ضوابط می‌پردازند.

از منظر دیگر، روش‌های پیشگیری از بن‌بست با استفاده از کنترل نظارتی را می‌توان به روش‌های متمرکز و روش‌های توزیع شده تقسیم‌بندی کرد. در روش‌های توزیع شده، از نتایج یک سیاست کنترل سوپرایزری متمرکز برای اهدافی مانند کاهش هزینه ارتباطی، بالابردن امنیت و کاهش بار محاسباتی سیستم استفاده می‌شود. هر یک از این روش‌ها به نوبه خود قابل تقسیم به روش‌های

---

<sup>1</sup> Resource-Transition Circuits

<sup>2</sup> Generalized Mutual Exclusion Constraints

مختلفی هستند که تحقیقات مختلفی روی آن‌ها انجام شده است. در این فصل به مطالعه روش‌های مختلف کنترل نظارتی برای پیشگیری از بن‌بست در شبکه‌های پتری می‌پردازیم.

## ۲-۳ معادلات محدودیت و رویکردهای کنترل نظارتی متمرکز

همانطور که در مقدمه نیز اشاره شد، چهار رویکرد کلی رویکرد زبانی<sup>۱</sup> [۷۱]، رویکرد مبتنی بر تحلیل فضای حالت و گراف دسترسی [۷۲، ۷۳]، رویکرد مبتنی بر تحلیل‌های ساختاری<sup>۲</sup> [۷۴] و رویکرد مبتنی بر شباهت<sup>۳</sup> [۷۵] برای طراحی کنترل نظارتی جهت پیشگیری از بن‌بست در شبکه‌های پتری مطرح شده است. هر یک از این رویکردها به انواع مختلفی از معادلات محدودیت و زبان محدودیت نیاز دارند. در حالت کلی، سه نوع معادله محدودیت بر روی یک شبکه پتری قابل تعریف است. محدودیت‌های زبانی، محدودیت‌های نشانه‌گذاری‌های ممنوع<sup>۴</sup> و محدودیت‌های انحصار متقابل تعمیم یافته. محدودیت‌های زبانی [۷۱]، دنباله رخداد‌های یک سیستم را به یک زبان خاص محدود می‌سازد. محدودیت‌های نشانه‌گذاری‌های ممنوع [۷۸] به طور واضح به ارائه یک لیست از نشانه‌گذاری‌های غیرقانونی<sup>۵</sup> می‌پردازد. این نوع محدودیت‌ها بیشتر در رویکرد سوپروایزری مبتنی بر تحلیل گراف دسترسی کاربرد دارند. محدودیت‌های انحصار متقابل برای اولین بار توسط گیوا و همکاران [۷۹] مطرح شدند. این نامعادلات، روشی را برای بیان محدودیت‌های استفاده همزمان از منابع اشتراکی در میان فرآیندهای مختلف یک شبکه پتری مبتنی بر سیستم‌های تولیدی ارائه می‌دهند. این محدودیت‌ها در حل یک مسئله کنترل نظارتی مبتنی بر تحلیل گراف دسترسی [۷۲] و نیز حل یک مسئله کنترل نظارتی مبتنی بر رویکرد ساختاری [۷۴] مورد استفاده قرار گرفته‌اند.

---

<sup>1</sup> Language Based

<sup>2</sup> Structural Analysis-Based Approach

<sup>3</sup> Similarity-Based Approach

<sup>4</sup> Forbidden Markings Constraints

<sup>5</sup> Illegal Markings

بطور کلی، GMEC ها نامعادلات خطی برای محدود نمودن جمع وزنی نشانه‌های مربوط به مجموعه‌ای از مکان‌ها می‌باشند. با استفاده از یک GMEC، نشانه‌گذاری‌های قانونی به شکل معادله ۳-۱ مطرح می‌شوند:

$$\mu(\vec{l}, b) = \{M \in \mathbb{N}^{|\mathcal{P}|} \mid \vec{l}^T \cdot M \leq b\} \quad (3-1)$$

که  $\mu(l, b)$  نشانه‌گذاری‌های قانونی را نشان می‌دهد.  $l$  بردار وزنی از اعداد غیر منفی و  $b$  نشان‌دهنده یک عدد غیر منفی است. البته می‌توان بطور همزمان چندین معادله GMEC را بیان نمود:

$$\mu(L, \beta) = \bigcap_{i=1}^m \mu(\vec{l}_i, b_i) = \{M \in \mathbb{N}^{|\mathcal{P}|} \mid L^T \cdot M \leq \beta\} \quad (3-2)$$

که  $L$  ماتریسی است که هر سطر آن نشان دهنده بردار وزنی مربوط به یک محدودیت است. بردار ستونی  $\beta$  نیز نشان دهنده محدودیت‌ها وزنی تجمیعی در هر یک از GMEC ها است. در [۷۹] اثبات شده است که برای اعمال کنترل روی نشانه‌گذاری ممنوع  $M$  در یک شبکه پتری امن می‌توان از نامعادله (۳-۳) بعنوان نامعادله خطی مرتبط استفاده نمود:

$$\sum_{k=1}^n M(k) \leq n - 1 \quad (3-3)$$

که  $n$  نشان دهنده تعداد مکان‌های دارای نشانه در نشانه‌گذاری ممنوع  $M$  است.

یامالیدو و همکاران [۸۰] نشان دادند برای حل یک مجموعه GMEC در یک شبکه پتری معمولی، می‌توان با اضافه کردن مجموعه‌ای از مکان‌های کنترلی به شبکه پتری، یک سیستم حلقه بسته را بدست آورد که مجموعه نشانه‌گذاری‌های سیستم را در یک مجموعه قانونی تعریف شده توسط GMEC محدود می‌نماید.



این مجموعه ابتدا با اضافه نمودن یک متغیر جدید به نام معادله خطی (۳-۲) آن را به یک معادله تبدیل نمود:

$$LM + M_c = \beta. \quad (3-4)$$

که برداری  $1 \times n_c$  است که تعداد نشانه‌های مکان‌های کنترلی جدید را برای هر یک از  $n_c$  محدودیت نشان می‌دهد. این معادله نشان دهنده ثابت بودن مقدار  $LM + M_c$  در تمامی نشانه-گذاری‌های قانونی  $M \in \mu(L, \beta)$  است. این نتیجه، نشان دهنده وجود یک P-Invariant در حلقه‌ی بسته کنترلی است؛ یعنی داریم:

$$PI: \alpha. C' = [L \ I] \begin{bmatrix} C \\ C_c \end{bmatrix} = 0 \rightarrow L.C + C_c = 0 \quad (3-5)$$

که  $C'$  ماتریس برخورد شبکه پتری جدید را نشان می‌دهد. با استفاده از معادله (۳-۵) ماتریس برخورد سوپروایزر قابل محاسبه است:

$$C_c = -L.C \quad (3-6)$$

با جایگذاری نشانه‌گذاری‌های اولیه در معادله (۳-۴) داریم:

$$LM_0 + M_{c0} = b \rightarrow M_{c0} = b - LM_0 \quad (3-7)$$

که  $C_c$  و  $M_{c0}$  بترتیب ماتریس برخورد و نشانه‌گذاری اولیه شبکه ورودی هستند.

یامالیدو و همکاران [۸۰] روش‌هایی با نام تبدیل محدودیت<sup>۱</sup> را نیز برای تبدیل محدودیت‌های روی بردارهای اجرا، به محدودیت‌های انحصار متقابل تعمیم یافته بر روی نشانه‌گذاری‌ها معرفی نمودند. همچنین، آنها نشان دادند اگر  $L$  یا  $b$  دارای ضرایب منفی باشند نیز معادلات (۳-۶) و (۳-۷)

---

<sup>۱</sup> Constraint Transformation

قابل بکارگیری است. نونز و سانچز [۸۱] نیز روشی را برای تبدیل محدودیت‌های گزاره‌ای<sup>۱</sup> مبتنی بر رفتار به محدودیت‌های انحصار متقابل تعمیم یافته در شبکه‌های پتری امن بیان نمودند. آنان نشان دادند یک نگاشت بین محدودیت‌های عددی به شکل  $G(\Gamma_a) \rightarrow \Theta(M_b)$  و گزاره  $P(\Gamma_a - m_b \leq$  (۰) در شبکه‌های پتری برقرار است و می‌توان ضوابط گزاره‌ای که در هنگام اعتبارسنجی یک سیستم بیان می‌شود را به ضوابط نامعادلات خطی تبدیل نموده و با استفاده از روش معرفی شده توسط یامالیدو و همکاران [۸۰] حل نمود.

### ۳-۲-۱ رویکرد زبانی

یکی از روش‌های اعمال کنترل نظارتی، استفاده از زبان تعریف شده برای شبکه پتری مربوطه است. اساس رویکرد زبانی در شبکه‌های پتری، تخصیص برچسب به گذرها و تولید شدن برچسب مربوط به هر گذر در هنگام اجرا شدن آن است [۸۲]. اگر بتوانیم زبانی برای سیستم و زبانی برای محدودیت‌ها تعریف نماییم، می‌توان از این روش در ساخت سوپروایزر استفاده نمود. بدین ترتیب، دنباله اجرای گذرها، یک زبان شبکه پتری را نشان می‌دهد. برای تعریف زبان یک شبکه پتری به چهار چیز نیاز است:  $(N, \sigma, M_0, M_f)$ ، که  $N = (P, T, W)$  ساختار شبکه پتری را نشان می‌دهد،  $M_0$  و  $M_f$  بترتیب حالت شروع و حالات پایانی هستند.  $\sigma$  نیز تابع برچسب‌دهی است که به هر گذر یک برچسب را نسبت می‌دهد.

در [۷۱] به کنترل نظارتی شبکه‌های پتری با استفاده از رویکرد زبانی پرداخته است. برای این کار ابتدا با داشتن شبکه پتری کارخانه و شبکه پتری مربوط به ضوابط، ترکیب همروند آن‌ها محاسبه می‌شود. سپس در صورتی که سوپروایزر بدست آمده دارای حالات بن‌بست باشد، به اصلاح و پالایش

<sup>۱</sup> Propositional Constraints

حالات نامطلوب می‌پرازد. در مرحله اصلاح، برای هر گذر خطرناک  $t$  با برچسب  $e$  نشانه‌گذاری‌هایی که  $t$  را فعال می‌سازند محاسبه می‌شوند. این نشانه‌گذاری‌ها به دو دسته نشانه‌گذاری‌های مجاز ( $M_a$ ) و نشانه‌گذاری‌های غیرمجاز ( $M_u$ ) تقسیم می‌شوند. اگر  $M_a = \emptyset$  باشد، گذر  $t$  حذف خواهد شد. در غیراینصورت، برای هریک از نشانه‌گذاری‌های مجاز قوانینی به صورت نامعادلات برای هر یک از مکان‌ها بدست خواهد آمد. گذری با ورودی و خروجی مشابه با گذر  $t$ ، بانضمام یک حلقه با وزنی برابر با وزن حد نامعادله برای مکان‌های نشانه‌دار نشانه‌گذاری مجاز، جایگزین گذر  $t$  خواهد شد.

### ۳-۲-۲ رویکرد تحلیل گراف دسترسی

در رویکرد مبتنی بر تحلیل فضای حالت، گراف دسترسی، رفتار سیستم را بطور کامل منعکس می‌کند. هر چند محاسبات این روش بسیار پرهزینه است؛ همیشه یک سوپروایزر اعمال کننده زنده بودن با بیشینه مجازیت می‌تواند بدست آید. در [۸۳] یک روش ساده برای پیشگیری از بن‌بست بر اساس حذف حالات بن‌بست از گراف دسترسی شبکه‌های پتری پیشنهاد شده است. البته، روش پیاده‌سازی این سیاست در [۸۳] بیان نشده است. اوزام [۸۴] برای جلوگیری از انفجار فضای حالت در این رویکرد از تکنیک‌های کاهش مطرح شده در شبکه‌های پتری استفاده نموده است. این تکنیک‌ها حافظ ویژگی‌هایی همچون کرانداری، زنده بودن و بازگشت‌پذیری هستند.

یکی از نظریه‌های مطرح در کنترل نظارتی مبتنی بر تحلیل فضای دسترسی، استفاده از نظریه نواحی<sup>۱</sup> است. نظریه نواحی، در سال ۱۹۹۵ توسط بادول و همکاران [۸۵] برای کنترل نظارتی مدل-

---

<sup>۱</sup> Theory of Regions

های مبتنی بر اتوماتا مطرح شد. چندی بعد، افرادی همچون اوزام [۸۶] و غفاری و همکاران [۷۶] از این روش برای کنترل نظارتی مدل‌های مبتنی بر شبکه‌های پتری استفاده کردند.

اوزام [۸۶] با استفاده از نظریه نواحی به کنترل نظارتی یک شبکه پتری پرداخت. در این روش، ابتدا گراف دسترسی به دو ناحیه با نام‌های منطقه بن‌بست<sup>۱</sup> (DZ) (یا منطقه مرده<sup>۲</sup> یا منطقه ممنوع<sup>۳</sup>) و منطقه بدون بن‌بست<sup>۴</sup> (DFZ) (یا منطقه زنده<sup>۵</sup> (LZ) یا منطقه قانونی<sup>۶</sup>) افزاز می‌شود. منطقه بن‌بست شامل حالات بن‌بست، حالات بن‌بست جزئی و حالاتی است که ناگزیر به بن‌بست می‌انجامند. استفاده از نظریه نواحی منجر به افزودن مکان‌های کنترلی جدیدی به شبکه پتری می‌شود که از وقوع نشانه‌گذاری‌های درون DZ پیشگیری نموده و تمامی نشانه‌گذاری‌های درون DFZ را حفظ خواهند کرد. بنابراین این سیاست منجر به یک سوپروایزر پیشینه مجازیت خواهد شد. روش معرفی شده در [۸۶]، از انفجار فضای حالت در هنگام تحلیل گراف دسترسی مربوط به شبکه‌های پتری بزرگ-مقیاس رنج می‌برد. در [۸۴]، از تکنیک‌های کاهش مطرح در شبکه‌های پتری [۱۲، ۸۲] برای کنترل نظارتی مبتنی بر نظریه نواحی در سیستم‌های بزرگ-مقیاس استفاده کرده است.

مقاله غفاری [۷۶]، نمونه دیگری از تحقیقاتی است که از نظریه نواحی برای طراحی کنترل نظارتی مبتنی بر گراف دسترسی استفاده کرده است که به طراحی کنترل نظارتی اعمال کننده زنده بودن برای شبکه‌های پتری عمومی می‌پردازد. غفاری [۷۶] برای این منظور، گراف دسترسی یک شبکه پتری را به چهار بخش تقسیم می‌کند:

---

<sup>1</sup> Deadlock Zone

<sup>2</sup> Dead Zone

<sup>3</sup> Forbidden Zone

<sup>4</sup> Deadlock-Free Zone

<sup>5</sup> Live Zone

<sup>6</sup> Authorized Zone

- منطقه زنده شامل نشانه‌گذاری‌های خوب<sup>1</sup> (نشانه‌گذاری‌هایی که نشانه‌گذاری اولیه از آنها و از تمامی مابعدهای آنها دسترس‌پذیر است) و نشانه‌گذاری‌های خطرناک (نشانه‌گذاری‌هایی که به نشانه‌گذاری اولیه دسترسی دارند اما حداقل یکی از مابعدهای آنها به نشانه‌گذاری اولیه دسترسی ندارد).

- منطقه مرده شامل نشانه‌گذاری‌های بن‌بست<sup>2</sup> (نشانه‌گذاری‌هایی که هیچ مابعدی ندارند) و نشانه‌گذاری‌های بد (نشانه‌گذاری‌هایی که مابعد دارند، اما هیچیک از مابعدهای آنها به نشانه‌گذاری اولیه دسترسی ندارند).

نشانه‌گذاری‌های متعلق به منطقه زنده را نشانه‌گذاری‌های قانونی، و نشانه‌گذاری‌های متعلق به منطقه مرده را نشانه‌گذاری‌های غیرقانونی می‌نامیم. نشانه‌گذاری‌های غیرقانونی که مابعد مستقیم یک نشانه‌گذاری قانونی هستند را نشانه‌گذاری‌های غیرقانونی مرزی و یا نشانه‌گذاری‌های غیرقانونی اولین بار دیده شده می‌نامیم. نشانه‌گذاری‌های قانونی و نشانه‌گذاری‌های غیرقانونی مرزی بدین ترتیب قابل محاسبه هستند:

$$M_L = R(N, M_0) \cap R(-N, M_0) \quad (3-8)$$

$$M_{FBM} = R(N, M_0) - M_L \quad (3-9)$$

این روش ابتدا به یافتن موارد جداکننده نشانه‌گذاری / گذر<sup>3</sup> (MTSI) با استفاده از تحلیل مدل کارخانه می‌پردازد. یک مورد جداکننده نشانه‌گذاری / گذر یک جفت از نشانه‌گذاری قانونی  $M$  و گذر  $t$  است که در  $M$  فعال است و بصورت  $(M, t)$  نشان داده می‌شود. اگر  $M[t]M'$  درست باشد،  $M'$  در  $DZ$  قرار دارد. بنابراین، هنگامی که  $t$  در  $M$  اجرا می‌شود، سیستم از منطقه زنده  $LZ$  وارد  $DZ$  خواهد شد.

<sup>1</sup> Good Markings

<sup>2</sup> Deadlock Markings

<sup>3</sup> Marking/Transition Separation Instances

در این مورد  $M$  و  $t$  بترتیب نشانه‌گذاری خطرناک و گذر خطرناک نامیده می‌شوند. مجموعه MTSI ها،  $\Omega$  بصورت زیر تعریف می‌شود:

$$\Omega = \{(M, t) | M[t]M', M \in LZ, M' \in DZ\} \quad (3-10)$$

هنگامی که تمام MTSI های یک مدل شبکه پتری استخراج شدند، می‌توان یک سوپروایزر برای جدا کردن DZ از LZ با استفاده از منع اجرای گذر  $t$  در نشانه‌گذاری متناظر  $M$  برای هر MTSI طراحی کرد.

برای محاسبه بردار برخورد و نشانه‌گذاری اولیه مکان کنترلی  $p_c$ ، سه معادله دسترس‌پذیری<sup>۱</sup>، معادله چرخه<sup>۲</sup> و معادله جداسازی<sup>۳</sup> مطرح شده و با استفاده از برنامه‌نویسی خطی حل می‌شوند. این معادله، طراحی سوپروایزر بهینه با استفاده از نظریه نواحی<sup>۴</sup> (OSDTR) نامیده می‌شود.

۱- معادله دسترس‌پذیری: به منظور کنترل بهینه، هر نشانه‌گذاری  $M_l$  عضو  $M_L$  باید پس اضافه کردن  $P_c$  نیز قابل دسترسی باشد؛ یعنی:

$$M(p_c) = M_0(p_c) + N(p_c, \cdot) \overrightarrow{\Gamma_{M_l}} \geq 0, \forall M_l \in M_L \quad (3-11)$$

که در آن  $\Gamma_{M_l}$  هر مسیر در LZ از  $M_0$  به  $M_l$  است.  $\Gamma_{M_l}$  یک  $T$ -بردار است و  $\overrightarrow{\Gamma_{M_l}(t)}$  جمع جبری تمام موارد بروز  $T$  در  $\Gamma_{M_l}$  را نشان می‌دهد.  $\vec{\Gamma}$  بردار شمارش  $\Gamma$  نامیده می‌شود.

۲- معادله چرخه: هر مانیتور  $p_c$  باید معادله چرخه را برای هر مدار در LZ برآورده سازد؛ یعنی:

<sup>1</sup> Reachability Equation

<sup>2</sup> Circuit Equation

<sup>3</sup> Separation Equation

<sup>4</sup> Optimal Supervisor Design by the Theory of Regions

$$N(p_c, \cdot) \cdot \vec{\gamma} = 0, \forall \gamma \in \mathcal{CL}. \quad (3-$$

12)

که در آن  $\gamma$  هر مدار از LZ است.  $\vec{\gamma}$  یک  $T$  - بردار  $M \times 1$  است و  $\vec{\gamma}(t)$  نشان‌دهنده جمع جبری تمام موارد بروز  $t$  در  $\gamma$  است.  $\mathcal{C}_L$  نیز مجموعه تمام مدارهای موجود در LZ را نشان می‌دهد.

۳- معادله جداسازی: برای  $(M, t)$ ، مانیتور  $p_c$  باید از اجرای گذر  $t$  در  $M$  ممانعت بعمل آورد.

بنابراین، هر مانیتور  $p_c$  باید حداقل یک MTSI بصورت  $(M, t)$  در  $\Omega$  را حل کند، یعنی:

$$M(p_c) + N(p_c, \cdot) \vec{\Gamma}_M + N(p_c, t) \leq -1 \quad (3-$$

13)

به منظور کاهش هزینه محاسباتی نظریه نواحی، رزیگ و همکاران [۷۲] به یافتن MTSI‌های تکراری<sup>۱</sup>، با استفاده از مفهوم برش کمینه در شبکه‌های پتری پرداخته‌اند. لی و همکاران [۸۷] نیز از رویکرد کنترل سایفون‌ها، قبل از استفاده از نظریه نواحی و برای کاهش محاسبات آن استفاده کرده‌اند.

روش‌های مبتنی بر گراف دسترسی که تاکنون مطرح شد در مورد شبکه‌های پتری بدون حلقه کاربرد دارند. در مورد شبکه‌های پتری مبتنی بر سیستم‌های تولیدی دارای حلقه، چن و همکاران [۲۱] روشی را برای طراحی کنترل نظارتی ارائه داده‌اند. در این روش، ابتدا با داشتن کمینه نشانه-گذاری‌های قانونی تحت پوشش، به محاسبه جداکننده‌های نشانه‌گذاری/گذر بهینه پرداختند. سپس با استفاده از طراحی یک معادله ILP، به حل هر یک از این جداکننده‌ها با استفاده از این معادله پرداختند.

چن و همکاران [۸۸] با استفاده از مفهوم کمان بازدارنده داده<sup>۱</sup>، به محاسبه کنترل نظارتی پرداخته‌اند. این کمان دارای برچسب‌های عددی داده‌ای است که منجر به اجرا نشدن یک گذر فعال،

---

<sup>۱</sup> Redundant

در هنگامی که مکان‌های رابط دارای نشانه‌هایی همانند برچسب‌های کمان هستند خواهد شد. فرض کنیم  $\{a\}$  یک برچسب داده‌ای برای یک کمان بازدارنده  $(p, t)$  باشد و گذرهای خوب<sup>۲</sup> در نشانه-گذاری‌های قانونی، عضو  $MTSI(M, \cdot)$  نباشند. این کمان نباید منجر به کاهش یک گذر فعال خوب در یک نشانه‌گذاری قانونی شود. بنابراین در یک نشانه‌گذاری قانونی  $M(p) \neq a$  یعنی،  $M(p) \leq a - 1$  یا  $M(p) \geq a + 1$  است. برای تبدیل این دو محدودیت به یک محدودیت قابل استفاده در تمام نشانه‌گذاری‌های قانونی، از متغیر  $g \in \{0, 1\}$  استفاده می‌شود که نشان دهنده کوچکتر بودن نشانه‌گذاری  $M(p)$  از  $a - 1$  ( $g = 0$ ) و یا بزرگتر بودن نشانه‌گذاری  $M(p)$  از  $a + 1$  ( $g = 1$ ) هستند.  $O$  نیز یک عدد بزرگ است.

$$\beta - \sum_{i=1}^n l_i \cdot M_k(p_i) \leq O \cdot g_{k,j,q} + a_{j,q} - 1, \forall k \in N_{\varepsilon_{tq}}, \forall j \in N_{D_{tq}}, \forall t_q \in T_C,$$

$$\beta - \sum_{i=1}^n l_i \cdot M_k(p_i) \geq -O \cdot (1 - g_{k,j,q}) + a_{j,q} + 1, \forall k \in N_{\varepsilon_{tq}}, \forall j \in N_{D_{tq}}, \forall t_q \in T_C$$

(3-

14)

معادله سومی نیز برای برابر بودن تعداد نشانه‌های مکان  $p$  با  $\{a\}$  در هنگام داشتن یک گذر

خطرناک معرفی شده است:

$$\beta - \sum_{i=1}^n l_i \cdot M_k(p_i) = a_{j,q}, \forall (M_k, t_q) \in \Omega$$

(3-

15)

---

<sup>1</sup> Data Inhibitor Arcs

<sup>2</sup> Good Transitions



که  $\varepsilon_{tq}$  مجموعه نشانه‌گذاری‌های قانونی است که در آنها  $t_q$  فعال است. با استفاده از این معادلات یک ILP طراحی و حل می‌شود. بدین ترتیب  $\beta$  و  $L$  محاسبه می‌شوند و با جایگذاری در معادلات (۳-۶) و (۳-۷)، نشانه‌گذاری اولیه و بردار برخورد مکان‌های کنترلی بدست خواهند آمد.

چن و لی [۸۹] از مفهوم تقریباً مشابهی بنام کمان‌های بازدارنده بازه‌ای<sup>۱</sup> برای کنترل حالات بن‌بست استفاده نمودند. کمان‌های بازدارنده بازه‌ای، کمان‌های بازدارنده‌ای با برچسب بازه‌ای  $[a..b]$  هستند که از وقوع یک گذر فعال، در صورتی که مکان متناظر دارای نشانه در بازه کمان بازدارنده باشد، جلوگیری می‌کنند.

در [۸۹] از روشی برای حذف مکان‌های کنترلی تکراری نیز استفاده شده است. برای اینکار از سه متغیر دودویی جدید با نام‌های  $d, f$  و  $h$  استفاده شده است که  $f$  و  $d$  بترتیب بیانگر صدق کردن یک نشانه‌گذاری ممنوع و  $h$  نیز تجمیع این دو را نشان می‌دهد. معادلات (۳-۱۶) و (۳-۱۷) بترتیب بیانگر پیشگیری نشدن از یک FBM با استفاده از مانیتور فعلی و پیشگیری شدن از آن با استفاده از مانیتور فعلی هستند. تابع هدف برنامه ILP نیز بیشینه شدن  $h$  را در نظر دارد.

$$\beta - \sum_{i=1}^n l_i \cdot M_k(p_i) \leq \sum_{i=1}^n l_i \cdot C(p_i, t_q) - 1 \quad (3-16)$$

$$a \leq \beta - \sum_{i=1}^n l_i \cdot M_k(p_i) \leq b \quad (3-17)$$

در هر بار تکرار الگوریتم، آن دسته از FBM‌ها که دارای  $h_j = 1$  هستند، حذف شده، و الگوریتم تا تهی شدن این مجموعه، به تکرار حل معادلات ادامه می‌دهد.

---

<sup>1</sup> Interval Inhibitor Arcs

روش‌هایی که تاکنون و براساس رویکرد مبتنی بر تحلیل فضای دسترسی مطرح شدند، تنها در مورد ضوابط پویا و بطور خاص برای اعمال زنده بودن کاربرد دارند. باسیل و همکاران [۶۱] به ارائه یک الگوریتم تکرار شونده برای بدست آوردن کنترل نظارتی مبتنی بر شبکه‌های پتری و تضمین برقرار بودن ضوابط ایستا و ضوابط رفتاری پرداختند. الگوریتم ۳-۱، روند این روش را نشان می‌دهد.

الگوریتم ۳-۱- روش تکرار شونده<sup>۱</sup> تضمین ضوابط ایستا و پویا [۶۱]

۱- گراف دسترسی تقطیع شده  $RG'$  را بدست آورید. برای این کار با شروع از نشانه‌گذاری اولیه، بطور تکرار شونده نشانه‌گذاری‌های مابعدی ایجاد می‌شوند که ضوابط ایستا را تضمین می‌کنند و به گراف اضافه می‌شوند.

۲- بر حسب خواص رفتاری که لازم است برقرار باشند، بطور تکرار شونده به هرس کردن گراف  $G'$  پرداز:

۲-۱- تمامی حالاتی که به یک جزء قویا همبند با کاردینالیته<sup>۲</sup> ۱ تعلق دارند، را هرس کنید (این اجزا نشان دهنده حالات بن‌بست هستند).

۲-۲- برای تضمین زنده بودن، اجزای قویا همبندی را که گذرهای درون آنها مساوی با تمامی گذرهای شبکه نیستند را هرس کنید.

۲-۳- برای تضمین بازگشت‌پذیری، تمامی اجزای قویا همبندی را که حالت  $M_0$  در آنها نیست را هرس کنید.

۲-۴- برای تضمین کنترل‌پذیری، هر نشانه‌گذاری  $M$  را که در آن یک گذر  $t \in T_u$  فعال بوده و مابعد  $M$ ، عضو گراف هرس شده فعلی نیست را هرس کنید.

---

<sup>1</sup> Iterative  
<sup>2</sup> Cardinality

۳- اگر در گراف  $RG'$  با استفاده از مراحل بیان شده در قسمت ۲، هرسی اتفاق افتاده است، به قسمت ۲ بازگردید. در غیر اینصورت  $G'$  را بعنوان خروجی نمایش دهید.

### ۳-۲-۳ رویکرد ساختاری

محاسبات مربوط به رویکرد مبتنی بر تحلیل فضای دسترسی ناکاراست؛ زیرا نیاز به محاسبه گراف دسترسی یک شبکه دارد، که از مشکل انفجار فضای حالت رنج می‌برد. علاوه بر این، مانیتورهای آن با استفاده از حل تعداد زیادی از نامعادلات بدست می‌آید که از مسئله پیچیدگی محاسباتی رنج می‌برد. رویکرد ساختاری، با استفاده از خواص ساختاری یک سیستم، مانند سایفون‌ها و P-invariantها، به کنترل خواص رفتاری می‌پردازد.

سایفون‌ها نقش اساسی در تحلیل و کنترل سیستم‌های مبتنی بر شبکه‌های پتری دارند. اگر یک سایفون در یک نشانه‌گذاری تهی از نشانه شود، در تمامی نشانه‌گذاری‌های مابعد نیز تهی از نشانه باقی خواهد ماند. بنابراین، کنترل سایفون و پیشگیری از بدون نشانه شدن آن جهت پیشگیری از بن-بست در یک سیستم بسیار مهم است. تحقیقات مختلفی به رویکرد مبتنی بر سایفون برای تشخیص و پیشگیری از بن‌بست در شبکه‌های پتری مبتنی بر سیستم‌های تولیدی پرداخته‌اند. تریکاس و همکاران [۹۰] با طراحی یک ILP به پیشگیری از بن‌بست در شبکه‌های  $S^4PR$  پرداختند. اوزام و همکاران [۹۱] به یک روش برای کنترل نظارتی شبکه‌های پتری  $S^4PR$  پرداختند. این روش، ابتدا به تغییر وزن کمان‌های فرآیندهای یک  $S^4PR$  می‌پردازند. سپس یک مکان جدید به شبکه اضافه می‌شود که گذرهای ورودی و خروجی آن به ترتیب اجتماع گذرهای ورودی و گذرهای خروجی مکان‌های بیکار شبکه است. نشانه‌گذاری اولیه این مکان، کمینه نشانه‌گذاری‌ای است که به ازای آن شبکه غیر زنده خواهد بود. اثبات می‌شود این شبکه، دارای گراف دسترسی معادل با گراف دسترسی اولیه است. برای این شبکه غیر زنده جدید، به یافتن مکان‌های کنترلی اعمال کننده زنده بودن می‌پردازد.

هو و همکاران [۹۲] به تشخیص و پیشگیری از حالات بن‌بست در  $S^3PGR^2$  پرداختند. مارچتی و مونیر-کاردن [۹۳] شرط کافی برای زنده بودن کلاس خاصی از شبکه‌های پتری با نام گراف رخداد وزندار یکه<sup>۱</sup> را بیان نمودند که در زمان چند جمله‌ای قابل اعتبارسنجی است. گراف رخداد وزندار فرمالیزمی قابل تبدیل به شبکه‌های پتری است. یک مسیر را در یک شبکه پتری در نظر بگیرید. اگر حاصل تقسیم تجمیع وزن کمان‌های ورودی مکان‌های درون مسیر بر تجمیع وزن کمان‌های خروجی از مکان‌های درون مسیر برابر یک باشد، به این مدل یک گراف رخداد وزندار یکه گفته می‌شود. اگر حاصل این تقسیم برای تمامی مدارهای درون یک شبکه پتری بیشتر یا برابر با یک باشد، این شبکه زنده است. لیو و بارکاوی [۹۴] به ارائه یک تحقیق جامع از سایفون‌ها، ویژگی‌های آنها، نحوه محاسبه آنها، محاسبه سایفون‌ها برای کلاس‌های خاصی از شبکه‌های پتری و با پیچیدگی چندجمله‌ای، رویکردهای کنترل نظارتی مبتنی بر سایفون و پیشگیری از بن‌بست با استفاده از آن پرداختند.

یک سایفون را می‌توان با استفاده از قرار دادن آن در یک P-Invariant کنترل نمود. بدین ترتیب، تعداد نشانه‌های درون سایفون ثابت مانده و بی‌نشانه نخواهد شد. برای این منظور، یک مکان کنترلی برای ایجاد P-Invariant به شبکه مدل شونده اضافه می‌شود. در [۹۵] یک P-Invariant, I, بگونه‌ای طراحی می‌شود که:

$$I^T \cdot C = 0, I \cdot M_0 > 0, I(p) \leq 0, \forall p \in P \setminus D \quad (3-$$

18)

لی و زو [۹۶] با معرفی مفهوم سایفون‌های مقدماتی<sup>۲</sup> و سایفون‌های غیرمقدماتی، به کاهش تعداد سایفون‌های مورد نیاز برای کنترل پرداختند. سایفون‌های مقدماتی سایفون‌هایی هستند که T-

<sup>1</sup> Unitary Weighted Event Graph

<sup>2</sup> Elementary Siphons

بردار مشخصه آن‌ها مستقل از دیگر سایفون‌ها باشد. T-بردار مشخصه سایفون‌های غیرمقدماتی نیز با استفاده از ترکیب خطی T-بردار مشخصه سایفون‌های مقدماتی قابل محاسبه باشد. آنها نشان دادند با کنترل سایفون‌های کمینه مقدماتی، سایفون‌های غیرمقدماتی نیز کنترل می‌شوند. برای کنترل هر یک از این سایفون‌ها ابتدا یک معادله GMEC تشکیل می‌شود. نشانه‌گذاری‌های قانونی این معادله، تمامی نشانه‌گذاری‌های دسترس‌پذیری هستند که حداقل در یکی از مکان‌های سایفون مربوطه دارای نشانه‌گذاری بزرگتر از 0 باشند. بدین ترتیب نامعادله خطی  $I.M \geq 0$  بدست می‌آید. با ضرب طرفین معادله در -1، معادله به فرم معادله [۷۹] تبدیل می‌شود. که با استفاده از روش معرفی شده در [۸۰] قابل حل است.

روش‌های کنترل نظارتی مبتنی بر سایفون [۹۵، ۹۶] محدودیت‌های بسیاری را بر سیستم کنترل شونده اعمال می‌کنند. سوپروایزر بدست آمده از این روش‌ها معمولاً بیشینه مجازیت نیست. پیرودی و همکاران [۹۷] با معرفی مفهوم سایفون‌های ضروری<sup>۱</sup> و سایفون‌های مغلوب<sup>۲</sup>، تنها به کنترل سایفون‌های ضروری می‌پردازند. سایفون‌های ضروری، مجموعه کمینه‌ای از سایفون‌های کنترل نشده هستند که با کنترل آن‌ها تمامی نشانه‌گذاری‌های غیرقانونی، که در آنها حداقل یک سایفون بدون نشانه وجود دارد، پیشگیری می‌شوند. این رویکرد کنترل سایفون انتخابی، با یک راه حل تکرار شونده پیشرفت می‌کند. در هر تکرار، روابط بین سایفون‌های کنترل نشده و نشانه‌گذاری بحرانی (که در آن حداقل یک سایفون خالی است) شناسایی شده و مجموعه‌ای از سایفون‌هایی که این نشانه‌گذاری‌ها را پوشش می‌دهند انتخاب می‌شوند. اگر سایفون‌های انتخاب شده کنترل شوند، همه نشانه‌گذاری‌های بحرانی نیز ممانعت می‌شوند. بدین ترتیب، تمام سایفون‌های کنترل نشده، کنترل می‌شوند. پیرودی و همکاران [۹۸] به ارائه یک روش بهبود یافته برای جلوگیری از شمارش کامل سایفون‌ها پرداختند که

<sup>۱</sup> Essential Siphons

<sup>۲</sup> Dominated Siphons

تا حدود زیادی منجر به کاهش زمان کلی محاسبات خواهد شد. اگرچه برای تمام مثال‌های ارائه شده توسط پیرودی و همکاران [۹۷, ۹۸] سوپروایزر بیشینه مجازیت وجود دارد، بطور نظری اثبات نشده است که این روش می‌تواند بیشینه مجازیت را به همراه داشته باشد.

در [۹۷, ۹۸] با استفاده از اطلاعات نشانه‌گذاری‌ها، سوپروایزر بهتری از نظر محدود نکردن رفتار قانونی سیستم، نسبت به [۹۵, ۹۶]، بدست آمده است. رویکردهایی که مستقیماً به کنترل سایفون می‌پردازند، الزاماً رویکردهایی تکرار شونده هستند. زیرا پس از افزودن مکان‌های کنترلی متناظر با سایفون‌ها به یک شبکه پتری، ممکن است سایفون‌های کنترل نشده جدیدی معرفی شوند. بنابراین این روال تا زمانی که هیچ سایفون کنترل نشده‌ای در سیستم وجود نداشته باشد ادامه خواهد یافت.

روش دیگری توسط لیو و همکاران [۹۹] برای کاهش ساختار کنترل نظارتی ارائه شده است. آنها به تعریف مفهوم سایفون قابل کنترل پرداختند و بجای سایفون کمینه از مفهوم سایفون قابل کنترل، برای کنترل نظارتی استفاده نمودند. یک سایفون قابل کنترل در شبکه‌های  $S^3PR$  زیرمجموعه‌ای از سایفون‌های اکیدا کمینه<sup>۱</sup> هستند که مجموعه مکمل آنها، مجموعه مکمل تمامی دیگر سایفون‌های اکیدا کمینه را پوشش می‌دهد. تعداد سایفون‌های قابل کنترل در شبکه‌های  $S^3PR$  حداکثر برابر با  $Min(|P_A|, |T|)$  است که  $P_A$  مکان‌های عملیاتی یک شبکه پتری را نشان می‌دهد. لیو و همکاران [۹۹] اثبات کردند با افزودن مکان‌های کنترلی به سایفون‌های قابل کنترل، می‌توان زنده بودن را در سیستم تضمین نمود. کاهش تعداد سایفون‌های قابل کنترل نسبت به سایفون‌های اکیدا کمینه، به داشتن یک کنترل نظارتی با ساختار ساده‌تر می‌انجامد.

---

<sup>۱</sup> Strictly Minimal siphons

دیدبان و علا [۱۰۰, ۱۰۱] روشی را برای کاهش محدودیت‌های محاسبه کنترل نظارتی در شبکه‌های پتری امن ارائه دادند. در [۱۰۱] دیدبان و علا روشی کارآ برای محاسبه کنترل نظارتی در شبکه‌های پتری امن و بر اساس مفهوم حالت مافوق ارائه داده‌اند. یک حالت مافوق بیانگر یک حالت کامل و یا بخشی از آن است. بعنوان مثال در یک شبکه پتری امن با مکان‌های  $p_1$ ,  $p_2$  و  $p_3$ , حالت  $p_1p_2$  یک حالت مافوق برای حالت  $p_1p_2p_3$  است. واضح است که هر حالت مافوق می‌تواند شامل بیش از یک حالت نمونه باشد و برای یک حالت نمونه نیز ممکن است بیش از یک حالت مافوق وجود داشته باشد. در حالت کلی، تعداد حالت‌های مافوق ممنوع، نسبت به حالت‌های ممنوع بسیار کمتر است و با تنظیم معادلات کنترل نظارتی برای این حالات، می‌توان کاهش چشمگیری در تعداد معادلات داشت.

استفاده از روش [۱۰۱] بویژه اگر تعداد محدودیت‌ها زیاد باشد، منجر به کاهش زیادی در تعداد نامعادلات نخواهد شد. البته، این روش بعنوان روشی پایه‌ای در روش‌های مبتنی بر P-Invariant [۱۰۲] [۱۰۳] مورد استفاده قرار گرفته است. دیدبان و علا [۱۰۲] با معرفی مفهوم P-Invariant جزئی<sup>۱</sup>، به راهکاری برای کاهش بیشتر تعداد محدودیت‌های مورد نیاز برای محاسبه یک سوپروایزر اعمال کننده زنده بودن در یک شبکه پتری امن و پایا می‌پردازند. در یک P-Invariant، جمع وزنی زیرمجموعه‌ای از مکان‌ها  $P_i = \{p_1, p_2, \dots, p_r\} \subseteq P$  با ضرایب وزنی  $l_i$  برای مکان‌های مرتبط بصورت معادله (۳-۱۹) است:

$$l_1m_1 + l_2m_2 + \dots + l_rm_r = b \quad (3-$$

19)

در مورد زیر مجموعه‌ای از مکان‌های موجود در P-Invariant  $I$  معادله به صورت (۳-۲۰) است:

---

<sup>1</sup> Partial P-Invariant

$$l_1 m_1 + l_2 m_2 + \dots + l_L m_L \leq b \quad (3-$$

20)

سپس قوانین کاهش دهنده‌ای برای کاهش معادلات محدودیت معرفی شدند. بعنوان مثال: اگر محدودیت‌های  $\{(p_{i1}p_1p_2 \dots p_n, n), (p_{i2}p_1p_2 \dots p_n, n)\}$  برقرار باشند و هیچیک از حالات مافوق  $p_{i1}p_{i2}p_1p_2 \dots p_{n-1}, \dots, p_{i1}p_{i2}p_2 \dots p_n, p_{i1}p_{i2}p_1p_2 \dots p_n$  در مجموعه حالات مافوق قانونی قرار نداشته باشند، می‌توان دو محدودیت را ادغام نمود:

$$m_{i1} + m_{i2} + m_1 + m_2 + \dots + m_r \leq n$$

دیدبان و همکاران [۱۰۳] با استفاده از مفاهیم P-Invariant گوسی جزئی<sup>۱</sup> و P-Invariant شبه گوسی جزئی<sup>۲</sup>، به تعمیم ویژگی‌های معرفی شده در [۱۰۲] برای شبکه‌های پتری امن پرداخته‌اند. اهمیت شبکه‌های پتری امن در قابلیت تبدیل آسان آنها به کدهای کنترلی سخت افزاری همانند نمودار تابع ترتیبی<sup>۳</sup> (SFC) و PLC است [۱۰۴, ۱۰۵].

از دیگر کارهای انجام شده در زمینه کنترل نظارتی با رویکرد ساختاری می‌توان به [۱۰۶] اشاره کرد. اوزام و زو [۱۰۶] یک روش تکرار شونده برای کنترل بن‌بست در سیستم‌های تولیدی ارائه دادند. در هر تکرار، یک مکان کنترلی برای جلوگیری از دسترس‌پذیر بودن یک FBM از طریق ساخت یک PI بدست می‌آید. با اینکه روش ارائه شده در [۱۰۶] به راحتی قابل استفاده است، نمی‌تواند بطور کلی بهینگی را در یک شبکه پتری مبتنی بر سیستم‌های تولیدی تضمین کند. از طرفی، در این روش محدودیت جمع وزنی نشانه‌ها با استفاده از تحلیل‌های دستی و نه بصورت قانون کلی بدست آمده است.

