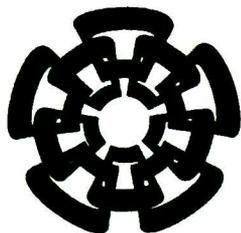


CT-786-SS1
Don. 2014



Centro de Investigación y de Estudios Avanzados
del Instituto Politécnico Nacional
Unidad Guadalajara

Secuencia Detectabilidad Estructural en Redes de Petri Interpretadas

**CINVESTAV
IPN
ADQUISICION
LIBROS**

Tesis que presenta:
Saúl Alonso Nuño Sánchez

para obtener el grado de:
Doctor en Ciencias

en la especialidad de:
Ingeniería Eléctrica

Director de Tesis
Dr. Antonio Ramírez Treviño

CLASIF..	CT 00690
ADQUIS..	CT-786-SSI
FECHA:	22-07-2014
PROCED.	Don. - 2014
	\$

ID: 214229-2001

Secuencia Detectabilidad Estructural en Redes de Petri Interpretadas

**Tesis de Doctorado en Ciencias
Ingeniería Eléctrica**

Por:

Saúl Alonso Nuño Sánchez

Maestro en Ciencias en Ingeniería Eléctrica

Centro de Investigación y de Estudios Avanzados del IPN

Unidad Guadalajara 2007-2009

Becario de CONACYT, expediente no. 35348

Director de Tesis

Dr. Antonio Ramírez Treviño

Resumen.

Esta tesis aborda el problema de secuencia detectabilidad estructural en Sistemas de Eventos Discretos (*SED*) modelados con redes de Petri interpretadas (*RPIs*), esto es, la posibilidad de recuperar la secuencia de disparo de transiciones en una red de Petri interpretada (*RPI*) usando la información de salida cuando el marcado inicial es desconocido. Basándose en la estructura de la *RPI*, se proponen cuatro relaciones dedicadas a capturar la confusión entre transiciones. Dichas relaciones dependen de algunas estructuras de la *RPI* como lo son los *P* – *semiflujos*, *T* – *semiflujos*, lugares de atribución y lugares de selección. Por lo tanto, el enfoque que es presentado explota la información estructural de la *RPI* con la finalidad de determinar la secuencia detectabilidad estructural de una *RPI*. Posteriormente se presenta un observador de secuencias, para cuya construcción se hace uso de *RPIs* coloreadas. El observador es una copia de la *RPI* original con algunos lugares adicionales. Asimismo, se demuestra que a partir del marcado de estos lugares adicionales se puede obtener la secuencia de disparo de transiciones una vez que el observador ha convergido.

También en este trabajo se aborda el problema de la diagnosticabilidad, esto es, con la posibilidad de detectar y localizar eventos de falta a partir de la información de salida en un tiempo finito. Se presenta una definición de diagnosticabilidad basada en la indistinguibilidad entre transiciones. Además, se presentan condiciones suficientes y necesarias para la diagnosticabilidad en *RPI* pertenecientes a la clase de las redes de libre elección usando la información estructural de la *RPI*. La caracterización de diagnosticabilidad proporcionada utiliza las mismas relaciones usadas en la caracterización de la Secuencia Detectabilidad Estructural, en lugar de realizar un análisis del grafo de alcanzabilidad de la *RPI*.

Abstract.

This thesis is concerned with the structural sequence detectability problem in Interpreted Petri Nets (*IPN*), i.e. the possibility of recovering the firing transition sequence in *IPN* using the output information when the initial marking is unknown. Based on the *IPN* structure, four relationships are proposed which are devoted to capture the confusion over the transitions. These relationships depend on *IPN* structures such as *P – semiflows*, *T – semiflows*, attribution places and selection places. Thus, the approach herein presented exploits the *IPN* structural information in order to determine the structural sequence detectability of an *IPN*. Afterwards a sequence observer is presented, where the observer is proposed as a Colored *IPN* that is a copy of the original *IPN* with some extra places. It is shown that from the marking of the added can be obtained the actual firing transition sequence with a mapping..

Also this work deals with the diagnosis problem, i.e. with the possibility of detecting and locating faulty events from the knowledge of output information in within a finite delay. Based on the indistinguishability between transitions the diagnosability is defined. Moreover, sufficient and necessary conditions for diagnosability in *IPN* belonging to the Free Choice class using the *IPN* structural information are presented. The provided diagnosability characterization employs the same relationships used in Structural Sequence Detectability characterization, instead of the reachability graph analysis of the *IPN*.

ÍNDICE GENERAL

Abreviaturas	7
Glosario	8
1 Introducción	11
1.1 Revisión de la literatura	12
1.1.1 Diagnóstico	14
1.2 Objetivos	18
1.3 Organización de la Tesis	18
2 Conceptos Básicos	19
2.1 Introducción	19
2.2 Redes de Petri	20
2.3 Redes de Petri interpretadas	24
2.3.1 Reglas de Evolución y Marcado de una <i>RPI</i>	25
2.4 Propiedades de las <i>RP</i>	28
2.4.1 Clasificación de las <i>RP</i>	31
2.5 Consideraciones importantes	32
2.5.1 Lenguajes y secuencias de disparo	33
2.6 Ejemplo	34
3 Caracterización de la Secuencia Detectabilidad Estructural	39
3.1 Introducción	39
3.2 Definición de la Secuencia Detectabilidad Estructural y de la Secuencia Detectabilidad	40
3.3 Secuencia Detectabilidad Estructural en <i>ME</i>	43
3.4 Secuencia Detectabilidad Estructural en <i>GM</i>	50
3.5 Secuencia Detectabilidad Estructural en <i>LE</i>	56
3.6 Secuencia Detectabilidad en las <i>RPI</i>	68
3.7 Conclusiones	71

4	Observador de la secuencia de transiciones disparada	73
4.1	Introducción	73
4.2	Observador de Vector de Parikh	76
4.3	Observador Coloreado de la Secuencia de Transiciones Disparada .	83
4.4	Conclusiones	88
5	Diagnosticabilidad en las <i>RPI</i> de la clase de las <i>LE</i>	89
5.1	Introducción	89
5.2	Caracterización de la Diagnosticabilidad en <i>RPI</i> en la clase de las <i>LE</i>	91
5.3	Algoritmos .	95
5.4	Ejemplos .	99
5.5	Conclusiones .	102
6	Conclusiones y trabajo futuro	103
6.1	Conclusiones .	103
6.2	Trabajo Futuro	104
7	Apéndice A	113

ÍNDICE DE FIGURAS

2.1	Una <i>RP</i> inicialmente marcada	21
2.2	<i>RP</i> con transiciones fuente y sumidero.	24
2.3	Una <i>RP</i> con un par de transiciones fork-join.	24
2.4	Una <i>RPI</i> .	27
2.5	<i>RP</i> no viva.	28
2.6	<i>RP</i> no acotada.	29
2.7	<i>RP</i> que es 2-acotada.	29
2.8	<i>RP</i> repetitiva.	30
2.9	<i>RP</i> no pura.	31
2.10	Transiciones en una <i>RPI</i> .	33
2.11	<i>RPI</i> donde una secuencia de símbolos de salida tiene más de una secuencia de transiciones asociada.	34
2.12	Sistema de vehículos.	35
2.13	Modelo en <i>RPI</i> para el sistema de vehículos.	36
2.14	Grafo de alcanzabilidad para la <i>RPI</i> que modela el sistema de los vehículos.	38
3.1	Una <i>RPI</i> que no es <i>SSD</i> .	42
3.2	Una <i>ME</i> que es <i>SSD</i> .	48
3.3	Una <i>ME</i> con <i>T – componentes</i> indistinguibles.	48
3.4	Arreglo <i>H</i> inicial.	49
3.5	Arreglo <i>H</i> final. Este permanece sin cambios cuando el Algoritmo 48 es ejecutado.	50
3.6	Una <i>RPI</i> binaria y viva que contiene paralelismo, la cual no es <i>SSD</i> .	51
3.7	Un <i>GM</i> que no es estructuralmente secuencia detectable.	54
3.8	Un <i>GM</i> que es <i>SSD</i> .	55
3.9	Una <i>RPI</i> de libre elección que no es <i>SSD</i> .	57
3.10	Esta figura presenta ejemplos para analizar la independencia de las condiciones que caracterizan a la <i>SDD</i> .	58

4.1	<i>RPI</i> que es <i>SSD</i> ; pero no marcado detectable.	74
4.2	<i>RPI</i> Q que es <i>SSD</i> .	80
4.3	<i>PVO</i> $Q^N(\Phi M_0)$.	81
4.4	<i>PVO</i> $Q^N(\Phi M_1)$.	82
4.5	<i>PVO</i> $Q^N(\Phi M_2)$.	82
4.6	<i>PVO</i> $Q^N(\Phi M_3)$.	83
4.7	Una <i>LE</i> que es <i>SSD</i> y marcado detectable.	85
4.8	La estructura de <i>RPI</i> para una $N_i \in PVO^N(\Phi M_0)$ o para el $CPVO^N$, para la <i>RPI</i> mostrada en la Figura 4.7.	86
5.1	Una <i>LE</i> que no es <i>SD</i> ; pero sí diagnosticable.	92
5.2	Una <i>LE</i> que no es <i>SD</i> ni tampoco diagnosticable.	93
5.3	Una <i>RPI</i> de la clase de las <i>LE</i> que no diagnosticable.	100
5.4	Una <i>RPI</i> diagnosticable.	102
7.1	Una <i>LE</i> con un lugar de distribución p_i que reside dentro de un par de transiciones fork-join.	115
7.2	Una <i>LE</i> con un lugar de distribución p_i que reside por fuera de un par de transiciones fork-join.	116

AGRADECIMIENTOS

Primeramente a Dios quien me ha permitido vivir hasta este momento una vida lo suficientemente larga y con salud para culminar con mis estudios.

A mi esposa Lucía Karina Ramírez Valle por proporcionarme su apoyo incondicional, concederme cariño, amor y comprensión, por alentarme a seguir adelante y superarme, por cuidarme cuando ha sido necesario, por la paciencia que ha tenido a lo largo de mis años de estudio, entre otros.

Al CONACYT por el apoyo económico que me otorgó durante todos mis años de estudio, el cual fue de vital importancia en mi vida y desarrollo.

También CINVESTAV por proporcionarme medios necesarios para el cumplimiento de mi investigación.

Al Dr. Antonio Ramírez Treviño por compartir su sabiduría conmigo, brindarme de su tiempo para encaminarme en el cumplimiento de los objetivos de la tesis desarrollada y por las largas discusiones en la elaboración de los artículos sometidos. Al Dr. Javier Ruiz por el tiempo dedicado a la corrección de artículos y seguimiento que se le dio a los mismos. También al resto de los doctores que me dieron una sólida formación.

A mis sinodales por sus valiosos comentarios para mejorar el contenido y redacción de esta tesis y de la presentación para su defensa, en especial a Renato por su pronta respuesta.

A mis padres, Sahara Sánchez Jiménez y Luis Nuño García, por haberme inculcado excelentes valores, por ser unos buenos modelos a seguir, por velar para que mi salud estuviera en buenas condiciones, por todo el apoyo que me proporcionaron cuando vivía bajo el mismo techo que ellos y por la asistencia que incondicionalmente me siguen brindando, entre muchas otras cosas más.

A mis hermanos Luis, Noé y Benjamín quienes me han ayudado de diversas maneras, tanto en actividades lúdicas como en incitarme al trabajo y a la superación personal y profesional.

A mis amigos Riemann, Marcos, Eduardo, Luque, Ross, Carmen, Rocío, Aguayo, Lizette, Oscar, Edgar, Iván, Samuel, Arnulfo, Jesús, Molus, Marina, Gerardo, Car-

los, Laura, Yolo, Sofía, Hebert, Luisa, Ileana y el resto de las personas que me han dado su apoyo y amistad.

A Griffin e Isis por su compañía en mis horas de trabajo.

A todos ustedes, muchas gracias. . .

ABREVIATURAS

<i>CFTSO</i>	Observador Coloreado de Secuencia de Disparo de Transiciones
<i>FA</i>	Autómata Finito
<i>GM</i>	Grafo Marcado
<i>ILP</i>	Programación Lineal Entera
<i>LE</i>	Redes de Libre Elección
<i>ME</i>	Máquina de Estado
<i>PVO</i>	Observador de Vector de Parikh
<i>RPI</i>	Red de Petri Interpretada
<i>RP</i>	Red de Petri
<i>SD</i>	Secuencia Detectable
<i>SED</i>	Sistemas de Eventos Discretos
<i>SSD</i>	Estructuralmente Secuencia Detectable.

GLOSARIO DE SÍMBOLOS.

(a) Conjuntos

\forall	Para todo (Cuantificador universal)
\exists	Existe (Cuantificador existencial)
\in	Es un elemento de
\notin	No es un elemento de
$>$	Mayor que
\geq	Mayor o igual que
$<$	Menor que
\leq	Menor o igual que
\neq	Diferente de
$\{x \in A P(x)\}$	El conjunto de todos los $x \in A$ tal que satisfacen la condición $P(x)$
$A \cup B$	Unión de los conjuntos A y B
$A \cap B$	Intersección de los conjuntos A y B
$A \times B$	Producto Cartesiano de A y B
$ A $	Cardinalidad del conjunto A
\emptyset	Conjunto Vacío
\mathbf{N}	Conjunto de todos los números naturales
\mathbf{Z}	Conjunto de todos los números enteros
\mathbf{Z}^+	Conjunto de todos los números enteros positivos
$f : X \rightarrow Y$	Es un mapeo de X a Y

(b) Redes de Petri

P	Conjunto de lugares de una RP
p, p_i	Lugar, i -ésimo lugar
T	Conjunto de Transiciones
t, t_j	Transición, j -ésima transición
$W(p_i, t_j)$	Peso del arco que va del lugar p_i a la transición t_j
$W(t_j, p_k)$	Peso del arco que va de la transición t_j al lugar p_k

GLOSARIO DE SÍMBOLOS

$\bullet t$	Conjunto de lugares de entrada a t
t^\bullet	Conjunto de lugares de salida a t
$\bullet p$	Conjunto de transiciones de entrada a t
p^\bullet	Conjunto de transiciones de salida a t
M	Marcado de una RP
M_0	Marcado inicial de una RP
$M(p_i)$	Marcado del i -ésimo lugar
(N, M_0)	RP inicialmente marcada
$R(N, M_0)$	Conjunto de alcanzabilidad de una RP
$\mathcal{L}(N, M_0)$	Lenguaje de disparos de una RP
$M_i \xrightarrow{t_j} M_j$	El marcado M_j es alcanzado al disparar la transición t_j
σ	Secuencia de disparos de transiciones ($\sigma = t_i t_j \dots t_q$)
$\vec{\sigma}$	Vector de Parikh de σ
Φ	Matriz de observación
C	Matriz de incidencia de un RP
A^T	Matriz transpuesta de la matriz A
■	Fin de la demostración
μ_0	Conjunto de marcados iniciales vivos y binarios
$\Lambda(Q, M_0)$	Conjunto de las secuencias de símbolos de salida
$\Lambda^n(Q, M_0)$	Conjunto de las secuencias de símbolos de salida de longitud mayor o igual que n
β^+	Cerradura positiva de Kleene
β^*	Cerradura estrella de Kleene
\approx_I	Relación indistinguibilidad
\approx^+	Relación hacia adelante
\approx^-	Relación hacia atrás
\approx_p	Relación en concurrencia
\approx^o	Relación cíclica
$Q^N(\Phi M_k)$	PVO en el marcado M_k
T_{reg}	Conjunto de transiciones de comportamiento regular

GLOSARIO DE SÍMBOLOS

T_f	Conjunto de transiciones de falta
T_f^i	Conjunto de transiciones de falta de la clase i -ésima
T_{\approx_I}	Conjunto de transiciones indistinguibles
t_j^i	la transición t_j de la clase de falta i -ésima
$\in \sigma$	pertenece a la secuencia σ
$\notin \sigma$	no pertenece a la secuencia σ
\hat{M}_k^i	Marcado de los lugares originales de la red
\tilde{M}_k^i	Marcado de los lugares adicionales de la red
$M_k^i \bar{P}^i$	Marcado restringido a los lugares adicionales
\hat{y}_k^i	Salida de los lugares originales de la red
\tilde{y}_k^i	Salida de los lugares adicionales de la red

1. INTRODUCCIÓN

Los Sistemas de Eventos Discretos (*SED*) son sistemas dinámicos que se caracterizan por tener un espacio de estados discreto, es decir, que la cantidad de estados puede ser finita o infinita, pero numerable. El funcionamiento de este tipo de sistemas está caracterizado por una sucesión finita o infinita de estados, delimitados por eventos. La ocurrencia de dichos eventos se presenta de manera asíncrona.

Una amplia variedad de sistemas que han sido creados por el hombre pueden ser modelados como *SED*, tales como: los sistemas logísticos, los sistemas de manufactura flexible, los sistemas de tráfico, sistemas computacionales, sistemas de redes, sistemas de servicios, sistemas de agentes, sistemas robóticos, cadenas de suministro, protocolos de comunicación, entre muchos otros.

Algunas de las áreas de investigación más importantes dentro de los *SED* son el estudio de las propiedades de controlabilidad, observabilidad, diagnosticabilidad e identificación de modelos. Dichas propiedades son de vital importancia para tener sistemas más confiables. Sin embargo, muchos de los trabajos que se han realizado en dichas áreas toman como hipótesis principal la evento detectabilidad, es decir, la posibilidad de distinguir la ocurrencia de un evento, solamente en un paso, usando tan sólo la información de entrada y salida que proporciona el sistema. Este trabajo se enfoca al estudio de una propiedad más general nombrada secuencia detectabilidad estructural, es decir, la posibilidad distinguir la ocurrencia de secuencias de eventos en base a la información de salida que proporciona el sistema, distinción que se consigue después de una ocurrencia finita de pasos.

Existen diversos formalismos que permiten describir o modelar el comportamiento de los *SED* y estudiar las propiedades anteriormente señaladas, tales como autómatas [11] [20], lenguajes formales [36] [20], álgebras de proceso [21], redes de Petri [31] [33] [13], entre otros. Cada formalismo tiene sus propias ventajas y desventajas con respecto a la representación, compacidad, claridad y facilidad de análisis.

En el presente documento los *SED* son modelados y analizados a través de las redes de Petri (*RPs*) debido a que permiten capturar de manera sencilla las prin-

cipales características de los *SED* como lo son la concurrencia, la sincronización, las exclusiones mutuas, las relaciones causales, entre muchas otras.

1.1. Revisión de la literatura

En la literatura hay pocos trabajos que traten exclusivamente sobre el tema de la detección de secuencia de eventos. Por ejemplo, en [32] la reconstrucción de las secuencias de eventos ha sido estudiada en los autómatas de estados finitos (*FA*) bajo el nombre de invertibilidad. En este artículo, los autores caracterizan la propiedad de invertibilidad para los casos en el que el estado inicial es conocido o desconocido. Con el fin de verificar que un *FA* posee la propiedad de invertibilidad cuando el estado inicial es desconocido, es necesaria la construcción de un observador y utilizar algunos algoritmos de complejidad polinomial con respecto al número de estados del *FA*, para verificar otras propiedades del *FA* como lo son la estabilidad y la no ambigüedad. El concepto de invertibilidad no es equivalente al concepto propuesto de secuencia detectabilidad estructural, porque en los *FA* invertibles se permite que después de distinguir la secuencia de eventos que se ha dado hasta una longitud k de palabra, puedan existir incertidumbres en secuencias de mayor longitud (es decir, que en secuencias posteriores no se pueda conocer de forma única la secuencia de eventos), aunque se debe poder determinar posteriormente la secuencia y posiblemente volverá a tener incertidumbre.