<sup>1</sup> Quasi-Partial Invariant

<sup>2</sup> Semi-Quasi-Partial Invariant

<sup>3</sup> Sequential Function Chart



اوزم و زو [۱۰۷] با استفاده از تکنیک‌های کاهش مطرح در شبکه‌های پتری [۱۲] به ساده-سازی یک شبکه پتری یک سیستم بزرگ-مقیاس، قبل از اعمال کنترل نظارتی می‌پردازند. روش‌های همجوشی مکان‌های سری<sup>۱</sup>، همجوشی مکان‌های موازی<sup>۲</sup>، همجوشی گذرهای سری<sup>۳</sup>، همجوشی گذرهای موازی<sup>۴</sup>، حذف مکان‌های حلقه<sup>۵</sup> و حذف گذرهای حلقه<sup>۶</sup> روش‌های کاهش هستند که مورد استفاده این مرجع قرار گرفته‌اند. آنها بیان کردند در محاسبه نشانه‌گذاری اولیه و گذرهای ورودی و خروجی یک مکان کنترلی، تنها گذرهای ورودی و خروجی مربوط به مکان‌های درون  $PI$  دخیل هستند. بنابراین معادلات (۳-۶) و (۳-۷)، به معادلات (۳-۲۱) و (۳-۲۲) ساده‌سازی شدند:

$$C_c = -L_{PI} \cdot C_{PI} \quad (3-21)$$

$$M_{c0} = b - L_{PI} M_{PI0} \quad (3-22)$$

که  $C_{PI}$  و  $M_{PI0}$  بترتیب نشان‌دهنده تصویر ماتریس برخورد و نشانه‌گذاری اولیه شبکه پتری تحت کنترل در مکان‌های  $PI$  هستند.  $L_{PI}$  نیز بیانگر بردار کاهش یافته‌ای از  $L$  است که در آن وزن-های صفر حذف شده و تنها وزن‌های مربوط به مکان‌های درون  $PI$  مورد احتساب قرار گرفته‌اند.

زارعی و همکاران [۱۰۸] [۱۰۹] به طراحی سوپروایزر بیشینه مجازیت در شبکه‌های پتری پرداختند. در [۱۰۹] با داشتن مجموعه نشانه‌گذاری‌های قانونی و مجموعه نشانه‌گذاری‌های غیرقانونی، شرط دسترس‌پذیری نشانه‌گذاری‌های قانونی، بصورت صدق کردن آنها در تمامی نامعادلات خطی (۳-۲۳) بیان می‌شود:

<sup>1</sup> Fusion of Serial Places

<sup>2</sup> Fusion of Parallel Places

<sup>3</sup> Fusion of Serial Transitions

<sup>4</sup> Fusion of Parallel Transitions

<sup>5</sup> Elimination of Self-Loop Places

<sup>6</sup> Elimination of Self-Loop Transitions

$$\sum_{i=1}^n M(p_i) \cdot L(p_i) \leq \beta \quad (3-$$

23)

شرط دسترس ناپذیر بودن نشانه‌گذاری‌های غیرقانونی بصورت صدق نکردن آنها در حداقل یک نامعادله خطی (۳-۲۴) بیان می‌شود.

$$\sum_{i=1}^n M'(p_i) \cdot L(p_i) > \beta \quad (3-$$

24)

چن و همکاران [۴۸] شرایط ممنوعیت نشانه‌گذاری غیرقانونی و شرط دسترس‌پذیر بودن نشانه‌گذاری‌های قانونی را با معادلات ذیل بدست می‌آورند:

شرط ممنوعیت: یک  $M \in M_{FBM}$ ، توسط یک PI پیشگیری می‌شود اگر:

$$\sum_{i=1}^n l_i M(p_i) = \beta + 1. \quad (3-$$

25)

شرط دسترس‌پذیری:  $M' \in M_L$  دسترس‌پذیر است اگر:

$$\sum_{i=1}^n l_i M'(p_i) \leq \beta. \quad (3-$$

26)

با ترکیب این دو شرط خواهیم داشت:

$$\sum_{i=1}^n l_i (M'(p_i) - M(p_i)) \leq -1. \quad (3-$$

27)

معادله (۳-۲۷) نشان می‌دهد، برای پیشگیری از بن‌بست نیاز به حل  $|M_L| \times |M_{FBM}|$  نامعادله

داریم.

چن و همکاران [۴۸] اثبات کردند مفاهیم پوشش برداری می‌تواند تعداد نامعادلات را کاهش

دهد:

قضیه ۱-۳- دو نشانه‌گذاری  $M, M' \in M_{FBM}$  را در نظر بگیرید که  $M \geq_A M'$ . اگر  $M'$  با استفاده از یک PI پیشگیری شود، نشانه‌گذاری  $M$  نیز پیشگیری خواهد شد.

قضیه ۲-۳- دو نشانه‌گذاری  $M, M' \in M_L$  را در نظر بگیرید که  $M \geq_A M'$ . اگر  $M$  توسط یک PI پیشگیری نشود،  $M'$  نیز پیشگیری نخواهد شد.

قضیه ۳-۳- فرض کنید  $M \in M_{FBM}^*$  و  $M_1, M_2 \in M_L$ . با فرض اینکه  $M_1 \geq_M M_2$ ، اگر یک PI طراحی شود که از  $M$  پیشگیری نموده و از  $M_1$  ممانعت بعمل نیاورد،  $M_2$  نیز ممانعت نخواهد شد [۴۸].

قضایای ۱-۳ و ۲-۳ نشان‌دهنده این هستند که تنها مجموعه کمینه نشانه‌گذاری‌های ممنوع تحت پوشش،  $M_{FBM}^*$ ، و مجموعه کمینه نشانه‌گذاری‌های قانونی پوشش‌دهنده،  $M_L^*$ ، برای محاسبات مکان‌های کنترلی کافی هستند. قضیه ۳-۳ نیز نشان می‌دهد اگر تنها زیرمجموعه‌ای از  $M_L^*$  که مجموعه نشانه‌گذاری‌های  $M$ -مرتبط هستند حفظ شوند تمامی نشانه‌گذاری‌های قانونی حفظ می‌شوند. بنابراین تنها نیاز به بررسی آن‌ها در طراحی یک مکان کنترلی برای هر نشانه‌گذاری متعلق به  $M_{FBM}^*$  داریم.

روش معرفی شده توسط چن و همکاران [۴۸] به کاهش تعداد معادلات سوپروایزر اعمال کننده زنده بودن برای شبکه‌های پتری مبتنی بر سیستم‌های تولیدی می‌پردازد. البته بدلیل استفاده از تعداد زیادی از معادلات ILP پیچیدگی محاسباتی این روش بالاست. از طرفی، معمولاً تعدادی از FBMها می‌توانند با مانیتورهای یکسانی پوشش داده شوند. این روش مانیتورهای یکسان را شناسایی

نکرده و منجر به داشتن یک سوپروایزر با ساختار پیچیده خواهد شد. همچنین الگوریتم ۳-۴ تنها در صورتی که سوپروایزر بیشینه مجازیت وجود داشته باشد، آن را ارائه خواهد داد.

در مقاله دیگری که توسط چن و همکاران ارائه شد [۱۱۰]، آنها به مسئله کاهش محدودیت‌ها و حذف مانیتورهای تکراری پرداخته‌اند. برای این کار، آنها به طراحی یک ILP برای یافتن نشانه-گذاری‌های ممنوع که مانیتور یکسانی دارند پرداختند. این معادلات بطور کلی به حل مسئله کمینه تعداد مکان‌های کنترلی<sup>۱</sup> (MCP) می‌پردازند.

۱- معادله همپوشانی نشانه‌گذاری‌های ممنوع در مکان کنترلی: اگر  $M_k, M_j \in M_{FBM}^*$  باشند و P-Invariant مورد استفاده در  $M_k$  بتواند از وقوع  $M_j$  نیز پیشگیری کند، داریم:

$$\sum_{i \in N_A} l_{j,i} \cdot M_k(p_i) \geq \sum_{i \in N_A} l_{j,i} \cdot M_j(p_i) \quad (3-28)$$

که  $l_{j,i}$  ضرایب PI متناظر با  $M_j$  را برای مکان  $p_i$  نشان می‌دهند. با ساده‌سازی معادله ۳-۲۸ داریم:

$$\sum_{i \in N_A} l_{j,i} \cdot (M_k(p_i) - M_j(p_i)) \geq 0 \quad (3-29)$$

با استفاده از معادله (۳-۲۹) می‌توان معادله‌ای را برای تضمین پیشگیری از هر  $M_k \in M_{FBM}^*$  با استفاده از یک PI طراحی شده برای پیشگیری از  $M_j$  - در صورت امکان - بیان نمود:

$$\sum_{i \in N_A} l_{j,i} \cdot (M_k(p_i) - M_j(p_i)) \geq -0 \cdot (1 - f_{j,k}). \quad (3-30)$$

<sup>۱</sup> Minimal number of Control Places Problem

که  $O$  یک عدد مثبت بسیار بزرگ است و  $f_{j,k}$  یک متغیر دودویی و بیانگر پیشگیری از  $M_k$  توسط  $PI_j$  (در صورت ۱ بودن) یا عدم امکان پیشگیری از  $M_k$  توسط  $PI_j$  (در صورت ۰ بودن) است.

۲- معادله اجتناب از نشانه‌گذاری‌های ممنوع دیگر

$$f_{j,k} \leq h_j \quad (3-$$

31)

که  $h_j$  یک متغیر دودویی است.  $h_j = 1$  نشانگر این است که مکان کنترلی  $PI_j$  برای پیشگیری از حالت ممنوع  $M_j$  در نظر گرفته شده است.  $h_j = 0$  نیز بیانگر این است که مکان کنترلی  $PI_j$  تکراری بوده و حذف می‌شود و  $M_j$  توسط مکان کنترلی متناظر با نشانه‌گذاری ممنوع دیگری پیشگیری خواهد شد. معادله ۳-۳۱ نشان دهنده این است که تنها اگر  $h_j$  مربوط به یک نشانه‌گذاری یک باشد، مکان کنترلی متناظر می‌تواند در اجتناب از نشانه‌گذاری‌های ممنوع دیگر استفاده شود.

۳- معادله اجتناب از وقوع هر یک از نشانه‌گذاری‌های ممنوع  $M_j$

$$h_j + \sum_{k \in N_{FBM}^*, k \neq j} f_{k,j} \geq 1 \quad (3-$$

32)

این معادله بیانگر این است که هر نشانه‌گذاری ممنوع  $M_j$  یا باید بطور مستقل و با استفاده از مکان کنترلی متناظر خود پیشگیری شود ( $h_j = 1$ ) و یا باید توسط مکان کنترلی یک نشانه‌گذاری ممنوع دیگر پیشگیری شده باشد (برای یک  $M_k$ ،  $f_{k,j} = 1$  باشد). تابع هدف نیز کمینه کننده تعداد نشانه‌گذاری‌های ممنوعی است که بصورت مستقل کنترل می‌شوند (نشانه‌گذاری‌های  $M_j$  که  $h_j \neq 0$  است).

روش مطرح شده در [۱۱۰] به یافتن سوپروایزر بیشینه مجازیت می‌پردازد که از نظر ساختاری

نیز بهتر از روش [۴۸] است. البته این روش در مقایسه با روش [۴۸] از مسئله پیچیدگی محاسباتی

رنج می‌برد و تعداد متغیرها و محدودیت‌های استفاده شده در این روش بترتیب  $(|P_A| + |M_{FBM}^*|)$  و  $(|M_L^*| + 2|M_{FBM}^*| - 1)$  هستند.

هوانگ و همکاران [۱۱۱] روش دیگری را برای تشخیص مکان‌های کنترلی تکراری و حذف نشانه‌گذاری‌های ممنوع متناظر آنها در MCPP ارائه دادند. آنها با حذف متغیرهای دودویی استفاده شده جهت مشخص‌سازی تکراری بودن یا نبودن یک نشانه‌گذاری ممنوع، معادلات MCPP را به معادله (۳-۳۳) تغییر دادند.

$$\text{Max } d_{m,n} \cdot M_n^T$$

s. t.

$$d_{l,j} \cdot M_j^T \leq -1, \forall l \in \{1,2,\dots, |M_L^*|\}, j \in \{1,2,\dots, |M_{FBM}^*|\}, (l,j) \neq (m,n) \quad (3-$$

33)

که  $M_j \in M_{FBM}^*$  و  $d_{l,j}$  نیز بردار نشان‌دهنده تفریق  $M_j$  از  $M_l \in M_L^*$  است. همچنین، آنها یک روش الگوریتمیک برای تشخیص نشانه‌گذاری‌های ممنوع تکراری ارائه دادند. استفاده از این روش الگوریتمیک، برای یافتن راه حل یک مسئله نسبت به استفاده از روش مبتنی بر ILP این مزیت را داراست که نیاز به حل مسائل بهینه‌سازی نداشته و راه حل مسئله بصورت قطعی بیان می‌شود، البته معمولاً از نظر سرعت نسبت به روش مبتنی بر ILP نیاز به زمان بیشتری دارد.

چن و همکاران [۱۱۲] معادله دیگری برای طراحی کمینه تعداد P-Semiflow<sup>۱</sup>ها (MPP) ارائه دادند. این روش نیز همانند روش [۱۱۰] به محاسبه کنترل نظارتی با بیشینه مجازیت و کمینه ساختار می‌پردازد. تعداد متغیرها و محدودیت‌های مورد استفاده این روش بترتیب  $(|P_A| + n_l \cdot (|M_L^*| + 2|M_{FBM}^*|) + |M_{FBM}^*|)$  و  $(|M_{FBM}^*| + 2)$  است. که  $n_l$  نشان دهنده کمینه تعداد

<sup>۱</sup> Minimal Number of P-Semiflows Problem

GMEC های لازم است. در واقع  $n_i$  نشان دهنده تعداد نشانه گذاری های غیرقانونی است که مستقلاً کنترل می شوند.

چن و همکاران [۷۴] به طراحی ILP برای اجتناب از تکرار در پیشگیری از نشانه گذاری های ممنوع توسط یک مکان کنترلی پرداخته اند. این مسئله با نام مجموعه کمینه نشانه گذاری های ممنوع<sup>۱</sup> (MFPP) بصورتی است که اگر  $I$  یک P-Invariant طراحی شده باشد، برای نشانه گذاری قانونی  $M'_i$  داریم:

$$\sum_{i \in \mathbb{N}_A} l_i \cdot M'_i \leq \beta \quad (3-34)$$

یک نشانه گذاری ممنوع  $M \in M_{FBM}^*$  با استفاده از  $I$  پیشگیری می شود اگر:

$$\sum_{i \in \mathbb{N}_A} l_i \cdot M_i \geq \beta + 1 \quad (3-35)$$

با تعریف تابع  $f_k$  برای وجود رابطه پیشگیری از  $M_k \in M_{FBM}^*$  توسط  $I$  داریم:

$$\sum_{i \in \mathbb{N}_A} l_i \cdot M_k(p_i) \geq \beta + 1 - O \cdot (1 - f_k) \quad (3-36)$$

نهایتاً، یک معادله ILP بصورت ذیل خواهیم داشت:

$$\text{Max } f = \sum_{i \in \mathbb{N}^*_{FBM}, k \neq j} f_k$$

$$s. t. \quad \text{Equations (3-35)-(3-36)}$$

<sup>1</sup> Maximal Number of Forbidden Markings

در [۷۴] روش دیگری با نام MFPP2 نیز برای کاهش متغیرها و معادلات مربوط به MFPP1 بیان شد که با ترکیب معادلات ۳-۲۱ و ۳-۲۷ بدست آمده است.

کاردن و پیرودی [۱۱۳] الگوریتم شاخه و حد<sup>۱</sup> را برای یافتن کمینه مجموعه GMEC ها برای کنترل نشانه‌گذاری‌های غیرقانونی معرفی کردند. این الگوریتم، با داشتن نشانه‌گذاری‌های قانونی و FBM ها، FBM ها را به  $c$  زیرمجموعه افراز می‌کند ( $c$  از قبل تعیین شده نیست). نهایتاً یک نشانه‌گذاری پیشگیری نمی‌شود (قانونی است)، اگر در تمامی GMEC ها صدق کند. در غیر اینصورت، اجازه داده نمی‌شود.

فرض کنیم  $\Pi = \{\Pi_1 \dots \Pi_c\}$ ، یک افراز روی مجموعه  $M_{FBM}$  باشد که برای آنها مجموعه معادلات GMEC طراحی شده است  $\Sigma = M_{FBM} \setminus \Pi$  نیز مجموعه باقیمانده از نشانه‌گذاری‌های غیرقانونی باشد که هنوز پیشگیری نشده‌اند. تابع  $SingelGMEC(\Pi, j)$  به پیشگیری از نشانه‌گذاری‌های غیرقانونی درون مجموعه  $\Pi_j$  می‌پردازد و به نشانه‌گذاری‌های درون  $\Pi_c \dots \Pi_j + 1$  اجازه اجرا می‌دهد. این تابع به معرفی دو مجموعه معادلات مبتنی بر GMEC می‌پردازد. مجموعه معادلات (۳-۳۷) به نشانه‌گذاری‌های قانونی و نشانه‌گذاری‌های درون  $\Pi_c \dots \Pi_j + 1$  اجازه اجرا می‌دهد. مجموعه معادلات (۳-۳۸) از اجرای نشانه‌گذاری‌های  $\Pi_j$  پیشگیری می‌کند.

singleGMEC :

$$\text{Min } J_j$$

s. t.

$$\sum_{k=1}^n L_{jk} M_{i,k} - b_j \leq 0, \quad i \in \mathbb{N}_{LZ \cup \Pi_{j+1} \cup \dots \cup \Pi_c} \quad (3-)$$

37)

---

<sup>1</sup> Branch and Bound



$$\sum_{k=1}^n L_{jk} M_{i,k} - b_j \geq 1, i \in \mathbb{N}_{\Pi_j} \quad (3-$$

38)

$$L_{jk} \geq 0, b_j \geq 0, k = 1, \dots, n.$$

$J_j$  نشان دهنده تابع هدف برای افراز  $J$  است. این تابع هدف را بسته به مسئله می توان تعریف نمود. بعنوان نمونه می توان از  $J_j = \sum_{k=1}^n L_{jk} M_{i,k} + b_j$  استفاده کرد. الگوریتم کلی شاخه و حد به یافتن افرازهای بهینه ممکن برای مجموعه نشانه گذاری های ممنوع می پردازد. هر یک از این افرازها با استفاده از یک PI پیشگیری می شود که ضرایب و محدودیت GMEC متناظر آن با استفاده از  $L_j$  و  $b_j$  بدست می آید. روند کلی این روش، در الگوریتم ۲-۳ نشان داده شده است.

الگوریتم ۲-۳- الگوریتم شاخه و حد [۱۱۳]:

۱- مقدار دهی اولیه:  $\Pi = \{\Pi_1\}$ ,  $\Pi_1 = \{M\}$ ,  $\Sigma = M_{FBM} \setminus \Pi_1$ .

۲- در هر مرحله، با داشتن  $\Pi$  و  $\Sigma$ ، singleGMEC را حل کنید تا متغیرهای  $[J, L, b]$  بدست آیند.

۳- در صورت تهی شدن  $\Sigma$ ، اگر  $J$  و  $c$  بدست آمده کمتر از  $c^*$  و  $J^*$  است، این دو پارامتر را به-روزرسانی کنید.

۴- اگر  $\Sigma \neq \emptyset$  است و هنوز نشانه گذاری های FBM وجود دارند که پیشگیری نشده اند، یک نشانه گذاری  $M \in \Sigma$  را بطور تصادفی انتخاب کنید. هر بار  $M$  را به یکی از  $c$  افراز اضافه کرده و singleGMEC را حل کنید و اگر  $c$  و  $J$  بدست آمده، بهتر از  $J^*$ ,  $c^*$  است، آنها را به-روز-رسانی کنید.

۵- مسئله singleGMEC را با اضافه کردن مجموعه جدید  $\{M\}$  و داشتن  $c + 1$  مجموعه در افراز نیز حل کنید و اگر  $c$  و  $J$  بدست آمده، بهتر از  $J^*$ ,  $c^*$  است، آنها را به-روزرسانی کنید.

الگوریتم شاخه و حد، در هر مرحله یک  $M \in M_{FBM}$  که هنوز پیشگیری نشده است را انتخاب می‌کند. افزاز بهینه جزئی که تاکنون بدست آمده است را به یکی از  $c + 1$  مجموعه ذیل به‌روزرسانی می‌کند.

$$\Pi^{(i)} = \{\Pi_1, \Pi_2, \dots, \Pi_i \cup \{M\}, \dots, \Pi_c\}, \quad i = \{1, 2, \dots, c\}$$

$$\Pi^{(c+1)} = \{\Pi_1, \Pi_2, \dots, \Pi_c, \{M\}\},$$

و از افزازی که تابع هزینه *singleGMEC* را کمینه کند برای ادامه مراحل استفاده می‌کند.

اکثر مقالاتی که تاکنون بررسی شدند، برای پیشگیری از بن‌بست، رویکرد کنترل نظارتی مبتنی بر P-Invariant را با فرض قابل کنترل و قابل مشاهده بودن تمامی گذرها مطرح کرده‌اند. [۱۱۴]، [۱۱۵] به کنترل نظارتی در شرایط غیرقابل کنترل بودن برخی از گذرها پرداختند. کوئین و همکاران [۱۱۴] یک روش پیشگیری از بن‌بست مبتنی بر سایفون‌های ابتدایی را در مورد شبکه‌های پتری مبتنی بر سیستم‌های تولیدی مطرح کردند که گذرهای غیرقابل کنترل و غیرقابل مشاهده را نیز پوشش می‌دهد. ونگ و همکاران [۱۱۵]، نیز به ارائه سوپروایزر بهینه، در صورت امکان وجود گذرهای غیر قابل کنترل می‌پردازند. این روش نیز مبتنی بر GMEC است که با استفاده از یک روش پایین به بالا و پیچیدگی محاسباتی چند جمله‌ای، محدودیت‌های ناسازگار را به محدودیت‌های سازگار تبدیل می‌کند.

روش‌هایی که تاکنون از نظر گذشتند، منجر به یافتن سوپروایزر بیشینه مجازیت، در صورت وجود آن، خواهند شد. هیچ یک از این روش‌ها در صورتی که اساسا سوپروایزر بیشینه مجازیت برای یک مسئله وجود نداشته باشد، پاسخی را ارائه نمی‌دهند. چن و همکاران [۱۱۶] به یافتن سوپروایزر

بیشترین مجازیت<sup>۱</sup>، در صورت وجود نداشتن سوپروایزر بیشینه مجازیت می‌پردازند. آنها به ارائه مفهومی به نام M-برابری<sup>۲</sup> می‌پردازند. یک نشانه‌گذاری قانونی  $M_1$  را M-برابر با یک نشانه‌گذاری قانونی  $M_2$  گوئیم، اگر به ازای تمامی مکان‌های  $p_i$  دارای نشانه در نشانه‌گذاری غیرقانونی  $M$ ، تعداد نشانه‌های دو نشانه‌گذاری برابر باشند یعنی :

$$\forall p_i, M(p_i) \neq 0 \rightarrow M_1(p_i) = M_2(p_i) \quad (3-$$

39)

در آنصورت، اگر  $M_1$ ، پس از پیشگیری از نشانه‌گذاری غیرقانونی  $M$  دسترس‌پذیر نباشد،  $M_2$  نیز دسترس‌پذیر نخواهد بود. مجموعه تمام نشانه‌گذاری‌های قانونی M-برابر با نشانه‌گذاری قانونی  $M_j$  را با  $N_{M,M_j}$  نشان می‌دهیم. چن و همکاران با استفاده از مفهوم M-برابری، به ارائه یک روش مبتنی بر برنامه‌نویسی خطی برای طراحی یک سوپروایزر بیشترین مجازیت پرداختند. تابع هدف این ILP، کمینه کردن تعداد نشانه‌گذاری‌های قانونی است که پس از اعمال کنترل نظارتی دسترس‌پذیر نخواهند بود. متغیر  $f_j \in \{0,1\}$  نیز نشان دهنده دسترس‌پذیر بودن ( $f_j = 0$ ) و یا دسترس‌ناپذیر بودن ( $f_j = 1$ ) نشانه‌گذاری  $M_j$  پس از اضافه نمودن مکان کنترلی جدید است.

$$\text{Min } \sum_{M_j \in \mu_L} N_{M,M_j} \cdot f_j$$

$$s. t. \sum_{i \in N_{A,M}(p_i) \neq 0} l_i \cdot (M_j(p_i) - M(p_i)) \leq 0 \cdot f_j - 1 \quad (3-$$

40)

در حالت کلی، نمی‌توان یک کنترل نظارتی خطی برای اعمال محدودیت روی نشانه‌گذاری‌های یک شبکه پتری یافت. کاردن و همکاران [۱۱۷] شرط وجود یک طبقه‌بند خطی برای جداسازی نشانه‌گذاری‌های ممنوع از نشانه‌گذاری‌های قانونی را قرار نداشتن هیچ یک از نشانه‌گذاری‌های ممنوع

<sup>1</sup> Most Permissive

<sup>2</sup> M-Equality

در پوشش محدب<sup>۱</sup> نشانه‌گذاری‌های قانونی بیان کرده‌اند. یک کنترل نظارتی بعنوان طبقه‌بند خطی می‌تواند با استفاده از تعیین مجموعه‌ای از محدودیت‌های انحصار متقابل تعمیم یافته به پیشگیری از حالات ممنوع بپردازد. البته اثبات می‌شود که تمام محدودیت‌های غیر خطی را نیز می‌توان با استفاده از اجتماع زیرمجموعه‌ای از GMEC‌ها پیاده‌سازی نمود. در این صورت سوپروایزر بدست آمده را نمی‌توان الزاما با استفاده از شبکه‌های پتری پیاده‌سازی مرد. بلکه می‌باست با استفاده از گزاره‌های منطقی پیاده‌سازی شده و ارضاشدن هر یک از این محدودیت‌ها را بصورت برخط، بررسی نمود [۱۱۸].

در صورتی که این سوپروایزر خطی وجود نداشته باشد، رویکردهای [۱۱۹، ۱۲۰] مجموعه نشانه‌گذاری‌های قانونی را به چندین زیر مجموعه افراز می‌نمایند. برای هر یک از این زیرمجموعه‌ها، نامعادلات خطی برای الزام پیشگیری از وقوع تمامی نشانه‌گذاری‌های غیرقانونی و حفظ تمامی نشانه‌گذاری‌های قانونی درون این زیرمجموعه تنظیم می‌شود. این افراز در مجموعه نشانه‌گذاری‌های قانونی بصورتی خواهد بود که فضای پوشش محدب زیر مجموعه‌های  $M_L$  با پوشش محدب نشانه‌گذاری‌های غیرقانونی اشتراکی نداشته باشد. نهایتاً اجتماع نامعادلات خطی، منجر به داشتن سوپروایزر بهینه خواهد شد. این نوع سوپروایزر نوع خاصی از سوپروایزر کلی معرفی شده در [۱۲۱] است.

همانطور که بیان شد در حالت کلی، محدودیت‌های حاصل از اجتماع مجموعه GMEC‌ها، قابل تبدیل به یک سوپروایزر مبتنی بر شبکه‌های پتری نیستند. در [۱۱۴] با ایجاد یک سوپروایزر دو لایه، به کنترل اجتماع محدودیت‌های انحصار متقابل کراندار پرداخته است. این الگوریتم، با نام تغییردهنده مانیتور<sup>۲</sup> بدین ترتیب است:

---

<sup>1</sup> Convex Hall

<sup>2</sup> Monitor Switcher

الگوریتم ۳-۳- الگوریتم تغییر دهنده مانیتور، جهت محاسبه کنترل نظارتی برای اجتماع محدودیت‌های خطی [۱۱۴]:

ورودی: یک مدل شبکه پتری از کارخانه، و محدودیت Or-And یا اجتماع  $r$  مجموعه از

$$L_{OA} = \{(L_i, b_i), 1 \leq i \leq r\}$$

خروجی: مدل حلقه بسته از شبکه پتری

۱- ثابت  $K$  را به گونه‌ای انتخاب نمایید که از وزن محدودیت‌های مربوط به تمامی نشانه-گذاری‌های شبکه بزرگتر باشد.

۲- برای هر یک از محدودیت‌های تکی مانند  $(L_{ij}, b_{ij})$  یک مانیتور  $p_{cij}$  را به شبکه اضافه نمایید بطوری که  $C(q_{ij}, t) = -L_{ij}^T \cdot C(\cdot, t)$  و  $M_0(L_{ij}) = b_{ij} - L_{ij}^T \cdot M_0 + B$  باشد.

۳- تعداد  $r$  مکان کنترلی دیگر  $q'_r$ ، به همراه  $r(r-1)$  گذر  $t_{ij}$  را به شبکه اضافه نمایید، بصورتیکه  $Pre(q'_j, t_{ij}) = 1$  و  $Post(q'_j, t_{ij}) = 1$  باشد. تمامی مکان‌های کنترلی  $p_{cij}$  محاسبه شده در مرحله قبل را نیز با کمان‌های ورودی و خروجی با وزن  $b$  به تمامی گذرهای تازه افزوده شده متصل کنید.

۴- برای مکان کنترلی  $p_{cl}$  که در معادله  $M_0(p_{cl}) \geq b$  صدق می‌کند، نشانه‌گذاری را برابر با  $M(p_{cl}) = M(p_{cl}) - b$  قرار داده و در مورد دیگر مانیتورها، نشانه‌گذاری را بصورت  $M(p_{ci}) = 0$  قرار دهید. نهایتاً شبکه پتری بدست آمده را بعنوان خروجی اعلام نمایید.

در [۱۱۴] شرایط کراندار بودن یک اجتماع محدودیت‌های انحصار متقابل بصورت صدق کردن در معادله برنامه‌نویسی خطی ذیل بیان شده است:

$$\begin{cases} B_{ij,x} = \text{Max}_{i,j}^T \cdot (M_0 + Cy) - b_{ij} \\ \text{s.t. } M_0 + Cy \geq 0, L_x^T \cdot (M_0 + Cy) \leq b_x, x \neq i \end{cases} \quad (3-$$

41)

چن و همکاران [۱۲۲] به ارائه یک سوپروایزر برای اعمال محدودیت‌های غیرخطی از نامعادلات غیرخطی‌ای که بصورت تجمیعی قابل جداسازی هستند پرداختند. فرض کنید این محدودیت‌ها به شکل  $f_1(M_1) + \dots + f_n(M_n)$  باشند که  $f_i$  تابع غیرخطی بر روی تعداد نشانه‌های مکان  $p_i$  است. برای اعمال هر یک از محدودیت‌های  $f_i(M_i)$  یک مکان متناظر با  $p_i$  را به شبکه اضافه می‌شود. هر یک از گذرهای متصل به  $p_i$  نیز با چندین گذر جایگزین می‌شود. تعداد این گذرهای جایگزین، برابر با تعداد ترکیبات متفاوت اعداد  $x - y = z$  است که  $1 \leq z \leq \text{bound}(p)$  است. منظور از  $\text{bound}(p)$ ، کران بالای تعداد نشانه‌های موجود در  $p$  است. این روش برای کنترل بن‌بست کاربرد ندارد؛ زیرا معادلات کنترل بن‌بست منجر به داشتن مجموعه محدودیت‌هایی است که بصورت نامعادلات خطی بیان می‌شوند. همچنین نیاز به افزایش بیش از اندازه ساختار یک شبکه دارد.

### ۳-۲-۴ روش‌های مبتنی بر شباهت

رویکردهای دیگری نیز برای کنترل نظارتی اعمال کننده زنده بودن در شبکه های پتری مطرح شده‌اند که کلیت سه رویکرد اصلی را ندارند. این رویکردها را تحت عنوان رویکردهای مبتنی بر شباهت دسته‌بندی می‌کنیم. این روش‌ها عمدتاً به یافتن شباهت بین یک سوپروایزر جدید برای سیستم و کنترل نظارتی قبلی (در صورت تغییر ضوابط سیستمی) و یا به یافتن شباهت بین کنترل نظارتی یک سیستم با کنترل نظارتی سیستم دیگر می‌پردازند.

لی و همکاران [۱۲۳، ۱۲۴] یک روش برای به‌روزرسانی مکان‌های کنترلی، در هنگام تغییرات خاصی در ضوابط، را ارائه دادند. فرض کنید  $L'M \leq B$  یک GMEC شامل ضوابط جدید باشد.

سیستم حلقه بسته جدید بصورت  $[M', M_{0c}']$  است که  $M$  و  $M_{0c}' = B' - L.M$  بترتیب، حالات و نشانه‌گذاری اولیه سوپروایزر را نشان می‌دهد. اگر  $M_{0c}' \geq 0$  باشد، ضابطه جدید را سازگار با حالات شبکه می‌گوییم. اگر ضابطه جدید با نشانه‌گذاری‌های شبکه سازگار نباشد، در صورتیکه دنباله‌ای از اجراها،  $\Pi$ ، وجود داشته باشند که  $[M, M'_{c0}] [\Pi] [M', M'_s]$  و شرط  $[M', M'_s] > 0$  برقرار باشد، ضابطه را قابل قبول می‌گویند. لی و همکاران [۱۲۴] نشان دادند اگر  $L' = L$  باشد، سوپروایزر جدید تنها نیاز به به‌روزرسانی حالت اولیه دارد. و اگر  $B' = B$  باشد، تنها نیازمند به‌روزرسانی ماتریس برخورد سوپروایزر است. برای این شرایط، یک معادله ILP برای تأمین شرایط سوپروایزر جدید ارائه شده است. همچنین، اگر ضابطه جدید با توجه به ضابطه قبل قابل قبول نباشد، دست یافتن به یک ضابطه قابل قبول با استفاده از افزایش منابع سیستمی را مطرح می‌کند.

سلیمی و همکاران [۷۵] با بررسی زنده بودن گذرهای یک شبکه پتری تحت یک سیاست کنترل نظارتی، به اثبات تضمین زنده بودن شبکه مشابه که تحت کنترل نظارتی مشابهی قرار گرفته است می‌پردازند. در این مرجع، مفهوم شباهت شبکه‌های پتری عمومی [۱۲۵]، به شبکه‌های پتری کنترل شونده تعمیم داده شده است. فرض کنیم  $N(P, T, F, W)$  و  $N' = (P', T', F', W')$  دو ساختار شبکه پتری باشند و تابع  $\alpha: T \rightarrow T'$  یک تابع تزریق<sup>۱</sup> باشد (نگاشتی از هر عضو  $T$  به یک عضو  $T'$  وجود دارد. اما نگاشت عکس الزاما برقرار نیست). این تابع تزریق را می‌توان به زیرمجموعه‌ای از گذرهای درون  $T$  (به جای تمامی گذرهای درون  $T$ ) و یا به دنباله‌ای از گذرهای درون  $T$  تعمیم داد.  $T'^* \subseteq T'$  زیر مجموعه‌ای از گذرهای  $T'$  هستند که تصویر یکی از گذرهای درون  $T$  تحت تابع تزریق هستند. بدین ترتیب، تابع معکوس تزریق  $\alpha^{-1}: T'^* \rightarrow T$  نیز تعریف می‌شود. می‌گوییم شبکه  $N'$  تحت سیاست سوپروایزری  $\rho'$ ، شبکه  $N$  تحت سیاست سوپروایزری  $\rho$  را شبیه‌سازی می‌کند اگر و تنها

<sup>۱</sup> Injection Function

اگر: (۱) تناظری بین نشانه‌گذاری‌های اولیه دو شبکه وجود داشته باشد. (۲) اگر دو نشانه‌گذاری متناظر  $M_1$  و  $M_1'$  تحت سیاست‌های  $\rho$  و  $\rho'$  دسترس‌پذیر باشند و اگر  $M_1 \xrightarrow{\delta} M_2$ ، آنگاه یک نشانه‌گذاری  $M_2'$  دسترس‌پذیر تحت سیاست  $\rho'$  وجود دارد که  $M_1' \xrightarrow{\delta'} M_2'$ .

سپس سلیمی و همکاران [۷۵] اثبات می‌کنند اگر شبکه پتری  $N$  مشابه شبکه پتری  $N'$  باشد، سیاست  $\rho$  برای شبکه  $N(M_0)$  یک سیاست اعمال‌کننده زنده بودن است اگر و تنها اگر مجموعه گذرهای  $T^{**}$  تحت سیاست سوپروایزری  $\rho'$  اعمال شده بر شبکه  $N'(M_0')$  زنده باشند.

### ۳-۳ رویکردهای کنترل نظارتی غیر متمرکز

برخی از محققین، به مسئله کنترل نظارتی توزیع شده در سیستم‌های مقیاس‌پذیر پرداخته‌اند. توزیع یک ضابطه کلی به ضوابطی در سوپروایزرهای محلی، با استفاده از یک هماهنگ‌کننده مرکزی [۱۲۶] یا ارتباط بین سوپروایزرهای محلی [۱۲۷] مطرح شده‌اند. در [۱۲۸] نیز به همجوشی<sup>۱</sup> سوپروایزرها با استفاده از مفهوم رنگ و نوع در نشانه‌ها می‌پردازد. و نامعادلات خطی را با رویکرد ماتریسی در شبکه پتری رنگی حل می‌کند.

یورداج و انتساکلیس [۱۲۹، ۱۳۰] و باسیل و همکاران [۱۲۹، ۱۳۰] نیز استفاده از کنترل نظارتی توزیع شده بدون داشتن هیچ گونه ارتباط بینابینی و یا هماهنگ‌کننده مرکزی را مطرح کرده‌اند. این روش‌ها منجر به کاهش ارتباطات کنترلی میان سوپروایزر و کارخانه شده و یک سیستم ایمن‌تر و کارآتر از نظر اقتصادی را فراهم می‌کند.

---

<sup>۱</sup>Fusion



باسیل و همکاران [۱۳۰] ایده‌ای را برای توزیع‌پذیر نمودن محاسبات کنترل نظارتی در سایت-های از قبل تعیین شده‌ای که با زیرمجموعه‌ای از مکان‌های یک شبکه پتری نشان داده می‌شوند مطرح کردند. فرضیات محدود کننده روش ارائه شده در [۱۳۰] عبارتند از: (۱) وزن‌های GMEC ها  $I_{ik}$  اعداد مثبت هستند). (۲) گذرها قابل کنترل هستند. (۳) مجموعه پشتیبان هر GMEC توزیع شده، مجموعه تک مقداره است.

الگوریتم پیشنهاد شده در [۱۳۰] با داشتن مجموعه‌ای از GMEC ها به شکل  $L.M \leq b$ ، به تعیین بیشینه جعبه داخلی عددی<sup>۱</sup> برای مکان‌های شبکه‌های پتری می‌پردازد. اگر این بیشینه جعبه داخلی عددی برابر با  $u^* = [b(p1), b(p2), \dots, b(pn)]$  باشد، سوپروایزر توزیع شده می‌تواند برای هر سایت  $S_i$  به ایجاد محدودیت روی نشانه‌گذاری‌های مکان  $p_i$  (بعنوان تنها عضو مجموعه پشتیبان  $S_i$ ) بپردازد. هر یک از GMEC های توزیع شده به شکل  $M(p_i) \leq u^*(i)$  بیان می‌شوند.

این مرجع در فاز اول محاسبات، به محاسبه بیشینه مقادیر قانونی نشانه‌های هر یک از مکان‌ها می‌پردازد. برای این کار، هر بار یک مکان را به قسمی انتخاب می‌نماید که  $\frac{b-\tau L.1}{L(p_i)}$  در مورد آن کمینه است. بیشینه ظرفیت این مکان برابر  $\tau$  خواهد بود.  $\tau$  کمینه  $\tau(x)$  روی محدودیت‌ها است که از معادله (۳-۴۲) قابل محاسبه است:

$$\tau = \text{Min}_{x \in \{1, \dots, n_c\}} \tau(x), \text{ where } \tau(x) = \text{ceil} \left( \frac{b_x}{\sum_{p \in P} L_x(p)} \right) \quad (3-$$

42)

سپس  $L$  و  $b$  به‌روزرسانی خواهند شد. بدین ترتیب که  $L$  در ستون‌های متناظر مکان انتخاب شده برابر ۰ شده و  $b_j = b_j - \tau.L_j(p_i)$  خواهد شد.

<sup>۱</sup> Maximall Interger Inner Box

در [۱۳۰] اثبات می‌شود، اگر ترتیب بیشینه ظرفیت مکان‌های انتخاب شده بصورت معادله (۳)-  
 (۴۳) باشد، بیشینه ظرفیت واقعی مکان‌ها بدست آمده است. این معادله بیانگر این است که اگر در یک  
 مرحله  $\tau_{\mu+1}$  با  $\tau_{\mu}$  برابر شد باید در ادامه نیز همین مقدار بماند.

$$\tau_1 < \tau_2 < \dots < \tau_{\mu} = \tau_{\mu+1} \dots = \tau_n \quad (3-$$

43)

اگر شرط معادله (۳-۴۳) برقرار نباشد یعنی  $\tau_i = \tau_{i+1}, i < \mu$  باشد، فاز دوم الگوریتم شروع  
 خواهد شد. در این فاز برای  $\tau$  های میانی که  $\tau_i = \tau_{i+1}$  بود، باقیمانده  $b$  به ظرفیت مکان  $p_i$  اضافه خواهد  
 شد؛ یعنی:

$$\tau(p_i) = \tau(p_i) + \text{Min}_{x \in \{1, \dots, n_c\}} \frac{b_x - \tau L_x \cdot 1}{L_x(j)} \quad (3-$$

44)

باسیل و همکاران [۱۳۱] الگوریتم پیشنهاد شده در [۱۳۰] را تعمیم دادند و فرضیه محدود  
 کننده شماره (۳)، تک مقداره بودن مجموعه پشتیبان هر یک از GMEC های توزیع شده، را حذف  
 کردند. برای نیل به این هدف ابتدا همانند [۱۳۰]، GMEC های مربوط به بیشینه تعداد نشانه برای هر  
 یک از مکان‌ها بدست می‌آید. سپس GMEC مکان‌های متعلق به یک سایت بصورت وزنی (بر اساس  
 وزن  $L$  در هر مکان) جمع می‌شوند. بعنوان مثال اگر افزایش مکان‌های یک شبکه بصورت  
 $\{M | 3M(p_1) + M(p_2) + M(p_3) \leq 5\}$  کلی باشد و  $P = \{\{p_1, p_2\}, \{p_3\}\}$   
 باشد، راه حل تک مقداره سه GMEC توزیع شده با حدهای  $u^* = [1 \ 1 \ 1]$  را تولید می‌کند؛ یعنی

$$\{M | (M(p_1) \leq 1), (M(p_2) \leq 1), (M(p_3) \leq 1)\}$$

نهایتاً GMEC توزیع شده با توجه به GMEC کلی و افزایش مشخص شده، به شکل  
 $\{M | (3M(p_1) + M(p_2) \leq 4), (M(p_3) \leq 1)\}$  خواهد بود.