También en el campo de los autómatas en [53], [51], [52] se presenta el concepto de detectabilidad, concepto que en primer instancia suena similar al abordado en esta tesis. En estos trabajos se definen distintos tipos de detectabilidad basados en si es posible determinar los estados para todas las secuencias de eventos del modelo, para unas cuantas o para secuencias repetitivas, es decir, se apega más al concepto de marcado de detectabilidad en redes de Petri [44].

Otro concepto relacionado a la secuencia de detectabilidad estructural es el concepto de la distinguibilidad de transiciones usado en [45] para conseguir la propiedad de observabilidad con una asignación óptima de sensores en sistemas modelados con redes de Petri. No obstante, dicho trabajo se limita a modelos donde el marcado inicial es conocido y no se permite que dos lugares arrojen la misma información a la salida, por lo cual los algoritmos propuestos, para verificar la detectabilidad, únicamente requieren de una revisión de columnas por pares de la matriz de observación.

En [4] se presentan condiciones suficientes para determinar la secuencia de detectabilidad en una clase limitada de *RPs*. No obstante, las clases en donde el

CAPÍTULO 1. INTRODUCCIÓN

paralelismo está involucrado no están caracterizadas.

La literatura más relevante en torno a la detección de secuencias se resume en la siguiente tabla.

	Esta tesis	[44]	[4] ²	[45] ³
Marcado inicial conocido	✓	✓	✓	✓
Marcado inicial desconocido	✓	✓	✓	×
Símbolos de salida Repetidos	✓	✓	×	×
Polinomial con respecto al tiempo	✓ ⁴	✓	✓	✓
Caracteriza secuencia detectabilidad en Máquinas de Estado	✓	✓	✓	✓
Caracteriza secuencia detectabilidad en Grafos Marcados	✓	×	✓	✓
Caracteriza secuencia detectabilidad en redes de Libre Elección	✓	×	×	✓
Caracteriza secuencia detectabilidad otros tipos de redes	× ¹	×	×	✓
Presenta condiciones suficientes y necesarias	✓	✓	×	✓

Tabla 1.1: Cuadro comparativo de los trabajos relacionados con la secuencia de detectabilidad.

1 : En esta tesis no se abordó la secuencia detectabilidad en otros tipos de redes. Sin embargo, durante el trabajo de investigación se obtuvieron resultados para caracterizar otros tipos de redes, sólo falta formalizar y completar algunas demostraciones.

2 : Este trabajo también abarca el caso en el que las *RPIs* no son binarias.

3 : Los resultados de este trabajo no aplican para redes binarias de marcado desconocido. En las *RP* no binarias si existen transiciones indistinguibles que se puedan habilitar simultáneamente entonces no son secuencia detectable, en cambio en las redes binarias existen más estructuras involucradas en tal propiedad.

4 : Se tienen algoritmos de tiempo polinomial para la mayoría de las condiciones.

Aunque no es el tema principal, en lo que respecta a la literatura dedicada al estudio de la observabilidad, controlabilidad, diagnosticabilidad e identificación, el análisis de las secuencias de eventos juega un rol importante para la caracterización de dichas propiedades. A continuación se mencionan sólo algunos trabajos pertenecientes a dichas áreas donde se utiliza la secuencia detectabilidad o

propiedades parecidas.

1.1.1. Diagnóstico

La diagnosticabilidad consiste en la posibilidad de poder detectar en línea los posibles comportamientos de falta en un número finito de eventos después de la ocurrencia del evento de falta, entendiéndose como falta cualquier cambio inesperado del funcionamiento de un sistema que pueda llevar a un error. La entidad encargada de llevar esta labor de detección y localización de la falta recibe el nombre de diagnosticador.

Los trabajos que pueden encontrarse en el área de diagnóstico de faltas [48], [14], [23], [17], [10], [9], [25] basan sus resultados sobre el hecho que las secuencias que contienen una falta f_i no deben confundirse con algún otra secuencia, es decir, no pueden existir secuencias que produzcan la misma información de salida donde una contenga falta y la otra no. De ahí que sea de vital importancia poder determinar bajo qué condiciones las secuencias de eventos pueden confundirse entre sí.

Dentro del marco de los FA , en uno de los primeros trabajos publicados en el área por Sampath y colaboradores [48] y [47] se proporcionan condiciones necesarias y suficientes para la diagnosticabilidad de sistemas vivos. Estos autores determinan que un sistema es diagnosticable si y sólo si el modelo del FA no tiene ciclos F_i – *indeterminados* (ciclos que arrojen la misma información a la salida y no se pueden distinguir entre sí). Por su parte Lunze y Schroder en [27] extienden el concepto de diagnosticabilidad a autómatas estocásticos, y ellos demuestran que, en general, el diagnosticador para un autómata finito estocástico no puede ser representado por otro autómata finito estocástico. En [22] y [56], se propone una prueba para la diagnosticabilidad de máquinas de estado finito y vivas mediante el uso de algoritmos de tiempo polinomial, donde a partir del modelo del FA original se obtiene un autómata no determinista usando los eventos observables [22], o bien por cada tipo de falta [56], donde el objetivo del algoritmo es determinar la existencia de ciclos indistinguibles (similares a los ciclos F_i – *indeterminados*). Una extensión del trabajo anterior es presentada en [34], donde la presencia de lenguajes que no sean vivos y ciclos no observables son permitidos, para este caso también se construye un diagnosticador no determinista para cada tipo de falta, el cual determinará si la ocurrencia de una falta puede ser detectada después de un número finito de pasos, donde no deben existir ciclos indistinguibles. Sin embargo, el resultado de usar FA o lenguajes regulares con-

CAPÍTULO 1. INTRODUCCIÓN

lleva a que los modelos únicamente sean aptos para sistemas de tamaño pequeño y sin paralelismo. Debido a la limitación anteriormente señalada muchos autores han preferido el uso de las redes de Petri, pues permiten una mayor compacidad del modelo y cuentan con poder de expresión mayor. Sin embargo, aunque se tienen algoritmos de tiempo polinomial en FA , éstos al ser trasladados a las redes de Petri son equivalentes a algoritmos de tiempo no polinomial debido a que es necesaria la expansión total del grafo de alcanzabilidad.

Algunas investigaciones que optaron por el uso de las redes de Petri analizan el grafo de alcanzabilidad para buscar las secuencias de eventos que sean indistinguibles entre sí, y a través de ellas decidir si el sistema es diagnosticable. Por ejemplo, en [17], [10] y [9], se proponen algoritmos para probar la diagnosticabilidad de una red de Petri (RP) basados en un grafo de alcanzabilidad modificado y haciendo uso de un diagnosticador de alcanzabilidad. Los autores determinan que una RP es diagnosticable si y sólo si no hay ciclos indeterminados en el grafo de alcanzabilidad modificado (los cuales también son similares a los ciclos F_i – *indeterminados* mencionados con anterioridad). En [6] los resultados previos son extendidos a RP s no acotadas haciendo uso del grafo de cobertura. Sin embargo, el tamaño de estos grafos depende del número de faltas modeladas y del marcado inicial de la red de Petri Interpretada (RPI). Por lo tanto, estos grafos pueden crecer exponencialmente.

En [23] se propone un algoritmo para probar la diagnosticabilidad a través de explicaciones mínimas de las secuencias de eventos observables. Este trabajo se basa en el estudio de una versión reducida del grafo de alcanzabilidad construido mediante el uso de las explicaciones mínimas, en el cual se buscan ciclos que estén confundidos entre sí (nuevamente ciclos F_i – *indeterminados*). No obstante, las explicaciones mínimas para una secuencia de disparo de transiciones pueden no ser únicas, por lo que se requiere de una exploración en un grafo no determinista.

Otros autores, en lugar de desarrollar el grafo de alcanzabilidad para analizar la propiedad de diagnosticabilidad de una RP , han optado por aprovechar la información que proporciona la estructura de la RP . Por ejemplo, en [40], se proponen condiciones suficientes para la diagnosticabilidad de una RP que sea evento detectable.

Dotoli y colaboradores en [14] usan la programación lineal entera (ILP) para determinar si una secuencia de disparo de transiciones contiene transiciones de falta. Desafortunadamente, para redes de Petri de gran tamaño, a la ILP le toma un periodo de tiempo considerable para resolver el problema. Lefebvre y colaboradores en [25] hacen uso de la variación en el marcado para calcular la

CAPÍTULO 1. INTRODUCCIÓN

secuencia de disparo de transiciones y aislar las faltas. Este enfoque se basa en la hipótesis de que se tiene pleno conocimiento de la variación en el marcado de todos los lugares, lo cual no siempre es posible.

Cabe señalar que muchos de los resultados que permiten encontrar las secuencias indistinguibles entre sí y con dichas secuencias emitir un juicio acerca de la diagnosticabilidad de una *RP* son de tiempo no polinomial (pues se requiere la expansión parcial o total del grafo de alcanzabilidad), o bien únicamente son suficiencias que se enfocan a clases muy limitadas de sistemas modelados con *RPs*. Por tal motivo el tener una caracterización de la secuencia detectabilidad y la secuencia detectabilidad estructural es benéfico para la diagnosticabilidad.

Observabilidad

Un sistema es observable si es posible inferir el estado actual del sistema y cuál fue su estado inicial a partir de la información de salida que proporciona el sistema, donde la entidad para llevar a cabo esta tarea recibe el nombre de observador.

Gran parte de la literatura en el área de la observabilidad hacen uso del concepto de evento detectabilidad propuesto en [43] para el diseño de sus observadores o para la caracterización de la propiedad de observabilidad [38], [41].

En el trabajo de Aguirre y colaboradores [2] a través de algoritmos de tiempo polinomial se propone una asignación óptima de sensores en el sistema para garantizar que el modelo de dicho sistema tiene la propiedad de evento detectabilidad y es por ende observable, para después poder determinar sus estados con los observadores asintóticos propuestos en [1].

Li y Hadjicostis en [26] abordan el problema de reconstruir la secuencia de disparo de transiciones de *RPIs* no binarias a través de un observador no centralizado. El algoritmo que se propone, basado en búsquedas en profundidad de un grafo [46], está enfocado a las *RPIs* donde la variación en el marcado de todos los lugares sea conocido, con lo que les es posible la detección de los eventos.

Aunque los resultados anteriores son bastante interesantes, éstos usan la propiedad de evento detectabilidad para calcular la secuencia de disparo de transiciones. Sin embargo, esta propiedad es muy restrictiva, por lo que muchas clases de redes de Petri no pueden ser analizadas, o bien muchos casos reales no cumplen con dichas características. Por tal motivo la secuencia detectabilidad permite extender los resultados ya existentes en el área de la observabilidad.

Controlabilidad

A grandes rasgos la controlabilidad es la propiedad que permite manipular el comportamiento de un sistema llevándolo de su estado inicial a un estado final requerido. En este caso, el controlador es la entidad encargada de llevar a cabo las acciones necesarias para alcanzar el estado requerido. Un lector interesado puede consultar [36], [55], [35], para profundizar más en el tema y analizar diversos tipos de controladores.

En [37], [50] se aborda el problema de control usando la técnica de regulación para los *SED*, la cual trata de la construcción de un controlador que permita que la salida del sistema sea igual a la salida de una especificación modelada con máquinas de estado, dicho de otra forma, se busca encontrar una secuencia de eventos cuya información a la salida iguale a la salida que establezca la especificación. Posteriormente, en [49] se propone un esquema de control por regulación en el caso de existir múltiples especificaciones. Sin embargo, los enfoques de control por regulación llegan a tener problemas cuando existen múltiples secuencias de eventos que puedan producir la misma información de salida que la especificación, por lo que se tendría que tomar una decisión de cuál de ellas (si son controlables) debe llevar a cabo el controlador, o bien a través de las ecuaciones que han sido propuestas para este tipo de técnica se podrían obtener muchas soluciones espurias. Por tal, motivo el hecho que se garantice que ninguna secuencia se puede confundir con otra, puede ayudar a mejorar los resultados existentes en el área de control por regulación y evitar este tipo de acontecimientos no deseados.

Identificación

La identificación es el proceso de obtener el modelo matemático de un sistema a partir de la observación del comportamiento del sistema, el cual es descrito a través de la información de entrada y salida que éste proporciona durante su evolución.

Para poder llevar a cabo la identificación son necesarias diversas hipótesis de trabajo. Una de ellas es que todos los estados del sistema sean medibles y proporcionen información de salida diferente, lo cual hace trivial a la identificación y en la realidad casi ningún sistema cumple con dichas especificaciones. Otra hipótesis más realista, como la utilizada en [30], [28], [29] es que no todos los estados del sistema pueden ser detectados, sin embargo el modelo del sistema a identificar deberá tener la propiedad evento detectabilidad, es decir, la ocurrencia de cada evento tiene que poder ser detectada de manera única haciendo uso de la

información de la salida del sistema.

Es claro que con una caracterización de las secuencias de eventos se podría ampliar la clase de sistemas que pueden ser identificados.

1.2. Objetivos

Los objetivos principales de esta tesis son los siguientes:

- Caracterizar la secuencia detectabilidad estructural y la secuencia detectabilidad en *RPIs* pertenecientes a la clase de las redes de libre elección.
- Presentar algoritmos (de preferencia de tiempo polinomial) que permitan probar si una *RPI* de libre elección presenta la propiedad de secuencia detectabilidad estructural, evitando la expansión del grafo de alcanzabilidad.
- Para las redes que presenten la propiedad de secuencia detectabilidad estructural, proponer un observador que permita recuperar la secuencia de disparo de transiciones.
- Utilizar la teoría de secuencia detectabilidad para caracterizar la diagnosticabilidad en *RPIs* de libre elección.
- Presentar algoritmos de tiempo polinomial que permitan probar si una *RPI* de libre elección es diagnosticable.

1.3. Organización de la Tesis

Esta tesis se encuentra organizada de la siguiente manera. El Capítulo 2 presenta los conceptos básicos sobre *RP* y *RPI*, requeridos a lo largo de este trabajo. En el Capítulo 3 se define formalmente la secuencia detectabilidad y la secuencia detectabilidad estructural. También en el Capítulo 3 se proporciona la caracterización de la secuencia detectabilidad estructural y secuencia detectabilidad en las *RPIs* de libre elección, así como algoritmos para verificar dichas propiedades. El Capítulo 4 muestra el diseño del Observador de Vector de Parikh (*PVO*) y un Observador Coloreado de Secuencia de Disparo de Transiciones (*CFTSO*). Posteriormente, en el Capítulo 5 se aporta la caracterización de la diagnosticabilidad de las *RPIs* pertenecientes a la clase de las redes de libre elección, haciendo uso de las mismas relaciones empleadas en la caracterización de la secuencia detectabilidad. Finalmente, las conclusiones de esta tesis son presentadas en el Capítulo 6.

2. CONCEPTOS BÁSICOS

Resumen: En este capítulo se presentan brevemente los conceptos básicos de RP y RPI estableciendo la nomenclatura utilizada a través del presente trabajo. La definición de RPI es antecedida por una presentación de las RPs y algunas estructuras de la RP como los P – semiflujos, T – semiflujos, lugares de selección, entre otros. Asimismo en este capítulo se presentan algunas propiedades de las RPs y su clasificación. Posteriormente se muestra un ejemplo ilustrativo donde se modela un sistema de vehículos con una RPI.

2.1. Introducción

Los formalismos utilizados con mayor frecuencia para modelar los *SED* son los autómatas y las *RPs*. Los autómatas son de gran utilidad para obtener representaciones gráficas sencillas, cuando el sistema no presenta paralelismo y exhibe pocos estados. Las *RPs* son una herramienta apropiada para modelar los *SED* cuando es necesario capturar paralelismo, concurrencia, sincronización, exclusiones mutuas, entre otras relaciones de interés. Además, permiten obtener una representación que es fácil de entender, es posible capturar lenguajes que los autómatas finitos no aceptan, por lo que las *RPs* resultan ser una herramienta adecuada para modelar y analizar los *SED*.

Este capítulo se organiza de la siguiente forma: En la Sección 2.2 se presenta la definición formal de una *RP* y la definición de algunas estructuras de interés como lo son los P – *semiflujos* y los T – *semiflujos*. La Sección 2.3 proporciona la definición de las *RPIs*, así como sus reglas de evolución y marcado. Posteriormente, en la Sección 2.4 se exponen algunas propiedades de las *RPs* que se usarán a través de esta tesis, como también una clasificación de las mismas. La sección 2.5 establece las suposiciones básicas que delimitan el tipo de redes que se pueden analizar con los resultados propuestos. Finalmente, en la Sección 2.6 se muestra un ejemplo para analizar las propiedades y estructuras abordadas a lo largo de este capítulo.

2.2. Redes de Petri

Las *RP*s son una herramienta formal para el modelado y análisis de los *SED*, ya que permiten analizar la dinámica de los *SED* por medio de la evolución de marcas dentro de la red. El modelado de los *SED* usando *RP*s no está en el alcance de esta tesis, sin embargo existen diversos trabajos que pueden ser consultados en lo que respecta a técnicas de modelado [42], [13], [54].

En esta sección se presentan los conceptos básicos de *RP* y *RPI*. Un lector interesado en profundizar en el tema de las *RP*s y sus propiedades puede consultar [31], [12].

Definición 1. Una red de Petri (*RP*) es un dígrafo bipartita definido por la 4-tupla $N = (P, T, F, W)$, donde:

- $P = \{p_1, p_2, \dots, p_n\}$ es un conjunto finito de lugares,
- $T = \{t_1, t_2, \dots, t_m\}$ es un conjunto finito de transiciones, donde $P \cup T \neq \emptyset$ y $P \cap T = \emptyset$,
- $F \subseteq (P \times T) \cup (T \times P)$ es el conjunto de arcos (relación de flujo de la red).
- $W : F \rightarrow \mathbb{N}$ es una función de peso, donde \mathbb{N} representa el conjunto de los números naturales.

Un marcado es una función $M : P \rightarrow \mathbb{Z}^+$ que asigna a cada lugar p_i de la *RP* un número entero positivo, llamado el número de marcas que reside dentro de cada lugar p_i . Cabe señalar que el marcado M puede ser representado a través de un vector de dimensión n , donde el i -ésimo elemento representa el número de marcas en el lugar p_i de la red. M_0 representa la distribución inicial de marcas en la red.

A lo largo de esta tesis una *RP* con un marcado inicial dado será denotada por (N, M_0) .

Gráficamente, los lugares son representados por círculos, las transiciones a través de cajas, los arcos mediante flechas y las marcas con puntos negros o números enteros dentro de cada lugar.

Ejemplo 2. En la Figura 2.1, una *RP* (N, M_0) con tres lugares y cuatro transiciones es presentada. El conjunto de los lugares es $P = \{p_1, p_2, p_3\}$ y el conjunto de transiciones es $T = \{t_1, t_2, t_3, t_4\}$, su marcado inicial es $M_0 = [1 \ 0 \ 0]^T$ y el peso de cada arco es de uno.

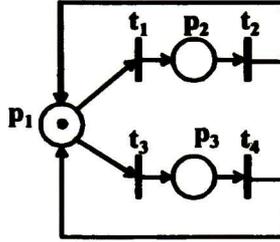


Figura 2.1: Una RP inicialmente marcada

La matriz de incidencia C de N , cuya dimensión es de $n \times m$, está definida por $C(i, j) = W(t_j, p_i) - W(p_i, t_j)$. Si $W(p_i, t_j)$ o $W(t_j, p_i)$ no está definido para un lugar p_i en específico y una transición t_j , entonces se asigna el valor de cero.