سپس در صورت نیاز ارتباطاتی ایجاد می‌شود که اجرای یک گذر در یک بخش را به بخش دیگری که به این اطلاعات نیاز دارد مخابره می‌کند. در نهایت قابل قبول بودن غیرمتمرکز<sup>۱</sup> بررسی خواهد شد. اگر راه حل به صورت غیرمتمرکز قابل قبول نبود، با استفاده از دو رویکرد استفاده از مخابره اطلاعات بین سوپروایزرها یا استفاده از انتقال محدودیت در بخش‌ها، زیر مجموعه‌ای از سیستم را که سازگار است بدست می‌آورد.

باسیل و همکاران [۱۳۲] به طراحی یک ILP برای کنترل نظارتی توزیع شده پرداخته‌اند. آنها از افزاز گذرها برای مشخص‌سازی هر یک از پیمان‌های کنترلی که باید توسط سوپروایزر مجزا کنترل شوند استفاده کرده‌اند. دلیلی که توسط آنها در مزیت این روش نسبت به روش‌های مبتنی بر افزاز مکان‌ها [۱۳۰, ۱۳۱] بیان شده است این است که گذرها نشان دهنده رخدادهای سیستمی هستند اما مکان‌ها عموماً نشان دهنده برقراری پیش‌شرط هستند. از نظر فیزیکی شناسایی گذرهای متعلق به یک سایت آسانتر از شناسایی مکان‌ها است. معادلات ILP تعریف شده در [۱۳۲] شامل ۱۰ دسته معادلات هستند:

معادله دسترس‌پذیری نشانه‌گذاری‌های قانونی، معادله حذف یک حالت ممنوع، معادله الزام حذف هر حالت ممنوع که بیانگر این است که حداقل یک مکان کنترلی برای حذف هر یک از حالات ممنوع مشخص شده است، معادله محاسبه توپولوژی شبکه سوپروایزر، معادله ارضا شدن نامعادله در نشانه‌گذاری اولیه، معادلات پیشگیری از وجود داشتن کمان‌هایی بین سوپروایزرهای یک سایت با گذرهای غیر متعلق به آن سایت، معادلات مرتبط با شرایط کنترل‌پذیری و مشاهده‌پذیری در یک سایت و معادله تعلق داشتن هر مانیتور تنها به یک سایت.

---

<sup>۱</sup>D-admissibility

تابع هدف نهایی نیز به بیشینه کردن تعداد نشانه‌گذاری‌های قانونی پوشش داده شده توسط سوپروایزرهای موجود در سایت‌های مختلف و کاهش وزن کمان‌های اتصال دهنده سوپروایزر و گذرهای سیستم کنترل شونده توجه دارد.

در [۱۳۲] مجموعه حالات مجاز توسط سوپروایزر توزیع شده، یک پیکربندی از زیرگراف دسترس‌پذیری که با تمام نیازمندی‌های رفتاری سازگار باشد را بدست نمی‌آورند. در حالیکه، با استفاده از کنترل نظارتی توزیع شده می‌توان تمامی حالات قانونی را که توسط یک سوپروایز متمرکز حفظ می‌شوند را حفظ نمود. برای تأمین این مسئله، باسیل و همکاران [۱۳۳] ابتدا به مشخص‌سازی نشانه‌گذاری‌های قانونی کاندید می‌پردازند و سپس از یک سوپرایز توزیع شده برای اعمال آن استفاده می‌کنند. در این مرجع با معرفی یک روش شاخه و حد بصورت سیستماتیک و کاراً به جستجو در تمامی زیرمجموعه‌های نشانه‌گذاری‌های قانونی برای یافتن بیشینه زیرمجموعه‌ای که محدودیت‌ها را ارضا می‌کند، پرداخته شده است.

باسیل و همکاران [۱۱۸] نیز روش شاخه و حد را برای یافتن زیرمجموعه‌های نشانه‌گذاری‌های قانونی بکار بردند. در هر سایت، مجموعه نشانه‌گذاری‌های قانونی به سه زیر مجموعه  $M_L = M_{L+} \cup M_{L-} \cup M_{Lx}$  افزایش می‌شود.  $M_{L+}$  نشانه‌گذاری‌های قانونی که در این سایت مجاز شوند،  $M_{L-}$  نشانه‌گذاری‌های قانونی که در این سایت مجاز نمی‌شوند و  $M_{Lx}$  نیز نشانه‌گذاری‌های قانونی که به  $M_{L+}$  و  $M_{L-}$  نگاشت نشده‌اند را نشان می‌دهد. الگوریتم کنترل نظارتی توزیع شده ابتدا یک نشانه‌گذاری قانونی را انتخاب می‌کند و تا زمانی که نشانه‌گذاری‌های قانونی وجود دارند که مجاز بودن آنها الزامی نشده است، چهار مرحله کلی انجام می‌شود: پیش‌پردازش، B-امکان‌پذیری<sup>۱</sup>، D-امکان‌پذیری<sup>۲</sup> و به روزرسانی شاخه. یک زیرمجموعه از نشانه‌گذاری‌های قانونی B-امکان‌پذیری است اگر خواص رفتاری

---

<sup>۱</sup> B-Feasibility

<sup>۲</sup> D-Feasibility

مورد نیاز (مانند زنده بودن، بازگشت پذیری و کنترل پذیری) را داشته باشد. یک زیرمجموعه از نشانه-گذاری های قانونی D-امکان پذیر است اگر بتواند با استفاده از یک کنترل نظارتی توزیع شده مجاز شود.

در مرحله پیش پردازش، برای مسئله فعلی،  $M_{L+}$  و  $M_{L-}$  را بدست می آورد و اگر این دو مجموعه مجزا نبودند مسئله فعلی و این گره، از الگوریتم شاخه و حد حذف می شود. زیرا راه حل امکان پذیری ندارد.

خروجی مرحله پیش پردازش زیرگراف دسترس پذیری است که یک جزء قویا همبند است. زیرگراف دسترس پذیری بدلیل امکان وجود گذرهای غیرقابل کنترل، الزاما B-امکان پذیر نیست. در این مرحله بطور بازگشتی، حالاتی از جزء قویا همبند که یکی از خواص رفتاری مورد نیاز را تأمین نمی کنند هرس می شوند.

در مرحله سوم، با طراحی معادلات ILP همانند مراجع [۱۳۴, ۱۳۵] به ایجاد یک راه حل D-امکان پذیر می رسیم. نهایتاً، طبق الگوریتم شاخه و حد، یک نشانه گذاری قانونی جدید M که تاکنون بررسی نشده است یافته شده و برای آن به حل مسئله برای دو شاخه  $\Pi_- = \{M_{L+}, M_{L-} \cup \{M\}$  و  $\Pi_+ = \{M_{L+} \cup \{M\}, M_{L-}\}$  می پردازد.

البته تحقیقاتی که برای اعمال کنترل سوپروایزری بصورت توزیع شده صورت گرفته اند، در زمینه محاسبه توزیع شده گراف دسترسی پیشنهادی نداشته اند. این رساله به استفاده از روش های سنتز برای محاسبات پیمانهای کنترل سوپروایزری می پردازد.

### ۳-۴ جمع بندی

در این فصل به بررسی و مرور کارهای انجام شده در زمینه معرفی سیاست‌های کنترل نظارتی مبتنی بر شبکه‌های پتری پرداختیم. ابتدا سیاست‌های محاسبه کنترل نظارتی متمرکز را بیان کردیم. این سیاست‌ها به سه دسته رویکرد زبانی، تحلیل فضای حالت و روش‌های ساختاری تقسیم شده و در هر بخش کارهای انجام شده مطرح شدند.

در زمینه کنترل نظارتی توزیع شده نیز به مرور منابع، پرداختیم. روش‌های مختلف این رویکرد را توزیع‌پذیری مبتنی بر مکان و یا توزیع‌پذیری مبتنی بر گذر بررسی کردیم. البته، تحقیقی در زمینه محاسبه توزیع شده گراف دسترسی، بعنوان گلوگاه محاسبات سوپروایزر در سیستم‌های مقیاس‌پذیر، انجام نشده است.

فصل ۴: نمونه‌هایی از کاربرد شبکه‌های پتری

در مدل‌سازی سیستم‌های گسسته رخداد

## ۴-۱ مقدمه

بسیاری از سیستم‌های مورد استفاده در اطراف ما بنحوی از انواع سیستم‌های گسسته رخداد بوده و یا می‌توانند بعنوان یک سیستم گسسته رخداد تلقی شوند. سیستم‌های مدیریت ترافیک در شبکه‌های حمل و نقل [۱۳۴]، سیستم‌های تولیدی منعطف [۶۵]، سیستم‌های کاوش فرآیند [۱۳۵] و شبکه‌های کامپیوتری [۱۱] از جمله این سیستم‌ها هستند. شبکه‌های پتری از مهمترین فرمالیزم‌ها برای مدلسازی و کنترل این سیستم‌ها هستند. در این فصل، به ارائه مثال‌هایی از این سیستم‌ها و مدلسازی و تحلیل آن‌ها با استفاده از شبکه‌های پتری می‌پردازیم.

## ۴-۲ فرآیندکاوی پیمانهای با استفاده از شبکه‌های پتری

یکی از حوزه‌های کاربرد شبکه‌های پتری، استفاده از آنها در فرآیندکاوی است [۱۳۵]. فرآیندکاوی، با بررسی داده‌های لاگ فایل‌ها - که ترتیب انجام فرآیندهای مختلف را درج می‌کنند - به بازسازی روند انجام فرآیندها و روابط میان آنها می‌پردازد. این روابط شامل: ترتیب (seq، .)، انتخاب انحصاری (xor، ^)، توازی (&، and) و انتخاب هر کدام یا هر دو (|، or) است. هر سطر از لاگ فایل، نشان‌دهنده ترتیب انجام یک فرآیند از شروع تا پایان است. با خواندن هر یک از این سطرها، مدل شبکه پتری آن بدست آمده و با مدل شبکه پتری سطرهای دیگر ترکیب می‌شود. روش‌های متداول فرآیندکاوی روی لاگ فایل با اندازه بزرگ قابل اعمال نیستند. از این روش‌هایی برای شکستن لاگ فایل به زیرلاگ فایلها ارائه شده است. روش‌های متداول، عمدتاً لاگ فایل را بصورت عمودی خرد می‌کنند. این عمل نیازمند روش‌هایی برای تشخیص نقاط شکست است، بصورتی که بتوان نهایتاً، زیرمدل‌های بدست آمده را ترکیب نمود.



در این بخش، به ارائه یک روش شکستن لاگ فایل‌ها به زیرلاگ فایل‌ها بصورت افقی می‌پردازیم. مؤلفه بارز این روش تجزیه لاگ فایل، داشتن روش تقسیم راحت و روش ترکیب نهایی دشوار است. بلعکس، در شیوه شکستن لاگ فایل‌ها بصورت عمودی، روش تقسیم سخت بوده و روش ترکیب نهایی با استفاده از قانون ساده ای فراهم می‌شود.

در روش افقی، یک مدل شبکه پتری برای هر زیرلاگ فایل بدست می‌آید. با استفاده از قوانینی که برای ترکیب سطرها ارائه خواهیم داد، به ترکیب دو به دو ی زیرلاگ فایل‌ها تا رسیدن به مدل کامل از کل لاگ فایل می‌پردازیم. قوانین ترکیبی که مورد استفاده در کار ما بوده‌اند از این قرار هستند:

- $\text{Compose}(a \times b, a \times b) = a \times b$ , هر یک از روابط:  $x$
- $\text{Compose}(a \cdot b, a \& b) = a \& b$
- $\text{Compose}(a \cdot b, a|b) = a|b$
- $\text{Compose}(a \& b, a|b) = a|b$
- $\text{Compose}(a^b, a|b) = a|b$

بعنوان مثال، یک لاگ فایل ساده را در نظر بگیرید که شامل ۶ اجرای<sup>۱</sup> مختلف از یک فرآیند است. این مثال در جدول ۴-۱ نشان داده شده است.

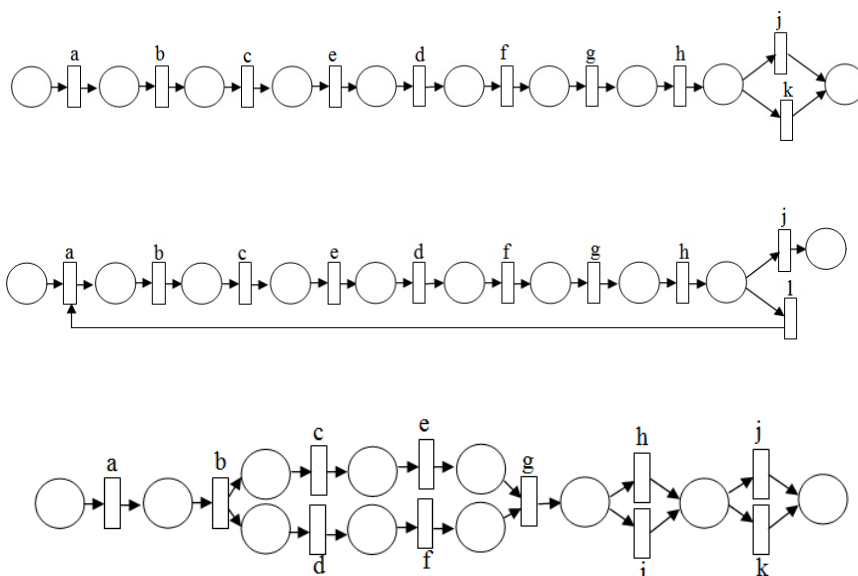
جدول ۴-۱- مثالی از یک لاگ فایل ساده، شامل ۶ اجرای مختلف از یک فرآیند.

Case 1	abcdfghj
Case 2	abcdfghk
Case 3	abcdfghlabcd fghj

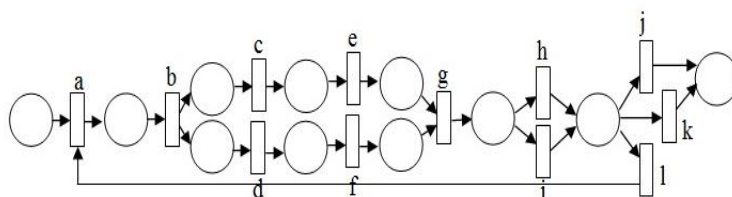
<sup>۱</sup> Run

Case 4	abcedfgik
Case 5	abcedfgij
Case 6	abdfceghk

با تقسیم سیستم به زیرسیستمهایی با دو اجرا و فرآیندکاوی هر یک از زوج اجراها، به سه مدل شبکه پتری می‌رسیم که در شکل ۴-۱ نشان داده شده‌اند. با ترکیب این زیرمدل‌ها با استفاده از قوانین ترکیب بیان شده، مدل کلی فرآیند بدست خواهد آمد. این مدل در شکل ۴-۲ نشان داده شده است.



شکل ۴-۱ زیر مدل‌های شبکه پتری بدست آمده پس از فرآیند کاوی برای زیرلاگ فایل‌های {1,2}، {3,4} و {5,6}.



شکل ۴-۲ - مدل شبکه پتری بدست آمده پس از ترکیب زیرمدلهای شکل ۱-۴ با استفاده از روش ترکیب مبتنی بر قانون.

## ۴-۳ ترکیب وب سرویس‌ها

وب سرویس‌ها از مهمترین معماری‌های مبتنی بر سرویس هستند که با گسترش تکنولوژی اینترنت، توسعه بیشتری داشته‌اند. البته یک سرویس نمی‌تواند تمام تقاضاهای کاربران را پوشش دهند بنابراین سرویس‌های فراهم شده توسط تأمین کنندگان متفاوت، در اشتراک با هم قرار می‌گیرند. البته گاه سرویس‌های مرتبط با هم، از نظر تعداد و نحوه دریافت آرگومان‌های ورودی و خروجی با هم سازگار نیستند. از اینرو ترکیب معنادار این وب سرویس‌ها برای ایجاد یک سرویس یکپارچه ضروری است. در این بخش به ارائه یک روش ترکیب وب سرویس‌ها با استفاده از شبکه‌های پتری خواهیم پرداخت.

بطور کلی یک وب سرویس مجموعه‌ای از پیمانه‌های توزیع شده و با هم‌بستگی پایین است که با فرستادن پیام با هم ارتباط دارند. رفتار نهایی یک وب سرویس با استفاده از رفتار ترکیبی پیمانه‌های آن مشخص می‌شود.

تعریف ۴-۱- وب سرویس: بصورت فرمال، یک وب سرویس یک ۶ تایی  $S = (\text{NameS}; \text{Desc};$

$\text{Loc}; \text{URL}; \text{CS}; \text{N})$  است که:

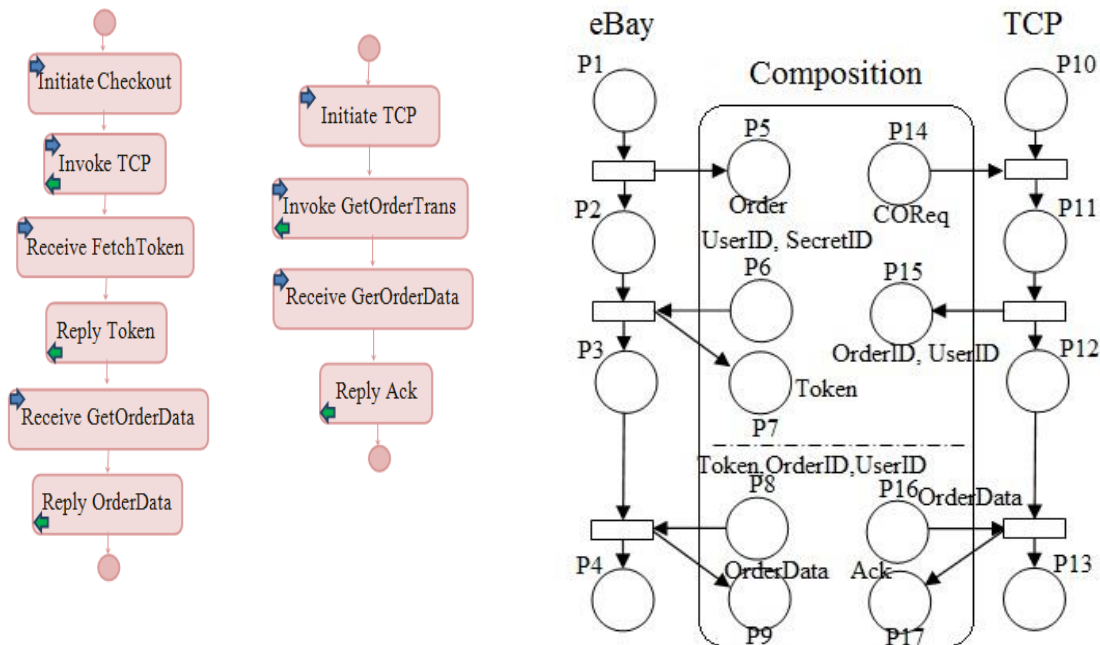
- $\text{NameS}$ : نام وب سرویس را به عنوان یک شناسه منحصر به فرد نشان می‌دهند.
- $\text{Desc}$ : یک توضیح مختصر از خدمتی است که با استفاده از وب سرویس ارائه می‌شود.

- Loc: مکان سروری را نشان می‌دهد که وب سرویس بر روی آن قرار دارد.
  - URL: آدرس اینترنتی برای فراخوانی یک وب سرویس را نشان می‌دهد.
  - CS: مجموعه‌ای از اجزای سرویس را نشان می‌دهد. اگر  $CS = NameS$  باشد، سرویس S یک سرویس پایه است. در غیر اینصورت یک سرویس ترکیبی است.
  - N: مدلی از وب سرویس است که رفتار داینامیک آن را نشان می‌دهد.
- عملگرهای جبری برای تجمیع وب سرویس‌ها و ایجاد وب سرویس ترکیبی بیان شده‌اند. گرامر شبه BNF این ترکیب بدین شرح است:

$$S ::= \lambda \mid X \mid SOS \mid S \diamond S \mid S \oplus S \mid \cup S \mid S \parallel_C S \mid Ref(S; a; S)$$

- $\lambda$ : یک سرویس تهی را نشان می‌دهد. یعنی؛ سرویسی که هیچ عملیاتی را انجام نمی‌دهد.
  - X: یک سرویس را نشان می‌دهد که برای می‌تواند یک سرویس اتمیک و یا پایه باشد.
  - $S_1 \circ S_2$ : یک سرویس ترکیبی است که سرویس  $S_1$  و سپس سرویس  $S_2$  را انجام می‌دهد.
  - $S_1 \diamond S_2$ : عملگر ترتیب دلخواه دو وب سرویس پایه را نشان می‌دهد.
  - $S_1 \oplus S_2$ : عملگر انتخاب را نشان می‌دهد. در زمان فراخوانی یک سرویس ترکیبی، یکی از سرویس‌های پایه  $S_1$  و یا  $S_2$ ، فراخوانی خواهد شد.
  - $\cup S$ : عملگر تکرار را نشان می‌دهد.
  - $S_1 \parallel_C S_2$ : ترکیب دو سرویس پایه را بصورت موازی انجام می‌دهد و امکان ارتباط بین سرویس‌ها با استفاده از مجموعه‌ای از پیام‌های C بوجود دارد.
  - $Ref(S_1; a; S_2)$ : که رفتاری همانند  $S_1$  دارد. البته بجز عملیاتی از  $S_1$  که با a برچسب خورده باشد که برای آنها عملیات مربوط به سرویس  $S_2$  انجام خواهد شد.
- در شکل ۳-۴ مثالی از دو وب سرویس مرتبط نشان داده شده است. این شکل که برگرفته از مثال معرفی شده در مرجع [۱۳۶] است، روند ارتباط وب سرویس‌های eBay و TPC را نشان می‌دهد.

دهد. در شکل ۴-۴ نیز مدل موجودیت شبکه پتری برای دو وب سرویس eBay و TCP و برطبق منطق اجرایی آنها نشان داده شده است. ترکیب این دو سرویس منجر به سرویس جدیدی خواهد شد. البته این دو سرویس با هم سازگار نبوده و تعداد ورودی و خروجی‌های برابری ندارند. در نتیجه روش‌های ترکیب مطرح شده در مورد سرویس‌های سازگار در این قسمت کاربرد ندارد. در [۱۳۶] ترکیب سرویس‌های ناسازگار با استفاده از یک مبادله گر پیام میانی<sup>۱</sup> انجام می‌شود که نقشی همانند آداپتور بین دو دستگاه سخت افزاری غیر همگن را ایفا می‌کند.



شکل ۴-۳- وب سرویس‌های eBay و TCP و نحوه برهم کنش آنها با سرویس دیگر [۱۳۶]. شکل سمت چپ، نمودار عملکرد eBay و شکل سمت راست، نمودار TCP را نشان می‌دهد.

شکل ۴-۴- مدل موجودیت شبکه پتری برای وب سرویس‌های eBay و TCP.

در این قست، ابتدا به معرفی یک روش ترکیب برای کلاس خاصی از شبکه‌های پتری بنام موجودیت شبکه‌های پتری<sup>۲</sup> می‌پردازیم. نگاهی بین این روش‌های ترکیب با جبر سرویس‌ها فراهم می‌کنیم. در صورت نبودن یک معادل برای برخی از عملگرهای جبر سرویس‌ها در روش‌های ترکیب

<sup>۱</sup> Message Mediator

<sup>۲</sup> Petri net Entity

موجودیت شبکه‌های پتری، به ارائه یک روش ترکیب می‌پردازیم. این روش‌ها می‌تواند برای ترکیب سرویس‌های ناسازگار باهم و یا ترکیب سرویس‌های سازگار باهم به کار رود؛ بدون اینکه نیازی به استفاده از مبادله گر پیام میانی داشته باشد.

تعریف ۲-۴ یک شبکه پتری برچسب دار، متشکل از  $NT = (N, L)$  است که  $N$  شبکه پتری معمولی و  $L$  تابع برچسب  $L \subseteq (P \times a \times P)$  و  $a$  مجموعه الفبا است.

تعریف ۳-۴ -P-نقطه دسترسی در یک شبکه پتری، یک مجموعه  $\rho \subseteq P(P)$  است که  $\forall M, M' \in \rho, M \subseteq M' \Rightarrow M = M'$ . این رابطه به این معناست که -P-نقطه دسترسی دارای زیرمجموعه ای از مکان‌هاست که هیچ یک زیرمجموعه دیگری نیست [۱۳۷].

تعریف ۴-۴ -عملگر ادغام مکان‌ها [۱۳۷]: فرض کنید  $N = (P, T)$  یک شبکه روی الفبای  $act$  و با -P-نقاط دسترسی  $\rho_1, \rho_2$  باشد. عملگر -P- ادغام برای  $N$  روی دو نقطه دسترسی، منجر به یک شبکه پتری جدید  $N' = p\text{-merge}(N, \rho_1, \rho_2) = N'(S', T')$  می‌شود که:

$$S' = S \setminus S^{12} + S_1 \times S_2, \text{ Where } S^{12} = S_1 \cup S_2$$

$$T' = \{(Q_1, a, Q_2) | M_1 \text{ in } \rho_1, M_2 \text{ in } \rho_2\}, \text{ where :}$$

$$Q_1 = \bullet T \setminus S^{12} \cup \bullet T \cap M_1 \times M_2 \cup M_1 \times (\bullet T \cap M_2)$$

$$Q_2 = T \bullet \setminus S^{12} \cup T \bullet \cap M_1 \times M_2 \cup M_1 \times (T \bullet \cap M_2) \quad (4-1)$$

تعریف ۴-۵ یک موجودیت شبکه پتری، یک شبکه پتری با اطلاعاتی در مورد ترکیب‌های انجام شونده در آینده است که بصورت  $E = (NL, h, l, d)$  تعریف می‌شود که:

$NL$ : یک شبکه پتری برچسب دار است.

$h, d, l$ : -P-نقاط دسترسی هستند که بترتیب نقاط دسترسی سر، میان و انتهایی نامیده می‌-

شوند [۲۳].

فرض کنید  $E_1 = (N_1, h_1, l_1, d_1)$  و  $E_2 = (N_2, h_2, l_2, d_2)$  دو موجودیت شبکه پتری باشند که  $N_i = (P_i, T_i)$  و الفبای برچسب دهی دو شبکه از مجموعه act انتخاب می شود. جبر موجودیت شبکه پتری شامل عملگرهایی است که روی وب سرویس ها نیز قابل بکارگیری است:

$$\text{Iteration: } \cup E = \langle p\_merge(N_1, h_1, l_1), h_1, l_1, d_1 \rangle$$

$$\text{Sequence: } E_1 \odot E_2 = \langle (N_1 \ l_1 \oplus \ h_2 \ N_2), h_1, l_1, d_1 \cup d_2 \rangle$$

$$\text{Choice: } E_1 \oplus E_2 = \langle (N_1 \ h_1 \oplus \ h_2 \ N_2), h_1 \cup h_2, l_1 \cup l_2, d_1 \cup d_2 \rangle$$

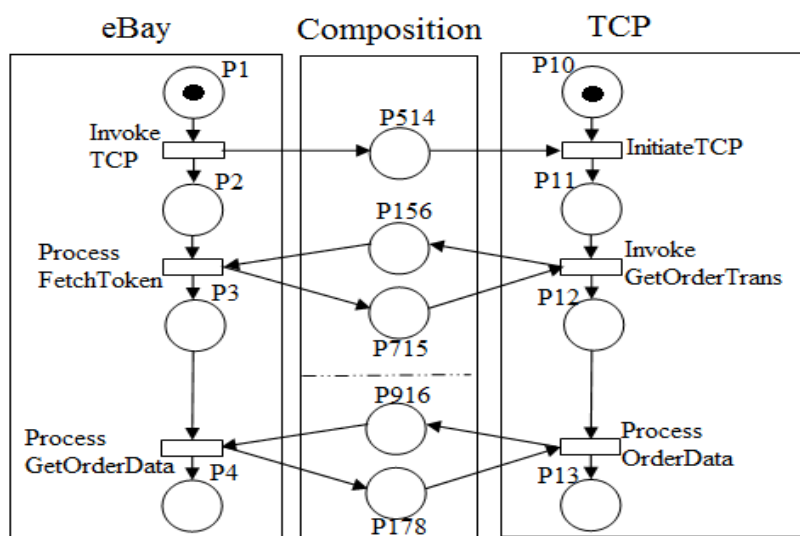
$$\text{Parallel: } E_1 \parallel E_2 = \langle (N_1 \cup N_2), h_1 \otimes h_2, l_1 \otimes l_2, d_1 \otimes d_2 \rangle$$

که  $\otimes$  ضرب کارتزین دو عملوند خود را نشان می دهد.

در مورد جبر سرویس ها، عملگر ترکیب دیگری نیز قابل تعریف است که در موجودیت شبکه های پتری تعریف نشده است. عملگر توازی با امکان ارسال پیام، در جبر سرویس ها را می توان بدین ترتیب تعریف نمود:

$$\text{Parallel with Message Passing: } E_1 \parallel_c E_2 = \langle (E_1 \ h'_1 \cup \ h'_2 \oplus \ l'_1 \cup \ l'_2 \ E_2), h''_1 \cup h''_2, l''_1 \cup l''_2, d_1 \cup d_2 \cup h'_1 \cup h'_2 \cup l'_1 \cup l'_2 \rangle$$

که  $h'_i$  ( $l'_i$ ) نشان گر سر و انتهای از  $E_i$  هستند که دارای پیام است و  $h''_i$  ( $l''_i$ ) نیز سر و انتهای از  $E_i$  را نشان می دهند که دارای پیام نیستند. برای سادگی، می توانیم یال هایی از ادغام را که دارای رؤوسی از هر دو شبکه هستند را بعنوان یال زائد حذف کنیم زیرا اطلاعات اضافه ای را نشان نمی دهند. اعمال عملگر توازی با امکان ارسال پیام بین دو سرویس، بر روی وب سرویس های شکل ۴-۴، شبکه پتری شکل ۴-۵ را نتیجه می دهد. عملگر ترکیب جدیدی می تواند بدون حضور یک مبادله گر، ارتباط میان دو وب سرویس را برقرار کند.



شکل ۴-۵- مدل موجودیت شبکه پتری ترکیبی برای وبسرویس های eBay و TCT.

## ۴-۴ شبکه های ریلی

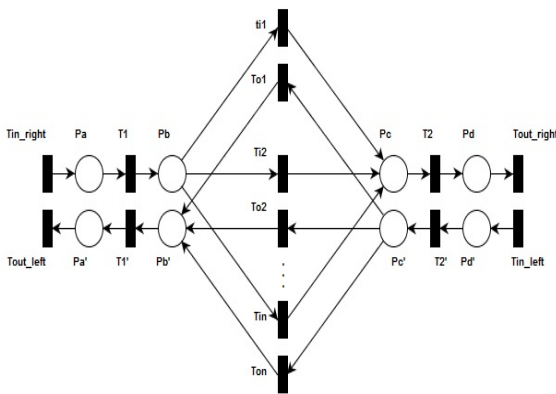
شبکه های ریلی بدلیل ایمنی بالا و مصرف انرژی پایین، سهم بزرگی در حمل و نقل افراد و کالاها برعهده دارند. با افزایش سطح نقل و انتقالات، کنترل شبکه ریلی اعم از کنترل گیت های ورود و خروج در ایستگاه ها پیچیده تر شده و نیاز به خودکارسازی آن احساس می شود. در خودکارسازی نیازمند روش هایی هستیم که بتوانند به تحلیل نیازمندی ها، طراحی، پیاده سازی و ارزیابی کارآیی این سیستم های پیچیده پردازند. در این بخش، به ارائه یک مدل فرمال مبتنی بر شبکه های پتری برای کنترل و تحلیل شبکه های ریلی خواهیم پرداخت. این مدل می تواند رخدادهای مهم یک شبکه ریلی را شبیه سازی کند. همچنین، این مدل به حل مسئله جلوگیری از برخورد و تصادم قطارها با وجود امکان قراردادن هم زمان بیش از یک قطار در یک خط ریلی می پردازد.

### ۴-۴-۱ مدل سازی شبکه

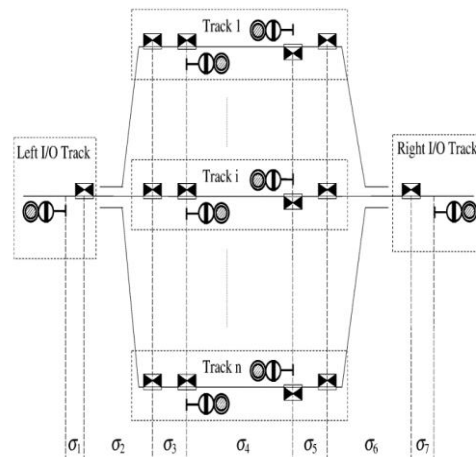
شبکه های ریلی، متشکل از ایستگاه ها و خطوط ریلی ارتباطی میان آنهاست. معمولاً، بین دو ایستگاه چندین خط ریلی ارتباطی قرار دارد. این خطوط در مسیر دو طرفه، ارتباط قطارها از یک



ایستگاه به ایستگاه بعد و بلعکس را برقرار می‌کنند. از این رو مدیریت ورود قطار به یک خط ریلی بسیار مهم است تا از تصادم پیشگیری شود. در مدل‌سازی یک شبکه‌های ریلی، ارتباط بین دو ایستگاه با استفاده از شبکه پتری کنترل می‌شود. یک ارتباط بین ایستگاهی ریلی  $n$ -خطی را در نظر بگیریم که شمای آن در شکل ۴-۶ آمده است.

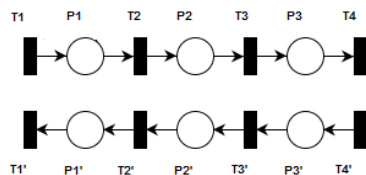


شکل ۴-۷- مدل تجریدی یک ایستگاه راه آهن با  $n$  خط ریلی.



شکل ۴-۶- شمای یک ارتباط بین ایستگاهی ایجاد شده با  $n$  خط ریلی [۱۳۴].

این شکل نشان‌دهنده محل قرارگیری سنسورها و سمافورها، در نقطه ورودی و خروجی هر ایستگاه و در نقاط ورود و خروج خطوط ریلی بین ایستگاهی است. مدل در نظر گرفته شده برای هر یک از خطوط ریلی در شکل ۴-۷ نمایش داده شده است. شبکه نشان‌دهنده حرکت به سمت راست در خط ریلی را  $N_1$  و شبکه نشان‌دهنده حرکت به سمت چپ را  $N_2$  می‌نامیم. هر یک از این دو شبکه یک فرآیند شبکه پتری تقریباً زنده، تقریباً بازگشت‌پذیر و تقریباً کراندار است.



شکل ۴-۸- مدل شبکه پتری یک خط ریلی واقع در یک ایستگاه.

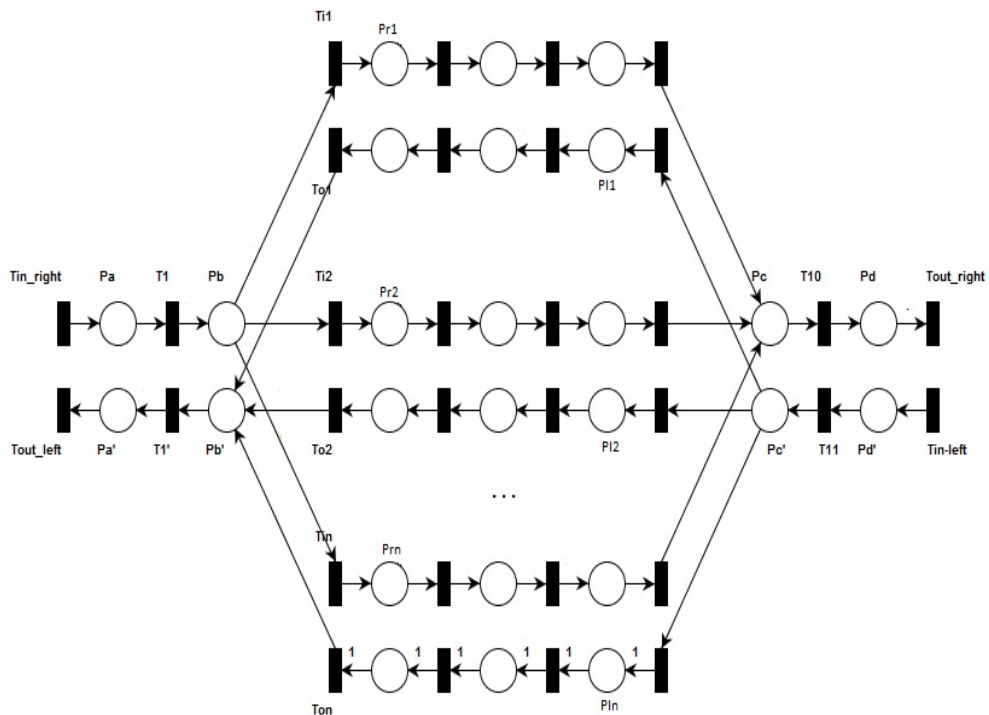
مدل کلی ایستگاه از سه بخش ورود به ایستگاه، ورود به یکی از  $n$  خط ریلی واقع در ایستگاه و خروج از ایستگاه (ورود به ایستگاه مجاور) تشکیل می‌شود که در شکل ۴-۸ آمده است. این شبکه پتری را  $N_3$  می‌نامیم. در این شکل، در موقع نشانه‌دار شدن مکان  $p_b$ ، گذرهای  $T_{i1}, T_{i2}, \dots, T_{in}$  فعال هستند. به محض رویداد یکی از گذرها، مکان  $p_b$  غیر نشانه‌دار خواهد شد و گذرهای  $T_{i1}, T_{i2}, \dots, T_{in}$  از حالت فعال خارج می‌شوند. مدل بدست آمده از پالایش شبکه پتری ایستگاه با استفاده از شبکه پتری خطوط ریلی نیز تقریباً زنده، تقریباً کراندار و تقریباً بازگشت‌پذیر است. این مدل ترکیبی در شکل ۴-۹ نشان داده شده است. برای بدست آوردن این مدل کلی از  $n$  پالایش استفاده شده است که:

$$M_1 = N_3 (T_{i1} \rightarrow N_1, T_{o1} \rightarrow N_2)$$

$$M_2 = M_1 (T_{i2} \rightarrow N_1, T_{o2} \rightarrow N_2)$$

...

$$N = M_n = M_{n-1} (T_{in} \rightarrow N_1, T_{on} \rightarrow N_2)$$



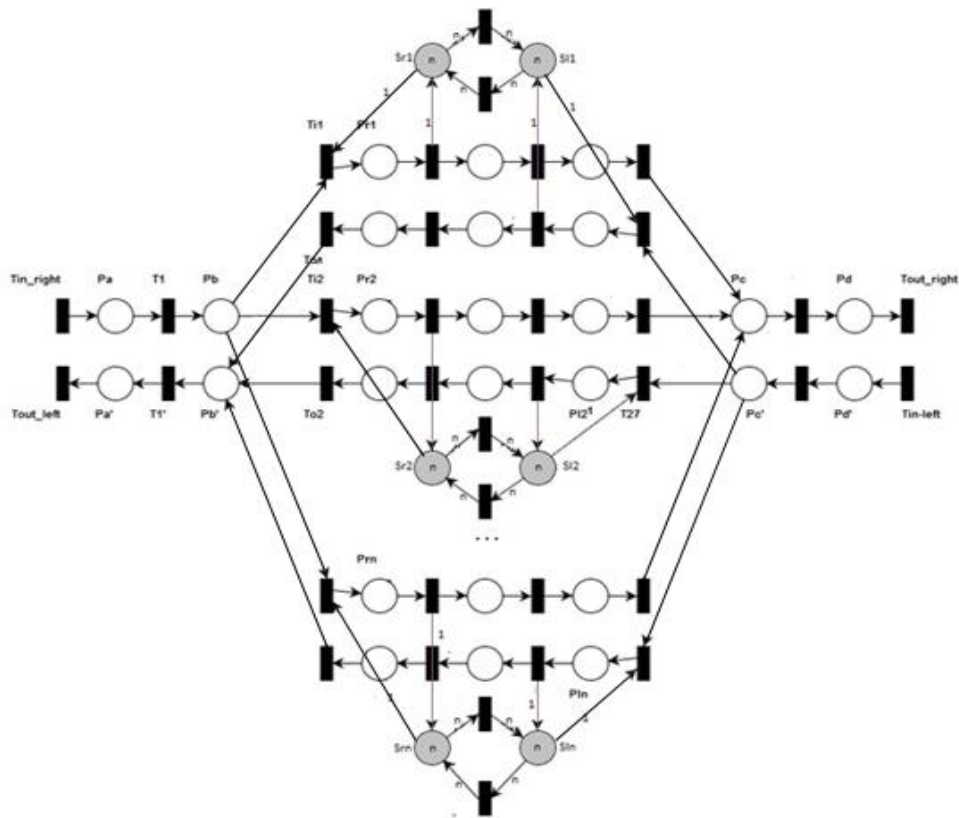
شکل ۴-۹- مدل شبکه پتری ایستگاه راه آهن با  $n$  خط ریلی پس از پالایش.

## ۴-۴-۲ تحلیل شبکه

بدلیل بهره‌گیری از طراحی پیمانهای و استفاده از عملگر پالایش که حافظ ویژگی‌های زنده بودن، کراندار و بازگشت‌پذیری است، مدل بدست آمده نهایی، تقریباً زنده، تقریباً کراندار و تقریباً بازگشت‌پذیر است و در نتیجه سیستم فاقد پدیده نامطلوب بن‌بست است. البته محدودیت‌های خاص سیستمی، برای جلوگیری از تصادم قابل تعریف است:

- حداکثر تعداد قطارهایی که در یک خط ریلی، هم جهت با هم، امکان حرکت دارند  $n$  است.
- اگر یک یا بیشتر قطار در یک جهت از خط ریلی قرار داشته باشند، قطاری در جهت مخالف نمی‌تواند وارد خط ریلی شود.

محدودیت اول بعنوان زیرشاخه‌ای از محدودیت دوم و در قالب آن قابل پیاده‌سازی است. برای بیان محدودیت دوم در هر یک از خطوط ریلی از دو مانیتور استفاده می‌کنیم. مانیتور اول، محدودیت ورود قطار به جهت راست و مانیتور دوم، محدودیت ورود قطار به جهت چپ خط ریلی را نشان می‌دهد. محدودیت اول بصورت  $R_1 + nR_2 \leq n$  و محدودیت دوم بصورت  $nR_1 + R_2 \leq n$  نمایش داده می‌شود که  $R_1$  و  $R_2$  بترتیب تعداد قطارهای موجود در جهت راست و جهت چپ خط ریلی را نشان می‌دهند. بدین ترتیب، اگر قطاری با جهت حرکتی سمت راست وارد خط ریلی شود،  $n-1$  قطار دیگر که همسو با قطار اول هستند نیز می‌توانند وارد خط ریلی شوند، و قطاری با جهت مخالف امکان ورود به این خط ریلی را ندارد مگر اینکه تمامی قطارها با جهت حرکت سمت راست از خط ریلی رد شده باشند. پس قطار مخالف یا باید از خطوط ریلی مجاور برای عبور استفاده کند و یا منتظر بماند. شبیه‌سازی این مدل در نرم افزار PIPE [۱۳۸] نیز نشان دهنده زنده بودن، کراندار بودن، بازگشت‌پذیری و عدم تصادم در مدل است. شکل ۴-۱۰ نشانگر مدل شبکه پتری به دست آمده در نرم افزار PIPE پس از اعمال کنترل نظارتی با توجه به نیازمندی‌های خاص مسئله است.