Ejemplo 3. Considere nuevamente la RP (N, M_0) mostrada en la Figura 2.1. Su matriz de incidencia es

$$C = \begin{bmatrix} -1 & 1 & -1 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

La notación $\bullet t_j = \{p_x | (p_x, t_j) \in F\}$ y $t_j^\circ = \{p_x | (t_j, p_x) \in F\}$ representan el conjunto de los lugares de entrada y el conjunto de los lugares de salida, respectivamente, de la transición t_j ; similarmente $\bullet p_i = \{t_x | (t_x, p_i) \in F\}$ y $p_i^\circ = \{t_x | (p_i, t_x) \in F\}$ denotan el conjunto de las transiciones de entrada y el conjunto de las transiciones de salida, respectivamente, del lugar p_i .

Los lugares de entrada de un conjunto $H = \{t_a, t_b, \dots, t_j\}$ de transiciones se define como $\bullet H = \bullet t_a \cup \bullet t_b \cup \dots \cup \bullet t_j$. Los lugares de salida de un conjunto H de transiciones se define como $H^\circ = t_a^\circ \cup t_b^\circ \cup \dots \cup t_j^\circ$. Las transiciones de entrada de un conjunto $G = \{p_a, p_b, \dots, p_i\}$ de lugares se definen como $\bullet G = \bullet p_a \cup \bullet p_b \cup \dots \cup \bullet p_i$. Las transiciones de salida de un conjunto G de lugares se define como $G^\circ = p_a^\circ \cup p_b^\circ \cup \dots \cup p_i^\circ$.

Definición 4. Sea N una RP. Los vectores X_i (Y_i) tal que $CX_i = 0$, $X_i \geq 0$ ($Y_i^T C = 0$, $Y_i \geq 0$) son llamados T -semi flujos (P -semi flujos), donde $X_i \geq 0$ ($Y_i \geq 0$) si cada elemento del vector X_i (Y_i) es mayor o igual a cero; pero no

CAPÍTULO 2. CONCEPTOS BÁSICOS

todos son igual a cero. El soporte de un T – semiflujo X_i (P – semiflujo Y_i), denotado por $\langle X_i \rangle$ ($\langle Y_i \rangle$), es el conjunto de transiciones $T_i = \{t_j | X_i(j) > 0\}$ (el conjunto de lugares $P_i = \{p_j | Y_i(j) > 0\}$).

Definición 5. Un P – semiflujo (T – semiflujo) se dice ser mínimo si el máximo común divisor de sus entradas es 1 y si no es una combinación lineal de otros P – semiflujos (T – semiflujos).

Definición 6. Una 4-dupla (P', T', F', W') es una subred de N si $P' \subseteq P$, $T' \subseteq T$, $F' = F \cap ((P' \times T') \cup (T' \times P'))$, W' es la función de peso restringida a los lugares y transiciones de la subred.

Definición 7. Sea N una RP . La subred $\mathcal{T}_i = \{(P_i, T_i, F_i, W_i), M_{0i}\}$ de N generada por el soporte del T – semiflujo X_i es llamada T – componente si $P_i = (\bullet \langle X_i \rangle \cup \langle X_i \bullet)$, $T_i = \langle X_i \rangle$ y F_i, W_i, M_{0i} son las relaciones de flujo, la función de peso y el marcado inicial, restringidos a P_i y T_i respectivamente.

Definición 8. Sea N una RP . La subred $\mathcal{P}_i = \{(P_i, T_i, F_i, W_i), M_{0i}\}$ de N generada por el soporte del P – semiflujo Y_i^T es llamada P – componente si $T_i = (\bullet \langle Y_i \rangle \cup \langle Y_i \bullet)$, $P_i = \langle Y_i \rangle$ y F_i, W_i, M_{0i} son las relaciones de flujo, la función de peso y el marcado inicial, restringidos a P_i y T_i respectivamente.

Ejemplo 9. Considere nuevamente la $RP (N, M_0)$ mostrada en la Figura 2.1. El conjunto de entradas y salidas para todo lugar y transición es:

Entradas	Salidas
$\bullet p_1 = \{t_2, t_4\}$	$p_1^\bullet = \{t_1, t_3\}$
$\bullet p_2 = \{t_1\}$	$p_2^\bullet = \{t_2\}$
$\bullet p_3 = \{t_3\}$	$p_3^\bullet = \{t_4\}$
$\bullet t_1 = \{p_1\}$	$t_1^\bullet = \{p_2\}$
$\bullet t_2 = \{p_2\}$	$t_2^\bullet = \{p_1\}$
$\bullet t_3 = \{p_1\}$	$t_3^\bullet = \{p_3\}$
$\bullet t_4 = \{p_3\}$	$t_4^\bullet = \{p_1\}$

Si $H = \{t_1, t_3\}$ y $G = \{p_1, p_3\}$, entonces su conjunto de entradas y salidas es:

Entradas	Salidas
$\bullet H = \{p_1\}$	$H^\bullet = \{p_2, p_3\}$
$\bullet G = \{t_2, t_3, t_4\}$	$G^\bullet = \{t_1, t_3, t_4\}$

Los P – semiflujos y T – semiflujos de la RPI son:

CAPÍTULO 2. CONCEPTOS BÁSICOS

<i>P</i> – semi flujos	<i>T</i> – semi flujos
$Y_1^T = [1 \ 1 \ 1]$	$X_1 = [1 \ 1 \ 0 \ 0]^T$
	$X_1 = [0 \ 0 \ 1 \ 1]^T$

Algunos lugares y transiciones de las *RP*s son clasificados de acuerdo a sus conjuntos de entrada y de salida.

Definición 10. *Tipos de lugares.*

1. **Lugares de atribución.** *Un lugar es llamado de atribución si tiene más de una transición en su conjunto de entrada.*
2. **Lugares de selección.** *Un lugar es llamado de distribución si tiene más de una transición en su conjunto de salida.*

Por ejemplo, para la *RP* mostrada en la Figura 2.1 el lugar p_1 es de atribución y de distribución.

Definición 11. *Tipos de transiciones.*

1. **Transición de tipo Fork.** *Una transición es llamada de tipo Fork si tiene más de un lugar en su conjunto de salida.*
2. **Transición de tipo Join.** *Una transición es llamada de tipo Join si tiene más de un lugar en su conjunto de entrada.*
3. **Transición fuente.** *Una transición es llamada fuente si no tiene ningún lugar de entrada. Este tipo de transiciones siempre están habilitadas.*
4. **Transición sumidero.** *Una transición es llamada sumidero si no tiene ningún lugar de salida. Este tipo de transiciones consume marcas; pero no las crea.*

Por ejemplo, para la *RP* mostrada en la Figura 2.3 la transición t_1 es de tipo Fork y las transiciones t_4 y t_5 son de tipo Join. En el caso *RP* presentada en la Figura 2.2 la transición t_1 es fuente y la transición t_3 es sumidero.

Definición 12. *Sea A el conjunto de T – componentes de una *RP*. El conjunto A es una T – cobertura si cada transición de la red pertenece a un T – componente de A . Una red está cubierta por T – componentes si tiene una T – cobertura.*

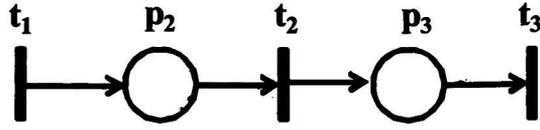


Figura 2.2: RP con transiciones fuente y sumidero.

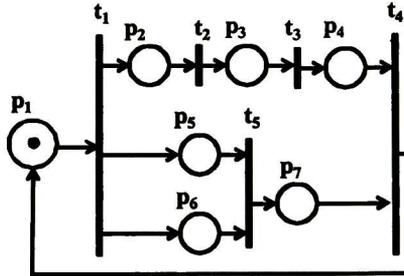


Figura 2.3: Una RP con un par de transiciones fork-join.

Definición 13. Sea TCb cualquier T – cobertura de una RP (ver [12]), las transiciones t_i, t_j (t_i y t_j pueden ser la misma) forman un par de transiciones fork-join si $|t_i^\bullet| > 1$, $|\bullet t_j| > 1$ y no existe un P – semiflujo Y tal que $\exists p_k \in t_i^\bullet, Y(p_k) > 0$ y $\forall p_q \in \bullet t_j, Y(p_q) = 0$.

Por ejemplo, considere la RP mostrada en la Figura 2.3, las transiciones t_1 y t_5 no forma un par de transiciones fork-join debido a que existe un P – semiflujo $Y = [1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0]$, donde $Y(p_2) = 1$ ($p_2 \in t_1^\bullet$) y $Y(p_5) = Y(p_6) = 0$ ($p_5, p_6 \in \bullet t_5$). En cambio, las transiciones t_1, t_4 si forman un par de transiciones fork-join.

2.3. Redes de Petri Interpretadas

Las RP s tienen sus limitaciones ya que no capturan los comandos de entrada por parte algún usuario al sistema ni muestran el valor que arrojan los sensores del sistema, por ello para introducir esta información en el modelo se han creado

CAPÍTULO 2. CONCEPTOS BÁSICOS

diversas extensiones. Entre dichas extensiones se encuentran las redes de Petri interpretadas (*RPIs*) las cuales permiten capturar las salidas de los sensores con los que cuenta el sistema. Los sensores son utilizados para medir un estado (por ejemplo, un sensor de nivel para saber que un tanque de agua alcanzó el estado lleno) o para medir un evento (por ejemplo, un sensor de contacto utilizado ampliamente en la robótica para conocer que se ha realizado un contacto con un objeto). Para representar la salida emitida por un sensor se asigna una etiqueta (también llamada símbolo de salida) ya sea a los lugares de la *RP* o a sus transiciones (símbolos de entrada), de acuerdo a si se mide un estado o el inicio de un evento, respectivamente. A continuación se definen formalmente a las *RPIs*.

Definición 14. Una red de Petri interpretada (*RPI*) es una dupla $Q = (N, \Phi)$ donde:

- N es una estructura de *RP*.
- Existe una matriz Φ de observación de dimensión $q \times n$ de números enteros, tal que $y_k = \Phi M_k$ es un mapeo del marcado M_k a un vector de observación q - dimensional.