شکل ۴-۱۰- مدل شبکه پتری یک ایستگاه راه آهن  $n$  خطی پس از اعمال کنترل نظارتی

## ۴-۵ جمع بندی

در این فصل به ارائه مدل شبکه پتری برای نمونه‌هایی از سیستم‌های گسسته رخداد پرداختیم. فرآیندکاوی در سیستم‌های دارای لاگ فایل را با استفاده از یک روش تقسیم و حل که لاگ فایل را بصورت افقی به چندین زیرلاگ فایل تبدیل می‌کند، حل کردیم. وب سرویس‌ها نیز از جمله سیستم‌هایی هستند که برقرای ارتباط میانی آنها را می‌توان بعنوان وقوع یک رخداد در نظر گرفت. مدلسازی ترکیب وب سرویس‌ها با استفاده از عملگرهای سنتز شبکه‌های پتری از دیگر مثال‌های کاربردی این فصل بود. نهایتاً به مسئله کنترل خودکار یک شبکه حمل و نقل ریلی با استفاده از مدل شبکه‌های پتری اشاره شد.

## فصل ۵: محاسبات پیمانهای کنترل نظارتی

## ۵-۱ مقدمه

همانطور که در فصل ۳ بیان شد، روش‌های متفاوتی برای محاسبه توزیع شده کنترل نظارتی معرفی شد. در این سوپروایزرها نیز محاسبه پیمانهای گراف دسترسی که می‌تواند بعنوان یک بخش پایه‌ای منجر به کاهش پیچیدگی محاسباتی یک کنترل نظارتی شود، از این نظر مغفول مانده است. در این فصل به محاسبه پیمانهای کنترل نظارتی با استفاده از عملگرهای انتخاب، ترتیب و برگ‌برگ-سازی خواهیم پرداخت. در هر بخش، به ارائه قضایایی در زمینه امکان محاسبه پیمانهای کنترل نظارتی تحت این عملگرها و اثبات آنها خواهیم پرداخت. همچنین برای عملگر ادغام مکان‌ها اثبات می‌کنیم دانش حاصل از اطلاعات ادغام مکان‌ها می‌تواند در شرایط حفظ نشدن زنده بودن نیز منجر به کاهش محاسبات کنترل نظارتی شود.

## ۵-۲ محاسبات پیمانهای با استفاده از عملگر انتخاب

عملگر انتخاب، رابطه تصادم میان دو فرآیند شبکه پتری را نشان می‌دهد. فرض کنید  $B_1$  و  $B_2$  دو پیمان از یک سیستم باشند که با استفاده از عملگر انتخاب باهم در ارتباطند. در این بخش اثبات می‌کنیم کنترل نظارتی سیستم کلی ( $B = B_1 \square B_2$ ) می‌تواند بصورت پیمانهای محاسبه شود.

قضیه ۵-۱- فرض کنید  $B_1$  و  $B_2$  دو پیمان باشند که با استفاده از عملگر انتخاب در ارتباطند.  $B$  نیز نشانگر سیستم کلی پس از سنتز دو پیمان اولیه است. نگاشتی بین هر یک از نشانه-گذاری‌های  $B$  و نشانه‌گذاری‌های  $B_1$  یا  $B_2$  وجود دارد.

اثبات.

فرض کنید  $R_1(B_1, M_{01})$  و  $R_2(B_2, M_{02})$  گراف دسترسی دو پیمانانه باشند و فرض کنید  $M_{0i}'$  بجز در مکان ورودی، برابر با  $M_{0i}$  باشد  $(M_{0i}'(p_{ei}) = 0)$ . شبکه  $B_i$  در نشانه‌گذاری  $M_{0i}'$  گذری برای اجرا نخواهد داشت. گراف دسترسی شبکه  $B$  که از سنتز دو پیمانانه بدست آمده است، به شکل ذیل است:

$R(B, M_0) = \{M_0\} \cup R_1' \cup R_2'$  که  $M_0$  با استفاده از معادله (۲-۱۱) بدست می‌آید.  $R_1' = (0, M_1, M_{02}')$  و  $R_2' = (0, M_{01}', M_2)$ . این گزاره درست است زیرا اگر  $B_1$  برای اجرا انتخاب شود، تا زمان اتمام فرآیند  $B_1$ ، در  $M_{02}'$  باقی می‌ماند و بالعکس (شبکه  $B_i$  در نشانه‌گذاری  $M_{0i}'$  گذری برای اجرا نخواهد داشت و بدلیل ویژگی رقابت میان دو فرآیند، نشانه‌ای در مکان ورودی متصل به گذر ورودی این فرآیند وجود ندارد تا آن را فعال سازد). بنابراین نگاشتی بین نشانه‌گذاری‌های فرآیند شبکه پتری  $B$  و نشانه‌گذاری‌های  $B_1$  یا  $B_2$  وجود دارد.

■

قضیه ۵-۲- نشانه‌گذاری  $M_i$  یک FBM در  $B_i$  است، اگر و تنها اگر نشانه‌گذاری متناظر آن در  $M, B$  یک FBM در  $B$  باشد.

اثبات.

الف. اگر  $M_1 \in M_{FBM1}$  در  $B_1$  باشد، آنگاه  $(0, M_1, M_{02}') \in M_{FBM}$  در  $B$  است. زیرا،  $M_{02}'$  نمی‌تواند گذری را اجرا کند.  $M_1$  نیز یک FBM است و منجر به یک نشانه‌گذاری بد خواهد شد. بنابراین  $(0, M_1, M_{02}') \in M_{FBM}$  در  $B$  است. این نتیجه برای  $M_2 \in M_{FBM2}$  نیز برقرار است.

ب. اگر  $M \in M_{FBM}$  در  $B$  باشد، آنگاه،  $M$  به یکی از الگوهای  $(0, M_1, M_{02}')$  یا  $(0, M_{01}', M_2)$  نگاشته می‌شود. فرض کنیم  $M = (0, M_1, M_{02}')$ . با استفاده از برهان خلف داریم: فرض کنید نشانه‌گذاری متناظر با  $M$  در  $B_1$ ،  $M_1$ ، FBM از  $B_1$  نیست. بنابراین، این نشانه‌گذاری

پس از اجرای چند گذر به  $M_{01}$  دسترسی خواهد داشت. بنابراین  $M$  نیز نشانه‌گذاری بد نخواهد بود. (نشانه‌گذاری اولیه هر یک از پیمانها باید خود قانونی باشد). این نتیجه، یک تناقض است. برای  $M = (0, M_{01}', M_{02})$  نیز نتایج بطریق مشابهی قابل اثبات است.

■

قضیه ۳-۵- نشانه‌گذاری  $M_i$  یک نشانه‌گذاری قانونی در  $B_i$  است، اگر و تنها اگر نشانه‌گذاری متناظر آن در  $B, M$ ، یک نشانه‌گذاری قانونی در  $B$  باشد.

■ این قضیه به طریقی مشابه با روش اثبات قضیه ۲-۵ قابل اثبات است.

قضیه ۴-۵- یک مکان کنترلی، یک نشانه‌گذاری عضو مجموعه  $FBM_i$  در  $B_i$  را کنترل می‌کند اگر و تنها اگر، این مکان، نشانه‌گذاری متناظر را در مجموعه نشانه‌گذاری‌های غیرقانونی  $FBM$  مربوط به  $B$  کنترل کند.

اثبات.

الف. مکان‌های کنترلی یک  $FBM_i$  در  $B_i$  می‌توانند نشانه‌گذاری  $FBM$  متناظر را در  $B$  کنترل کنند: از قضیه ۳-۵ می‌دانیم مجموعه نشانه‌گذاری‌های قانونی در  $B$  از اجتماع نشانه‌گذاری‌های قانونی متناظر در  $B_1$  و  $B_2$  و نشانه‌گذاری اولیه بدست می‌آید (نشانه‌گذاری اولیه باید قانونی باشد، در غیر اینصورت تمامی نشانه‌گذاری‌ها غیرقانونی بوده و سیستم تحت نظارت دارای هیچ نشانه‌گذاری نخواهد بود). فرض کنیم  $M \in M_{FBM}^* = (0, M_1, M_{02}')$ . (بر طبق قضیه ۱-۳، از نشانه‌گذاری‌های  $FBM$  می‌دانیم که در  $M_{FBM}^*$  قرار ندارد صرف نظر می‌کنیم). پوشش نشانه‌گذاری  $M$  تنها دارای مکان‌های عملیاتی  $M$  است که نشانه‌گذاری‌های آنها تنها با نشانه‌گذاری‌های مکان‌های عملیاتی در نشانه‌گذاری‌های قانونی مقایسه خواهد شد.  $M_{02}'$  هیچ مکان عملیاتی نشانه‌داری ندارد. سه حالت برای  $M_l \in M_l$  متصور است:



حالت ۱-  $M_l = M_0 = (1, 0, M_{01}', 0, M_{02}')$  در این صورت هر دو نشانه‌گذاری  $M_0$  و  $M$

در مکان‌های  $B_2$  دارای مقادیر یکسانی هستند. معادله (۳-۲۷) بدین ترتیب خواهد بود:

$$\sum_{i=1}^{|P_1|} l_i(M'(p_i) - M(p_i)) + \sum_{i=1}^{|P_1|} l_i(0 - 0) \leq -1$$

که فقط به نشانه‌گذاری مکان‌های  $B_1$  وابسته است.

حالت ۲- نشانه‌گذاری متناظر  $M_l$  در  $B_1$  قرار دارد. بنابراین،  $M_l = (0, M_{l1}, M_{l20}')$  بر طبق

قضیه ۳-۳ تنها مکان‌های عملیاتی از  $M_l$  در نظر گرفته می‌شوند که در  $M \in M_{FBM}$  دارای نشانه هستند. مکان‌های  $M_{02}'$  در  $M$  نشانه‌دار نیستند. بنابراین در نشانه‌گذاری آنها  $M_l$  در نظر گرفته نمی‌شود.

حالت ۳- نشانه‌گذاری متناظر  $M_l$  در  $B_2$  قرار دارد. بنابراین  $M_l = (0, M_{l10}', M_{l2})$  بر طبق

قضیه ۳-۳ تنها مکان‌های عملیاتی از  $M_l$  در نظر گرفته می‌شوند که در  $M \in M_{FBM}$  دارای نشانه باشند. مکان‌های  $M_{02}'$  در  $M$  نشانه‌دار نیستند. در نتیجه نشانه‌گذاری‌های آنها در  $M_l$  در نظر گرفته نمی‌شوند.

ب- مکان کنترلی  $B$  می‌تواند نشانه‌گذاری‌های متناظر  $FBM_i$  در  $B_i$  را نیز کنترل کند. این

قسمت همانند قسمت الف قضیه قابل اثبات است.

■

از قضایای ۱-۵ الی ۴-۵ نتیجه می‌گیریم اگر برهم‌کنش پیمان‌های یک سیستم با استفاده از

عملگر انتخاب باشد، اعمال پیمان‌های کنترل نظارتی از نظر معنایی هم‌ارز با اعمال کنترل نظارتی در مدل کلی سیستم است.

## ۳-۵ محاسبات پیمان‌های کنترل نظارتی توسط عملگر ترتیب

عملگر ترتیب، رابطه ترتیب میان دو فرآیند شبکه پتری را نشان می‌دهد. فرض کنید  $B_1$  و  $B_2$  دو پیمانۀ از یک سیستم باشند که با استفاده از عملگر ترتیب با هم در ارتباطند. در این بخش اثبات می‌کنیم کنترل نظارتی سیستم کلی ( $B = B_1 \gg B_2$ ) می‌تواند بصورت پیمانۀ ای محاسبه شود.

قضیه ۵-۵- فرض کنید  $B_1$  و  $B_2$  دو پیمانۀ باشند که با استفاده از عملگر ترتیب در ارتباطند.  $B$  نیز سیستم کلی پس از سنتز دو پیمانۀ اولیه است. نگاشتی بین نشانه‌گذاری‌های  $B$  و نشانه‌گذاری‌های  $B_1$  یا  $B_2$  می‌توان یافت.

اثبات.

فرض کنید  $M_1$  و  $M_2$  بترتیب مجموعه تمام نشانه‌گذاری‌های دسترس‌پذیر در دو پیمانۀ  $B_1$  و  $B_2$  باشند. گراف دسترسی شبکه  $B$  که از سنتز دو پیمانۀ بدست آمده است، دارای نشانه‌گذاری‌های ذیل است:

$$R(B, M_0) = \{M_0\} \cup (M_1, M'_{02}) \cup (M'_{01}, M_2)$$

در هنگام شروع  $B$ ، فرآیند کلی دارای نشانه‌گذاری  $M_0$  است. در این حالت مکان ورودی فرآیند  $B_1$  نشانه‌دار است. در این زمان فرآیند  $B_2$  دارای نشانه‌گذاری  $M'_{02}$  خواهد بود و تمامی مکان‌های عملیاتی آن بدون نشانه هستند. در این حالت، هیچ گذری در  $B_2$  قابلیت اجرا نخواهد داشت. تا هنگام اتمام کامل فرآیند  $B_1$  و نشانه‌دار شدن مکان  $p_{x1}$  نشانه‌گذاری‌های  $B$  به فرم  $(M_1, M'_{02})$  هستند. پس از اتمام کامل  $B_1$ ، اجرای گذر  $\varepsilon$  منجر به نشانه‌دار شدن مکان ورودی  $B_2$  خواهد شد. در این هنگام  $B_1$  دارای نشانه‌گذاری  $M'_{01}$  خواهد شد و قابلیت اجرا ندارد. تا زمان اتمام کامل  $B_2$ ، نشانه‌گذاری‌های  $B$  به فرم  $(M'_{01}, M_2)$  هستند. پس از اتمام کامل فرآیند  $B_2$ ، با اجرای گذر  $t_a$  سیستم دوباره به حالت اولیه برخورد گشت؛ جایی که امکان فعال سازی گذرهای  $B_1$  وجود دارد و فرآیند  $B_2$  تا زمان اتمام روند  $B_1$ ، در  $M'_{02}$  باقی می‌ماند و بلعکس (شبکه  $B_i$  در نشانه‌گذاری  $M'_{0i}$  گذری برای

اجرا نخواهد داشت). بنابراین نگاشتی بین نشانه‌گذاری‌های  $B$  و نشانه‌گذاری‌های  $B_1$  یا  $B_2$  وجود دارد. ■

قضیه ۵-۶- فرض کنیم نشانه‌گذاری  $M_i$  یک FBM در  $B_i$  باشد، در این صورت نشانه‌گذاری متناظر آن در  $B, M$ ، یک FBM در  $B$  است و بلعکس. اثبات.

الف. اگر  $M_1 \in M_{FBM1}$  در  $B_1$ ، آنگاه  $(M_1, M_{02}') \in M_{FBM}$  در  $B$  است. زیرا،  $M_{02}'$  نمی‌تواند گذری را اجرا کند.  $M_1$  نیز یک FBM است و منجر به یک نشانه‌گذاری بد خواهد شد. این نتیجه برای  $M_2 \in M_{FBM2}$  نیز برقرار است.

ب. اگر  $M \in M_{FBM}$  در  $B$  باشد،  $M$  با یکی از الگوهای  $(0, M_1, M_{02}')$  یا  $(0, M_{01}', M_2)$  نگاشته می‌شود. فرض کنیم  $M = (0, M_1, M_{02}')$ . با استفاده از برهان خلف داریم: فرض کنید نشانه‌گذاری مرتبط با  $M$  در  $B_1, M_1$ ، یک نشانه‌گذاری قانونی است. بنابراین، این نشانه‌گذاری پس از اجرای چند گذر به  $M_{f1}$  دسترسی خواهد داشت. سپس سیستم وارد حالت  $(0, M_{01}', M_{02})$  می‌شود. این حالت متناظر با حالت  $M_{02}$  در  $B_2$  است. نشانه‌گذاری اولیه  $B_2$  الزاما زنده است. بنابراین  $M$  نیز نشانه‌گذاری بد نخواهد بود. زیرا حداقل یکی از مابدهای آن به  $M_{02}$  و سپس به  $M_0$  دسترسی دارد. که این یک تناقض است. در مورد  $M = (0, M_{01}', M_{02})$  نیز نتایج بطریق مشابهی قابل اثبات است. ■

قضیه ۵-۷- نشانه‌گذاری  $M_i$  یک نشانه‌گذاری قانونی در  $B_i$  است، اگر و تنها اگر نشانه‌گذاری متناظر آن در  $B, M$ ، یک نشانه‌گذاری قانونی در  $B$  باشد.

این قضیه به طریقی مشابه با روش اثبات قضیه ۵-۶ قابل اثبات است. ■

قضیه ۵-۸- یک مکان کنترلی، یک FBM در  $B_i$  را کنترل می‌کند اگر و تنها اگر، این مکان FBM را در  $B$  کنترل کند.

اثبات.

■ این قضیه به روشی مشابه با اثبات قضیه ۵-۴، قابل اثبات است.

از قضایای ۵-۵ الی ۵-۸ نتیجه می‌گیریم اگر پیمان‌های یک سیستم با استفاده از عملگر ترتیب برهم کنش داشته باشند، می‌توان کنترل نظارتی را بصورت پیمان‌های اعمال نمود.

## ۵-۴ اثبات محاسبات پیمان‌های کنترل نظارتی در برهم‌کنشی

### که امکان اجرای همزمان ندارد

تعریف ۵-۳- اجرای همزمان: دو پیمان شبکه پتری مبتنی بر سیستم‌های تولیدی، دارای ویژگی اجرای همزمان هستند اگر و تنها اگر حالتی در برهم کنش دو پیمان وجود داشته باشد که هر دو پیمان دارای مکان‌های عملیاتی نشانه‌داری باشند که برخی از گذرهای آنها را فعال سازد.

بطور کلی، می‌توان اثبات نمود اگر پیمان‌های مختلف سیستم دارای برهم‌کنشی باشند که اجرای همزمان آنها را ناممکن می‌سازد، محاسبات کنترل نظارتی بصورت پیمان‌های قابل اعمال است. به طریقی مشابه با قضایای ۵-۲ (۵-۳) می‌توان اثبات کرد اگر برهم‌کنشی قابلیت اجرای همزمان پیمان‌ها را فراهم نسازد، نشانه‌گذاری‌های غیرقانونی (قانونی) در پیمان‌ها متناظر با نشانه‌گذاری‌های غیرقانونی (قانونی) سیستم کلی است. همچنین، به طریقی مشابه با قضیه ۵-۴ می‌توان اثبات کرد مکان‌های کنترلی که برای کنترل نشانه‌گذاری‌های ممنوع در پیمان‌ها استفاده شده‌اند، قابلیت کنترل نشانه‌گذاری‌های ممنوع در سیستم کلی منتج از این برهم‌کنش را نیز دارند.

بعنوان مثال، برهم کنش‌های عمل-پیشوند و گذرگاه فرعی که در [۴۰] بیان شده است نیز دارای قابلیت اجرای همزمان پیمانها نیستند. در نتیجه، کنترل نظارتی آنها بصورت پیمانهای قابل محاسبه است.

در مورد اثبات امکان انجام محاسبات اعتبارسنجی ویژگی بازگشت‌پذیری در یک برهم کنش بصورت پیمانهای، به معادله ۳-۸ نشانه‌گذاری‌های قانونی توجه کنید. طبق این معادله، نشانه‌گذاری‌ای قانونی است که قابلیت دسترسی به نشانه‌گذاری اولیه را داشته و حداقل یکی از مابعدهای آن نیز این قابلیت را داشته باشد. این تعریف دقیقاً معادل تعریف بازگشت‌پذیری ارائه شده در تعریف ۲-۸ است. بنابراین قضایایی که برای اثبات محاسبات پیمانهای ویژگی زنده بودن بیان شد، در مورد ویژگی بازگشت‌پذیری نیز قابل اعمال هستند.

قضیه ۵-۹- یک مکان کنترلی، مکان غیرکراندار  $p_{ji}$  در فرآیند  $B$  را کنترل می‌کند اگر و تنها اگر مکان غیر کراندار متناظر را در  $B_i$  کنترل نماید.

اثبات.

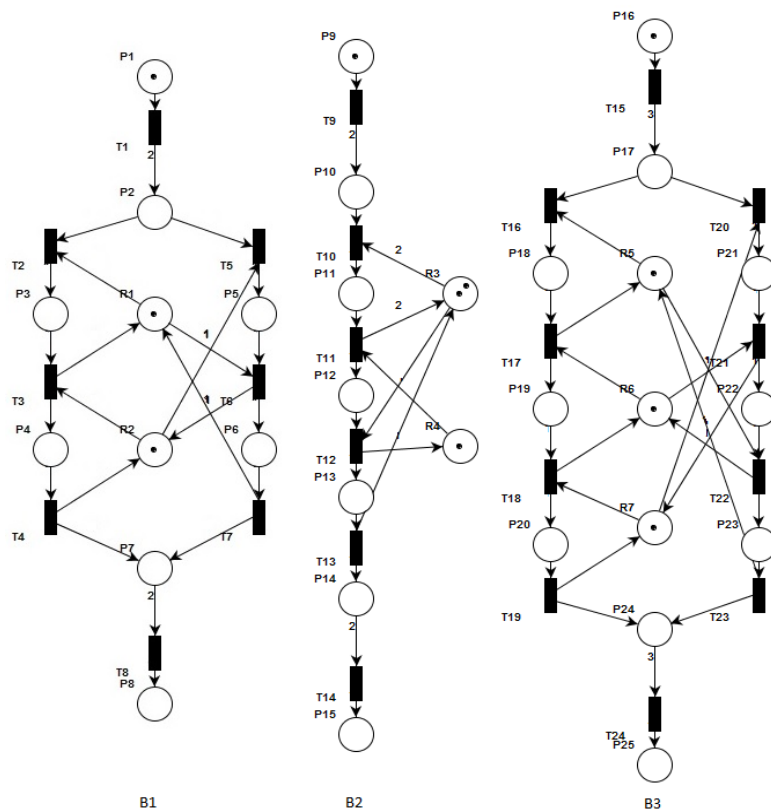
بدلیل عدم امکان اجرای همزمان پیمانها، یک مکان غیرکراندار در  $B_i$ ، در  $B$  نیز غیر کراندار است و بالعکس. محدودیت‌های اعمال شونده برای بیشینه ظرفیت این مکان در  $B_i$  می‌تواند برای تضمین کراندار بودن این مکان در  $B$  نیز استفاده شود و بالعکس.



## ۵-۴-۱ مثال کاربردی

سیستمی متشکل از سه پیمان پایه‌ای  $B_1$ ،  $B_2$  و  $B_3$  را در نظر بگیرید که بصورت  $B_3 \gg (B_1 \square B_2)$  با هم برهم‌کنش دارند. شکل ۵-۱ این پیمانها را نشان می‌دهد. گراف دسترسی سیستم کلی دارای ۷۵ حالت و ۱۰۶ کمان است. تحلیل فضای حالت بزرگ می‌تواند زمانبر

باشد. از آنجایی که برهم‌کنش پیمان‌ها، اجازه اجرای همزمان آنها را نمی‌دهد، می‌توان محاسبات کنترل نظارتی را بصورت پیمان‌های انجام داد. با اعمال کنترل نظارتی به مدل شبکه پتری شبکه  $B_1$  با ۱۵ نشانه‌گذاری قانونی و یک نشانه‌گذاری ممنوع، مکان کنترلی  $C_1$  به سیستم اضافه خواهد شد. گراف دسترسی مربوط به شبکه پتری متناظر با پیمان  $B_2$  دارای ۱۲ حالت و ۱۲ کمان است. یکی از حالات این گراف ممنوع است. اضافه نمودن مکان کنترلی  $C_2$  منجر به اعمال خوش رفتاری در این شبکه خواهد شد. گراف دسترسی مربوط به پیمان  $B_3$  دارای ۵۶ نشانه‌گذاری است که ۳۸ نشانه-گذاری قانونی و بقیه ممنوع هستند. این شبکه تنها دارای ۶ FBM کمینه تحت پوشش است که البته تعداد سه تا از این FBMها تکراری بوده و توسط FBMهای دیگری قابل کنترلند. با استفاده از معادلات (۳-۶) و (۳-۷)، ماتریس برخورد و نشانه‌گذاری اولیه هر یک از مکان‌های کنترلی محاسبه شده و در جدول ۵-۱ نشان داده شده‌اند. پس از اعمال کنترل نظارتی بر پیمان‌ها، بر طبق برهم‌کنش مطرح شده سیستم کلی بدست خواهد آمد که نیازی به اعمال کنترل نظارتی برای تضمین زنده بودن آن نیست.



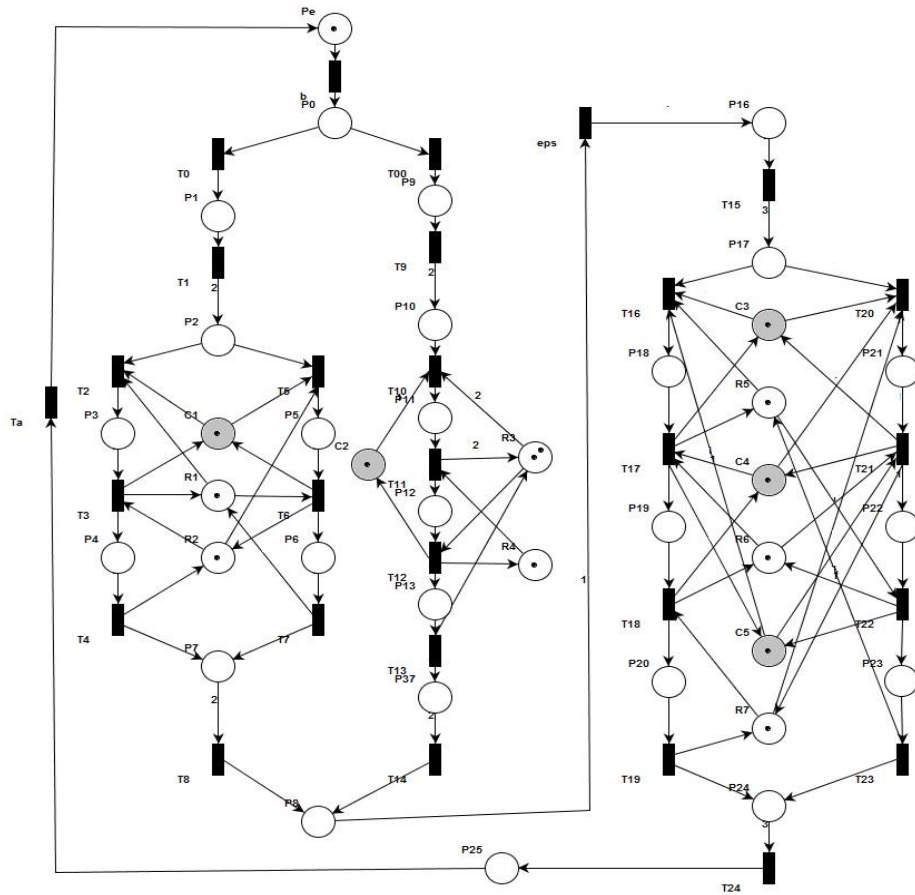
شکل ۱-۵ مدل شبکه پتری پیمانه‌های  $B_1$ ،  $B_2$  و  $B_3$  در سیستم مثال بخش ۵-۴-۱.

جدول ۱-۵- نشانه‌گذاری اولیه و ماتریس برخورد مکان‌های کنترلی محاسبه شده در مثال بخش ۵-۴-۱.

پیمانه	مکان کنترلی	نشانه‌گذاری اولیه	گذرهای مابعد	گذرهای ماقبل
$B_1$	$C_1$	1	$\{t_2, t_5\}$	$\{t_3, t_6\}$
$B_2$	$C_2$	1	$\{t_{10}\}$	$\{t_{12}\}$
$B_3$	$C_3$	1	$\{t_{16}, t_{20}\}$	$\{t_{17}, t_{21}\}$
	$C_4$	1	$\{t_{17}, t_{20}\}$	$\{t_{18}, t_{21}\}$
	$C_5$	1	$\{t_{16}, t_{21}\}$	$\{t_{17}, t_{22}\}$

با آزمایش این سیستم با استفاده از نرم افزار PIPE [۱۳۹] نیز مدل کلی سیستم، زنده، کراندار

و بازگشت پذیر است. این مدل در شکل ۲-۵ نشان داده شده است.



شکل ۲-۵ مدل خوش رفتار شبکه پتری برای سیستم کلی پس از اعمال کنترل نظارتی روی پیمانها و سنتز آنها.

در این شکل مکان های کنترلی با استفاده از رنگ طوسی از دیگر مکان های سیستم مجزا شده-

اند. اعمال کنترل نظارتی متمرکز برای این سیستم در کامپیوتری با مشخصات پروسسور Intel(R)

Pentium(R) Dual CPU T2410 @ 2.00GHz و ۲ گیگابایت حافظه RAM و سیستم عامل ۳۲

بیتی ویندوز ۱۰، ۲/۵۴۰ ثانیه و اعمال آن بصورت پیمانهای ۱/۸۷۹ ثانیه زمان برده است.



## ۵-۵ محاسبات پیمانه‌ای کنترل نظارتی با استفاده از عملگر

### برگ‌برگ‌سازی

عملگر برگ‌برگ‌سازی، امکان اجرای موازی فرآیندهای تشکیل دهنده یک سیستم، بدون ارتباط پیامی میان آن‌ها را فراهم می‌سازد. در این بخش به اثبات امکان انجام محاسبات پیمانه‌ای کنترل نظارتی در مورد عملگر برگ‌برگ‌سازی خواهیم پرداخت.

قضیه ۵-۱۰- فرض کنید  $B_1$  و  $B_2$  دو پیمانه باشند که با استفاده از عملگر برگ‌برگ‌سازی برهم‌کنش دارند و  $B$  سیستم کلی بدست آمده از سنتز این دو پیمانه است. اعمال پیمانه‌ای کنترل نظارتی بر  $B$  هم‌ارز با اعمال سوپروایزر روی سیستم کلی است.

اثبات

فرض کنید  $R_1(B_1, M_{01})$  و  $R_2(B_2, M_{02})$  بترتیب گراف‌های دسترسی  $B_1$  و  $B_2$  بوده و  $M_{0i}' = (0, M_{0i})$ . گراف دسترسی سیستم کلی به شکل  $R(B, M_0) = \{M_0\} \cup \{M_f\} \cup R_1' \times R_2'$  است. مکان‌های  $p_e$  و  $p_x$  بترتیب تنها در نشانه‌گذاری‌های  $M_0$  و  $M_f$  دارای نشانه هستند.

فرض کنید مکان‌های  $B_1$  ( $B_2$ ) دارای نشانه‌گذاری  $M_{j1}$  ( $M_{l2}$ ) هستند. اجرای گذر فعال  $t_{k1}$  ( $t_{k2}$ ) در  $B_1$  ( $B_2$ ) سیستم کلی را در نشانه‌گذاری  $(0, M_{(j+1)1}, M_l, 0)$  ( $(0, M_{j1}, M_{(l+1)2}, 0)$ ) قرار خواهد داد. بنابراین نگاهی بین نشانه‌گذاری‌های  $B$  و نشانه‌گذاری‌های  $B_1$  و  $B_2$  وجود دارد.

نشانه‌گذاری  $M = (0, M_1, M_2, 0)$  یک FBM در  $B$  است، اگر و تنها اگر  $M_1$  یک نشانه-گذاری FBM در  $B_1$  و یا  $M_2$  یک نشانه‌گذاری FBM در  $B_2$  باشد. زیرا در آن صورت نمی‌توان از پیمانه‌های مربوطه به نشانه‌گذاری اولیه آن دست یافت و  $M_0$  قابل دسترس نخواهد بود. نکته قابل توجه اینکه که برای FBM بودن یک نشانه‌گذاری در  $B$ ، FBM بودن یکی از نشانه‌گذاری‌های متناظر

در  $B_i$  کافی است. زیرا، در آن صورت نشانه‌گذاری دیگر امکان پیش رفتن تا چند مرحله و ایجاد مابعدها را دارد. اما بدلیل همزمانی نهایی در اجرای گذر  $\varepsilon_2$ ، و تأمین نشدن نشانه‌گذاری یکی از مکان‌های آن، امکان بازگشت به  $M_0$  وجود ندارد.

یک مکان کنترلی، نشانه‌گذاری  $M$  در  $B$  را کنترل می‌کند اگر مکان کنترلی متناظری وجود داشته باشد که FBM متناظر را در پیمانانه مربوطه کنترل کند. بنابراین معادله (۳-۲۷) می‌تواند بصورت ذیل نوشته شود:

$$\sum_{i=1}^{|P1|} l_i(M'(p_i) - M_1(p_i)) + \sum_{j=1}^{|P2|} l_j(M'(p_j) - M_2(p_j)) \leq -1$$

این معادله می‌تواند به دو معادله تقسیم شود (زیرا مکان‌های شبکه کلی قابل تقسیم به مکان‌های یکی از دو پیمانانه است و پیمانانه‌ها جز در حالت شروع و پایان برهم‌کنشی با یکدیگر ندارند):

$$\sum_{i=1}^{|P1|} l_i(M'(p_i) - M_1(p_i)) \leq -1$$

و

$$\sum_{j=1}^{|P2|} l_j(M'(p_j) - M_2(p_j)) \leq -1$$

این معادلات با استفاده از اضافه کردن مکان‌های کنترلی در پیمانانه‌های  $B_1$  و  $B_2$  برآورده می‌شوند.

■

قضیه ۵-۱۰ نشان می‌دهد کنترل نظارتی پیمانانه‌هایی که دارای برهم‌کنش برگ‌برگ‌سازی هستند، می‌تواند بصورت پیمانانه‌ای محاسبه شود. اعمال کنترل نظارتی پیمانانه‌ای از نظر محاسباتی بسیار کارآتر از محاسبه کنترل نظارتی در سیستم کلی است.

روش‌های پیمانانه‌ای مطرح شده در این ۴ بخش، از سیاست‌های کنترل نظارتی بیشینه مجازیت استفاده می‌کنند. البته در این روش‌ها بدلیل استفاده از ویژگی  $M$ -مرتبط بودن نمی‌توان از روش‌های

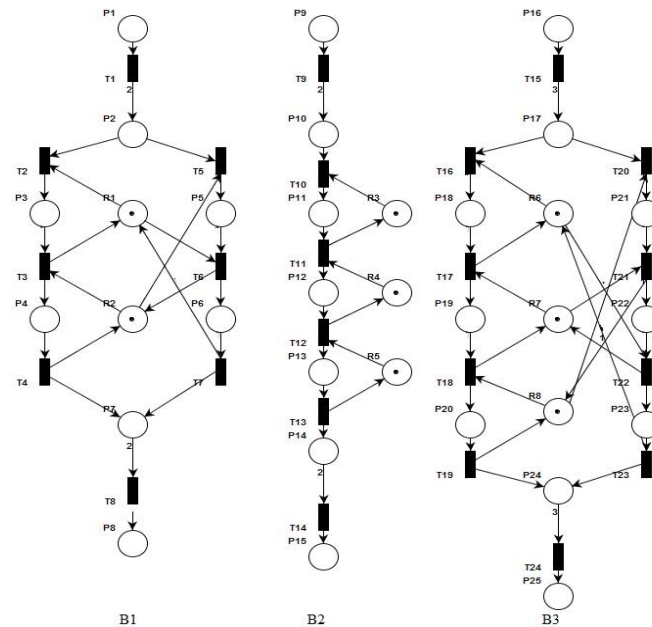
کاهش ساختار مانند MCPP استفاده نمود. در مورد سه بخش ابتدایی، با استفاده از مثال عملی به مشاهده بهبود پیچیدگی محاسباتی روش‌های کنترل نظارتی پیمان‌های نسبت به روش‌های کنترل نظارتی متمرکز پرداختیم. برای بحث دقیقتر در مورد پیچیدگی محاسباتی روش‌های پیمان‌های، پیچیدگی زمانی این روش‌ها را با روش‌های متمرکز مقایسه می‌کنیم. در صورت استفاده از سیاست کنترل نظارتی [۴۸] برای محاسبه پیمان‌های یک سوپروایزر، جدول ۵-۲ به بررسی این پیچیدگی‌ها می‌پردازد. البته، در مورد عملگر برگ‌برگ‌سازی تعداد نشانه‌گذاری‌های غیرقانونی (نشانه‌گذاری‌های قانونی) سیستم کلی، با ضرب تعداد نشانه‌گذاری‌های غیرقانونی (نشانه‌گذاری‌های قانونی) در پیمان‌های مختلف مرتبط است که منجر به کاهش بیشتر بار محاسباتی آن خواهد شد. بنابراین می‌توان از روش‌های کنترل نظارتی پیمان‌های برای کنترل سیستم‌های بزرگ مقیاس استفاده نمود. این روش‌ها علاوه بر کاهش پیچیدگی محاسباتی، از مسئله انفجار فضای حالت نیز پیشگیری می‌نمایند.

جدول ۵-۲- مقایسه پیچیدگی زمانی روش‌های کنترل نظارتی پیمان‌های و متمرکز.

سیاست	پیچیدگی زمانی	پیچیدگی حافظه
کنترل نظارتی پیمان‌های	$\sum_{i=1}^{Modules}  M_{FBM_i}^*  \cdot  M_{L_i}^*  \cdot  P_{A_i} $	$\sum_{i=1}^{Modules}  P_{A_i}  \cdot ( M_{L_i}^*  +  M_{FBM_i}^* )$
کنترل نظارتی متمرکز	$ M_{FBM}^*  \cdot  M_L^*  \cdot  P_A $	$ P_A  \cdot ( M_L^*  +  M_{FBM}^* )$

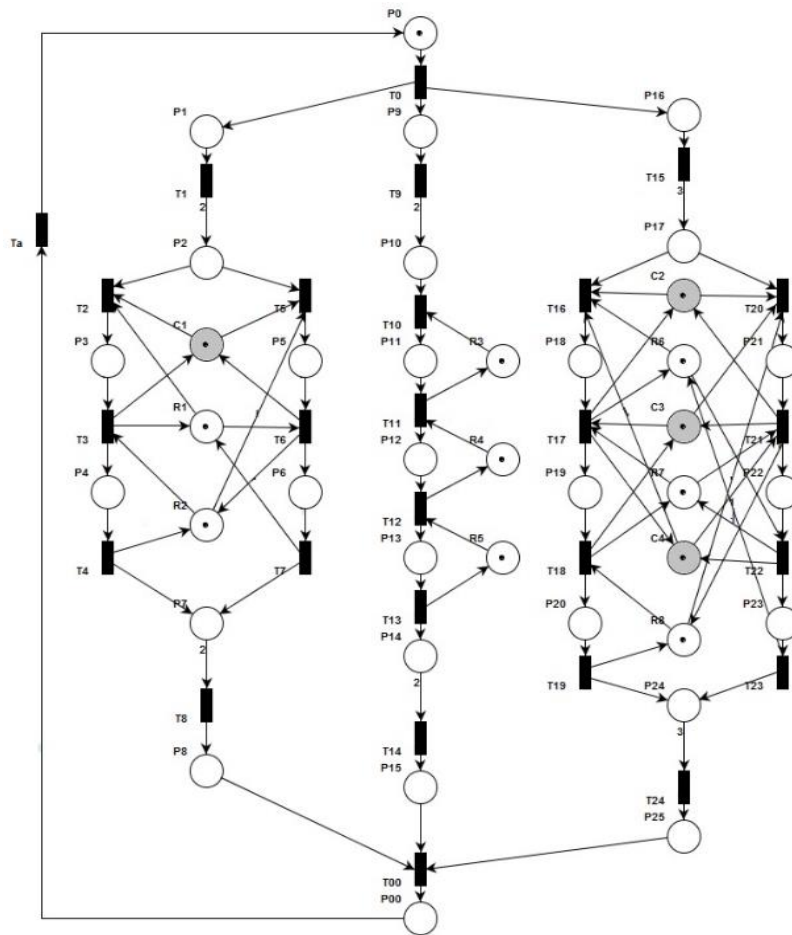
به همین سبب استفاده از کنترل نظارتی پیمان‌های بر روی این عملگر بسیار مقرون به صرفه‌تر از استفاده از کنترل نظارتی متمرکز است. در ادامه، مثالی را بررسی می‌کنیم که محاسبات سوپروایزری در آن با استفاده از کنترل نظارتی متمرکز در کامپیوتری با مشخصات پروسسور Intel(R) Pentium(R) Dual CPU T2410 @ 2.00GHz و ۲ گیگابایت حافظه RAM و سیستم عامل ۳۲ بیتی ویندوز ۱۰، دچار مسئله انفجار فضای حالت خواهد شد. اما با استفاده از کنترل نظارتی پیمان‌های با سرعت بالا و در زمان ۸/۶۲۹ ثانیه قابل حل است.

## ۵-۵-۱ مثال کاربردی



شکل ۳-۵ مدل شبکه پتری پیمانانه‌های پایه‌ای  $B_1$ ،  $B_2$  و  $B_3$  در مثال بخش ۵-۵-۳.

سیستمی را در نظر بگیرید که از سه پیمانانه پایه  $B_1$ ،  $B_2$  و  $B_3$  تشکیل شده است که با استفاده از عملگر برگ‌برگ‌سازی برهم کنش دارند. شکل ۳-۵ این پیمانانه‌ها را نشان می‌دهد. گراف دسترسی سیستم کلی دارای ۱۰۹۵۰ حالت است. تحلیل این گراف دسترسی بزرگ می‌تواند منجر به انفجار فضای حالت شود. سوپروایزر اعمال کننده زنده بودن می‌تواند بصورت پیمانانه‌ای محاسبه شود. پیمانانه‌های  $B_1$ ،  $B_2$  و  $B_3$  بترتیب دارای ۱۷، ۱۴ و ۴۶ حالت و دارای ۱، ۰ و ۶ عدد FBM کمینه تحت پوشش هستند. تحلیل این شبکه‌های کوچک بسیار کارآتر از تحلیل سیستم کلی با گراف دسترسی بزرگ است. پس از اعمال کنترل نظارتی به پیمانانه‌های غیرزنده  $B_1$  و  $B_3$  بترتیب ۱ و ۳ مکان کنترلی به این دو پیمانانه اضافه خواهد شد. سنتز پیمانانه‌های زنده نهایی، منجر به سیستمی خواهد شد که نمای آن در نرم‌افزار PIPE [۱۳۹] در شکل ۴-۵ نشان داده شده است. در این شکل مکان‌های کنترلی  $C_1 - C_4$  با رنگ طوسی از دیگر مکان‌ها مجزا شده‌اند. این سیستم، بدون بن‌بست است.



شکل ۴-۵ مدل شبکه پتری سیستم کلی پس از اعمال کنترل نظارتی روی پیمانهای مجزا و سنتز آنها.