2.3.1. Reglas de Evolución y Marcado de una *RPI*

El marcado de una *RPI* evoluciona de acuerdo a las siguientes reglas:

Una transición $t_j \in T$ de una *RPI* está habilitada en el marcado M_k si para cada lugar de entrada de t_j (es decir, cada lugar p_i tal que $(p_i, t_j) \in F$) está marcado con al menos $W(p_i, t_j)$ marcas; es denotado por $M_k \xrightarrow{t_j}$. El disparo de una transición habilitada remueve $W(p_i, t_j)$ marcas de cada lugar de entrada p_i de t_j y agrega $W(t_j, p_k)$ marcas a cada lugar de salida p_k de t_j . Cuando una transición habilitada t_j es disparada en el marcado M_k entonces un nuevo marcado M_{k+1} es alcanzado. Este hecho es representado como $M_k \xrightarrow{t_j} M_{k+1}$. El nuevo marcado M_{k+1} puede ser calculado usando la siguiente ecuación fundamental

$$\begin{aligned} M_{k+1} &= M_k + C \vec{t}_j \\ y_k &= \Phi M_k \end{aligned} \tag{2.1}$$

donde C se define de la misma manera que en una *RP* y \vec{t}_j es un vector, donde $\vec{t}_j(i) = 1$ si y sólo si $i = j$ y $\vec{t}_j(i) = 0$ en cualquier otro caso.

CAPÍTULO 2. CONCEPTOS BÁSICOS

La notación $M_0 \xrightarrow{t_a} M_1$ puede ser extendida a una secuencia de transiciones $M_0 \xrightarrow{\sigma} M_{k+1}$, donde $\sigma = t_a t_b \dots t_j$ representa la secuencia de disparos $M_0 \xrightarrow{t_a} M_1 \xrightarrow{t_b} M_2 \dots \xrightarrow{t_j} M_{k+1}$ y las transiciones están habilitadas en su respectivo marcado. Además, M_{k+1} se dice ser alcanzable desde el marcado M_0 . Por tal motivo la ecuación de estado puede ser reescrita como

$$\begin{aligned} M_{k+1} &= M_0 + C\vec{\sigma} \\ y_k &= \Phi M_k \end{aligned} \tag{2.2}$$

donde $\vec{\sigma}$ es un vector de dimensión $|T|$ en el cual la i -ésima entrada representa el número de ocurrencias de t_i en σ . El vector $\vec{\sigma}$ también es conocido como el vector de Parikh de la secuencia σ .

El conjunto de alcanzabilidad de (N, M_0) , es denotado por $R(N, M_0)$, el cual es el conjunto de todos los posibles marcados alcanzables desde M_0 , disparando únicamente transiciones habilitadas.

El grafo en el que cada nodo representa cada uno de los posibles marcados alcanzables desde M_0 es llamado grafo de alcanzabilidad de la RP .

Ejemplo 15. Considere la RPI (Q, M_0) mostrada en la Figura 2.4. Para este caso la matriz de observación es

$$\Phi = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

donde cada columna representa un lugar en la RPI y cada renglón los diferentes símbolos de salida. Las letras que aparecen junto a los lugares representan los lugares medibles y su símbolo de salida asociado, por ello los lugares p_1 , p_4 y p_8 tienen el mismo vector de observación (las columnas en la matriz Φ asociadas a dichos lugares tienen el mismo valor). Note que la columna asociada al lugar p_2 es el vector de observación $\vec{0}$, esto debido a que el lugar no es medido por ningún sensor. Para este ejemplo $y_0 = \Phi M_0 = [1 \ 0 \ 0 \ 0 \ 0]^T$

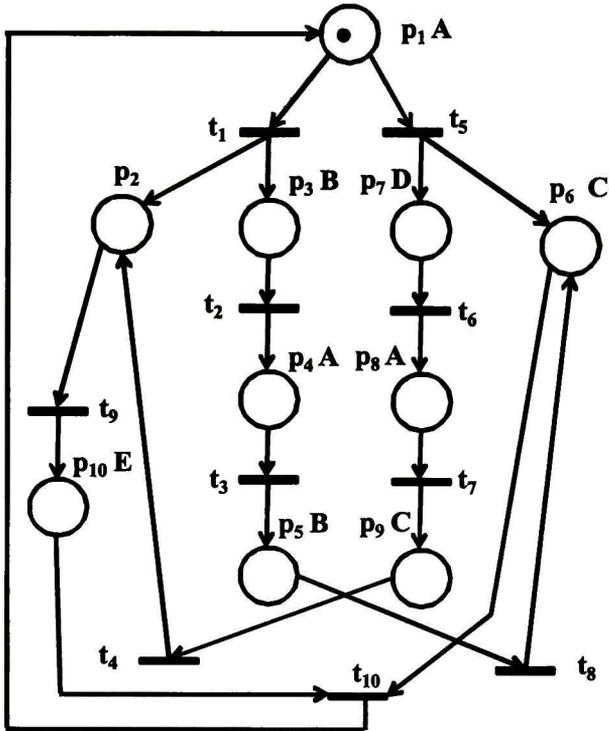


Figura 2.4: Una RPI.

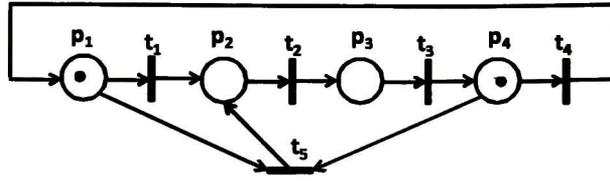


Figura 2.5: *RP* no viva.

2.4. Propiedades de las *RP*

Algunas propiedades importantes de las *RP* que serán utilizadas a través de este son las siguientes:

Definición 16. Una *RP* (N, M_0) es llamada viva (o de forma equivalente M_0 es un marcado vivo de N) si, sin importar el marcado que haya sido alcanzado a partir de M_0 , siempre es posible disparar cualquier transición de N durante su evolución por alguna secuencia de disparo de transiciones posterior.

Ejemplo 17. La *RP* que se muestra en la Figura 2.5 no es viva, debido a que después de disparar la transición t_5 es imposible volver a dispararla. En cambio, la *RP* de la Figura 2.1 si es viva.

Definición 18. Una *RP* es acotada si para cada lugar p existe un número natural b tal que $M_k(p) \leq b, \forall M_k \in R(N, M_0)$. Obviamente, una *RP* es acotada si y sólo si $R(N, M_0)$ es finito.

Definición 19. Una *RP* es llamada b – acotada si ningún lugar tiene una cota mayor que b , donde la cota de un lugar p es el número máximo marcas en algún marcado alcanzado.

Ejemplo 20. La *RP* mostrada en la Figura 2.6 no es acotada, pues el disparo de la transición t_1 siempre produce marcas en el lugar p_3 . En cambio, la *RP* de la Figura 2.1 si es acotada.

Definición 21. Una *RP* (N, M_0) es segura (binaria o 1-acotada) si el máximo número de marcas que puede residir en cada lugar es de 1 para cada $M_k \in R(N, M_0)$.

CAPÍTULO 2. CONCEPTOS BÁSICOS

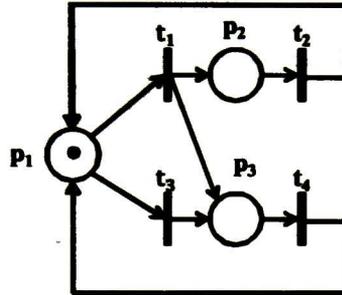


Figura 2.6: *RP* no acotada.

Ejemplo 22. La *RP* de la Figura 2.7 es acotada. Sin embargo, no es segura ya que el número máximo de marcas que reside en cada lugar es de 2. En cambio, las *RP*s de las Figuras 2.1, 2.3, 4.7 si son binarias.

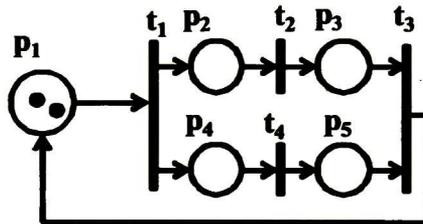


Figura 2.7: *RP* que es 2-acotada.

Definición 23. Una *RP* es cíclica si $\forall M_k \in R(N, M_0), \exists \sigma$ tal que $M_k \xrightarrow{\sigma} M_0$.

Definición 24. Una *RP* es repetitiva a partir de un marcado M_k si existe una secuencia σ tal que $M_k \xrightarrow{\sigma} M_k$ y el vector de Parikh de $\vec{\sigma}$, construido a partir de σ no debe contener elementos nulos. Si el vector de disparo $\vec{\sigma}$ contiene elementos nulos se dice ser parcialmente repetitiva.

Ejemplo 25. La *RP* mostrada en la Figura 2.8 no es cíclica, porque no es posible volver a alcanzar el marcado inicial. Sin embargo, si es repetitiva debido a que

CAPÍTULO 2. CONCEPTOS BÁSICOS

después de alcanzar el marcado $M_1 = [1\ 0\ 0\ 1]^T$, con el disparo de t_1 , existe la secuencia $\sigma = t_2 t_3 t_1$ tal que $M_1 \xrightarrow{\sigma} M_1$. En cambio, la RP de la Figura 2.1 es cíclica.

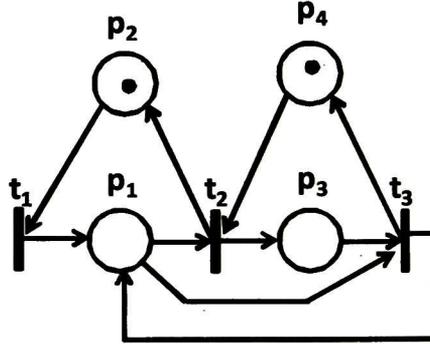


Figura 2.8: RP repetitiva.

Definición 26. Una RP es pura si $\forall p \in P, p^* \cap {}^*p = \emptyset$, es decir una RP es pura si no existe ninguna transición que tenga un lugar que sea al mismo tiempo de entrada y salida de la transición.