## ۵-۶ استفاده از عملگر ادغام مکان‌ها در کاهش محاسبات

### کنترل نظارتی

محاسبات کنترل نظارتی پیمانهای، منجر به کاهش اندازه گراف دسترسی مورد بررسی و در نتیجه پیشگیری از وقوع پدیده انفجار فضای حالت خواهد شد. همچنین، پیچیدگی محاسباتی الگوریتم‌های کنترل نظارتی تا حد زیادی به تعداد نشانه‌گذاری‌های قانونی و نشانه‌گذاری‌های غیرقانونی مرزی وابسته است. با استفاده از کنترل نظارتی پیمانهای و کاهش اندازه این دو مجموعه،

پیچیدگی محاسباتی الگوریتم‌های کنترل نظارتی تا حد زیادی کم خواهد شد. یکی از برهم کنش‌های متداول میان پیمان‌های یک سیستم تخصیص منابع، برهم کنش ادغام مکان‌های منبع میان دو یا چند فرآیند است [۵۲]. در این بخش ثابت می‌کنیم، با استفاده از شرایط حفظ زنده بودن در این عملگر می‌توان در یک سیاست کنترل نظارتی مبتنی سایفون تعداد سایفون‌های مورد کنترل را کاهش داد.

قضیه ۵-۱۱- در یک شبکه  $S^3PGR$  اگر یک سایفون شامل یک و تنها یک مکان اشتراکی باشد، این سایفون، الزاماً تله نشانه‌دار نیز هست.

اثبات.

طبق تعریف سایفون، برای یک سایفون مانند  $S$  داریم:  $\cdot S \subseteq S^*$ . فرض کنید  $S$  شامل یک و تنها یک مکان اشتراکی باشد، داریم:  $\cdot S = \cdot p_r \cup \cdot P_o$  که  $p_r$  مکان اشتراکی مربوطه و  $P_o$  زیرمجموعه‌ای از مکان‌های عملیاتی و مکان‌های بیکار است که در سایفون شرکت دارند. بنابراین  $\cdot S = T_{\text{release}} \cup \cdot P_o$ . همچنین داریم:  $S^* = T_{\text{allocation}} \cup P_o^*$  که  $T_{\text{allocation}}$  و  $T_{\text{release}}$  بترتیب، نشان دهنده مجموعه گذرهای ورودی به مکان اشتراکی  $p_r$  که نشان دهنده آزاد شدن آن توسط یک فرآیند هستند، و مجموعه گذرهای خروجی از مکان اشتراکی  $p_r$  که نشان دهنده تخصیص یافتن یک منبع  $p_r$  به روندی از یک فرآیند هستند می‌باشند. بنابراین داریم:  $T_{\text{release}} \cup \cdot P_o \subseteq T_{\text{allocation}} \cup P_o^*$  (I)

از طرفی، در یک شبکه  $S^3PGR$  گذرهای مربوط به تخصیص داده (آزاد) شدن مکان منبع الزاماً دارای یک ما قبل (مابعد) در مکان‌های عملیاتی است. زیرا عملیات تخصیص داده شدن و یا آزاد شدن منبع توسط یک مرحله از یک فرآیند انجام می‌شود که با استفاده از مکان‌های عملیاتی نشان داده

می‌شود. همچنین، مکان‌های کنترلی به کنترل مکان‌های عملیاتی می‌پردازند. بنابراین داریم:

$$(II) T_{\text{allocation}} \subseteq \bullet P_0 \text{ و } T_{\text{release}} \subseteq P_0^*$$

با استفاده از (I) و (II) نتیجه می‌گیریم  $S = S^*$ . بنابراین این سایفون یک تله نیز هست. از

طرفی از آنجا که این تله دارای یک مکان اشتراکی است، پس الزاماً نشانه‌دار است. ■

قضیه ۵-۱۲- در یک شبکه پتری  $S^3PGR$ ، کنترل سایفون‌هایی که دارای حداقل دو مکان

اشتراکی هستند، برای اعمال زنده بودن کافی است.

اثبات.

طبق تعریف ۲-۱۹ می‌دانیم اگر پیمانه‌های یک سیستم دارای ۳ شرط باشند، اعمال عملگر

ادغام مکان‌ها در پیمانه‌ها حافظ زنده بودن است. یک شبکه پتری  $S^3PGR$  را در نظر بگیرید (شبکه

پتری  $N'$ ). اگر به تعداد فرآیندهای این شبکه، مکان‌های منبع را تکرار کنیم و هر یک از این مکان‌های

تکراری را به یک فرآیند تخصیص دهیم، مدل شبکه پتری تجزیه شده<sup>۱</sup> بدست می‌آید (شبکه پتری

$N$ ). اعمال عملگر ادغام مکان‌های منبع یکسان روی شبکه پتری  $N$ ، شبکه پتری  $N'$  را نتیجه می‌دهد.

این مکان‌ها از پیمانه‌های متفاوت در شبکه پتری  $N$  انتخاب شده‌اند، بنابراین غیر مجاورند. از طرفی

شرط دوم حفظ زنده بودن عملگر ادغام مکان‌ها نیز برقرار است؛ یعنی شبکه پتری  $N'$  که یک شبکه

پتری  $S^3PGR$  است، قابل تبدیل به پیمانه‌های  $SM$  است. (در مورد شبکه‌های  $S^4PR$  این شرط برقرار

نیست. زیرا در این شبکه‌ها در هر مرحله از هر فرآیند می‌توان بیش از یک منبع را مورد استفاده قرار

داد، بنابراین پیمانه‌های بدست آمده الزاماً  $SM$  نیستند). همچنین شبکه پتری  $N$  دارای ویژگی

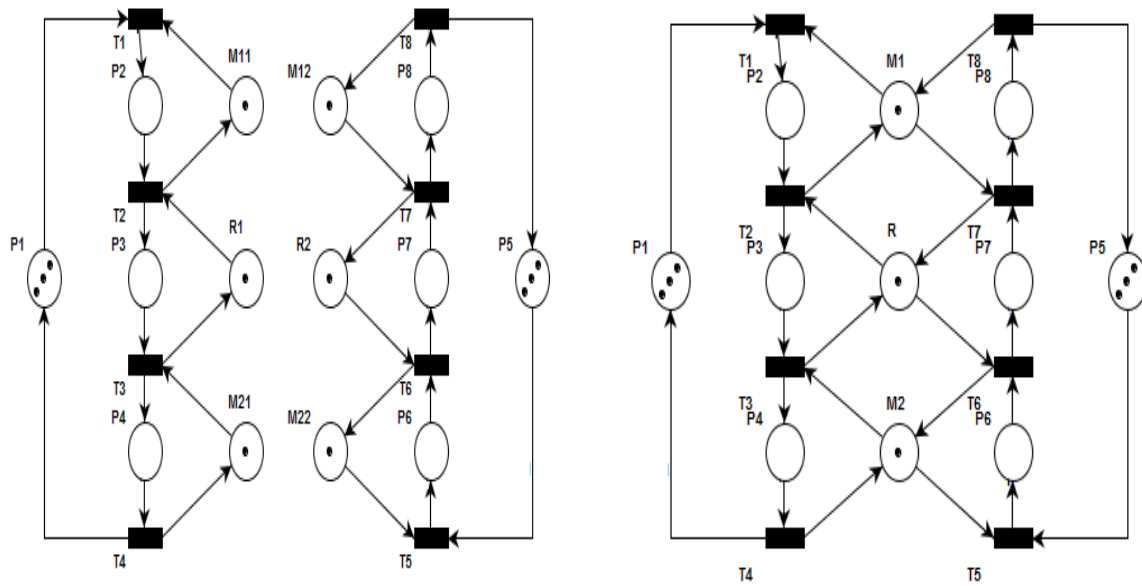
سایفون دارای تله نشانه‌دار با کاهش تکرار مجموعه است. زیرا، در هر یک از پیمانه‌های  $N$ ، سایفون-

هایی که دارای مکان منبع نیستند، دارای تله نشانه‌دار هستند (هر یک از فرآیندهای یک شبکه پتری

<sup>۱</sup> Decomposed PN

$S^3PGR$  یک ماشین وضعیت قویا همبند است. بنابراین زنده بوده و سایفون‌های آن دارای تله نشانه‌دار است). بعلاوه، سایفون‌هایی که دارای یک مکان منبع هستند طبق قضیه (۵-۱۱) دارای تله نشانه‌دار هستند. بنابراین، تنها نیاز به کنترل سایفون‌های کمینه‌ای است که شامل حداقل دو مکان کنترلی هستند.

■



شکل ۵-۶- مدل تجزیه شده شبکه پتری شکل ۵-۵ به دو پیمانه.

شکل ۵-۵- یک مدل شبکه پتری  $S^3PR$  (زیر کلاسی از شبکه‌های پتری  $S^3PGR$ ).

اساس اثبات قضیه (۵-۱۲) بر تجزیه یک شبکه پتری  $S^3PGR$  به پیمانه‌های تشکیل دهنده بنا شده است. شکل ۵-۵ یک شبکه پتری  $S^3PGR$  و شکل ۶-۵ تجزیه این شبکه پتری به پیمانه‌ها را بر اساس روال تجزیه پیشنهادی ارائه می‌نماید. بر اساس قضیه ۵-۱۲ پس از محاسبه سایفون‌های دارای دو یا بیشتر مکان کنترلی، به محاسبه سایفون‌هایی از این مجموعه می‌پردازیم که کمینه، مقدماتی و ضروری هستند. سپس، برای هر یک از این سایفون‌ها به محاسبه مکان کنترلی برای جلوگیری از بدون نشانه شدن آنها خواهیم پرداخت. برای این منظور از معادله (۵-۱) استفاده می‌کنیم. در این معادله،  $S(p_i) - 1$  بترتیب ضرایب و محدودیت معادله  $GMEC$ ، یعنی  $L$  و  $b$ ، را



نشان می‌دهند. بدین ترتیب به محاسبه مکان‌کنترلی با استفاده از معادلات (۳-۶) و (۳-۷) می‌پردازیم.

$$\sum_{i=1}^n S(p_i) \cdot M \geq 1 \Rightarrow \sum_{i=1}^n -S(p_i) \cdot M \leq -1 \quad (5-1)$$

پس از کنترل تمامی سایفون‌های کمینه، مقدماتی، ضروری و بدون تله توسط PI، و افزودن مکان‌های کنترلی به شبکه پتری، ممکن است سایفون‌های کنترل نشده جدیدی ایجاد شوند. بصورت تکرار شونده به کنترل این سایفون‌ها می‌پردازیم. الگوریتم ۵-۱ روند محاسبه کنترل نظارتی کاهش یافته برای یک شبکه پتری  $S^3PGR$  با استفاده از ادغام مکان‌ها را نشان می‌دهد.

الگوریتم ۵-۱- محاسبات کنترل نظارتی برای شبکه‌های پتری  $S^3PGR$

ورودی: یک شبکه پتری  $(N, M_0) S^3PGR$

خروجی: شبکه پتری کنترل شده

۱-  $S_1 = \emptyset$  نشان‌دهنده مجموعه سایفون‌های شامل حداقل دو مکان اشتراکی شامل مکان‌های

منبع و مکان‌های کنترلی است که تا کنون بررسی شده‌اند.

۲- تمامی سایفون‌های شامل حداقل دو مکان اشتراکی را محاسبه نموده و در مجموعه  $S_2$  قرار

دهید. از مجموعه  $S_2$  سایفون‌هایی را که تاکنون بررسی شده‌اند را حذف کنید.  $S_2 = S_2 \setminus S_1$ .

۳- مجموعه سایفون‌های بررسی شده را به روز رسانی کنید.  $S_1 = S_1 \cup S_2$ .

۴- سایفون‌های غیرکمینه و تکراری و مغلوب را از  $S_2$  حذف کنید.

۵- اگر  $S_2 = \emptyset$  به مرحله ۶ بروید. در غیر اینصورت:

۵-۱- هر یک از سایفون‌های درون مجموعه  $S_2$  را با طراحی یک PI کنترل کنید. مکان‌های

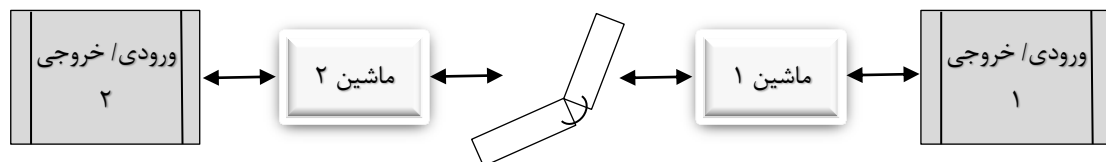
کنترلی جدید را به شبکه پتری اضافه کنید و به مرحله ۲ بروید.

۶- شبکه پتری جدید را بعنوان یک شبکه پتری زنده اعلام کنید.

در این الگوریتم  $S_1$  و  $S_2$  بترتیب بیانگر مجموعه تمام سایفون‌های غالب دارای بیش از یک مکان اشتراکی که تاکنون پردازش شده‌اند و مجموعه سایفون‌های پردازش نشده در یک مرحله هستند. الگوریتم ۵-۱ دارای پیچیدگی محاسباتی  $O(|T_i|(|P_i| + |P_{ri}|) + I(2^{2|P_o|}(2^{|P_r|} - 1) - 1))$  است. در صورتیکه پیچیدگی محاسباتی الگوریتم بدون در نظر گرفتن قضیه ۵-۱۲  $O(I(2^{2|P_i|} + |P| \times |RG(N)| + (|P|)^3 + 12))$  است. که بویژه عبارت اول این پیچیدگی نسبت به عبارت دوم پیچیدگی الگوریتم بدست آمده با قضیه ۵-۱۲ بسیار زیاد است. بنابراین می‌توان از نتایج بدست آمده از قضیه ۵-۱۲ در کاهش پیچیدگی محاسباتی کنترل نظارتی در سیستم‌های بزرگ مقیاس استفاده نمود.

## ۵-۶-۱ مثال کاربردی

بعنوان مثال روشن کننده، سیستم تولیدی بیان شده در [۸۶] را در نظر بگیرید. این سیستم دارای دو ماشین  $M_1$  و  $M_2$  و یک بازوی رباتیکی است. هر یک از ماشین‌ها و ربات دارای بافری با ظرفیت ۱ برای پردازش و نگهداری قطعات هستند. دو نوع قطعه  $P_1$  و  $P_2$  با استفاده از بافرهای ورودی/خروجی به این خط وارد و از آن خارج می‌شوند (در زمان شروع کار سیستم، هیچ قطعه‌ای وجود ندارد). روند تولیدی انجام شده روی قطعه نوع  $P_1$  (بصورت: ورود به خط، پردازش توسط ماشین  $M_1$  (M2)، انتقال به ماشین  $M_2$  (M1) توسط ربات، پردازش توسط ماشین  $M_2$  (M1) و خروج از خط است. شمای کلی این سیستم در شکل ۵-۷ نشان داده شده است.



شکل ۵-۷- شمای کلی یک سیستم تولیدی خودکار.

شبکه پتری متناظر با این روند در شکل ۵-۵ آمده است. این مدل، یک شبکه پتری  $S^3PR$  (که زیرکلاسی از  $S^3PGR$  است) می‌باشد. با استفاده از قضایای (۵-۱۱) و (۵-۱۲)، برای کنترل سیستم کلی تنها نیاز به کنترل سایفون‌های کمینه دارای بیش از یک مکان منبع است. این سایفون‌ها شامل لیست ذیل هستند:

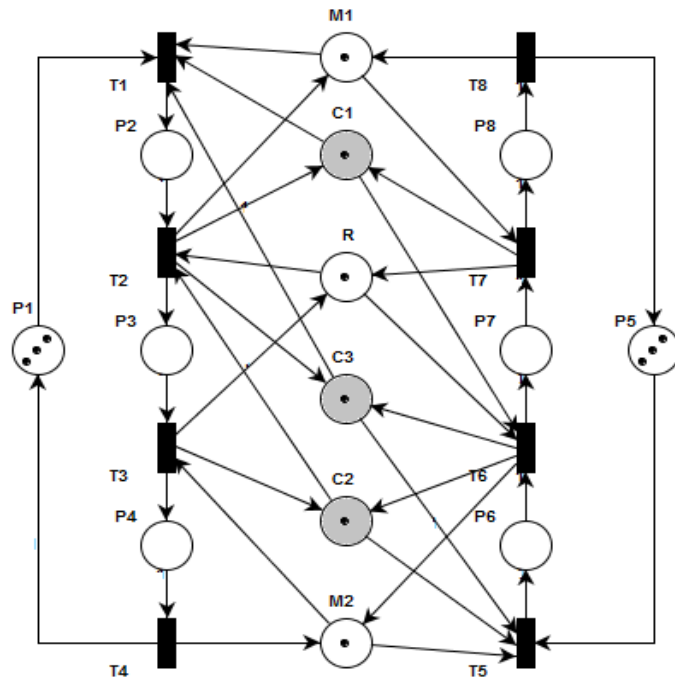
$$S_1 = \{P_7, M_2, P_4, R\}, S_2 = \{P_8, M_1, P_3, R\}, S_3 = \{P_4, M_1, M_2, P_8, R\}$$

سایفون  $S_3$  یک سایفون تکراری است. زیرا  $T$ -بردار مشخصه آن، با استفاده از ترکیب خطی دو سایفون دیگر قابل دسترسی است. با استفاده از روش [۹۷] این شبکه تنها دارای دو نشانه‌گذاری غیرقانونی است که در آنها حداقل یکی از سایفون‌های پایه‌ای بدون نشانه خواهد شد. در جدول ۵-۳، علامت \* در هر سلول جدول نشان‌دهنده بدون نشانه شدن سایفون مربوطه در نشانه‌گذاری متناظر است. بنابراین دو مکان کنترلی به شبکه پتری شکل ۵-۵ افزوده می‌شوند. با افزودن این دو مکان جدید به شبکه پتری، ۴ سایفون جدید که دارای حداقل دو مکان از مجموعه مکان‌ها منبع و کنترلی هستند ایجاد می‌شود. تمامی این سایفون‌ها در یک نشانه‌گذاری یکسان بدون نشانه می‌شوند. بنابراین یکی از این سایفون‌ها بعنوان نماینده انتخاب می‌شود:  $S = \{P_7, P_3, C_1, C_3\}$ .

جدول ۵-۳- جدول نشان‌دهنده بدون نشانه شدن هر یک از سایفون‌های کمینه پایه‌ای شکل ۵-۶ در یک نشانه‌گذاری غیرقانونی.

	$S_1$	$S_2$
$M_1 = [0,0,1,1,1,0,0,1,0,0,2]$		*
$M_2 = [1,1,1,0,2,1,0,0,0,0,0]$	*	

با افزودن مکان کنترلی حاصل به شبکه پتری، شبکه پتری زنده خواهد شد. سیستم کنترل شده در شکل ۵-۸ نشان داده شده است. شبیه‌سازی این سیستم، با استفاده از نرم‌افزار PIPE [۱۳۹] نیز نشان‌دهنده زنده بودن این سیستم است.



شکل ۵-۸- مدل شبکه پتری کنترل شده شکل ۵-۵.

## ۵-۷ جمع بندی

تحلیل گراف دسترسی، یکی از مهمترین گلوگاه‌های سیاست‌های محاسبه کنترل نظارتی در سیستم‌های بزرگ-مقیاس است. در این فصل به اثبات امکان انجام محاسبات کنترل سوپروایزری پیمانهای برای عملگرهای انتخاب، ترتیب و برگ‌برگ‌سازی پرداختیم. در مورد هر یک از این عملگرها اثبات نمودیم که امکان انجام محاسبات کنترل نظارتی بصورت پیمانهای وجود دارد. نتایج بدست آمده روی چند شبکه پتری، نشان‌دهنده مزیت استفاده از روش‌های کنترل نظارتی پیمانهای، بویژه برای عملگر برگ‌برگ‌سازی، نسبت به روش انجام محاسبات در سیستم کلی است. همچنین، با استفاده از عملگر ادغام مکان‌ها نشان دادیم، برای یک شبکه  $S^3PGR$  می‌توانیم محاسبه کنترل نظارتی را روی مجموعه کاهش یافته‌ای از سایفون‌های کمینه انجام دهیم.

فصل ۶: معرفی قضایای مربوط به حفظ زنده

بودن توسط عملگر ادغام گذرها

## ۶-۱ مقدمه

همانطور که در بخش ۲-۴-۵ یاد شد، عملگرهای مختلفی برای سنتز شبکه‌های پتری پیشنهاد شده‌اند. از جمله این عملگرها، عملگرهای ادغام هستند که به ادغام اجزای متفاوتی از یک شبکه پتری و ایجاد شبکه جدید می‌پردازد. این عملگرها بطور کلی به سه دسته عملگر قابل تقسیم هستند: عملگرهای ادغام مسیر [۵۷]، عملگر ادغام مکان‌ها [۵۲] و عملگر ادغام گذرها [۳۷، ۵۷]. عملگر ادغام گذرها با داشتن مجموعه‌هایی از گذرها که برای ادغام مشخص شده‌اند، هر مجموعه را با یک گذر نماینده جایگزین خواهد کرد. از آنجا که در اکثر مدل‌های مبتنی بر شبکه‌های پتری، گذرها نقش رخدادهای سیستمی را ایفا می‌کنند، ادغام گذرها منجر به همگام‌سازی گسسته بین رخدادهای متناظر خواهد شد. البته اگر ادغام گذرها، بین پیمان‌های مختلف یک سیستم گسسته رخداد اتفاق افتد، به یک روش پایین به بالا برای سنتز سیستم دست خواهیم یافت و در واقع می‌توان از این عملگر، بعنوان یک عملگر ترکیب استفاده نمود.

از دیدگاه ریاضی و نحوی<sup>۱</sup> می‌توان هر دو گذر متفاوت از یک شبکه پتری را با هم ادغام نمود. البته از نظر مفهومی<sup>۲</sup>، ادغام دو گذر در صورتی ممکن و باارزش است که تفسیر آنها به یک واقعیت فیزیکی یکسان اشاره داشته و مفهوم شبکه ادغام شده و شبکه اولیه متناقض نباشند. بعنوان مثال در یک سیستم تولیدی، دو گذر خروج قطعه از یک ماشین و برداشته شدن قطعه توسط ربات متناظر با ماشین دارای مفهوم یکسان بوده و ادغام‌پذیرند. اما اگر دو ربات بطور همزمان وظیفه برداشت قطعات از ماشین را بر عهده داشته باشند، نمی‌توان گذرهای مربوط خروج قطعه از ماشین و برداشته شدن قطعه توسط یک ربات را ادغام نمود؛ زیرا مفهوم شبکه ادغام شونده فاقد معنای برداشته شدن قطعه توسط ربات دوم خواهد بود. در این موارد نیاز است ابتدا گذر مربوط به خروج قطعه از ماشین به دو

---

<sup>۱</sup> Syntax

<sup>۲</sup> Semantic

گذر تبدیل شود و سپس هر یک از گذرهای نتیجه با گذر مربوط به برداشته شدن قطعه از یکی از ربات‌ها ادغام شود.

در [۳۷] اثبات شده است که اگر شبکه پتری تحت مطالعه یک ماشین وضعیت قویا همبند باشد، عملگر ادغام گذر حافظ خوش رفتاری خواهد بود. جیاو و همکاران [۵۷] نیز به معرفی یک عملگر ادغام گذر پرداخته‌اند. در این مرجع اثبات شده است که اگر عملگر مربوطه، در شرایط خاصی به یک شبکه پتری معمولی اعمال شود، حافظ خوش رفتاری خواهد بود.

در این فصل شرایط جدیدی را برای حافظ خوش رفتار بودن این عملگر معرفی و اثبات خواهیم کرد. در ادامه به اعتبارسنجی مثال‌هایی از سیستم‌های تولیدی خودکار با استفاده از قضایای مطرح در این فصل می‌پردازیم. از آنجا که تحلیل خوش رفتاری این مثال‌ها با استفاده از قوانین ارائه شده در [۳۷، ۵۷] امکان‌پذیر نیست، اهمیت قوانین یاد شده برای تحلیل خوش رفتاری سیستم به اثبات خواهد رسید. البته این به معنای برتری مطلق قضایای ارائه شده در این فصل نسبت به قضایای ارائه شده در [۳۷، ۵۷] نخواهد بود بلکه نشان‌دهنده شرایط جدیدی است که می‌تواند در کنار شرایط بیان شده دیگر به تحلیل پیشین خوش رفتاری یک سیستم و کاهش بار محاسباتی تحلیل بیانجامد.

## ۶-۲ معرفی و اثبات قوانینی برای حفظ خوش رفتاری پس از

### اعمال ادغام

در این بخش تابع  $TR: \text{a set of transitions of } N \rightarrow \text{a set of transitions of } N'$  را تعریف

می‌کنیم که مجموعه‌ای از گذرهای شبکه  $N$  را بعنوان آرگومان<sup>۱</sup> ورودی دریافت می‌کند. خروجی این

---

<sup>۱</sup> Parameter

تابع مجموعه ادغام‌هایی است که هر یک از گذرهای آرگومان ورودی به آنها تعلق دارند. بعنوان مثال در شکل ۹-۲ داریم:

$$\text{TR}(\{t_{11}, t_{12}, t_{21}, t_{k1}\}) = \{T_1, T_2, T_k\}.$$

فرض کنیم  $N'$  شبکه‌ای باشد که از شبکه  $N$  و با اعمال عملگر ادغام گذرها بدست آمده است. در ادامه قضایایی برای حفظ خواص ساختاری و رفتاری شبکه  $N$  در شبکه  $N'$  بیان خواهد شد. قضیه ۶-۱: حفظ سایفون، تله، ویژگی  $ST$  (MST) در یک شبکه پتری پس از اعمال ادغام گذرها.

(۱) برای هر  $S \subseteq P$ ، اگر  $S$  یک سایفون (تله) در  $N'$  باشد، آنگاه  $S$  یک سایفون (تله) در  $N$  نیز هست. یک سایفون (تله) از  $N'$  یک سایفون (تله) در  $N$  است اگر برای هیچیک از مجموعه مکان‌های غیر سایفون مانند  $S$  در  $N$ ، شرط  $\text{TR}(S^*) \subseteq \text{TR}(*S)$  برقرار نباشد.

(۲) بطور کلی ویژگی  $ST$  (MST) در یک ادغام گذر حفظ نخواهد شد، اما تنها نیاز به بررسی در سایفون‌های جدید ایجاد شده دارد.

اثبات.

(۱) الف. فرض کنیم  $S$  یک سایفون از  $N$  باشد ( $*S \subseteq S^*$ ). نشان خواهیم داد که  $S$  یک سایفون از  $N'$  نیز هست. از آنجا که مجموعه مکان‌ها پس از اعمال ادغام گذرها ثابت می‌مانند، برای هر گذر  $t$  در  $N'$  که  $t \in *S$  باشد:

حالت ۱- اگر  $t \in T_u$  باشد، آنگاه  $t \in S^* \text{ in } N \Rightarrow t \in *S \text{ in } N$  (طبق ویژگی سایفون) و در نتیجه  $t \in S^* \text{ in } N'$ . در نتیجه  $S$  یک سایفون در  $N'$  نیز هست.



حالت ۲- اگر  $t \in T_m \Rightarrow \exists p \in S, (t, p) \in F'$  طبق تعریف ۲-۱۸ از عملگر ادغام گذرها داریم:  
 بنابراین  $t_{ik} \in S^*$  in  $N$  و طبق ویژگی سایفون داریم:  $\exists t_{ik} \in T_{mi}, (t_{ik}, p) \in F \Rightarrow t_{ik} \in S^*$   
 $(p, t_{ik}) \in F'$  در نتیجه بر طبق تعریف ادغام گذر  $(p, t) \in F'$  است. عبارتی  $t \in S^*$  بنابراین  $S$  یک  
 سایفون در  $N'$  نیز هست. اثبات حفظ تله‌ها نیز به طریق مشابهی امکان‌پذیر است.

ب. همانطور که در [۴۷] اثبات شد، اگر هیچ سایفون جدیدی ایجاد شده‌ای وجود نداشته باشد،  
 تمامی سایفون‌های  $N'$ ، سایفون  $N$  نیز هستند. اگر برای یک مجموعه از مکان‌های غیر سایفون مانند  $S$   
 در شبکه  $N$  داشته باشیم:  $TR(S^*) \subseteq TR(S)$  در آن صورت  $S$  یک سایفون جدیدی ایجاد شده در  
 $N'$  خواهد بود. بنابراین، اگر برای گذر ادغام شونده، برای هر  $S$  داشته باشیم  
 $t \in S^* \cup S^*, TR(S^*) \not\subseteq TR(S)$ ، آنگاه هیچ سایفون جدیدی ایجاد شده‌ای در  $N'$  وجود ندارد. در  
 نتیجه اگر شرط بیان شده در قضیه ۶-۱.۱ برقرار نباشد، سایفون‌های  $N'$  نیز سایفون‌های  $N$  خواهند  
 بود.

(۲) بدلیل حفظ سایفون و تله تحت عملگر ادغام گذر (همانطور که در قضیه ۶-۱.۱ بیان و  
 اثبات شد)، اگر  $N$  دارای ویژگی  $ST(MST)$  باشد و سایفون‌های جدید ایجاد شده در  $N'$  نیز دارای  
 تله (نشانه‌دار) باشند، آنگاه شبکه  $N'$  نیز دارای ویژگی  $ST(MST)$  است. به طریق مشابهی، این قضیه  
 در مورد تله نیز اثبات می‌شود. ■

قضیه ۶-۲: حفظ ویژگی‌های  $SM, MG, FC, AC$  در شبکه‌های پتری کلی پس از اعمال عملگر  
 ادغام گذرها.

(۱)  $N$  دارای ویژگی  $SM$  است اگر  $N'$  یک  $SM$  باشد.  $N'$  یک  $SM$  است اگر  $N$  یک  $SM$  بوده و  
 گذرهای ادغامی دارای مکان‌های مشترک نباشند.

(۲) شبکه  $N$  یک  $MG$  است اگر و تنها اگر  $N'$  یک  $MG$  باشد.

(۳) شبکه  $N$  یک شبکه FC است، اگر  $N'$  یک شبکه FC بوده و هیچ زوج گذر ادغام شونده‌ای دارای مکان ورودی مشترک نباشند.  $N'$  یک شبکه FC است اگر  $N$  یک شبکه FC بوده

$$\forall p, |p^*| > 1, \exists t_1 \in p^*, t_1 \in T_{mi} \Rightarrow p^* \subseteq T_{mi}$$

(۴) شبکه  $N$  یک شبکه AC است اگر  $N'$  یک شبکه AC باشد.  $N'$  یک شبکه AC است اگر  $N$

یک شبکه AC باشد و  $\forall p_1, p_2, \exists t_1 \in p_1^*, t_2 \in p_2^*, \{t_1, t_2\} \subseteq T_{mi} (i = 1, \dots, k) \Rightarrow$

$TR(p_1^*) \subseteq TR(p_2^*)$ . عبارت دیگر، اگر گذر مابعد یک مکان با گذر مابعد مکان دیگری

ادغام شود، تمام اجزای مابعد یکی از دو مکان باید با برخی از اجزای مابعد مکان دیگر ادغام

شوند.

اثبات.

(۱) الف. اگر  $N'$  یک SM باشد آنگاه  $|t^*| = |t^*| = 1 \Rightarrow \forall t \in T' \Rightarrow$  بر طبق قسمت ۴ از تعریف

۲-۱۸، دو حالت ممکن است.

حالت ۱- اگر  $t \in T_u$  باشد، آنگاه  $|t^*| = |t^*| = 1$  in  $N$  بنابراین  $N$  یک SM است.

حالت ۲- اگر  $t \in T_m$  باشد، آنگاه  $\text{Sum}(t_i^*) = 1, t_i \in T_{mi}$  و  $\text{Sum}(t_i^*) = 1, t_i \in T_{mi}$

$T_{mi}$  بنابراین  $N$  یک SM است.

ب. فرض کنیم  $N$  یک SM باشد و شرط  $\forall t_i, t_j \in T_{ml} \rightarrow t_i^* = t_j^*, t_i^* = t_j^*, l =$

$\{1, \dots, k\}$  را برآورده کند. ماقبل‌های گذر  $t_{ml}$  در  $N'$  بصورت  $t_{ml} = \cup (t_i^*), t_i \in T_{ml}$  است.

بنابراین،  $t_{ml} = t_i^* \rightarrow |t_{ml}^*| = 1$ . همین قضیه برای مابعد‌های گذر  $t_{ml}$  نیز برقرار است.

(۲) اگر  $N$  یک MG باشد، برای هر مکان  $p$  در  $N$ ، ماقبل  $p$  تنها دارای یک عضو است؛ یعنی

$p = \{t\}$  in  $N$ . عبارتی  $(t, p) \in F, \nexists t_i \in T$  in  $N (t_i, p) \in F$  بنابراین، بر طبق بخش

۳ از تعریف ۱۸-۲، اگر  $t \in T_u$ ، آنگاه  $(t, p) \in F'$  و اگر  $t \in T_{mi}$  آنگاه  $(t_i, p) \in F'$ .  
شرایط یکسانی برای  $p^*$  برقرار است.

عکس تمامی این مراحل نیز درست است. بنابراین از MG بودن  $N'$  می‌توان، MG بودن  $N$  را نتیجه‌گیری کرد.

۳ الف. با استفاده از برهان خلف فرض کنیم  $N'$  یک FC است و  $N$  این خاصیت را ندارد. بنابراین،  $|\cdot(p^*)| > 1, |\cdot(p^*)| > 1, \exists p \in P, |p^*| > 1$ . بدون از دست دادن کلیت قضیه، فرض کنیم

$$(p, t_1), (p, t_2), (p_2, t_2) \in F$$

$N'$  یک FC است تنها اگر  $p^* \in T_{mi}$  اما می‌دانیم که در  $N'$  کمان‌های  $(p, t_i), (p_2, t_i) \in F'$  وجود دارند و این نتیجه در تناقض با شرطی است که در قضیه مطرح شد.

ب. فرض کنیم شبکه  $N$  یک FC است. برای هر مکان  $p$  در این شبکه دو حالت امکان دارد.

حالت ۱-  $|p^*| = 1$  در نتیجه، بر طبق تعریف ۱۸-۲،  $|p^*| = 1$  in  $N'$ .

حالت ۲-  $|p^*| = 1, |\cdot(p^*)| > 1$ . اگر  $p^* \subseteq T_u$  in  $N$  آنگاه بر طبق تعریف ۱۸-۲ داریم:

$|\cdot(p^*)| = 1$  اگر  $\exists t \in p^*, t \in T_{mi}$  آنگاه  $|\cdot t_i| > 1$ . برای حفظ خاصیت FC در این حالت،

باید  $p^* = 1$  in  $N'$ . این قضیه بیان می‌کند که برای حفظ زنده بودن لازم است:  $p^* \subseteq T_{mi}$  (یعنی

تمام گذرهای مابعد  $p$  باید در ادغام یکسانی شرکت کنند).

به بیان دیگر، با توجه به تعریف ۲-۴.۳ ویژگی FC بودن زمانی برقرار است که هر مکان  $p$  تنها

یک گذر مابعد داشته باشد و یا گذرهای مابعد یک مکان، تنها دارای یک مکان ماقبل ( $p$ ) باشند. تنها

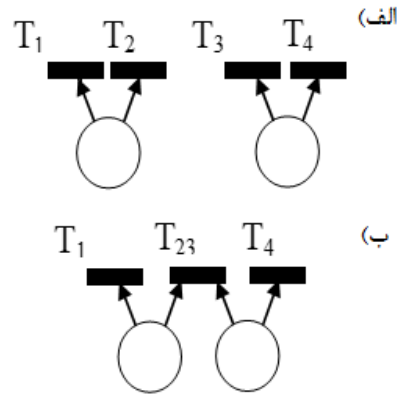
شرایطی که موجب می‌شود خاصیت FC بودن شبکه پتری  $N$  در شبکه پتری  $N'$  حفظ نشود این است

که مکانی در  $N$  دارای بیش از یک مابعد باشد که بدلیل FC بودن  $N$  هر یک از مابعدهای مکان تنها

دارای یک ماقبل هستند. اگر یکی از این مابعدها در ادغامی شرکت کند، در شبکه  $N'$  دارای بیش از

یک ماقبل خواهد شد که خاصیت FC بودن  $N'$  را نقض می‌کند؛ مگر آنکه تمامی مابعدهای مکان

مورد نظر در همان ادغام شرکت کنند که در نتیجه با اینکه گذرها دارای بیش از یک ماقبل شده‌اند، مکان مورد نظر تنها یک مابعد (گذر نماینده ادغام) است.



شکل ۶-۱ مثالی برای روشن سازی قضیه ۶-۲. الف) شبکه اولیه که SM, FC و AC است. ب) شبکه ادغام شده ( $T_2$  و  $T_3$  به گذر واحد  $T_{23}$  ادغام شده‌اند) که SM, FC و AC نیست.

۴) فرض کنیم در شبکه  $N'$  داریم  $p_1^* \cap p_2^* \neq \emptyset$ . دو حالت متصور است:

حالت ۱- در شبکه  $N$ ،  $p_1^* \cap p_2^* \neq \emptyset$ . بر طبق تعریف ویژگی AC در  $N$  داریم:

$p_1^* \cap p_2^* = p_1^* \text{ or } p_2^*$ . بدون از دست دادن کلیت قضیه فرض کنیم  $p_1^* \cap p_2^* = p_1^*$ . دو

حالت امکان دارد. (۱) در شبکه  $N$ ،  $p_1^* \subseteq T_u$ . بنابراین، در شبکه  $N'$ ،  $p_1^* \subseteq T_u$ . بنابراین

$(p_1^* \text{ in } N) = (p_1^* \text{ in } N')$ . بنابراین  $p_1^* \cap p_2^* = p_1^* \text{ in } N'$ . در نتیجه  $N'$  یک شبکه AC است. (۲)

در شبکه  $N$ ،  $\exists t \in p_1^*, t \in T_{mi}$ . در نتیجه  $(p, t) \in F \Rightarrow (p, t_i) \in F'$ . در شبکه  $N$ ،  $t \in p_2^*$ .

به اشتراک  $p_1^*$  و  $p_2^*$  تعلق دارد. بنابراین، در  $N'$ ،  $t_i \in F'$ . نشان دادیم که هر گذر(ادغام شونده یا

ادغام نشونده‌ای) در  $N'$  که به  $p_1^*$  متعلق باشد، به  $p_2^*$  نیز متعلق است. در نتیجه، در  $N'$  داریم:

$p_1^* \subseteq p_2^*$ . بنابراین  $N'$  یک AC است. روند یکسانی برای اثبات AC بودن  $N$  در هنگام AC بودن  $N'$

قابل اتخاذ است.

حالت ۲- در شبکه  $N'$ ،  $p_1^* \cap p_2^* \neq \emptyset$  است و در شبکه  $N$ ،  $p_1^* \cap p_2^* = \emptyset$  است. بنابراین در شبکه  $N$ ، یک  $t_1 \in p_1^*$  و  $t_2 \in p_2^*$  وجود دارد که در شبکه  $N'$  به یک گذر واحد  $t$  ادغام شده‌اند؛ بطوریکه در شبکه  $N'$  داریم  $p_1^* \cap p_2^* \neq \emptyset$ . اگر تمام گذرهای خروجی  $p_1$  در ادغام با برخی از گذرهای خروجی  $p_2$  شرکت کنند، آنگاه در شبکه  $N'$ ،  $p_1^* \subseteq p_2^*$  و بنابراین، یک شبکه AC است. در غیر اینصورت AC نیست.

به بیان دیگر، تنها شرایطی که ممکن است منجر به حفظ نشدن خاصیت AC بودن شبکه پتری  $N$  در شبکه پتری  $N'$  شود این است که دو مکان،  $p_1$  و  $p_2$ ، که در  $N$  دارای مابعد مشترک نیستند، گذرهای مابعد خود را در ادغامی به اشتراک بگذارند. در این حالت بدلیل آنکه اشتراک مابعد این دو مکان در شبکه  $N'$  تهی نیست، لازم است برای تأمین ویژگی AC بودن شبکه  $N'$  مابعدهای یکی از این دو مکان زیرمجموعه مابعدهای مکان دیگر در  $N'$  باشد؛ یعنی  $TR(p_1^*) \subseteq TR(p_2^*)$ . شکل ۶-۱ مثالی را نشان می‌دهد که شبکه  $N$  دارای ویژگی‌های  $SM, FC$  و  $AC$  است، اما شبکه  $N'$  هیچیک از این خواص را ندارد. ■

قضیه ۳-۶: حفظ P-Invariantها، پایایی، کراندار ساختاری، کراندار پتری عمومی پس از اعمال عملگر ادغام گذرها.

(۱)  $I$  یک P-Invariant در  $N$  است اگر و تنها اگر  $I$  یک P-Invariant در  $N'$  باشد.  $N$  پایا

است اگر و تنها اگر  $N'$  اینگونه باشد.  $N$  کراندار ساختاری است اگر و تنها اگر  $N'$  اینگونه

باشد.  $(N', M_0)$  کراندار است اگر  $(N, M_0)$  کراندار باشد [۵۷].

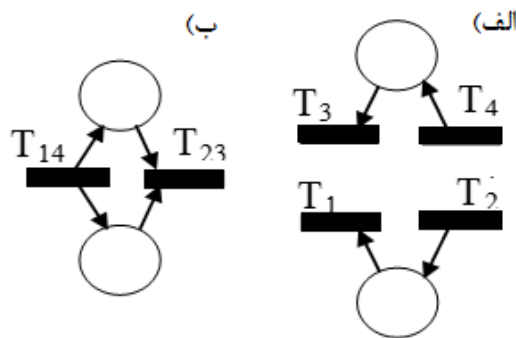
(۲) در [۵۷]، شروطی برای تبدیل T-Invariantهای شبکه  $N$  به T-Invariantهای شبکه  $N'$

بیان شده است.

قضیه ۴-۶: حفظ T-invariant ها، سازگاری، تکرارپذیری شبکه پتری عمومی پس از ادغام گذرها ( با برقراری شرایط خاص).

(۱) T-Invariant های شبکه N با تغییراتی، T-Invariant های N' را نتیجه می دهند اگر به شکل  $(\beta_0, \beta_1, \dots, \beta_k)$  باشند که  $\beta_0 = (b_1, b_2, \dots, b_{|T_u|})$  و  $\beta_i$  یک بردار با طول  $|T_{mi}|$  و مقادیر یکسان است (و بلعکس).

(۲) N سازگار (بازگشت پذیر) است اگر N' اینگونه باشد. N' سازگار (بازگشت پذیر) است اگر N سازگار (بازگشت پذیر) باشد و شرایط قضیه ۴-۶. ۱ را برآورده سازد [۵۷].



شکل ۲-۶ مثالی برای روشن سازی قضیه ۴-۶. (الف) شبکه اولیه که همبند نیست. (ب) شبکه ادغام شده  $(T_1, T_4)$  به  $T_{14}$  و گذرهای  $T_2, T_3$  به  $T_{23}$  ادغام شده اند) که قویا همبند است.

قضیه ۵-۶: حفظ قویا همبندی شبکه های پتری عمومی پس از اعمال عملگر ادغام گذرها: N' قویا همبند است اگر N اینگونه باشد.

اثبات.

گذرهای شبکه N' گذرهای ادغام شده و یا گذرهای ادغام نشده از شبکه N هستند. مسیرهایی از هر گره x به هر گره y در N' که دارای هیچ گذر ادغام شده ای نیستند، مسیرهایی از N نیز هستند. مسیرهایی که شامل یک یا چند گذر ادغام شده هستند، قابل تبدیل به چند زیرمسیر از مکان های مرتبط در N هستند. بدلیل قویا همبند بودن N، حداقل یک مسیر از x به y، با استفاده از یکی از این

زیرمسیره‌ها، در  $N$  قرار دارد. بنابراین  $N'$  نیز قویا همبند است. عکس این قضیه بطور کلی صحیح نیست. شکل ۶-۲ شبکه‌ای را نشان می‌دهد که همبند نیست اما شبکه ادغام شده بدست آمده از آن همبند و قویا همبند است.

قضیه ۶-۶: حفظ ویژگی‌های خالص بودن و معمولی بودن در شبکه پتری پس از اعمال عملگر ادغام گذرها

(۱) شبکه پتری  $N$  خالص است اگر  $N'$  یک شبکه پتری خالص باشد.  $N'$  خالص است اگر  $N$

$$\forall t_j, t_l \in T_{mi} \text{ in } N \rightarrow \bullet t_l \cap t_j^* = \emptyset$$

(۲) شبکه پتری  $N$  معمولی است اگر  $N'$  یک شبکه پتری معمولی باشد.  $N'$  معمولی است اگر  $N$

$$\forall t_i, t_j \in T_{mi} \text{ in } N \rightarrow \bullet t_i \cap \bullet t_j = \emptyset, t_i^* \cap t_j^* = \emptyset$$

اثبات.

(۱) طبق بخش ۴ از تعریف ۱۸-۲، شبکه  $N$  خالص است اگر  $N'$  اینگونه باشد. اما بطور کلی، اگر

$N$  خالص باشد نمی‌توان نتیجه گرفت  $N'$  خالص است. البته اگر اطمینان داشته باشیم برای

هر مکان  $p$ ، شرط  $p \subseteq \bullet t_l \cap t_j^*$ ،  $\exists t_j, t_l \in T_{mi} \text{ in } N$  برقرار نیست، می‌توان گفت حلقه‌های

جدید در طول اعمال عملگر ادغام گذرها ایجاد نخواهد شد. در نتیجه اگر  $N$  فاقد حلقه باشد،

$N'$  نیز این ویژگی را دارد.

(۲) الف. فرض کنیم  $N'$  معمولی است. طبق تعریف ۱۸-۲ وزن یال‌های  $N'$  از جمع وزن یال‌های

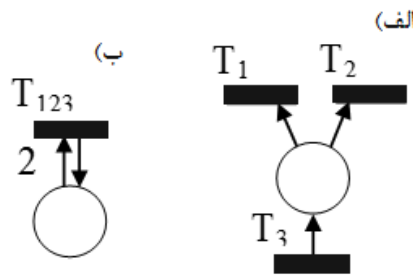
گذرهای ادغام شده در  $N$  بدست می‌آید. بنابراین، وزن یال‌های  $N$  نیز حداکثر یک است.

بنابراین  $N$  نیز معمولی است.

ب. طبق تعریف ۱۸-۲ وزن یال‌های  $N'$  برابر با وزن یال‌های  $N$  است بجز در مواردی که یال‌هایی به

شکل  $(t, p)$  و  $(t', p)$  و  $(p, t)$  و  $(p, t')$  در  $N$  وجود دارند و  $t, t' \in T_{mi}$ . که در این صورت وزن

یال  $(t_i, p)$   $((p, t_i))$  در شبکه  $N'$  از جمع وزن یال‌های متناظر در  $N$  بدست می‌آید و شرط معمولی بودن  $N'$  برقرار نیست. شکل ۳-۶ شبکه‌ای را نشان می‌دهد که معمولی و خالص است اما شبکه متناظر با آن پس از اعمال عملگر ادغام گذرها، این ویژگی‌ها را دارا نیست. ■



شکل ۳-۶ مثال روشن کننده قضیه ۶-۷. الف) شبکه اولیه، یک شبکه پتری معمولی و خالص است. ب) شبکه پتری ادغام شده ( $T_1, T_2, T_3$  به گذر واحد  $T_{123}$  ادغام می‌شود) معمولی یا خالص نیست.

قضیه ۶-۷: حفظ زنده بودن شبکه پتری پس از اعمال عملگر ادغام گذرها

موارد مختلفی با شروط متفاوت وجود دارند که حفظ شدن زنده بودن پس از اعمال عملگر ادغام گذرها را اثبات می‌کنند.

(۱) با دانستن اینکه یک ماشین وضعیتِ قویا همبند، خوش‌رفتار است و حفظ قویا همبندی ماشین وضعیت و ویژگی ماشین وضعیت پس از اعمال عملگر ادغام گذرها، ثابت می‌شود عملگر ادغام گذرها در ماشین وضعیتِ قویا همبند حافظ خوش‌رفتاری است [۳۷].

(۲) ویژگی زنده بودن پس از اعمال عملگر ادغام حفظ خواهد شد، اگر  $N$  دارای شروط ذیل باشد:

- $N$  شبکه پتری معمولی باشد.
- ویژگی قابلیت پوشش  $SM^1$  را دارا باشد.
- ویژگی  $MST$  را دارا باشد.
- تمامی سایفون‌های تازه ایجاد شده دارای تله نشانه‌دار باشند [۵۷].

<sup>1</sup> SM-Coverable



۳)  $N'$  زنده است اگر شبکه پتری  $N$ ،  $FC$  و  $MST$  بوده و شرایط مربوط به حفظ  $FC$  و حفظ  $MST$  در قضایای ۶-۲ و ۶-۱ برقرار باشد.

اثبات قسمت ۳:

اگر  $N$  ( $N'$ ) دارای ویژگی‌های  $FC$  و  $MST$  بوده و شروط حفظ  $FC$  و  $MST$  برقرار باشند،  $N'$

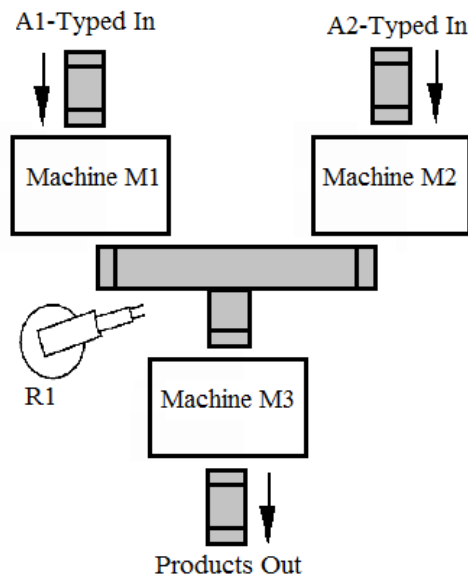
( $N$ ) نیز  $FC$  و  $MST$  است. بنابراین، شبکه  $N'$  ( $N$ ) زنده است (مشخصسازی ۲-۱). ■

مرتب‌ه اجرایی قضیه ۶-۷.۳ به بررسی  $FC$  و  $MST$  بودن شبکه پتری  $N$  و مرتبه اجرایی قضایای ۶-۲ و ۶-۱ بستگی دارد و برابر است با  $O(2|P| \times |T| + 2^{P-1} * P + T^2 \sum_{i=1}^P \binom{P}{i} * i^2 + 3|P| \times |T| \times |T_{mi}|)$  عبارت اول و دوم این پیچیدگی بترتیب نشان دهنده پیچیدگی بررسی  $FC$  بودن و  $MST$  بودن  $N$  است. عبارت سوم با ضریب یک و با ضریب ۲ بترتیب نشان دهنده پیچیدگی شرایط حفظ  $FC$  بودن و  $MST$  بودن هستند. با استفاده از قضیه ۶-۷ می‌توان به تحلیل پیشین ویژگی زنده بودن در مدل شبکه پتری سیستم‌های بزرگ مقیاس پرداخت و از پیچیدگی‌های تحلیل پسین بر حذر بود.

## ۶-۲-۱ مثال کاربردی برای حفظ زنده بودن توسط ادغام گذرها

یک سیستم تولیدی شامل سه ماشین متفاوت،  $M1$ ،  $M2$  و  $M3$  را در نظر بگیرید. ماشین‌های  $M1$  و  $M2$  بترتیب عملیات پیش‌پردازشی روی قطعات نوع  $A1$  و  $A2$  را انجام می‌دهند. هر یک از این دو ماشین دارای بافری با ظرفیت ۲ است که از زمان ورود قطعه تا خروج قطعه از ماشین مشغول می‌ماند. ماشین نوع ۳، قطعات پیش‌پردازش شده از دو نوع را دریافت می‌نماید و آنها را جهت تولید محصول نهایی مونتاژ می‌نماید. این ماشین دارای بافری با ظرفیت نامحدود است (بدلیل سرعت بیشتر این ماشین نسبت به دو ماشین قبل، این ماشین دارای بافر برای ذخیره‌سازی نیست و مواد ورودی را مسقیماً و توسط بازوی رباتیکی و نوار نقاله دریافت می‌کند). انتقال مواد خام، میانی و نهایی با استفاده

از نوارهای نقاله انجام می‌شود. سیستم نقاله دارای دو بافر با ظرفیت  $k$  برای هر یک از انواع مواد ورودی است. بازوی رباتیکی  $R1$  مواد پیش‌پردازش شده را از ماشین‌های  $M1$  و  $M2$  برداشته و در نوار نقاله مربوط ماشین  $M3$  قرار می‌دهد. این سیستم تولیدی در شکل ۴-۶ نمایش داده شده است.

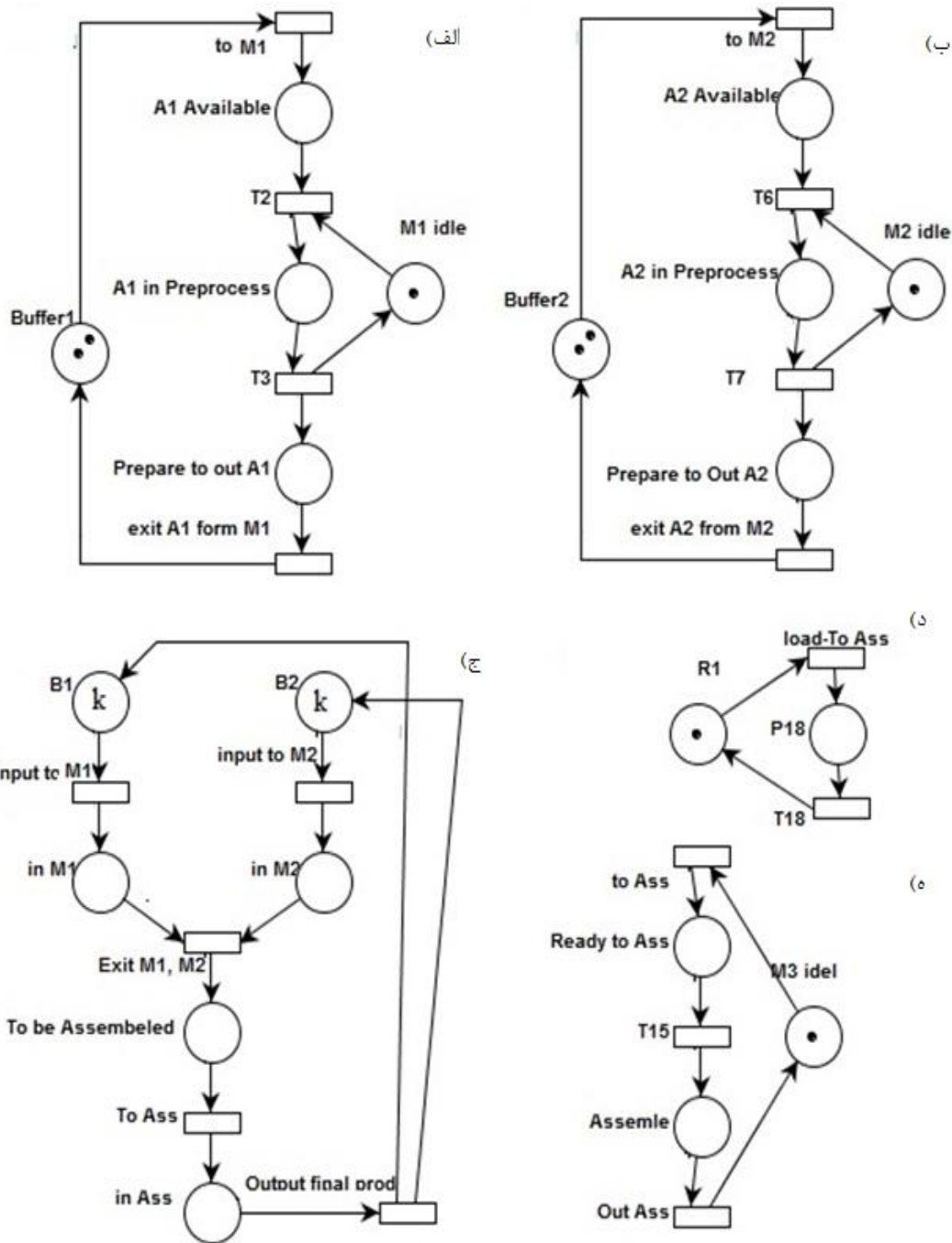


شکل ۴-۶ شمای کلی سیستم تولیدی مثال ۱-۲-۶.

مدل شبکه پتری هریک از ماشین‌ها، بازوی رباتیکی و سیستم نقاله در شکل ۵-۶ نشان داده شده است. در این شکل، تفسیر هر یک از مکان‌های استفاده شده در مدل‌های شبکه‌های پتری (بعنوان فعالیت‌ها) مشخص شده است. با در نظر گرفتن تفسیر هر یک از گذرها بصورت شروع و یا پایان یک فعالیت، تفسیر گذرها مشخص است. گذرهایی که از نظر مفهومی قابلیت ادغام دارند با استفاده از این تفسیر مشخص خواهد شد. بعنوان مثال گذرهای "to M1" در شکل ۵-۶ الف و "input to M1" در شکل ۵-۶ ب قابل ادغام هستند. مدل شبکه پتری نهایی پس از ادغام گذرها در شکل ۶-۶ آمده است.

در مورد تحلیل زنده بودن این شبکه، از آنجا که مدل‌های شبکه‌های پتری ماشین‌های  $M1$ ،  $M2$  و نقاله در شکل ۵-۶ ماشین وضعیت نیستند، قضایای مطرح شده در مراجع [۳۷] و [۵۷] نمی-

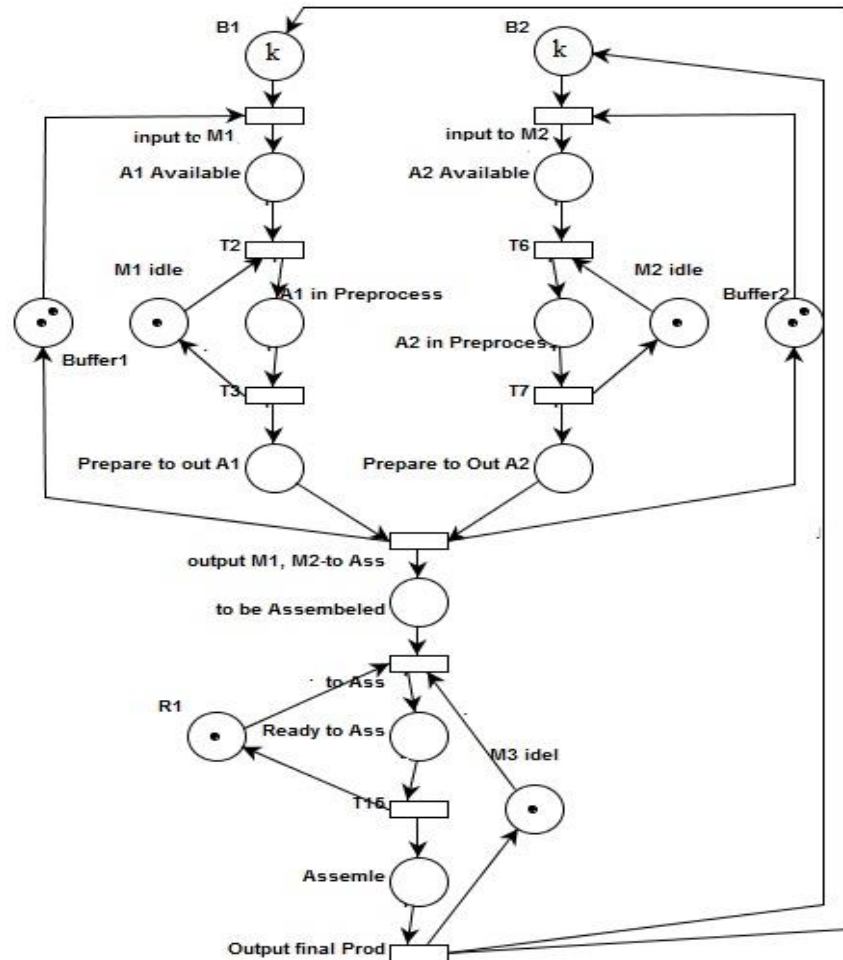
توانند به اثبات زنده بودن سیستم نهایی نشان داده شده در شکل ۶-۶ باشند. البته هر یک از پیمانها  
 FC، MST بوده و شرایط مربوط به حفظ FC و حفظ MST در قضایای ۶-۲ و ۶-۱ را دارا هستند.



شکل ۶-۵- مدل شبکه پتری پیمانهای پایه در سیستم تولیدی شکل ۶-۴. الف) مدل شبکه پتری ماشین  
 M1. ب) مدل شبکه پتری ماشین M2. ج) مدل شبکه پتری سیستم نقاله. د) مدل شبکه پتری بازوی رباتیکی R1. ه)  
 مدل شبکه پتری ماشین M3.

بنابراین طبق قضیه ۶-۷.۳ می‌توان بدون انجام تحلیل‌های پسین ثابت کرد مدل شبکه پتری

نهایی در شکل ۶-۶ زنده است.



شکل ۶-۶- شبکه پتری سیستم کلی شکل ۶-۴ پس از اعمال عملگر ادغام گذر روی پیمان‌های پایه در شکل

۵-۶

### ۶-۳ استفاده از عملگر ادغام در ترکیب همروند دو پیمان

در این بخش نوع خاصی از عملگر ادغام گذرها یعنی ادغام گذرهای غیر مجاور را شرح می‌دهیم.

این نوع عملگر ادغام گذرها می‌تواند برای ترکیب پیمان‌های مختلف سیستم با استفاده از

همگام‌سازی گذرهای آنها مورد استفاده قرار گیرد.

تعریف ۶-۱- گذرهای غیرمجاور: در یک شبکه پتری، دو گذر  $t_i, t_j$  ( $i \neq j$ ) را غیر مجاور

گوییم اگر و تنها اگر  $(\cdot t_i \cup t_i^*) \cap (\cdot t_j \cup t_j^*) = \emptyset$ .

قضیه ۶-۸: حفظ خالص بودن شبکه‌های پتری پس از اعمال عملگر ادغام بر روی گذرهای

غیرمجاور.

(۱) فرض کنید  $N'$  یک شبکه پتری باشد که از شبکه  $N$  با استفاده از عملگر ادغام بر روی

گذرهای غیر مجاور بدست آمده است.  $N$  خالص است، اگر و تنها اگر  $N'$  خالص باشد.

(۲)  $N'$  یک شبکه معمولی است اگر و تنها اگر  $N$  یک شبکه معمولی باشد.

اثبات.

(۱) اگر  $N'$  خالص باشد،  $N$  نیز خالص است. درستی این گزاره واضح است زیرا در عملگر ادغام

گذرها (بدون محدودیت غیر مجاور بودن گذرهای ادغامی) نیز این ویژگی برقرار است.

نشان می‌دهیم در عملگر ادغام گذرهای غیر مجاور، اگر  $N$  خالص باشد،  $N'$  نیز خالص

است: شبکه  $N$  فاقد حلقه است. گذرهای مجاور نیز امکان ادغام ندارند. بنابراین شرط

مطرح شده در قضیه ۶-۶. ۱ برقرار است. در نتیجه، حلقه جدیدی در  $N'$  ایجاد نخواهد شد

و  $N'$  نیز خالص است.

(۲) برطبق تعریف ۲-۱۸ از عملگر ادغام گذرها، اگر گذرهای  $t_{i1}, t_{i2}, \dots, t_{i|T_{mi}|}$  به گذر جدید

$t_i$  ادغام شوند، آنگاه  $W'(t_i, p_x) = \text{Sum}(W(t_{ij}, p_x))$  که  $t_{ij} \in T_{mi}$ . اما اگر  $t_i$

متشکل از مجموعه‌ای از گذرهای غیر مجاور باشد،  $W(t_{ij}, p_x)$  برای حداکثر یکی از گذرها

۱ است. بنابراین  $N'$  نیز یک شبکه معمولی است. ■

قضیه ۶-۹: حفظ ویژگی زنده بودن شبکه پتری پس از اعمال عملگر ادغام گذرهای غیر مجاور.

(۱) فرض کنید  $N'$  یک شبکه پتری باشد که از شبکه پتری  $N$  و پس از اعمال عملگر ادغام گذرهای غیر مجاور بدست آمده است. فرض کنید: (۱)  $N$  یک شبکه معمولی و  $AC$  است. (۲) شرایط حفظ ویژگی‌های  $MST$  و  $AC$  (قضیه ۶-۲ و قضیه ۶-۱) برقرار باشد. در آنصورت  $N$  زنده است، اگر و تنها اگر  $N'$  زنده باشد.

اثبات.

برطبق قضیه ۶-۲.۴ تحت شرایطی، ویژگی  $AC$  یک شبکه پس از اعمال عملگر ادغام گذرها حفظ خواهد شد. برطبق قضیه ۶-۱.۲، ویژگی  $MST$  پس از اعمال عملگر ادغام گذرها تحت شرایطی حفظ خواهد شد. بعلاوه، برطبق قضیه ۶-۸، معمولی بودن یک شبکه پس از اعمال عملگر ادغام گذرهای غیرمجاور حفظ خواهد شد. در نتیجه، اگر  $N$  ( $N'$ ) زنده، معمولی،  $MST$  و  $AC$  باشد، شبکه  $N'$  ( $N$ ) نیز معمولی،  $MST$  و  $AC$  است. بنابراین شبکه  $N'$  ( $N$ ) نیز زنده است (مشخصه‌سازی (۲-۲)).

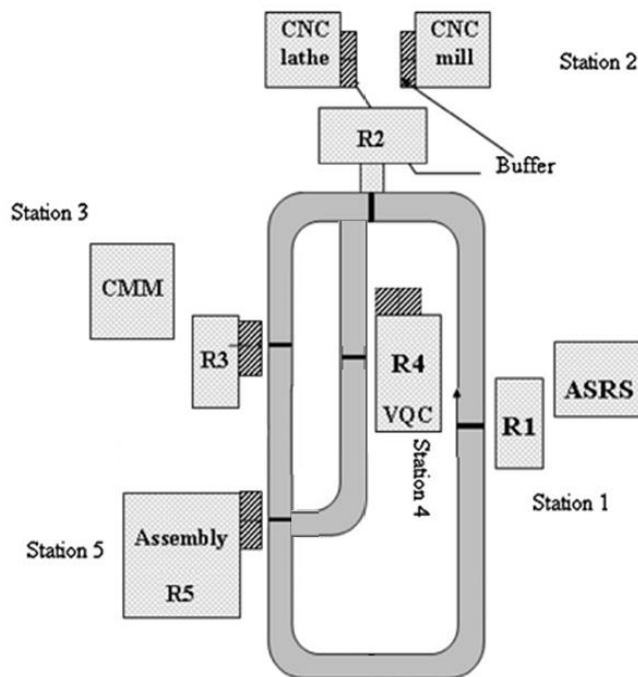
قضیه ۶-۹ می‌تواند به تحلیل پیشین ویژگی زنده بودن در مدل شبکه پتری سیستم‌های بزرگ مقیاس بپردازد که با بی‌نیازی از پیچیدگی‌های تحلیل پسین این ویژگی، کاهش محاسبات را در پی دارد. مرتبه اجرایی قضیه ۶-۹ به بررسی معمولی بودن،  $AC$  بودن و  $MST$  بودن شبکه پتری  $N$  و مرتبه اجرایی قضایای ۶-۲ و ۶-۱ بستگی دارد و برابر است با  $O(|P| \times |T| + |P|^2 \times |T|^2 + (|P| \times |T| \times \sum_{i=1}^k |T_{mi}|)^2 + 2^{P-1} * P + T^2 \sum_{i=1}^P \binom{P}{i} * i^2 + |P| \times |T| \times |T_{mi}|)$  عبارت اول، دوم و سوم این پیچیدگی بترتیب نشان دهنده پیچیدگی بررسی معمولی بودن،  $AC$  بودن و  $MST$  بودن  $N$  است. عبارت چهارم و پنجم نیز بترتیب نشان دهنده پیچیدگی شرایط حفظ  $AC$  بودن و  $MST$  بودن هستند.

همچنین در [۵۷] ثابت شده است می‌توان تناظری میان عملگر ادغام گذرهای غیر مجاور و عملگر ترکیب همروند برقرار نمود. بدین ترتیب می‌توان از قضایای مطرح شده در زمینه ادغام گذرهای غیرمجاور در یک شبکه پتری، در زمینه ترکیب دو شبکه پتری و در نتیجه در زمینه طراحی پیمانهای یک سیستم نیز استفاده نمود.

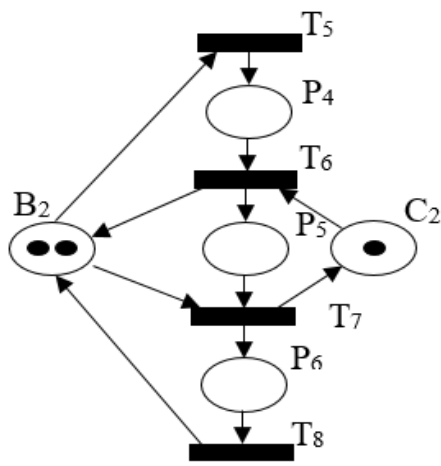
در [۵۷] ترکیب با استفاده ادغام گذرهای غیر مجاور در حالت خاصی حافظ ویژگی MST است: اگر دو شبکه  $N_1$  و  $N_2$  شبکه‌هایی با ویژگی ST (MST) باشند و تعداد گذرهای ادغام شونده در هر یک از دو شبکه برای هر ادغام برابر با ۱ باشند، شبکه  $N$  ترکیب شده نیز دارای ویژگی ST (MST) است.

### ۶-۳-۱ مثال کاربردی برای حفظ زنده بودن توسط ادغام گذرهای غیر

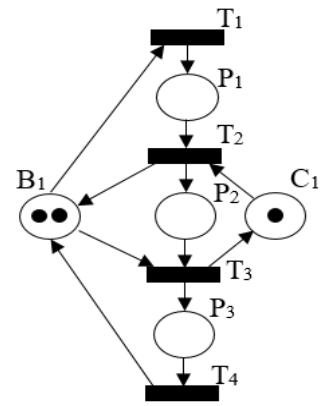
#### مجاور



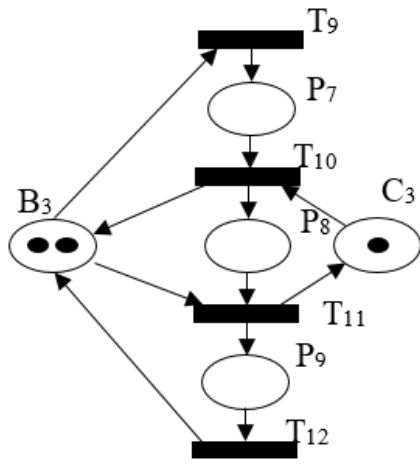
شکل ۶-۷ شمای یک کارخانه تولیدی. شکل برگرفته از مرجع [۵۴] با اعمال تغییرات.



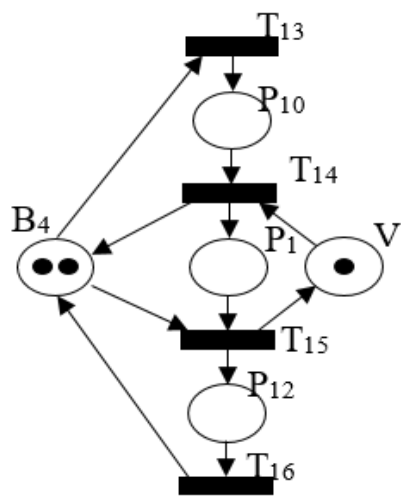
(ج)



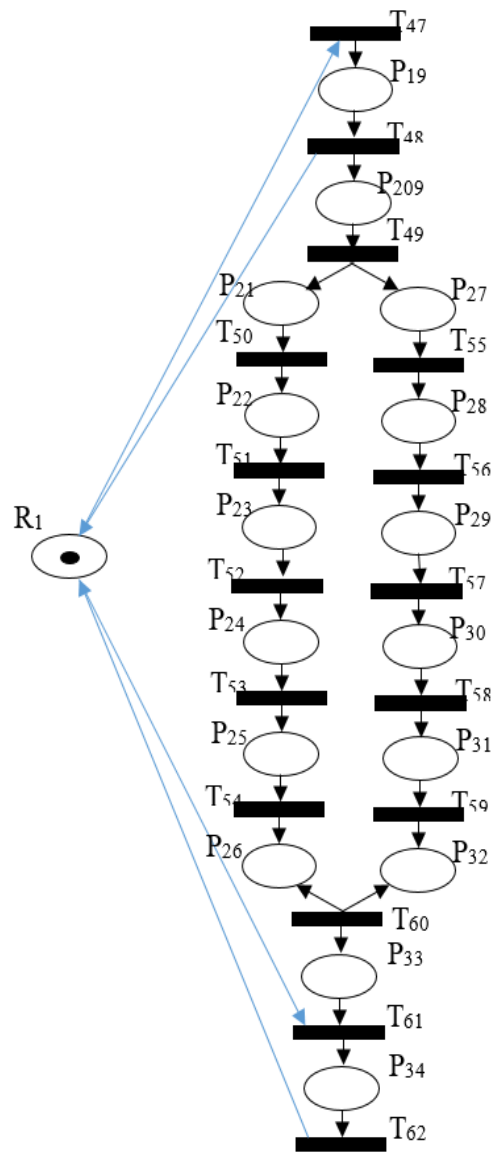
(الف)



(د)

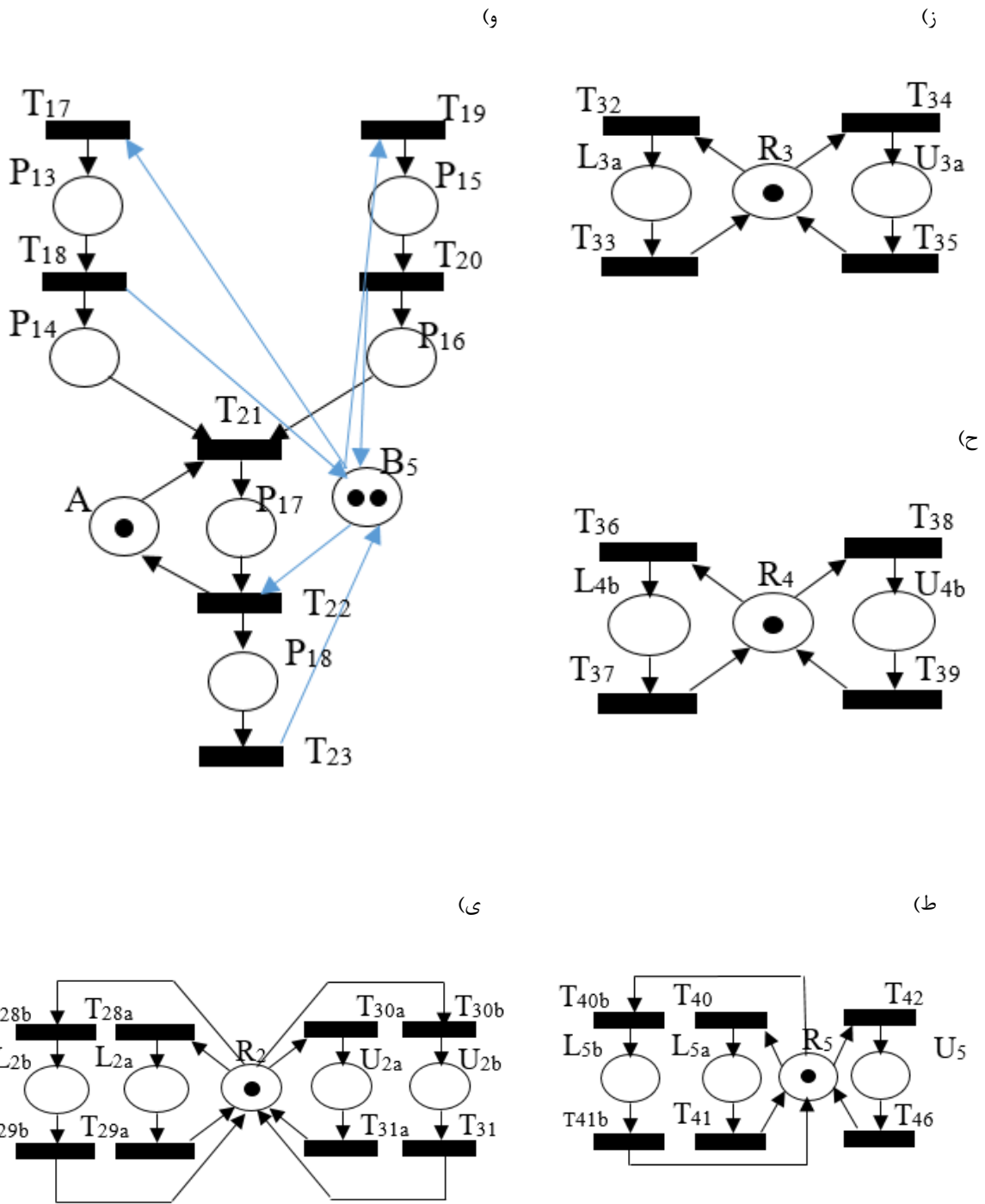


(هـ)



(ب)





شکل ۸-۶- مدل شبکه پتری پیماندهای یک سیستم تولیدی خودکار. الف) مدل شبکه پتری برای ماشین CNC-mill. ب) مدل شبکه پتری نقاله. ج) مدل شبکه پتری برای CNC-lathe. د) مدل شبکه پتری برای CMM. ه) مدل شبکه پتری VQC. و) مدل شبکه پتری ASS. ز) مدل شبکه پتری ربات ۳. ح) مدل شبکه پتری ربات ۴. ط) مدل شبکه پتری ربات ۵. ی) مدل شبکه پتری ربات ۲.

برای مدل‌سازی یک سیستم تولیدی خودکار صنعتی بزرگ-مقیاس، رویکرد تقسیم و غلبه قابل استفاده است. این بخش به ارائه نتایج تجربی برای اعمال عملگر ادغام گذرها در طراحی و اعتبارسنجی پیمانهای یک سیستم تولیدی خودکار می‌پردازد. مثال ارائه شده در این بخش، با اندکی تغییر از مثال مطرح شده در [۵۴] گرفته شده است. همانطور که در شکل ۶-۷ نشان داده شده است، این خط تولید شامل مجموعه‌ای از ایستگاه‌ها است که در حول یک نقاله قرار گرفته‌اند.

این ایستگاه دارای دو دستگاه ماشین برای فرز و تراش دادن است که بترتیب برای مواد خام نوع A و نوع B مورد استفاده قرار می‌گیرند.

جدول ۶-۱- تفسیر مکان‌های پیمان‌های نشان داده شده در شکل ۶-۸.

مکان	تفسیر
P11	VQC قطعه نوع B را بررسی می‌کند.
P12	R4 قطعه بررسی شده را از VQC برمی‌دارد.
P13	R5 قطعه نوع A را در ASS قرار می‌دهد.
P14	قطعه نوع A در ASS قرار دارد.
P15	R5 قطعه نوع B را در ASS قرار می‌دهد.
P16	مواد نوع B در ASS قرار گرفته‌اند.
P17	ASS یک قطعه نوع A را با دو قطعه نوع B مونتاژ می‌کند.
P18	R5 ماده نهایی را از ASS برمی‌دارد.
P19	R1 یک ماده خام را از ASRS بر روی نوار نقاله قرار می‌دهد.
P20	نقاله مواد خام را به CNC انتقال می‌دهد.
P21	بارکدخوان CNC قطعه نوع A را تشخیص می‌دهد.
P22	قطعه نوع A در CNC-mill قرار دارد.
P23	نقاله قطعه نوع A را به CMM منتقل می‌کند.
p1	R2 قطعه نوع A را در CNC-mill قرار داده است.
p2	CNC-mill در حال پردازش مواد خام نوع A است.
p3	R2 قطعه پردازش شده را از CNC-mill برمی‌دارد.
p4	R2 قطعه نوع B را در CNC-lathe قرار می‌دهد.
P5	CNC-lathe ماده خام نوع B را پردازش می‌کند.
P6	R2 قطعه پردازش شده را از CNC-lathe برمی‌دارد.
P7	R3 قطعه نوع A را در CMM قرار می‌دهد.
P8	CMM قطعه نوع A را کنترل می‌کند.
P9	R3 قطعه پردازش شده را از CMM برمی‌دارد.
P10	R4 قطعه نوع B را در VQC قرار می‌دهد.

P32	ماده نوع B در ASS قرار دارد.
Bi	بافر i در دسترس است. ( $i=\{1,2,3,4,5\}$ )
C1	CNC-mill در دسترس است.
C2	CNC-lathe در دسترس است.
C	CMM در دسترس است.
V	VQC در دسترس است.
A	ASS در دسترس است.
Ri	ربات i در دسترس است. ( $i=\{1,2,3,4,5\}$ )
Lia	Ri قطعه A را برمی‌دارد. ( $i=\{1,2,3,5\}$ )
Lib	Ri قطعه B را برمی‌دارد. ( $i=\{1,2,4,5\}$ )
Uia	Ri قطعه A را می‌گذارد. ( $i=\{2,3\}$ )
Uib	Ri قطعه نوع B را می‌گذارد. ( $i=\{2,4\}$ )
Ui	Ri مواد را می‌گذارد. ( $i=\{1,5\}$ )

P24	قطعه نوع A در CMM قرار دارد.
P25	نقاله مواد نوع A را به ASS منتقل می‌کند.
P26	مواد نوع A در ASS قرار دارند.
P27	بارکدخوان CNC قطعه نوع B را تشخیص می‌دهد.
P28	قطعه نوع B در CNC-lathe قرار گرفته است.
P29	نقاله قطعه نوع B را به VQC انتقال می‌دهد.
P30	قطعه نوع B در VQC قرار دارد.
P31	نقاله ماده نوع B را به ASS منتقل می‌کند.
P32	قطعه نوع V در ASS قرار دارد.
P33	عملیات مونتاژ
P34	R1 محصول را از نقاله به خروجی ASRS منتقل می‌کند.

جدول ۶-۲- گروه‌های گذرهای کاندید برای ادغام و برجسب ادغام متناظر آنها.

T60	T60, TT23, T43
T55	T55, T5, T28b
T6	T6, T29b
T7	T7, T30b
T56	T56, T8, T31b
T57	T57, T13, T36
T14	T14, T37
T15	T15, T38
T58	T58, T16, T39
T59	T59, T19, T40b
T20	T20, T41b

برجسب گذرها در پیمانها	برجسب گذر ادغام شده
T50, T1, T28a	T50
T2, T29a	T2
T3, T30a	T3
T51, T4, T31a	T51
T52, T9, T32	T52
T10, T33	T10
T11, T34	T11
T53, T12, T35	T53
T54, T17, T40a	T54
T18, T41a	T18
T22, T42	T22

سپس کنترل کیفیت مواد نوع A و کنترل بصری کیفیت مواد نوع B در ماشین‌های CMM و VQC انجام می‌شود. نهایتاً این دو نوع ماده در ایستگاه مونتاژ<sup>۱</sup>، مونتاژ خواهند شد. مدل شبکه پتری هر یک از ماشین‌های این خط تولید در شکل ۶-۸ نشان داده شده است. در این مدل‌ها، مکان‌ها نشان دهنده عملیات‌ها هستند (تفسیر هر یک از مکان‌ها در جدول ۶-۱ آمده است). گذرها شروع و پایان یک فعالیت را نشان می‌دهد. یعنی، یک گذر  $t$  بعنوان شروع فعالیت  $a$  در نظر گرفته می‌شود اگر  $t^* = a$  باشد و نشان دهنده اتمام آن فعالیت است اگر  $t = a^*$  باشد.

بنابراین، تفسیر گذرها نیز با استفاده از تفسیر مکان‌ها مشخص است و بیان نشده‌اند. گذرهایی با تفسیر مشابه، کاندید ادغام هستند. این گروه از گذرها در جدول ۶-۲ لیست شده‌اند. با اعمال عملگر ادغام گذرهای غیر مجاور در پیمان‌های مختلف سیستم و حذف مکان‌های بی‌فایده، مدل کلی سیستم در شکل ۶-۹ آمده است. برای اعتبارسنجی خوش‌رفتاری در شبکه کلی، از قضایای ۶-۱ الی ۶-۹ استفاده می‌کنیم تا از مزایای اعتبارسنجی پیمان‌های بهره‌مند شویم.

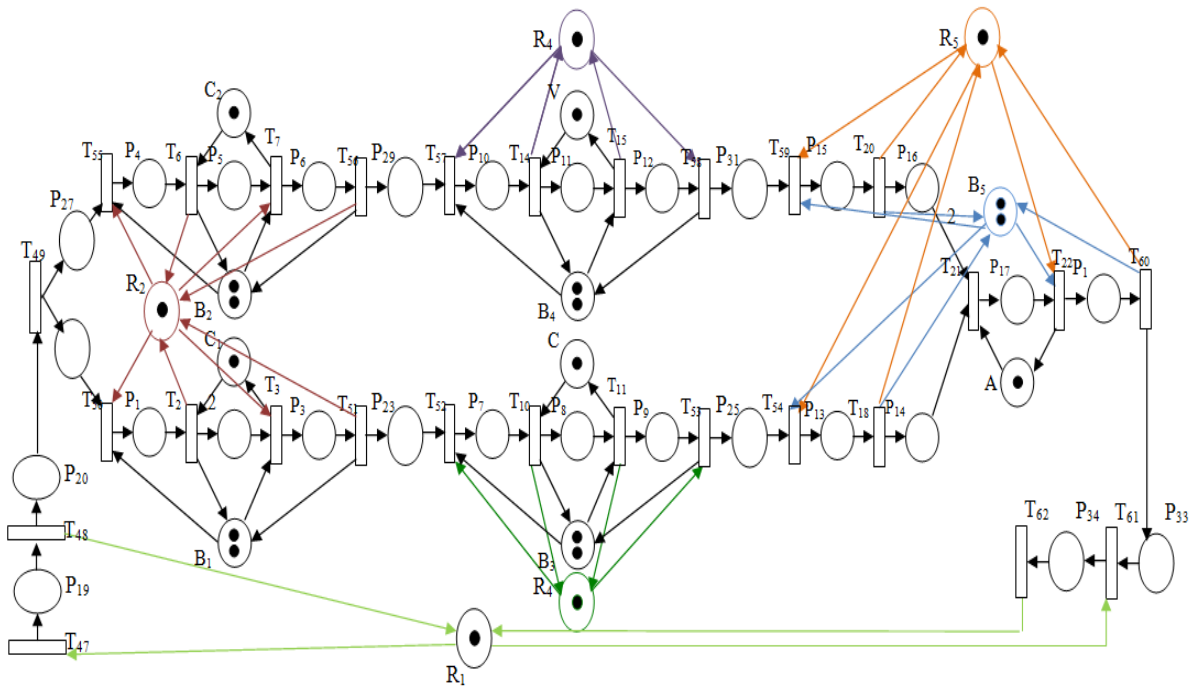
تمامی پیمان‌ها کراندار هستند، بنابراین شبکه نتیجه نیز کراندار است (قضیه ۶-۳). مدل شبکه پتری پیمان‌ها تکرارپذیر است و شرایط حفظ تکرارپذیری را دارند، در نتیجه سیستم نتیجه، تکرارپذیر است (قضیه ۶-۴). بعنوان مثال T-Invariant های پیمان‌های شکل ۶-۸. ب و ۶-۸. ی بترتیب برابر با  $[1\ 1\ 1\ 1\ 1\ 1\ 1]$  و  $[1\ 1\ 1\ 1\ 1\ 1\ 1]$  هستند که شرایط قضیه ۶-۴ را برآورده می‌سازند.