Ejemplo 27. La RP mostrada en la Figura 2.9 no es pura, debido a que la transición t_1 tiene como entrada y salida al lugar p_3 . En cambio la RP de la Figura 2.3 sí es pura.

Definición 28. Un camino en una RP N es una secuencia de nodos x_1, x_2, \dots, x_k los cuales satisfacen que $(x_1, x_2), (x_2, x_3), \dots, (x_{k-1}, x_k) \in F$. Un camino x_1, x_2, \dots, x_k se dice que conduce de x_1 a x_k .

Definición 29. Una RP N se dice ser débilmente conexa si para todo par de nodos (x, y) de la red N existe un camino que conduce de x a y , o bien uno que conduzca de y a x .

Definición 30. Una RP N se dice ser fuertemente conexa si para todo par de nodos (x, y) de la red N existe un camino que conduce del nodo x al nodo y .

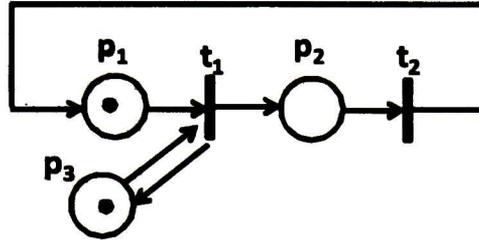


Figura 2.9: *RP* no pura.

Ejemplo 31. Considere nuevamente la *RP* mostrada en la Figura 2.2. Dicha red no es fuertemente conexa, pero sí débilmente conexa. La razón es que para el par de nodos (t_1, t_3) existe el camino t_1, p_2, t_2, p_3, t_3 ; pero para el par (t_3, t_1) no existe camino que conecte a t_3 con t_1 .

2.4.1. Clasificación de las *RP*

Definición 32. Las *RP* pueden ser clasificadas como:

- **Máquinas de Estado (*ME*).** Una *ME* es una *RP* fuertemente conexa donde cada transición tiene exactamente un sólo lugar entrada y un único lugar de salida, es decir $\forall t_j \in T, |\bullet t_j| = |t_j \bullet| = 1$.

Por ejemplo, la *RP* que se presentó en la Figura 2.1 es una *ME*.

- **Grafos Marcados (*GM*).** Un grafo marcado *GM* es una *RP* fuertemente conexa donde cada lugar tiene exactamente una sola transición de entrada y una transición de salida, es decir $\forall p_i \in P, |\bullet p_i| = |p_i \bullet| = 1$.

Por ejemplo, la *RP* expuesta en la Figura 2.3 es un *GM*.

- **Redes de Libre Elección (*LE*).** Una *LE* es una *RP* fuertemente conexa tal que si $p_i \bullet \cap \bullet p_j \neq \emptyset$, entonces $p_i \bullet = \bullet p_j, \forall p_i, p_j \in P$.

Por ejemplo, la *RP* mostrada en la Figura 2.4 es una *LE*.

- **Otro tipo de redes.** Son aquellas redes que no están incluidas en las definiciones previas.

Por ejemplo, la *RP* presentada en la Figura 2.8 entra en la clasificación de otros tipos de redes.

2.5. Consideraciones importantes

Los resultados expuestos en esta tesis toman como hipótesis principales los siguientes puntos:

1. Este trabajo se enfoca en redes puras y fuertemente conexas, en las cuales el marcado inicial es desconocido. Note que, como el marcado inicial es desconocido, pueden existir una gran cantidad de marcados iniciales. Por lo tanto, se utilizará la notación (Q, μ_0) para enfatizar que el marcado de la *RPI* es desconocido; pero que puede ser cualquiera tal que la *RPI* sea viva y binaria, es decir $\mu_0 = \{M_0^1, M_0^2, \dots, M_0^k\}$ es el conjunto de k marcados iniciales vivos y binarios para Q .
2. Todos los lugares de entrada de una transición tienen que tener un símbolo de salida asociado. Si $|\bullet t_j| > 1$ entonces $\forall p_i \in \bullet t_j, \Phi(\bullet, i) \neq \vec{0}$ (donde la i -ésima columna de Φ , $\Phi(\bullet, i)$, mapea el marcado del lugar p_i en elementos del vector de observación). Esta restricción tiene la finalidad de garantizar que el cambio en los símbolos de salida siempre será observable y evitar confusiones entre las transiciones de una sola entrada con las de múltiples entradas.

Por ejemplo, los lugares de entrada a la transición mostrada en la Figura 2.10.a) no cumplen con la restricción establecida en el punto 2), puesto que el lugar p_1 no se mide por ningún sensor.

3. Para cualquier transición t_j no está permitido que para cualquier $p_k \in \bullet t_j, p_l \in t_j^\bullet$, el marcado de estos lugares sea mapeado en un vector de observación igual, es decir $\Phi(\bullet, k) = \Phi(\bullet, l)$. Esta restricción permite garantizar que cualquier par de vectores de observación consecutivos, y_k y y_{k+1} , son diferentes entre sí (ver [43]). Esto significa que el disparo de cada transición se puede observar mediante un cambio en los símbolos de salida, aunque no se pueda saber cuál transición se disparó.

Por ejemplo, la transición t_1 de la Figura 2.10.a) tampoco cumple con la restricción 3) debido a que $p_1 \in \bullet t_1, p_3 \in t_1^\bullet$ y tanto p_1 como p_3 tienen asociado el mismo vector de observación $\Phi(\bullet, 1) = \Phi(\bullet, 3)$. En cambio, la transición mostrada en la Figura 2.10.b) sí es permitida.

Las siguientes definiciones están relacionadas con la secuencia de disparo de transiciones y con los símbolos de salida observados:

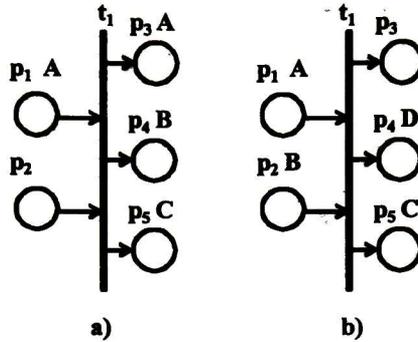


Figura 2.10: Transiciones en una RPI.

2.5.1. Lenguajes y secuencias de disparo

Definición 33. Una secuencia de disparo de transiciones de una RPI (Q, M_0) es una secuencia $\sigma = t_i t_j \dots t_k \dots$ tal que $M_0 \xrightarrow{t_i} M_i \xrightarrow{t_j} \dots, M_w \xrightarrow{t_k} \dots$. El conjunto de todas las secuencias de disparo de transiciones es llamado el lenguaje de disparo $\mathcal{L}(Q, M_0) = \{\sigma \mid \sigma = t_i t_j \dots t_k \dots \text{ y } M_0 \xrightarrow{t_i} M_i \xrightarrow{t_j} \dots, M_w \xrightarrow{t_k} \dots\}$. Análogamente, se tiene que $\mathcal{L}(Q, \mu_0) = \bigcup_{i=1}^k \mathcal{L}(Q, M_0^i)$.

Definición 34. Sea (Q, M_0) una RPI. El conjunto de todas las secuencias de símbolos de salida de (Q, M_0) será denotado por $\Lambda(Q, M_0)$. En particular, el conjunto de todas las secuencias de símbolos de salida de logitud mayor o igual a n será denotado por $\Lambda^n(Q, M_0)$, i.e. $\Lambda^n(Q, M_0) = \{\omega \in \Lambda(Q, M_0) : |\omega| \geq n\}$.

Definición 35. Si ω es una secuencia de símbolos de salida, entonces la secuencia de disparo de transiciones $\sigma \in \mathcal{L}(Q, M_0)$ cuyo disparo genera a ω es denotada por σ_ω .

Definición 36. Una secuencia de símbolos de salida de (Q, M_0) es una secuencia $\omega = (y_0)(y_1) \dots (y_n)$, $y_i = \Phi M_i$ y los marcados M_1, M_2, \dots, M_n son marcados consecutivos alcanzables a partir de M_0 . Si ω es una secuencia de símbolos de salida, entonces el conjunto de secuencias de disparo de transiciones $\sigma \in \mathcal{L}(Q, M_0)$ cuyo disparo genera ω es denotado por $S_{\sigma\omega} = \{\sigma_{\omega_1}, \sigma_{\omega_2}, \dots, \sigma_{\omega n}\}$.

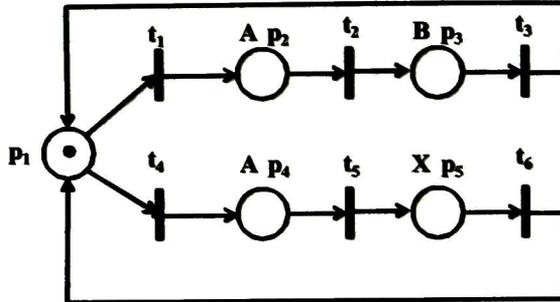


Figura 2.11: RPI donde una secuencia de símbolos de salida tiene más de una secuencia de transiciones asociada.

Ejemplo 37. Sea \$Q\$ la RPI mostrada en la Figura 2.11, donde

$$\Phi = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Suponga que se da la secuencia de símbolos de salida $\omega_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$ entonces para dicha secuencia el conjunto de secuencias de disparo de transiciones $\sigma \in \mathcal{L}(Q, M_0)$ cuyo disparo genera a ω_1 es $S_{\sigma\omega_1} = \{t_1, t_4\}$.

Para la secuencia de símbolos de salida $\omega_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$ sólo se tiene una secuencia de disparo de transiciones que genera a ω_2 la cual es $S_{\sigma\omega_2} = \{t_4 t_5\}$

2.6. Ejemplo

Un sistema está compuesto por dos vehículos V_1 y V_2 , una vía para cada vehículo, un interruptor M y cuatro sensores indicadores de posición (A, B, C y D). Este sistema es mostrado en la Figura 2.12.