پیمان‌ها ماشین وضعیت قویا همبند نیستند و در نتیجه قضایای اثبات شده در [۳۷] برای اعتبارسنجی زنده بودن شبکه کلی قابل استفاده نیست. از طرفی این پیمان‌ها قابل پوشش SM نیز نیستند. در نتیجه، قضایای مطرح شده در مرجع [۵۷] نیز قابل استفاده نخواهند بود. از طرفی این

---

<sup>1</sup> Assembly Station

پیمانه‌ها معمولی بوده و ویژگی MST و شرایط بیان شده در [۵۷] برای حفظ این ویژگی را دارا هستند.



شکل ۶-۹- سیستم کلی حاصل از ادغام گذرها در پیمانه‌های نشان داده شده در شکل ۶-۸.

همچنین این پیمانه‌ها AC بوده و شرایط لازم برای حفظ این ویژگی (قضیه ۶-۲، ۴) را دارا هستند. بعنوان مثال پیمانه‌های شکل ۶-۸، و و شکل ۶-۸، ط دارای ویژگی AC بوده و  $TR(A^*) \subseteq TR(R_5^*)$  و  $TR(R_5^*) = TR(B_5^*)$ . بنابراین سیستم کلی (شکل ۶-۹) که از ادغام گذرهای غیرمجاور بدست آمده است زنده است (قضیه ۶-۹).

## ۶-۴ جمع‌بندی

در این فصل به بیان کارهای انجام شده در زمینه معرفی و اثبات قضایای مرتبط با خوش- رفتاری در عملگرهای سنتز شبکه‌های پتری پرداختیم. از یک عملگر ادغام گذرهای تعریف شده در

مراجع استفاده نمودیم و در مورد آن به بررسی شرایط حفظ ویژگی زنده بودن و ویژگی‌های مرتبط با آن، مانند AC بودن، FC بودن و معمولی بودن، پرداختیم. سپس به طراحی پیمان‌های یک مثال سیستم‌های تولیدی با استفاده از عملگر ادغام گذرهای غیرمجاور پرداختیم. نتایج حاصل از قضایای مطرح و اثبات شده در این فصل، ما را از اعتبارسنجی سیستم کلی که می‌تواند زمانبر باشد، دور می‌سازد.

فصل ۷: بهبود سرعت در طراحی سوپروایزر

بیشینه مجازیت با ساختار کمینه

## ۷-۱ مقدمه

در فصل ۳ با مفهوم حالت مافوق که در [۱۰۱] تعریف شده است، آشنا شدیم. مفهوم حالت مافوق، در مورد نشانه‌گذاری‌های ممنوع کاربرد داشته و منجر به کاهش تعداد این نشانه‌گذاری‌ها خواهد شد. در این فصل به معرفی مفهوم بردار تخصیص برای مجموعه حالات قانونی می‌پردازیم. یک روش بازگشتی برای محاسبه هر یک از مفاهیم بردار تخصیص و حالت مافوق نیز ارائه خواهیم داد. استفاده از روش بازگشتی، محاسبات بالایی را در پی داشته و منجر به جستجو در فضای  $(|P|+1)$ -بعدی خواهد شد. استفاده از هرس کردن، باعث جستجو در بخش کوچکتري از فضا، خواهد شد. همچنین محاسبه این مفاهیم با استفاده از ILP مورد بررسی قرار می‌گیرد. سپس یک ILP برای محاسبه کنترل نظارتی بیشینه مجازیت با ساختار کمینه و پیچیدگی محاسباتی پایین پیشنهاد می‌شود. استفاده از ILP نسبت به محاسبه بازگشتی سریعتر است. همچنین در صورتی که تعداد نشانه‌گذاری‌های غیرقانونی زیادی به یک بردار تعمیم نگاشته شوند، این روش منجر به کاهش پیچیدگی محاسباتی خواهد شد.

## ۷-۲ بردار تخصیص و بردار تعمیم

تعریف ۷-۱: بردار  $|P|$ -بعدی  $G$  را یک حالت مافوق برای یک نشانه‌گذاری  $M$  گوئیم اگر و تنها اگر  $G \leq M$  باشد [۱۴۰].

تعریف ۷-۲: بردار  $|P|$ -بعدی  $D$  را یک بردار تخصیص برای یک نشانه‌گذاری  $M$  می‌گوئیم اگر و تنها اگر  $D \geq M$  باشد.

قضیه ۷-۱: اگر یک PI برای پیشگیری از بردار  $G$  در یک شبکه پتری ایجاد شود، PI تمامی نشانه‌گذاری‌های  $M_j \in M_{FBM}^*$  که بردار  $G$  حالت مافوق آنها است را نیز پیشگیری می‌نماید [۱۴۰].



قضیه ۲-۷: اگر یک PI که برای پیشگیری از وقوع نشانه‌گذاری ممنوع  $M_j \in M_{FBM}^*$  طراحی شده است، بردار  $D$  را که تخصیصی از یک نشانه‌گذاری قانونی  $M_i \in M_L^*$  است منع نکند، نشانه‌گذاری قانونی  $M_i$  را نیز منع نخواهد کرد.

اثبات.

طبق معادله (۲۷-۳)، ضرایب یک PI که از  $M_j$  پیشگیری می‌کند و  $D$  را حفظ می‌کند در معادله زیر صدق می‌کند:

$$\sum_{i=1}^n l_i \{D(p_i) - M_j(p_i)\} \leq -1$$

از طرفی طبق تعریف ۱-۷ می‌دانیم، اگر  $D$  یک بردار تخصیص برای نشانه‌گذاری قانونی  $M_i$  باشد، آنگاه  $D \geq M_i$  است. بنابراین داریم:

$$\sum_{i=1}^n l_i \{M_i(p_i) - M_j(p_i)\} \leq -1$$

این معادله بیانگر این است که  $M_i$  پس از اضافه کردن PI، منع نخواهد شد.

### ۳-۷ محاسبه مجموعه کمینه بردارهای تخصیص مجاز

قضیه ۲-۷ نشان می‌دهد به جای طراحی سوپروایزری برای حفظ نشانه‌گذاری‌های قانونی، می‌توان این سوپروایزر را برای بردار تخصیص آن بدست آورد. البته، بردارهای تخصیصی وجود دارند که مخصص یک نشانه‌گذاری غیرقانونی نیز هستند. با محاسبه این بردارها، باید آنها را از مجموعه بردارهای تخصیص نشانه‌گذاری‌های قانونی حذف نمود. بدین ترتیب مجموعه بردارهای تخصیص مجاز بدست می‌آید. با استفاده از این مجموعه می‌توان مجموعه کمینه بردارهای تخصیص مجاز را نیز بدست آورد. برای این کار بدین ترتیب می‌توان عمل کرد:

الگوریتم ۱-۷- محاسبه مجموعه کمینه بردارهای تخصیص قانونی

۱- برای تمامی نشانه‌گذاری‌های قانونی، مجموعه تمام بردارهای تخصیص را حساب کنید و در مجموعه  $D_{1L}$  قرار دهید.

۲- برای تمامی نشانه‌گذاری‌های غیرقانونی، مجموعه تمام بردارهای تخصیص را حساب کنید و در مجموعه  $D_{1F}$  قرار دهید.

۳- بردارهای تخصیصی از  $D_{1L}$  که مخصص حداقل یک نشانه‌گذاری غیرقانونی است را حذف کنید.  $D_{2L} = D_{1L} \setminus D_{1F}$ .

۴- برطبق مفهوم پوشش دادن، نشانه‌گذاری‌هایی از  $D$  را که توسط یک نشانه‌گذاری دیگر از این لیست پوشش داده می‌شود را حذف کنید. بدین ترتیب مجموعه کمینه بردارهای تخصیص مجاز بدست می‌آید.

برای محاسبه مجموعه تمام بردارهای تخصیص یک نشانه‌گذاری (در مرحله ۱ و ۲) می‌توان از یک راه حل بازگشتی، هرس کردن و یا برنامه‌نویسی خطی استفاده کرد. روند این روش در الگوریتم ۲-۷ آمده است.

الگوریتم ۲-۷- محاسبه بردارهای تخصیص برای یک نشانه‌گذاری قانونی

ورودی: یک نشانه‌گذاری قانونی  $M$ ، عدد  $n$ ، بردار  $\text{bound}$  که نشان‌دهنده کران هر یک از مکان‌هاست و  $D$  مجموعه تمام بردارهای تخصیص.

خروجی: مجموعه بردارهای تخصیص‌دهنده نشانه‌گذاری  $M$

۱- اگر  $n \geq |P|$

۱-۱- بردار  $M$  را به  $D$  افزوده و از تابع خارج شوید.

۲- اگر  $M(p_n) = \text{bound}(p_n)$

۱-۲- تابع را با پارامترهای  $M, n + 1, \text{bound}$ ، و  $D$  صدا بزن.

۳- در غیر اینصورت

$$\text{Temp} = M(p_n) \quad ۱-۳$$

۳-۲- به ازای نهایی  $i = \text{temp}$  تا  $i \leq \text{bound}(p_n)$

۳-۲-۱.  $M(p_n)$  را برابر  $i$  قرار داده و تابع را با پارامترهای  $D, \text{bound}, n+1, M$  صدا

بزنید.

تعداد نشانه در مکان  $p_i$  در این راه حل می‌تواند یکی از حالات  $M(i)$  تا  $\text{bound}(i)$  باشد که  $\text{bound}(i)$  نشان‌دهنده کران بالای تعداد نشانه‌های مکان  $i$  در تمامی نشانه‌گذاری‌های دسترس‌پذیر است. واضح است پیچیدگی این الگوریتم با توجه به تعداد مکان‌های یک شبکه پتری بصورت نمایی تغییر می‌کند. با استفاده از مفهوم هرس کردن یا استفاده از تنظیم یک برنامه‌نویسی خطی عددی می‌توان این تعداد را کاهش داد.

الگوریتم ۷-۱ نهایتاً تخصیص یافته‌ترین بردارهایی را که یک نشانه‌گذاری قانونی را تخصیص داده و مخصص هیچ نشانه‌گذاری غیرقانونی نباشند را می‌یابد. می‌توان با شروع از تخصیص یافته‌ترین بردار (که برابر با همان بردار  $\text{bound}$  است) هر بار این بردار را به نحوی تعمیم داد که هنوز مخصص نشانه‌گذاری قانونی مربوطه باشد. در اولین گره‌ای که درخت حاصل از تخصیص در هر یک از مکان‌ها شامل فرزندی باشد که هیچ یک از نشانه‌گذاری‌های غیرقانونی را تخصیص نمی‌دهد، این شاخه را هرس نموده و ادامه نمی‌دهیم. گره مربوطه به مجموعه بردارهای تخصیص یافته نشانه‌گذاری  $M_i$  اضافه می‌شود. نهایتاً، از مجموعه تمامی بردارهای تخصیص یافته بدست آمده برای تمامی نشانه-گذاری‌های قانونی، بردارهایی که توسط هیچ بردار دیگری از این لیست تخصیص نمی‌یابد را بعنوان مجموعه کمینه بردارهای تخصیص مجاز انتخاب می‌نماییم.

نتیجه این روش با نتیجه استفاده از الگوریتم ۷-۱ و روش بازگشتی یکسان است. البته، این روش منجر به کاهش بردارهای تخصیص مورد بررسی برای یافتن تخصیص یافته‌ترین بردارها می‌شود، زیرا از تعمیم بیشتر یک گره تخصیص یافته که تخصیص نشانه‌گذاری‌های غیرقانونی است پیشگیری می‌کند. این تعمیم بیشتر لازم نیست. زیرا، در مرحله ۳ از الگوریتم ۷-۱ تمامی این تعمیم‌ها (که بردار تخصیصی برای آنها وجود دارد) حذف خواهند شد.

روش استفاده از برنامه‌نویسی خطی، منجر به تنظیم یک ILP برای تشخیص تخصیص یافته‌ترین بردار مجاز در هر یک از نشانه‌گذاری‌های قانونی خواهد شد که مخصص هیچ یک از نشانه‌گذاری‌های غیرقانونی نیست.

Specification 1:

$$\text{Max } \sum_{i=1}^{|P|} D(p_i) \quad (7-2)$$

s. t.

$$\bigwedge_{i=1}^n \{D(p_i) \geq M_k(p_i)\} = 1 \quad (7-3)$$

$$\bigvee_{i=1}^n \{D(p_i) < M_j(p_i)\} = 1, \quad \forall M_j \in M_{FBM}^* \quad (7-4)$$

معادله (۷-۲) بعنوان تابع هدف، بیشینه کردن تعداد نشانه‌های مکان‌های بردار تخصیص  $D$  را در نظر دارد. بدین ترتیب تخصیص یافته‌ترین بردار مجاز یافته می‌شود. معادله (۷-۳) طبق تعریف ۷-۱ بیانگر ویژگی مخصص بودن  $D$  برای یک نشانه‌گذاری قانونی  $M_k$  است. معادله (۷-۴) نیز نشان دهنده مخصص نبودن  $D$  برای هیچیک از نشانه‌گذاری‌های غیرقانونی است.

نهایتاً با داشتن تمامی  $D$ های تخصیص یافته، که هر یک نماینده تخصیص بهینه مجاز برای یک نشانه‌گذاری قانونی است،  $D$ های تخصیص یافته‌ای را که توسط هیچ یک از بردارهای درون مجموعه

بردارهای تخصیص یافته، تخصیص نمی‌یابند را بدست می‌آوریم. بدین ترتیب، کمینه بردارهای تخصیص محاسبه می‌شود.

روش دومی نیز با استفاده از طراحی یک ILP کلی برای بدست آوردن تمامی بردارهای تخصیص کمینه پیشنهاد می‌شود. اگر یک نشانه‌گذاری قانونی توسط بردار تخصیص  $D$  پوشش داده شود داریم:

$$\bigwedge_{i=1}^n \{D(p_i) \geq M_k(p_i)\} = 1$$

اگر یک نشانه‌گذاری قانونی توسط بردار تخصیص  $D$  پوشش داده نشود داریم:

$$\bigvee_{i=1}^n \{D(p_i) < M_k(p_i)\} = 1$$

با استفاده از یک متغیر دودویی  $f$  برای نشان دادن پوشیده شدن یک نشانه‌گذاری قانونی توسط بردار تخصیص  $M$  ( $f = 1$ ) و عدم پوشش یک نشانه‌گذاری قانونی توسط این بردار ( $f = 0$ )، معادله (۵-۷) را خواهیم داشت.

$$f_k \cdot (\bigwedge_{i=1}^n \{D(p_i) \geq M_k(p_i)\}) + (1 - f_k) \cdot (\bigvee_{i=1}^n \{D(p_i) < M_k(p_i)\}) = 1, \forall M_k \in M_L^* \quad (7-5)$$

معادله (۵-۷) بطور خلاصه بصورت ذیل نوشته می‌شود:

$$\bigwedge_{i=1}^n \{D(p_i) \geq M_k(p_i)\} = f_k \quad (7-6)$$

که باید حداقل یک نشانه‌گذاری قانونی  $M_k$  توسط  $D$  پوشش داده شده باشد:

$$\sum_{k=1}^{|M_L^*|} f_k \geq 1 \quad (7-7)$$

بدین ترتیب معادله ILP بصورت (۸-۷) خواهد شد.

$$\text{Max } \sum_{k=1}^{|M_L^*|} f_k$$

s. t. (7 - 4), (7 - 5), (7 - 6), (7-7)

(7-8)

الگوریتم کلی بدین ترتیب خواهد شد.

الگوریتم ۷-۳- یافتن مجموعه کمینه بردارهای تخصیص قانونی

۱- بردار  $D^* = \emptyset$

۲- تا زمانیکه  $M_L \neq \emptyset$

۲-۱- با استفاده از ILP (۷-۸)، یک بردار تخصیص بهینه را که نماینده بیشترین تعداد نشانه-

گذاری قانونی است بدست آورید.

۲-۲-  $D$  را به مجموعه تخصیص یافته‌ترین بردارهای مجاز اضافه کنید. یعنی  $D^* = D^* \cup D$

$\{D\}$ . و تمامی نشانه‌گذاری‌های قانونی را که  $f_k$  متناظر با آنها برابر ۱ است را از لیست  $M_L$  حذف

کنید. به مرحله ۲ برگردید.

۳- بردار  $D^*$  را بعنوان خروجی اعلام کنید.

به همین ترتیب می‌توانیم یک روش مبتنی بر ILP را برای یافتن مجموعه کمینه حالات مافوق نشانه‌گذاری‌های غیرقانونی بیابیم. بسته به یک مدل شبکه پتری، یکی از این دو معیار کاهش بیشتری در محاسبات کنترل نظارتی ایجاد خواهد نمود. بعنوان مثال اگر تعداد نشانه‌گذاری‌های قانونی نسبت به تعداد نشانه‌گذاری‌های غیرقانونی بیشتر بوده و با استفاده از بردارهای تخصیص کمتری نشان داده شود، می‌توان از مجموعه کمینه بردارهای تخصیص به جای مجموعه کمینه نشانه‌گذاری‌های قانونی پوشش دهنده در یک سیاست کنترل نظارتی همانند MCPP استفاده کرد. البته استفاده همزمان از مفاهیم بردار تخصیص و حالت مافوق امکان‌پذیر نیست و نمی‌توان بردارهای تعمیم یک نشانه‌گذاری غیرقانونی را به نحوی محاسبه نمود که فقط تعمیم دهنده بردارهای تخصیص کمینه نبوده است اما

نشانه‌گذاری‌های قانونی‌ای وجود داشته باشند که توسط بردار تعمیم، تعمیم داده شده و توسط بردارهای تخصیص کمینه تخصیص داده می‌شوند.

## ۴-۷ محاسبه کنترل نظارتی کمینه ساختاری

در بخش قبل بیان شد که می‌توان با استفاده از مفهوم حالت مافوق و یا مفهوم بردار تخصیص به کاهش نشانه‌گذاری‌های مورد بررسی و در نتیجه کاهش محاسبات یک سیاست کنترل نظارتی پرداخت. در ادامه به معرفی یک سیاست کنترل نظارتی کمینه ساختاری می‌پردازیم که از مفهوم حالت مافوق استفاده نموده و آن را همزمان با محاسبه PI به کار می‌بندد. برای این قسمت نیز به طراحی یک ILP می‌پردازیم. این ILP به گونه‌ای طراحی می‌شود که مفاهیم کمینه نشانه‌گذاری‌های غیرقانونی تحت پوشش، کمینه بردارهای مافوق نشانه‌گذاری‌های غیرقانونی و کمینه بردارهای تعمیم غیرقانونی را دربردارد.

۱- حالت مافوق  $G$  باید، هیچ نشانه‌گذاری قانونی را پیشگیری نکند:

$$\sum_{i=1}^n l_i \{M_k(p_i) - G(p_i)\} \leq -1, \forall M_k \in M_L^* \quad (7-9)$$

۲- حالت مافوق  $G$ ، تعمیم هیچ نشانه‌گذاری قانونی نباشد:

$$\forall_{i=1}^n \{G(p_i) > M_k(p_i)\} = 1, \forall M_k \in M_L^* \quad (7-10)$$

۳- ضرایب یک PI طراحی شده برای پیشگیری از حالت مافوق  $G$  می‌تواند از نشانه‌گذاری غیرقانونی  $M_j$  نیز پیشگیری کند:

$$\sum_{i=1}^n l_i (M_j(p_i) - G(p_i)) \geq O(h_j - 1), \forall M_j \in M_{FBM}^* \quad (7-11)$$

در معادله (۷-۱۱)  $h_j$  یک متغیر دودویی است که نشان دهنده این است که نشانه‌گذاری  $M_j$  توسط حالت مافوق  $G$  ( $h_j = 1$ ) پوشش داده می‌شود یا خیر.

۴- حالت مافوق  $G$  باید حداقل یک  $M_f$  را پوشش دهد.

$$\sum_{j=1}^{|M_{FBM}^*|} h_j \geq 1 \quad (7-12)$$

۵- حالت مافوق  $\vec{0} \neq G$  است.

$$\sum_{i=1}^n G(p_i) \geq 1 \quad (7-13)$$

نهایتاً هدف بهینه‌سازی، افزایش تعداد نشانه‌گذاری‌های غیرقانونی است که با استفاده از حالت مافوق  $G$  پوشیده شده‌اند. بدین ترتیب ضرایب  $L_i$  نیز برای PI جهت کنترل  $G$  بدست خواهند آمد. روش ILP کلی در معادله (۷-۱۴) آمده است.

$$Max \sum_{j=1}^{|M_{FBM}^*|} h_j$$

s. t.

$$(7-9), (7-10), (7-11), (7-12), (7-13) \quad (7-14)$$

با دقت در معادله (۷-۱۴) مشخص است که این معادله برنامه‌نویسی غیرخطی است. زیرا در معادله (۷-۹) مجهول  $l_i$  در مجهول  $G$  ضرب می‌شود. البته، در این مرحله از کنترل نظارتی بردار  $G$  مستقیماً مورد نیاز نیست و تنها نیاز به دانستن  $L$  برای محاسبه مکان کنترلی است. بنابراین با حذف  $G$  از معادلات و استفاده کردن از مجهول جدید  $GprodL = L.G$  به جای آن معادله (۷-۱۴) به معادله برنامه‌نویسی خطی (۷-۱۵) تبدیل می‌شود:

$$Max \sum_{j=1}^{|M_{FBM}^*|} h_j$$

s. t.



$$\sum_{i=1}^n l_i \cdot M_k(p_i) - \text{GprodL}(p_i) \leq -1, \forall M_k \in M_L^*$$

$$\bigvee_{i=1}^n \{\text{GprodL}(p_i) > l_i \cdot M_k(p_i)\} = 1, \forall M_k \in M_L^*$$

$$\sum_{i=1}^n l_i \cdot M_j(p_i) - \text{GprodL}(p_i) \geq O(h_j - 1), \forall M_j \in M_{FBM}^*$$

$$\sum_{j=1}^{|M_{FBM}^*|} h_j \geq 1$$

$$\sum_{i=1}^n \text{GprodL}(p_i) \geq 1 \quad (7-15)$$

در ILP (۷-۱۵) معادله  $\bigvee_{i=1}^n \{\text{GprodL}(p_i) > l_{j,i} \cdot M_k(p_i)\} = 1, \forall M_k \in M_L^*$  که جایگزین

معادله (۷-۱۰) شده است، در صورتی برقرار است که ضرایب  $l_{j,i}$  در معادله ILP بصورت اعداد مثبت تعریف شوند.

از طرفی، معادله (۷-۱۵) قادر به محاسبه سوپروایزر بیشینه مجازیت و کمینه ساختاری در صورت وجود است. در صورتی که معادله بیشینه مجازیت وجود نداشته باشد، یعنی در هنگامیکه یک نشانه‌گذاری غیرقانونی در پوشش محدب مجموعه نشانه‌گذاری‌های قانونی قرار داشته باشد، معادله (۷-۱۵) هیچ راه حلی را نشان نخواهد داد. در این مواقع، بهتر است با دسترس‌ناپذیر نمودن برخی از نشانه‌گذاری‌های قانونی به‌مراه تمامی نشانه‌گذاری‌های غیرقانونی، یک سوپروایزر بیشترین مجازیت بدست آورد. برای نیل به این هدف، در یک مرحله پیش‌پردازش، تمامی نشانه‌گذاری‌های قانونی را که حداقل از یک نشانه‌گذاری غیرقانونی بزرگتر هستند حذف می‌نماییم. سپس، به محاسبه کنترل نظارتی برای حفظ نشانه‌گذاری‌های قانونی باقیمانده می‌پردازیم. روند کلی محاسبه کنترل نظارتی در الگوریتم ۷-۴ نشان داده شده است.

الگوریتم ۷-۴- محاسبه کنترل نظارتی بیشترین مجازیت (در صورت امکان بیشینه مجازیت) با

ساختار کمینه و پیچیدگی محاسباتی پایین

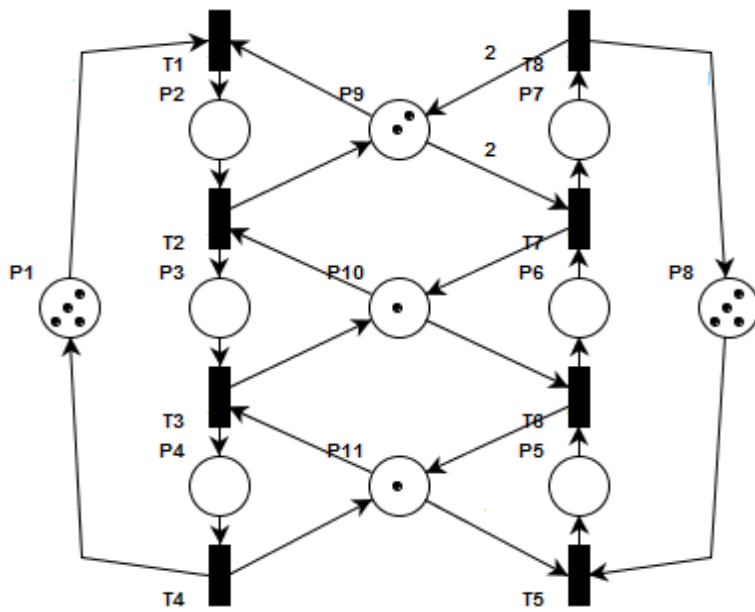
ورودی: یک مدل شبکه پتری از یک سیستم  $(N, M_0)$

خروجی: مدل کنترل شده سیستم

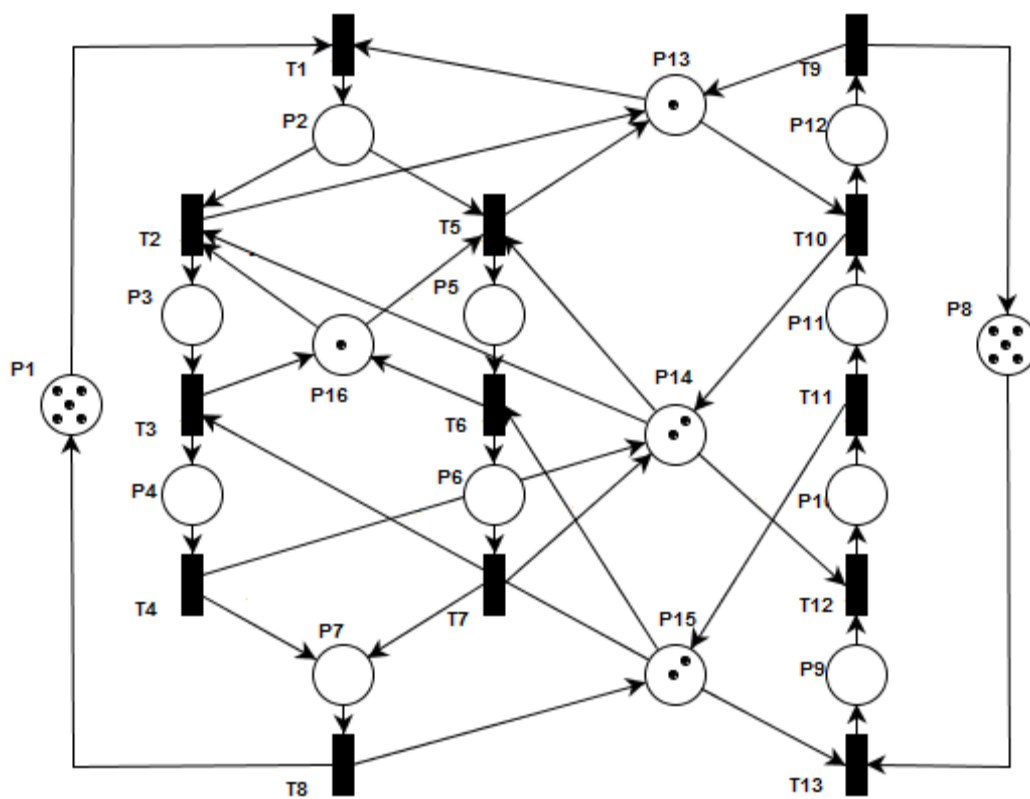
- ۱- گراف دسترسی شبکه پتری را بدست بیاورید. با استفاده از تحلیل گراف، نشانه‌گذاری‌های قانونی،  $M_L$ ، و نشانه‌گذاری‌های غیرقانونی مرزی  $M_{FBM}$  را محاسبه کنید.
- ۲- به منظور داشتن سوپروایزر بیشترین مجازیت در صورت وجود نداشتن سوپروایزر بیشینه مجازیت، تمامی نشانه‌گذاری‌های قانونی که حداقل از یک نشانه‌گذاری غیرقانونی بزرگتر هستند را از مجموعه نشانه‌گذاری‌های قانونی حذف کنید.
- ۳-  $V = \emptyset$  را مجموعه مکان‌های کنترلی اضافه شونده به شبکه پتری ورودی در نظر بگیرید.
- ۴- تا زمانی که  $M_{FBM} \neq \emptyset$
- ۴-۱- با استفاده از ILP (۷-۱۵)، ضرایب PI را برای یک حالت مافوق بهینه که نماینده بیشترین تعداد نشانه‌گذاری‌های غیرقانونی است بدست بیاورید.
- ۴-۲- با استفاده از این ضرایب  $L_i$  و معادله (۳-۶) و (۳-۷)، یک مکان کنترلی جدید را برای کنترل حالت مافوق بدست آمده مشخص کنید. این مکان کنترلی را به  $V$  اضافه کنید.
- ۴-۳- نشانه‌گذاری‌های غیرقانونی را که  $h_j$  متناظر با آنها برابر ۱ است را از لیست  $M_{FBM}$  حذف کنید. به مرحله ۲ بازگردید.
- ۵- مکان‌های کنترلی  $V$  را به شبکه  $(N, M_0)$  اضافه کنید تا شبکه کنترل شده با کمینه ساختار بدست آید.

## ۷-۵ نتایج تجربی

در این قسمت مثال‌هایی از شبکه‌های پتری را بیان می‌کنیم و به مقایسه سرعت الگوریتم کنترل نظارتی مبتنی بر حالت مافوق به روش تکرار شونده [۱۴۰] و روش مبتنی بر ILP می‌پردازیم.

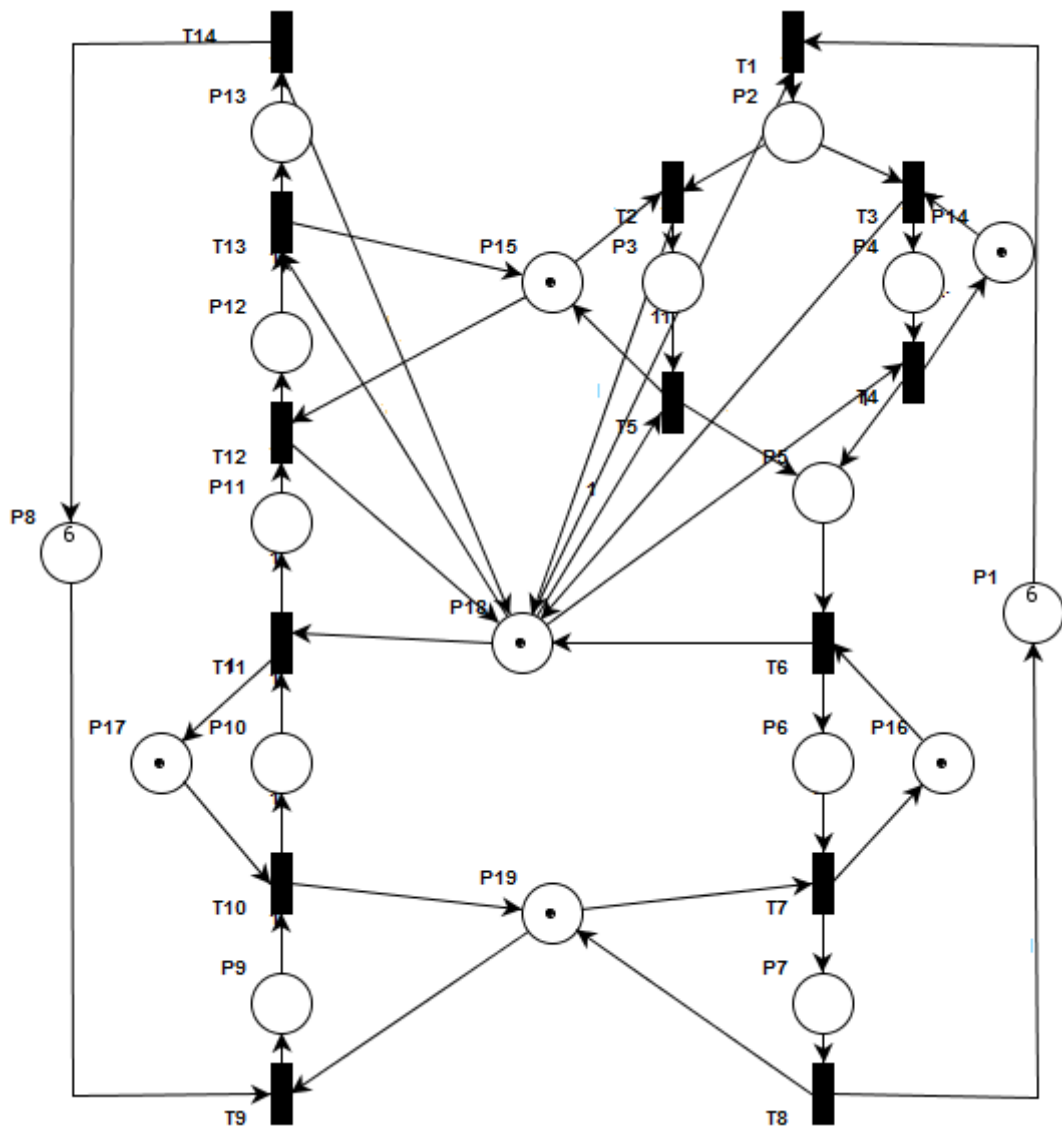


شکل ۱-۷ یک مدل شبکه پتری.

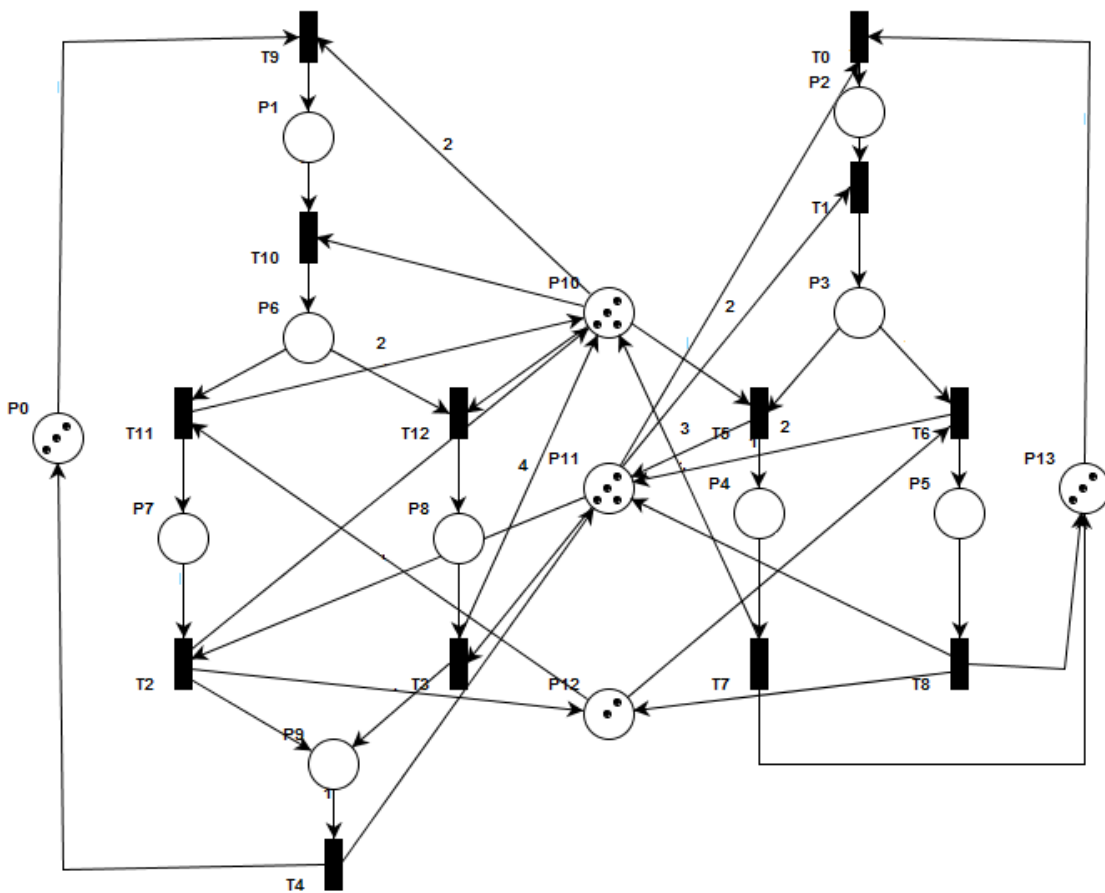


شکل ۲-۷ یک مدل شبکه پتری برگرفته از مرجع [۴۳].

جدول ۱-۷ به معرفی تعداد نشانه‌گذاری قانونی و غیر قانونی هر یک از مثال‌ها پرداخته و مقایسه‌ای از نظر سرعت محاسبه کنترل نظارتی برای هر یک از این مثال‌ها و با استفاده از الگوریتم تکرار شونده [۱۴۰] و الگوریتم مبتنی بر ILP برای محاسبه حالت مافوق می‌پردازد. نتایج بدست آمده از این جدول حاکی از بهبود سرعت در روش الگوریتم مبتنی بر ILP نسبت به الگوریتم تکرار شونده [۱۴۰] در محاسبه کنترل نظارتی مبتنی بر حالت مافوق است.



شکل ۳-۷- یک مدل شبکه پتری برگرفته از مرجع [۱۱۱].



شکل ۴-۷ یک مدل شبکه پتری برگرفته از مرجع [۴۲].

جدول ۱-۷ مقایسه سرعت الگوریتم کنترل نظارتی مبتنی بر حالت مافوق با استفاده از روش تکرار شونده [۱۴۰] و روش پیشنهادی مبتنی بر ILP.

تعداد نشانه گذاری قانونی	تعداد نشانه گذاری غیرقانونی مرزی	زمان روش تکرار شونده [۱۴۰]	زمان محاسبه روش پیشنهادی	شکل
۱۹	۷	۱/۲۱۵	۰/۹۷۶	شکل ۱-۷
۲۲۱	۱۱	۱۹۷/۲۱	۵/۰۰۱	شکل ۲-۷
۲۰۵	۵۴	۱۷۳۴/۹۸۲	۷/۰۳۲	شکل ۳-۷
۳۲۳	۳۸	۱۲۱/۷۵۴	۵/۸۲۵	شکل ۴-۷

این زمان‌ها روی کامپیوتر با مشخصات پروسسور Intel(R) Pentium(R) Dual CPU T2410 @ 2.00GHz و ۲ گیگابایت حافظه RAM بر روی سیستم عامل ۳۲ بیتی ویندوز ۱۰ بدست آمده است.

## ۶-۷ جمع بندی

در این فصل به معرفی یک روش کنترل نظارتی مطلوب که بیشینه مجازیت، کمینه ساختاری و با پیچیدگی محاسباتی پایین است پرداختیم. استفاده از بردار تخصیص موجب کمینه شدن نشانه-گذاری‌های قانونی مورد بررسی شده است. همچنین محاسبات حالات مافوق را در ضمن معادلات ILP محاسبات کنترل نظارتی انجام دادیم. نتایج تجربی نشان‌دهنده بهبود سرعت محاسبات در روش پیشنهادی نسبت به روش تکرارشونده [۱۴۰] است. با استفاده از روش‌های محاسبات کنترل نظارتی پیمان‌های که در دو فصل گذشته به آنها اشاره شد، می‌توان محاسبات مربوط به این روش را کاهش داد.

## فصل ۸: نتیجه‌گیری و پیشنهاد کارهای آتی

## ۸-۱ نتیجه‌گیری

در این رساله، به بررسی مسئله مدلسازی فرمال کنترل نظارتی در سیستم‌های پیچیده صنعتی پرداخته‌ایم. سیستم‌های صنعتی، بدلیل استفاده از دستورات کنترلی کامپیوتری، غالباً در کلاس سیستم‌های گسسته رخداد تقسیم‌بندی شده و یا می‌توانند به عنوان یک سیستم گسسته رخداد تجرید شوند. از طرفی، مقیاس بزرگ این سیستم‌ها، گاه استفاده از الگوریتم‌های کنترل نظارتی موجود را ناکارآمد می‌سازد.

فرمالیزم شبکه‌های پتری دارای قابلیت‌هایی از جمله داشتن ساختار فشرده، داشتن نمایش گرافیکی در کنار پایه و اساس ریاضیاتی، قابلیت تبدیل به کد قابل اجرای سخت افزاری بوده و یک فرمالیزم قدرتمند در مدلسازی و اعتبارسنجی سیستم‌های گسسته رخداد است. در این رساله، با بهره‌گیری از این فرمالیزم، به مدلسازی و تحلیل سیستم‌های تخصیص منابع پرداختیم.

یکی از مسائلی که در کنترل سیستم‌های گسسته رخداد با مقیاس بزرگ وجود دارد، مسئله پیچیدگی محاسباتی در طراحی کنترل نظارتی است. روش‌های مختلفی برای طراحی کنترل نظارتی توزیع شده مطرح شده‌اند. اما این روش‌ها نیز همانند روش‌های کنترل نظارتی متمرکز از گراف دسترسی سیستم کلی بعنوان ورودی استفاده می‌کنند. محاسبه گراف دسترسی در سیستم‌های بزرگ-مقیاس می‌تواند منجر به مسئله انفجار فضای حالت شود. از جمله کارهایی که در مسیر رساله به انجام رسید طراحی پیمان‌های کنترل نظارتی بود. فصل ۴ این رساله عمدتاً به مطرح ساختن و اثبات قضایایی در زمینه امکان انجام محاسبات کنترل نظارتی بصورت پیمان‌های و برای برخی از عملگرهای سنتز مطرح در شبکه‌های پتری می‌پردازد. اگر یک عملگر، امکان انجام محاسبات پیمان‌های کنترل نظارتی را در بین پیمان‌های تشکیل دهنده فراهم کند، گراف دسترسی نیز می‌تواند بصورت پیمان‌های محاسبه شود. در طراحی پیمان‌های، ابتدا پیمان‌های تشکیل دهنده سیستم شناسایی شده و مدل می-



شوند. سپس کنترل نظارتی برای اعمال زنده بودن بر روی هر یک از این پیمانها اعمال خواهد شد. نهایتاً پیمانهای زنده باهم سنتز شده و سیستم کلی را می‌سازند که الزاماً و بدون عملیات اعتبارسنجی دیگری زنده است.

همچنین در فصل ۴ اثبات کردیم اگر برهم‌کنش پیمانهای تشکیل دهنده یک سیستم با استفاده از عملگرهایی باشد که امکان اجرای هم‌زمان پیمانهای تشکیل دهنده را ایجاد نمی‌کنند، طراحی پیمانهای کنترل نظارتی می‌تواند مورد استفاده قرار گیرد. عملگرهای انتخاب و ترتیب از جمله مهمترین این عملگرها هستند.

همچنین، اثبات نمودیم که اگر برهم‌کنش پیمانهای یک سیستم با استفاده از عملگر برگ-برگ‌سازی انجام گیرد، اعمال پیمانهای کنترل نظارتی می‌تواند جایگزین اعمال کنترل نظارتی در سیستم کلی شود. نتایج تجربی بدست آمده، نشان دهنده کاهش بسیار در فضای حالت پیمانها نسبت به فضای حالت سیستم کلی است که منجر به کاهش سرشار محاسباتی کنترل نظارتی اعمال کننده زنده بودن در یک سیستم مبتنی بر شبکه‌های پتری خواهد شد.

روش‌های سنتز شبکه‌های پتری در صورتیکه حافظ درستی باشند، می‌توانند در محاسبه کنترل نظارتی پیمانهای مورد استفاده قرار گیرند. یکی از مهمترین روش‌های سنتز متصور در شبکه‌های پتری مبتنی بر سیستم‌های تخصیص منابع، ادغام مکان‌ها است. شرایط مناسب جهت حافظ درستی بودن عملگر ادغام مکان‌ها در تحقیقات پیشینان مورد بررسی قرار گرفته است. البته بسیار محتمل است که یک شبکه پتری دارای شرایط حفظ زنده بودن توسط عملگر ادغام نباشد. در این حالت، نیز استفاده از کنترل نظارتی پیمانهای می‌تواند منجر به کاهش محاسبات کنترل نظارتی شود. دلیل این کاهش، کاهش گراف دسترسی مورد بررسی نیست بلکه به کاهش تعداد سایفون‌های کنترل نشده مورد بررسی می‌پردازد. در این روش، به شناسایی سایفون‌های کمینه‌ای که پس از ادغام مکان‌ها ایجاد می‌شوند پرداختیم. اثبات کردیم که اگر پیمانهای شبکه‌های پتری زنده باشند (یا با اعمال کنترل

نظارتی، به یک شبکه پتری زنده تبدیل شوند) و سایفون‌های کمینه و پایه‌ای که دارای حداقل دو مکان مشترک هستند تحت کنترل قرار گیرند، شبکه نهایی یک شبکه زنده است.

فرمالیزم شبکه‌های پتری، بطور ذاتی فاقد روش‌های ترکیب‌پذیری است. البته تحقیقات بسیاری برای جبران این نقص انجام شده است. حاصل این تحقیقات، عملگرهای سنتر هستند که از جمله آنها می‌توان به عملگرهای ادغام و ترکیب اشاره کرد. از جمله کارهایی که در مسیر رساله به انجام رسید و در فصل ۵ به تفصیل به آن پرداخته‌ایم، مطرح ساختن و اثبات قضایایی در زمینه حفظ ویژگی‌های مهم، همانند زنده بودن، پس از اعمال عملگر ادغام گذرها در یک شبکه پتری است. در مسیر این اثبات، ابتدا به معرفی و اثبات شرایطی برای حفظ ویژگی‌های ساختاری نظیر بدون تصادم بودن، معمولی بودن و قویا همبند بودن یک شبکه پتری پس از اعمال عملگر ادغام گذرها پرداختیم. سپس، با استفاده از روش‌های مشخصه‌سازی، به نتیجه‌گیری در مورد حفظ زنده بودن با اعمال عملگر ادغام گذرها پرداختیم.

همچنین بدلیل وجود یک تناظر یک به یک بین عملگر همگام‌سازی و عملگر ادغام گذرهای غیرمجاور، از قضایای بدست آمده در زمینه حفظ ویژگی تحت عملگر ادغام گذرها، برای اثبات حفظ ویژگی در عملگر همگام‌سازی استفاده نمودیم. از آنجا که عملگر همگام‌سازی، نوعی از عملگر ترکیب چند پیمانانه است، از آن برای طراحی پیمانانه‌ای یک سیستم تولیدی خودکار استفاده نمودیم.

در فصل ۶ این رساله نیز به ارائه نتایجی در زمینه محاسبه کنترل نظارتی بیشینه مجازیت با ساختار کمینه و محاسبات پایین با استفاده از مفهوم حالت مافوق یک مجموعه نشانه‌گذاری پرداختیم. نتایج ارائه شده در قالب مثال بیانگر کاهش زمان محاسبه حالات مافوق در ضمن معادله ILP طراحی شده برای سوپروایزر نسبت به روش تکرار شونده‌است. همچنین، استفاده از این روش، در قالب روش‌های کنترل نظارتی پیمانانه‌ای می‌تواند منجر به کاهش بیشتر بار محاسباتی کنترل نظارتی نیز شود.

## ۸-۲ پیشنهاد کارهای آتی

نتایج استفاده از محاسبات پیمانهای در طراحی یک سوپروایزر اعمال کننده زنده بودن، گاه بسیار جالب توجه است. در این رساله اثبات شد در صورتیکه پیمانهای پایه‌ای یک سیستم دارای برهم‌کنش‌هایی از نوع ترتیب، انتخاب و برگ‌برگ‌سازی باشند، محاسبات کنترل نظارتی سیستم کلی می‌تواند بصورت پیمانهای انجام گیرد. همچنین، در مورد عملگر ادغام مکان‌ها، ثابت شد اگر سیستمی دارای شروط حفظ زنده بودن پس از اعمال این عملگر باشد، می‌توان محاسبات را به صورت پیمانهای به پیش برد.

محاسبه برخط کنترل نظارتی، محاسبه کنترل نظارتی مبتنی بر روش‌های برنامه‌نویسی خطی و با در نظر گرفتن گذرهای غیرقابل کنترل یا غیرقابل مشاهده، محاسبه کنترل نظارتی که الزاما از نظر ارتباطی کمینه است و توسعه روش‌های کنترل نظارتی در صورت وجود نداشتن سوپروایزر مبتنی بر محدودیت‌های انحصار متقابل، از جمله این مسائل هستند.

یافتن راهکارهایی برای محاسبات پیمانهای کنترل نظارتی با استفاده از عملگرهای سنتز دیگری که در شبکه‌های پتری تعریف شده‌اند و در شرایط برقرار نبودن شروط حفظ زنده بودن این عملگرها، می‌تواند در ادامه مسیر این رساله مورد توجه قرار گیرد.

همچنین معرفی عملگرهای جدید سنتر شبکه‌های پتری و اثبات سریع حافظ زنده بودن آنها با استفاده از رویکردی مشابه رویکرد فصل ۴ می‌تواند انجام گیرد.

در بسیاری از مواقع با سیستمی سروکار داریم که پیمانهای پایه‌ای آن از نظر فیزیکی مجزا نیستند. بنابراین تشخیص پیمانها و برهم‌کنش آنها و تجزیه مدل کلی به مدل پیمانها روند واضح و آسانی ندارد. از این رو پرداختن به تشخیص اتوماتیک پیمانها و تجزیه اتوماتیک مدل کلی به

زیرسیستم‌ها کاری بسیار ضروری است که در ادامه روند محاسبه کنترل نظارتی پیمان‌های و در جهت تمام اتوماتیک کردن آن باید مورد توجه قرار گیرد.

بررسی روشی برای کاهش سیستم کلی و احیانا حذف گذرهای زنده می‌تواند مورد بررسی قرار گیرد. بدین ترتیب با داشتن سیستم ساده‌تری که تنها در بردارنده گذرهای غیرزنده است، اعمال کنترل نظارتی بر سیستم ساده‌سازی شده می‌تواند مورد مطالعه قرار گیرد.

یافتن راهکارهایی برای ترکیب روش‌های اعمال کنترل نظارتی توزیع شده با روش‌های محاسبه پیمان‌های کنترل نظارتی نیز از جمله کارهایی است که می‌تواند در ادامه مسیر این رساله انجام شود.

محاسبه کنترل نظارتی مبتنی بر روش‌های برنامه‌نویسی خطی و با در نظر گرفتن گذرهای غیرقابل کنترل یا غیرقابل مشاهده نیز از مسائل باز این حوزه است.

همچنین استفاده از مفهوم تابع نگهبان در شبکه‌های پتری رنگی برای ایجاد کنترل نظارتی برای محدودیت‌های غیرخطی با توابع دلخواه می‌تواند مورد بررسی قرار گیرد.

از دیگر مسائل مهم و البته حل نشده در این زمینه، محاسبه برخط کنترل نظارتی است که نیاز به کاهش شدید بار محاسباتی در سیاست‌های مورد استفاده دارد. یکی از دیگر مسائل مهم در سیستم‌های تخصیص منابع، تخصیص منصفانه منابع است. مدلسازی انصاف در تخصیص منابع با استفاده از معادلات ریاضی و اعمال آن با استفاده از کنترل نظارتی، از دیگر مسائلی است که می‌تواند مورد بررسی و تحقیق قرار گیرد.

- [1] Desel J. and Esparza J. (1995), "Free Choice Petri Nets", Vol. Cambridge University Press.
- [2] Tricas F. and Ezpeleta J. (2006) "Computing Minimal Siphons in Petri Net Models of Resource Allocation Systems: A Parallel Solution" **IEEE Transactions on Systems , Man, and Cybernetics - Part A: Systems and Humans**, 36, 3, pp 532.
- [3] Miremadi S., (2012), "Symbolic Supervisory Control of Timed District Event Systems", **Chalmers University of Technology**.
- [4] Frey G. and Wagner F., (2006) "A Toolbox for the Development of Logic Controllers using Petri Nets", Discrete Event Systems, 2006 8th International Workshop on, Proceeding of IEEE, P473, Ann arbor, MI.
- [5] Wu Z. and Hsieh S. J. (2012) "A Realtime Fuzzy Petri Net Diagnoser for Detecting Progressive Faults in PLC Based Discrete Manufacturing System" **The International Journal of Advanced Manufacturing Technology**, 61, 1, pp 405.
- [6] Petit N., Camponogara E., Prati T. J., Farines J. M., de Queiroz M. H. (2015) "Automatic Test of Safety Specifications for PLC Programs in the Oil and Gas Industry" **2nd IFAC Workshop on Automatic Control in Offshore Oil and Gas Production OOGP 2015**, 48, 6, pp 27.
- [7] Haddad S., Kordon F., Pautet L., and Petrucci L. (2011). Introduction to Formal Models, pp 353, In: "Models and Analysis in Distributed Systems", S.Haddad, F. Kordon, L. Pautet, Petrucci L, John Wiley & Sons, Inc., United States.
- [8] Giua A., (1992), "Petri Nets as Discrete Event Models for Supervisory Control", **Rensselaer Polytechnic Institute**.
- [9] Jafarinejad F. and Pouyan A. A. (2016) "Distributed Computation of Liveness Enforcing Supervisory Control of Flexible Manufacturing Systems" **Advanced Manufacturing Systems**, 16, 1, pp1-15.
- [10] Shengbing J. and Kumar R., (2001) "Supervisory Control of Discrete Event Systems with CTL\* Temporal Logic Specifications", Decision and Control, 2001 Proceedings of the 40th IEEE Conference on, P4122, Florida, USA.
- [11] Younes O. S. (2016) "A Secure DHCP Protocol to Mitigate LAN Attacks" **Journal of Computer and Communications** 4, 1, pp 39.
- [12] Murata T. (1989) "Petri Nets: Properties, Analysis and Applications" **Proceedings of the IEEE**, 77, 4, pp 541.
- [13] Egerstedt M., Wardi Y., Haar S., and Theissing S. (2015) "Analysis and Design of Hybrid Systems ADHSA Hybrid-Dynamical Model for Passenger-flow in Transportation Systems" **IFAC-PapersOnLine**, 48, 27, pp 236.
- [14] Mecitoğlu F. and Söylemez M. T. (2013) "A UML Modelling Approach for a Railway Signalization System Simulator and SCADA System" **1st IFAC Workshop on Advances in Control and Automation Theory for Transportation Applications**, 46, 25, pp 77.

- [15] Cai H., Zhang C., Wu W., Ho T. k., and Zhang Z. (2014) "Modelling High Integrity Transport Systems by Formal Methods" **The 9th International Conference on Traffic and Transportation Studies (ICTTS 2014)**, 138, pp 729.
- [16] Henderson T. R., Roy S., Floyd S., and Riley G. F., (2006) "Ns-3 Project Goals", Proceeding from the 2006 workshop on ns-2: the IP network simulator, P 13, Pisa, Italy.
- [17] Varga A. and Hornig R., (2008) "An Overview of the OMNeT++ Simulation Environment", Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, P 1, Marseille, France.
- [18] Chen J. and Kumar R., (2014) "Failure Diagnosis of Discrete-Time Stochastic Systems Subject to Temporal Logic Correctness Requirements", Networking, Sensing and Control (ICNSC), 2014 IEEE 11th International Conference on, P42, Miami, USA.
- [19] Leung Pah Hang M. Y., Martinez-Hernandez E., Leach M., and Yang A. (2016) "Towards a Coherent Multi-Level Framework for Resource Accounting" **Journal of Cleaner Production**, 125, pp 204.
- [20] Zinoviev A., Zinovieva O., Ploshikhin V., Romanova V., and Balokhonov R. (2016) "Evolution of Grain Structure During Laser Additive Manufacturing. Simulation by a Cellular Automata Method" **Materials & Design**, 106, pp 321.
- [21] Chen Y., Li Z., Barkaoui K., and Uzam M. (2016) "Monitor Design with Multiple Self-Loops for Maximally Permissive Supervisors" **ISA Transactions**, 61, pp 129.
- [22] Kress-Gazit H., Wongpiromsarn T., and Topcu U. (2011) "Correct, Reactive Robot Control from Abstraction and Temporal Logic Specifications" **Special Issue of the IEEE Robotics and Automation Magazine on Formal Methods for Robotics and Automation**, 18, 3, pp 65.
- [23] Harrison P. G. and Strulo B. (jan. 2000) "SPADES- a Process Algebra for Discrete Event Simulation" **journal of Logic and Computation**, 10, 1, pp 2.
- [24] Schutter B. D. and Boom T., (2008) "Max-Plus Algebra and Max-Plus Linear Discrete Event Systems: An Introduction", Discrete Event Systems, 2008 WODES 2008 9th International Workshop on.
- [25] Ramadge P. J. G. and Wonham W. M. (1989) "The Control of Discrete Event Systems" **Proceedings of the IEEE**, 77, 1, pp 81.
- [26] Wysk R. A., Yang N. S., and Joshi S. (1991) "Detection of Deadlocks in Flexible Manufacturing Cells" **IEEE Transactions on Robotics and Automation**, 7, 6, pp 853.
- [27] Fanti M. P. and Zhou M. C. (2004) "Deadlock Control Methods in Automated Manufacturing Systems" **IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans**, 34, 1, pp 5.
- [28] Billington J., Wheeler G. R., and Wilbur-Ham M. C. (1988) "PROTEAN: A High-Level Petri Net Tool For the Specification and Verification of Communication Protocols" **IEEE Transactions on Software Engineering**, 14, 3, pp 301.
- [29] Falkman P., (2005), "Specification of Resource Allocation Systems: a Step towards a Unified Framework", Depart. Signal and Systems. Chalmers University.

[30] Lennartson B., Wigström O., Fabian M., and Basile F. (2014) "Unified Model for Synthesis and Optimization of Discrete Event and Hybrid Systems" **12th IFAC International Workshop on Discrete Event Systems**, 47, 2, pp 86.

[۳۱] دیاز م. (۱۳۹۲)، "شبکه های پتری: مدل های بنیادی، راستی آزمایی و کاربرد"، جلد اول، انتشارات دانشگاه صنعتی شاهرود، شاهرود، ایران.

[32] Leoni m. d., Maggi f. m., and Aalst W. M. P. (2015) "An Alignment-Based Framework to Check the Conformance of Declarative Process Models and to Preprocess Event-Log Data" **Information Systems**, 47, pp 258.

[33] Auer A., Dingel J., and Rudie K. (2014) "Concurrency Control Generation for Dynamic Threads Using Discrete Event Systems" **Science of Computer Programming**, 82, pp 22.

[34] Ghasemieh H., Remke A., and Haverkort B. R. (2016) "Survivability Analysis of a Sewage Treatment Facility Using Hybrid Petri Nets" **Performance Evaluation**, 97, pp 36.

[35] Petri C. A., (1962), "**Kommunikation mit Automaten** ," **Phys. Depart. Darmstadt University of Technology**.

[36] Marinescu D. C. (may 2013). Parallel and Distributed Systems, pp 21, In: "**Cloud Computing Theory and Practice**", Elsevier Inc.

[37] Jeng M. D. (1995) "Modular Synthesis of Petri Nets for Modeling Flexible Manufacturing Systems" **International Journal of Flexible Manufacturing Systems**, 7, 3, pp 287.

[38] Berthelot G. (1987). Transformations and decompositions of nets, pp 359, In: "**Petri Nets: Central Models and Their Properties: Advances in Petri Nets**", Brauer W, Reisig W, Rozenberg G, Springer Berlin Heidelberg, Berlin, Heidelberg.

[39] Jiao L., Cheung T.-Y., and Lu W. (2004) "On Liveness and Boundedness of Asymmetric Choice Nets" **Theoretical Computer Science**, 311, 1, pp 165.

[40] Huang H., Jiao L., and Cheng T. Y. (2012). Petri Net Processes and Extension Operators, pp In: "**Property Preserving Petri net Process Algebra in Software Engineering**", World Scientific Publishing Co. Pte. Ltd.

[41] Ezpeleta J., Tricas F., Garcia-Valles F., and Colom J. M. (2002) "A Banker's Solution for Deadlock Avoidance in FMS with Flexible Routing and Multiresource States" **IEEE Transactions on Robotics and Automation**, 18, 4, pp 621.

[42] Park J. and Reveliotis S. A. (2001) "Deadlock Avoidance in Sequential Resource Allocation Systems with Multiple Resource Acquisitions and Flexible Routings" **IEEE Transactions on Automatic Control**, 46, pp 1572.

[43] Huang Y. S., Jeng M., Xie X., and Chung D. H. (2006) "Siphon-Based Deadlock Prevention Policy for Flexible Manufacturing Systems" **IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans**, 36, 6, pp 1248.

[44] Zhong C. and Li Z. (2010) "Self-Liveness of a Class of Petri Net Models for Flexible Manufacturing Systems" **IET Control Theory & Applications**, 4, 3, pp 403.

[45] Liu G., Li Z., and Zhong C. (2010) "New Controllability Condition for Siphons in a Class of Generalised Petri Nets" **IET Control Theory & Applications**, 4, 5, pp 854

- [46] López-Grao J. P., and Colom J. M. (2013). Structural Methods for the Control of Discrete Event Dynamic Systems C the Case of the Resource Allocation Problem, pp 257, In: "**Control of Discrete-Event Systems**", Seatzu C, Lecture Notes in Control and Information Sciences (LNCIS).
- [47] Liao H., Lafortune S., Reveliotis S., Wang Y., and Mahlke S. (2013) "Optimal Liveness-Enforcing Control for a Class of Petri Nets Arising in Multithreaded Software" **IEEE Transactions on Automatic Control**, **58**, **5**, pp **1123**.
- [48] Chen Y., Li Z., Khalgui M., and Mosbahi O. (2011) "Design of a Maximally Permissive Liveness- Enforcing Petri Net Supervisor for Flexible Manufacturing Systems" **IEEE Transactions on Automation Science and Engineering**, **8**, **2**, pp **374**.
- [49] Aybar A. and İftar A. (2012) "Supervisory Controller Design to Enforce Some Basic Properties in Timed-Transition Petri Nets Using Stretching" **Nonlinear Analysis: Hybrid Systems**, **6**, **1**, pp **712**.
- [50] Luo J., Ni H., Wu W., Wang S., and Zhou M. (2015) "Simultaneous Reduction of Petri Nets and Linear Constraints for Efficient Supervisor Synthesis" **IEEE Transactions on Automatic Control**, **60**, **1**, pp **88**.
- [51] Elhog-Benzina D., Haddad S., and Hennicker R. (2012). Refinement and Asynchronous Composition of Modal Petri Nets, pp 96, In: "**Transactions on Petri Nets and Other Models of Concurrency V**", Springer Berlin Heidelberg.
- [52] Jiao L., Huang H. J., and Cheung T.Y. (2005) "Property-Preserving Composition by Place Merging" **World Scientific, Journal of Circuits, Systems and Computers** **14**, **4**, pp **793**.
- [53] Li J., Zhou M., and Dai X. (2012) "Reduction and Refinement by Algebraic Operations for Petri Net Transformation" **IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans**, **42**, **5**, pp **1244**.
- [54] Pouyan A. A., Shandiz H. T., and Arastehfar S. (2011) "Synthesis a Petri Net Based Control Model for a FMS Cell" **Computers in Industry**, **62**, **5**, pp **501**.
- [55] Huang H. J., Jiao L., Cheung T. Y., and Mak W. M. (2012). Composition Operators of PPPA, pp 61, In: "**property-preserving petri net process algebra in software engineering**", World Scientific Publishing Co .
- [56] Jiao L., Cheung T., and Lu W. (2005) "Handling Synchronization Problem in Petri net-based System Design by Property-Preserving Transition-Reduction" **Oxford, The Computer Journal**, **48**, **6**, pp **692**.
- [57] Huang H. J., Jiao L., and Cheung T. Y. (2005) "Property-Preserving Subnet Reductions for Designing Manufacturing Systems with Shared Resources" **Theoretical Computer Science**, **332**, **1**, pp **461**.
- [58] Huang H. J., Jiao L., and Cheung T. Y. (2012). Property-Preserving Refinements, pp 101, In: "**Property Preserving Petri net Process Algebra in Software Engineering**", World Scientific Publishing Co. Pte. Ltd.
- [59] Chao D. Y., MengChu Z., and Wang D. T., (1992) "Extending Knitting Technique To Petri Net Synthesis Of Automated Manufacturing Systems", Computer Integrated Manufacturing, 1992, Proceedings of the Third International Conference on.



- [60] Huang H. J., Cheung T. Y., and Wang X. L. (2007) "Applications of Property-Preserving Algebras to Component-Based Manufacturing System Design" **Journal of Information Science and Engineering**, **23**, pp 167.
- [61] Basile F., Cordone R., and Piroddi L. (2013) "Integrated Design of Optimal Supervisors for the Enforcement of Static and Behavioral Specifications in Petri Net Models" **Automatica**, **49**, **11**, pp 3432.
- [62] Chen Y. and Li Z. (2012). Structurally Minimal Supervisors, pp 109, In: "**Optimal Supervisor of Automatic Manufacturing Systems**", CRC Press.
- [63] Coffman E. G., Elphick M., and Shoshani A. (1971) "System Deadlocks" **ACM Comput Surv**, **3**, **2**, pp 67.
- [64] Wysk R. A., Yang N. S., and Joshi S. (1994) "Resolution of Deadlocks in Flexible Manufacturing Systems: Avoidance and Recovery Approaches" **Journal of Manufacturing Systems**, **13**, **2**, pp 128.
- [65] Yue H., Xing K., Hu H., Wu W., and Su H. (2016) "Petri net-Based Robust Supervisory Control of Automated Manufacturing Systems" **Control Engineering Practice**, **54**, pp 176.
- [66] Wu N. and Zhou M. (2005) "Modeling and Deadlock Avoidance of Automated Manufacturing Systems with Multiple Automated Guided Vehicles" **IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)**, **35**, **6**, pp 1193.
- [67] Jeng M. D. and Xie X. L. (2005). Deadlock Detection and Prevention of Automated Manufacturing Systems using Petri Nets and Siphons, pp 233, In: "**Deadlock Resolution in Computer-Integrated Systems**", Zhou M, Fanti MP, CRC Press, New York.
- [68] Giua A. and DiCesare F. (1994) "Petri Net Structural Analysis for Supervisory Control" **IEEE Transactions on Robotics and Automation**, **10**, **2**, pp 185.
- [69] Reveliotis S. (2016) "Real-Time Management of Complex Resource Allocation Systems: Necessity, Achievements and Further Challenges" **Annual Reviews in Control**, **41**, pp 147.
- [70] López-Mellado E., Ramírez-Treviño A., Lefebvre D., Ortmeier F., and Reveliotis S. (2015) "Real-Time Management of Complex Resource Allocation Systems: Necessity, Achievements and Further Challenges" **5th IFAC International Workshop on Dependable Control of Discrete Systems**, **48**, **7**, pp 50.
- [71] Giua A. (2013). Supervisory Control of Petri Nets with Language Specifications, pp 235, In: "**Control of Discrete-Event Systems: Automata and Petri Net Perspectives**", Seatzu C, Silva M, van Schuppen HJ, Springer London, London.
- [72] Rezig S., Achour Z., and Rezg N., (2014) "Control Synthesis Based on Reachability Graph with Minimal Cuts: Application to a Flexible Manufacturing System", Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA).

- [73] Huang Y. S. and Pan Y. L. (2011) "An Improved Maximally Permissive Deadlock Prevention Policy Based on the Theory of Regions and Reduction Approach" **IET Control Theory & Applications**, **5**, **9**, pp 1069.
- [74] Chen Y., Li Z., and Zhou M. (2012) "Behaviorally Optimal and Structurally Simple Liveness-Enforcing Supervisors of Flexible Manufacturing Systems" **IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans**, **42**, **3**, pp 615.
- [75] Salimi E., Somnath N., and Sreenivas R. S., (2015) "On Supervisory Policies that Enforce Liveness in Controlled Petri Nets that are Similar" IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM).
- [76] Ghaffari A., Rezg N., and Xiaolan X. (2003) "Design of a Live and Maximally Permissive Petri Net Controller Using the Theory of Regions" **IEEE Transactions on Robotics and Automation**, **19**, **1**, pp 137.
- [77] Li Z. and Zhou M. (2006) "Two-Stage Method for Synthesizing Liveness-Enforcing Supervisors for Flexible Manufacturing Systems Using Petri Nets" **IEEE Transactions on Industrial Informatics**, **2**, **4**, pp 313.
- [78] Holloway L. E. and Krogh B. H. (1990) "Synthesis of Feedback Control Logic for a Class of Controlled Petri Nets" **IEEE Transactions on Automatic Control**, **35**, **5**, pp 514.
- [79] Giua A., DiCesare F., and Silva M. (1992) "Generalized Mutual Exclusion Constraints on Nets with Uncontrollable Transitions", Systems, Man and Cybernetics, IEEE International Conference on, P974, Chicago, USA.
- [80] Yamalidou K., Moody J., Lemmon M., and Antsaklis P. (1996) "Feedback Control of Petri Nets Based on Place Invariants" **Automatica**, **32**, **1**, pp 15.
- [81] Núñez A. and Sánchez A. (2015) "Supervisory Control based on Behavioral Constraints using a Class of Linear Inequalities" **15th IFAC Symposium on Information Control Problems in Manufacturing**, **48**, **3**, pp 2189.
- [82] Peterson J. L. (1981), "**Petri Net Theory and the Modeling of Systems**", Prentice Hall International Inc, London.
- [83] Viswanadham N., Narahari Y., and Johnson T. L. (1990) "Deadlock Prevention and Deadlock Avoidance in Flexible Manufacturing Systems Using Petri Net Models" **IEEE Transactions on Robotics and Automation**, **6**, **6**, pp 713.
- [84] Uzam M. (2004) "The Use of the Petri Net Reduction Approach for an Optimal Deadlock Prevention Policy for Flexible Manufacturing Systems" **The International Journal of Advanced Manufacturing Technology**, **23**, **3**, pp 204.
- [85] Badouel E., Bernardinello L., and Darondeau P. (1995). Polynomial Algorithms for the Synthesis of Bounded Nets, pp 364, In: "**TAPSOFT '95: Theory and Practice of Software Development**", Mosses PD, Nielsen M, Schwartzbach MI, Springer Berlin Heidelberg, Berlin, Heidelberg.
- [86] Uzam M. (2002) "An Optimal Deadlock Prevention Policy for Flexible Manufacturing Systems Using Petri Net Models with Resources and the Theory of

Regions" **The International Journal of Advanced Manufacturing Technology**, **19**, **3**, pp **192**.

[87] Li Z., Zhou M., and Jeng M. (2008) "A Maximally Permissive Deadlock Prevention Policy for FMS Based on Petri Net Siphon Control and the Theory of Regions" **IEEE Transactions on Automation Science and Engineering**, **5**, **1**, pp **182**.

[88] Chen Y., Li Z., and Barkaoui K. (2014) "Maximally Permissive Petri Net Supervisors with a Novel Structure" **12th IFAC International Workshop on Discrete Event Systems (2014)**, **47**, **2**, pp **80**.

[89] Chen Y. and Li Z. (2014) "New Petri Net Structure and Its Application to Optimal Supervisory Control: Interval Inhibitor Arcs" **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, **44**, **10**, pp **1384**.

[90] Tricas F., Garcia-Valles F., Colom J. M., and Ezpeleta J., (2005) "A Petri Net Structure-Based Deadlock Prevention Solution for Sequential Resource Allocation Systems", Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain.

[91] Uzam M., Gelen G., and Saleh T. L. (2016) "Think-Globally-Act-Locally Approach with Weighted Arcs to the Synthesis of a Liveness-Enforcing Supervisor for Generalized Petri Nets Modeling FMSs" **Information Sciences**, **363**, pp **235**.

[92] Hu H. and Li Z. (2009) "Local and Global Deadlock Prevention Policies for Resource Allocation Systems using Partially Generated Reachability Graphs" **Computers & Industrial Engineering**, **57**, **4**, pp **1168**.

[93] Marchetti O. and Munier-Kordon A. (2009) "A Sufficient Condition for the Liveness of Weighted Event Graphs" **European Journal of Operational Research**, **197**, **2**, pp **532**.

[94] Liu G. and Barkaoui K. (2016) "A Survey of Siphons in Petri Nets" **Information Sciences**, **363**, pp **198**.

[95] Chu F. and Xie X.-L. (1997) "Deadlock Analysis of Petri Nets Using Siphons and Mathematical Programming" **IEEE Transactions on Robotics and Automation**, **13**, **6**, pp **793**.

[96] ZhiWu L. and MengChu Z. (2004) "Elementary Siphons of Petri Nets and Their Application to Deadlock Prevention in Flexible Manufacturing Systems" **IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans**, **34**, **1**, pp **38**.

[97] Piroddi L., Cordone R., and Fumagalli I. (2008) "Selective Siphon Control for Deadlock Prevention in Petri Nets" **IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans**, **38**, **6**, pp **1337**.

[98] Piroddi L., Cordone R., and Fumagalli I. (2009) "Combined Siphon and Marking Generation for Deadlock Prevention in Petri Nets" **IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans**, **39**, **3**, pp **650**.

[99] Liu H., Xing K., Wu W., Zhou M., and Zou H. (2015) "Deadlock Prevention for Flexible Manufacturing Systems via Controllable Siphon Basis of Petri Nets" **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, **45**, **3**, pp **519**.

- [100] Dideban A. and Alla H. (2005) "From Forbidden State to Linear Constraints for the Optimal Supervisory Control " **Control Engineering and Applied Informatics**, **7**, **3**, pp **48**.
- [101] Dideban A. and Alla H. (2008) "Reduction of Constraints for Controller Synthesis based on Safe Petri Nets" **Automatica**, **44**, **7**, pp **1697**.
- [102] Dideban A., Zareiee M., and Alla H. (2009) "Controller synthesis with very simplified linear constraints in Petri Net model" **2nd IFAC Workshop on Dependable Control of Discrete Systems**, **42**, **5**, pp **233**.
- [103] Dideban A., Zareiee M., and Alla H. (2013) "Controller Synthesis with Highly Simplified Linear Constraints" **Asian Journal of Control**, **15**, **1**, pp **1**.
- [104] Uzam M. and Jones A. H. (1998) "Discrete Event Control System Design Using Automation Petri Nets and Their Ladder Diagram Implementation" **The International Journal of Advanced Manufacturing Technology**, **14**, **10**, pp **716**.
- [105] Shih Sen P. and Meng Chu Z. (2004) "Ladder Diagram and Petri Net-Based Discrete-Event Control Design Methods" **IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)**, **34**, **4**, pp **523**.
- [106] Uzam M. and Zhou M. (2007) "An Iterative Synthesis Approach to Petri Net-Based Deadlock Prevention Policy for Flexible Manufacturing Systems" **IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans**, **37**, **3**, pp **362**.
- [107] Uzam M. and Zhou M. C. (2006) "An Improved Iterative Synthesis Method for Liveness Enforcing Supervisors of Flexible Manufacturing Systems" **International Journal of Production Research**, **44**, **10**, pp **1987**.
- [108] Zareiee M., Dideban A., Orouji A. A., and Alla H. (2012) "A Simple Petri Net Controller in Discrete Event systems" **14th IFAC Symposium on Information Control Problems in Manufacturing**, **45**, **6**, pp **188**.
- [109] Zareiee M., Dideban A., and Orouji A. A. (2014)"Safety Analysis of Discrete Event Systems Using a Simplified Petri Net Controller" **ISA Transactions**, **53**, **1**, pp **44**.
- [110] Chen Y. and Li Z. (2011) "Design of a Maximally Permissive Liveness-Enforcing Supervisor With a Compressed Supervisory Structure for Flexible Manufacturing Systems" **Automatica**, **47**, **5**, pp **1028**.
- [111] Huang B., Zhou M., and Zhang G. (2015) "Synthesis of Petri net supervisors for FMS via redundant constraint elimination" **Automatica**, **61**, pp **156**.
- [112] Chen Y. and Li Z. (2012) "On Structural Minimality of Optimal Supervisors for Flexible Manufacturing Systems" **Automatica**, **48**, **10**, pp **2647**.
- [113] Cordone R. and Piroddi L. (2013) "Parsimonious Monitor Control of Petri Net Models of Flexible Manufacturing Systems" **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, **43**, **1**, pp **215**.
- [114] Ma Z., Li Z., and Guia A. (2015) "Design of Optimal Petri Net Controllers for Disjunctive Generalize Mutual Exclusion Constraints" **IEEE Transactions on Automatic Control**, **60**, **7**, pp **1774**.

- [115] Wang S., You D., and Wang C. (2016) "Optimal Supervisor Synthesis for Petri Nets with Uncontrollable Transitions: A bottom-Up Algorithm" **Information Sciences**, **363**, pp 261.
- [116] Chen Y., Li Z., and Zhou M. (2012) "Most Permissive Liveness-Enforcing Petri Net Supervisors for Flexible Manufacturing Systems" **International Journal of Production Research**, **50**, **22**, pp 6357.
- [117] Cordone R., Nazeem A., Piroddi L., and Reveliotis S., (2012) "Maximally Permissive Deadlock Avoidance for Sequential Resource Allocation Systems using Disjunctions of Linear Classifiers", 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), P7244-7251.
- [118] Basile F., Cordone R., and Piroddi L. (2015) "A Bbranch and Bound Approach for the Design of Decentralized Supervisors in Petri Net Models" **Automatica**, **52**, pp 322.
- [119] Cordone R., Nazeem A., Piroddy L., and Reveliotis S. (2013) "Designing Optimal Deadlock Avoidance Policies for Sequential Resource Allocation Systems Through Classification Theory: Existence Results and Customized Algorithms" **IEEE Transactions on Automatic Control**, **58**, **11**, pp 2772.
- [120] Reveliotis S. R. and Nazeem A. (2013) "Optimal Linear Separation of the Safe and Unsafe Subspace of Sequential RAS as a Set-Covering Problem: Algorithmic Approach and Geometric Insights" **SIAM Journal of Control and Optimization**, **51**, **2**, pp 1707.
- [121] Nazeem A. and Revelotis S., (2011) "Designing Maximally Permissive Deadlock Avoidance Policies for Sequential Resource Allocation Systems Through Classification Theory", in proceeding of 7th IEEE conference on Automation, Science and Engineering, P405, Trieste, Italy.
- [122] Chen Y., Li Z., Barkaoui K., and Giua A. (2015) "On the Enforcement of a Class of Nonlinear Constraints on Petri Nets" **Automatica**, **55**, pp 116.
- [123] Li J., Zhou M., and Dai X., (2013) "Exploiting Robustness in Petri Net-Based Closed-Loop Systems to Accommodate Control Specification Changes", 2013 IEEE International Conference on Automation Science and Engineering (CASE), P71-76, Madison, WI.
- [124] Li J., Zhou M., Guo T., Gan Y., and Dai X. (2014) "Robust Control Reconfiguration of Resource Allocation Systems with Petri Nets and Integer Programming" **Automatica**, **50**, **3**, pp 915.
- [125] Best E. (1987). Structure Theory of Petri Nets: the Free Choice Hiatus, pp 168, In: "**Petri Nets: Central Models and Their Properties**", Brauer W, Reisig W, Rozenberg G, Springer Berlin Heidelberg, Berlin, Heidelberg.
- [126] Ye J., Li Z., and Giua A. (2015) "Decentralized Supervision of Petri Nets With a Coordinator" **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, **45**, **6**, pp 955.
- [127] Iordache M. V. and Antsaklis P. J. (2006) "Decentralized Supervision of Petri Nets" **IEEE Transactions on Automatic Control**, **51**, **2**, pp 376.

- [128] Menezes A. and Barasso G. C. (2011) "Fusion of Supervisors in Discrete Event Systems using Colored Petri Nets" **Buletinul Institutului Politehnic Din IAȘI**, 7, 4, pp 43.
- [129] Iordache M. V. and Antsaklis P. J., (2003) "Decentralized Control of Petri Nets with Constraint Transformations", American Control Conference, 2003 Proceedings of the 2003, P314.
- [130] Basile F., Giua A., and Seatzu C., (2007) "Supervisory Control of Petri Nets with Decentralized Monitor Places", 2007 American Control Conference, P4957, New York.
- [131] Basile F., Giua A., and Seatzu C., (2008) "Some New Results on Supervisory Control of Petri Nets with Decentralized Monitor Places", 17th IFAC World Congress, P531, Seoul, Korea.
- [132] Basile F., Cordone R., and Piroddi L., (2013) "Compact and Decentralized Supervisors for General Constraint Enforcement in Petri Net Models", 52nd IEEE Conference on Decision and Control, P7279, Florence, Italy.
- [133] Basile F., Cordone R., and Piroddi L. (2014) "Decentralized Monitors Design for Petri Net Models" **12th IFAC International Workshop on Discrete Event Systems (2014)**, 47, 2, pp 73.
- [134] Giua A. and Seatzu C. (2008) "Modeling and Supervisory Control of Railway Networks Using Petri Nets" **IEEE Transactions on Automation Science and Engineering**, 5, 3, pp 431.
- [135] Munoz-Gama J., Carmona J., and Aalst W. M. P. (2014) "Single-Entry Single-Exit Decomposed Conformance Checking" **Information Systems**, 46, pp 102.
- [136] Tan W., Fan Y., and Zhou M. (2009) "A Petri Net-Based Method for Compatibility Analysis and Composition of Web Services in Business Process Execution Language" **IEEE Transactions on Automation Science and Engineering**, 6, 1, pp 94.
- [137] Anisimov N. A. and Kovalenko A. A., (1996) "Asynchronous Composition of Petri Nets via Places", In Proceedings of the Andrei Ershov Second International Memorial Conference on Perspectives of System Informatics, P 214.
- [138] Dingle N. J., Knottenbelt W. J., and Suto T. (2009) "PIPE2: A Tool for the Performance Evaluation of Generalised Stochastic Petri Nets" **SIGMETRICS Perform Eval Rev**, 36, 4, pp 34.
- [139] Dingle N. J., Knottenbelt W. J., and Suto T. (2009) "PIPE2: A Tool for the Performance Evaluation of Generalised Stochastic Petri Nets" **Sigmetrics Performance Evaluation Review**, 36, 4, pp 34.
- [140] Zareiee M. and Dideban A. (2011) "Reducing the Number of Constraints in Non Safe Petri Net" **International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering**, 5, 1, pp 20.

## Abstract:

This monograph deals with the issue of safe allocation of resources in discrete event systems. Most of computer systems around us is a DES or can be abstracted as it. Employing parallel processes and shared resources, these systems are highly complex. Therefore, safe resource allocation of shared resources to parallel processes is a critical task to avoid deadlock and buffer overflow. For this reason, formal models, e.g. Petri nets, have been used for control and monitoring of resource allocation systems. These models can both simulate system and provide verification methods for modeled system.

There are three important criteria to assess supervisory control policies: optimality of behavior, structural efficiency, and computational efficiency. Several researches, e.g. Language-based approaches, reachability graph-based approaches, Siphon-based and similarity-based approaches have been completed to improve some or all of these three criteria. These algorithms face with high time and space complexity in large-scale systems. Reduction of number of controlled Siphons, introducing concepts of over state and vector covering include the work done dealing with this problem. This monograph provides Petri nets-based Supervisory Control in two related approaches. In the first approach, investigate and prove the feasibility of the calculation of modular supervision using synthesis operators of Petri nets. Utilization of modular design in calculation of a maximum supervisory control reduces the computational complexity of supervisor calculations. The furthermore, utilizing place-merging operators reduce the number of uncontrolled siphones that must be handled In addition, we prove liveness preservation of transition-merging operator in certain well-specified conditions. This approach can independently be used to prove the liveness preservation in synthesis operators. The second approach solves Supervisory Control problem by introducing a linear programming methods. The use of specification/ generalization vectors leads to a maximally permissive supervisory control with minimal structure and low computational time. The second approach can be used modularly to further reduce the computational time of supervisor calculation in large-scale systems.

Keywords: Discrete Event Dynamic Systems, Resource Allocation Systems, Supervisory Control, Formal Modeling, Petri Nets.



**Shahrood University of Technology**

**Faculty of Computer and IT Engineering**

PhD Dissertation in Artificial Intelligence

**A Formal Supervisory Control model for Resource Allocation in Complex  
Industrial Systems**

By: Fatemeh Jafarinejad

Supervisor:

Dr Ali A. Pouyan

Advisor:

Dr Morteza Zahedi

January 2017