CAPÍTULO 2. CONCEPTOS BÁSICOS

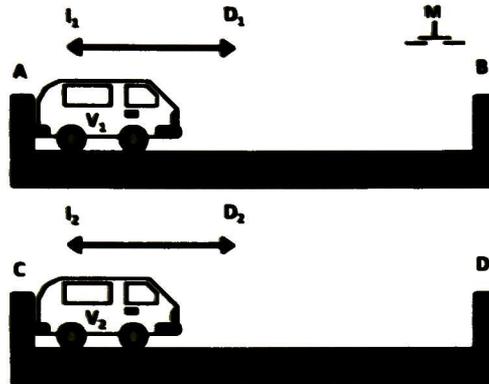


Figura 2.12: Sistema de vehículos.

Los vehículos V_1 y V_2 inicialmente están siendo detectados por los sensores etiquetados como A y B , es decir se encuentran en la posición extrema izquierda de su respectiva vía. Los vehículos son de juguete con motores eléctricos con alimentación por la vía, los cuales siempre que la vía está energizada avanzarán. Las vías independientes son energizadas a través de la activación del interruptor M , con lo cual ambos vehículos comienzan su desplazamiento hacia la derecha hasta que los sensores de posición C y D detecten que han llegado al final de su recorrido. Una vez que cualquier vehículo alcanza la posición extrema derecha de la vía inicia su recorrido de regreso hasta su posición inicial, donde espera hasta que el otro vehículo también alcance su posición inicial. Ya que los vehículos han alcanzado su posición inicial esperan en reposo hasta oprimir de nuevo el interruptor M para comenzar nuevamente el ciclo descrito.

El modelo en *RPI* que permite capturar el comportamiento de este sistema se exhibe en la Figura 2.2.

A la *RPI* se le puede asociar la siguiente interpretación:

Para los lugares:

p_1 : Vehículo 1 en reposo, localizado en la extrema izquierda de la vía y detectado por el sensor A .

p_2 : Vehículo 2 en reposo, localizado en la extrema izquierda de la vía y detectado por el sensor B .

CAPÍTULO 2. CONCEPTOS BÁSICOS

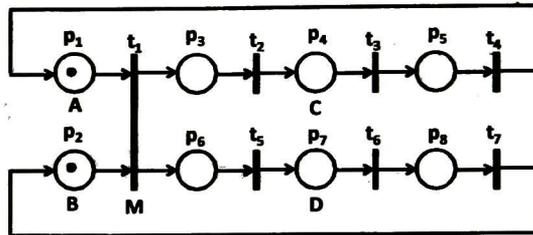


Figura 2.13: Modelo en *RPI* para el sistema de vehículos.

- p_3 : Vehículo 1 avanzando hacia la derecha.
- p_4 : Vehículo 1 ha alcanzado su posición extrema derecha y es detectado por el sensor *C*.
- p_5 : Vehículo 1 avanzando hacia la izquierda.
- p_6 : Vehículo 2 avanzando hacia la derecha.
- p_7 : Vehículo 2 ha alcanzado su posición extrema derecha y es detectado por el sensor *D*.
- p_8 : Vehículo 2 avanzando hacia la izquierda.

Para las transiciones:

- t_1 : Se oprime el interruptor *M*, inicio del recorrido hacia la derecha de los vehículos.
- t_2 : El vehículo 1 finaliza su recorrido hacia la derecha.
- t_3 : El vehículo 1 comienza su recorrido hacia la izquierda.
- t_4 : El vehículo 1 finaliza su recorrido hacia la izquierda y regresa a su posición inicial.
- t_5 : El vehículo 2 finaliza su recorrido hacia la derecha.
- t_6 : El vehículo 2 comienza su recorrido hacia la izquierda.
- t_7 : El vehículo 2 finaliza su recorrido hacia la izquierda y regresa a su posición inicial.

La matriz de incidencia de esta red es:

CAPÍTULO 2. CONCEPTOS BÁSICOS

$$C = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

Donde el mercado inicial $M_0 = [1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$

La matriz de observación para esta *RPI* es:

$$\Phi = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

El conjunto de entradas y salidas para todo nodo x de la *RPI* es:

Entradas	Salidas
$\bullet p_1 = \{t_4\}$	$p_1^\circ = \{t_1\}$
$\bullet p_2 = \{t_7\}$	$p_2^\circ = \{t_1\}$
$\bullet p_3 = \{t_1\}$	$p_3^\circ = \{t_2\}$
$\bullet p_4 = \{t_2\}$	$p_4^\circ = \{t_3\}$
$\bullet p_5 = \{t_3\}$	$p_5^\circ = \{t_4\}$
$\bullet p_6 = \{t_1\}$	$p_6^\circ = \{t_5\}$
$\bullet p_7 = \{t_5\}$	$p_7^\circ = \{t_6\}$
$\bullet p_8 = \{t_6\}$	$p_8^\circ = \{t_7\}$

Entradas	Salidas
$\bullet t_1 = \{p_1, p_2\}$	$t_1^\circ = \{p_3, p_6\}$
$\bullet t_2 = \{p_3\}$	$t_2^\circ = \{p_4\}$
$\bullet t_3 = \{p_4\}$	$t_3^\circ = \{p_5\}$
$\bullet t_4 = \{p_5\}$	$t_4^\circ = \{p_1\}$
$\bullet t_5 = \{p_6\}$	$t_5^\circ = \{p_7\}$
$\bullet t_6 = \{p_7\}$	$t_6^\circ = \{p_8\}$
$\bullet t_7 = \{p_8\}$	$t_7^\circ = \{p_2\}$

CAPÍTULO 2. CONCEPTOS BÁSICOS

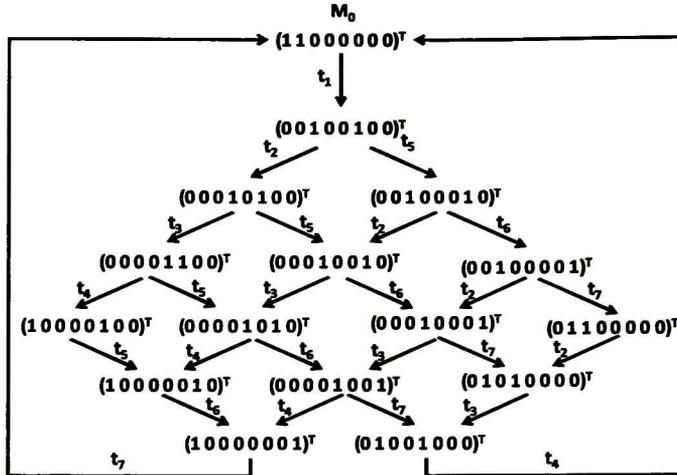


Figura 2.14: Grafo de alcanzabilidad para la *RPI* que modela el sistema de los vehículos.

Los *P* – *semiflujos* y *T* – *semiflujos* de la *RPI* son:

<i>P</i> – <i>semiflujos</i>	<i>T</i> – <i>semiflujos</i>
$Y_1^T = [1\ 0\ 1\ 1\ 1\ 0\ 0\ 0]$	$X_1 = [1\ 1\ 1\ 1\ 1\ 1\ 1]^T$
$Y_2^T = [0\ 1\ 0\ 0\ 0\ 1\ 1\ 1]$	

Los soportes de los *P* – *semiflujos* y *T* – *semiflujos* son:

$$\begin{aligned} \langle Y_1^T \rangle &= \{p_1, p_3, p_4, p_5\} \\ \langle Y_2^T \rangle &= \{p_2, p_6, p_7, p_8\} \\ \langle X_1 \rangle &= \{t_1, t_2, t_3, t_4, t_5, t_6, t_7\} \end{aligned}$$

Por sus características esta *RPI* es pura, viva, segura, cíclica y fuertemente conexas. Las propiedades de vivacidad, ciclicidad y acotamiento se pueden apreciar claramente en su grafo de alcanzabilidad mostrado en la Figura 2.14.

Debido a que cumple con la condición $\forall p_i \in P, |^*p_i| = |p_i^*| = 1$, entonces es un *GM*.

3. CARACTERIZACIÓN DE LA SECUENCIA DETECTABILIDAD ESTRUCTURAL

Resumen: En este capítulo se definen formalmente las propiedades de secuencia detectabilidad y secuencia detectabilidad estructural en *RPIs*. Asimismo, se proporciona una caracterización de ambas propiedades para la clase de las *LE*.

3.1. Introducción

Las principales líneas de investigación en los *SED* se enfocan al estudio de las propiedades de observabilidad, controlabilidad, diagnóstico e identificación. En el análisis de dichas propiedades la detección de las secuencias de disparo de transiciones juega un rol muy importante porque todas estas propiedades se basan en el conocimiento de las secuencias. Por ejemplo, si no se garantiza que las secuencias de disparo de transiciones son distinguibles entre sí, se podrían tener los siguientes problemas: una infinidad de modelos que resultarían de identificar un sistema, no poder diagnosticar la ocurrencia de faltas en diversos modelos, no poder conocer el estado actual de una *RPI* a partir del conocimiento de sus salidas, no poder llevar a cabo la controlabilidad de sistemas en cuyo modelo exista la confusión de secuencias, entre otros. Sin embargo, en la literatura existente sólo se ha proporcionado una caracterización para la detección de la ocurrencia de sólo un evento, o bien, cuando tratan con secuencias de eventos la caracterización se limita a una clase muy restringida de *RPIs*.

Los estudios que han abordado el problema de la distinguibilidad de las secuencias de eventos a través del grafo de alcanzabilidad de una *RPI* no son convenientes por la explosión de estados del mismo, por ello es necesaria una caracterización que haga uso de la estructura misma de la red para decidir si las secuencias de disparo de transiciones pueden ser distinguidas.

Por lo cual, este capítulo está dedicado a la caracterización de distinción de las secuencias de disparo de transiciones, en distintas clases de redes, a través de

CAPÍTULO 3. CARACTERIZACION DE LA SSD

un análisis estructural de la RPI , ya sea que el marcado inicial sea conocido o no.

Cabe señalar que en este capítulo no se caracteriza la secuencia detectabilidad estructural para redes no binarias porque ya se encuentra caracterizada en [45] en la Proposición 2, donde se caracteriza a las $RPIs$ que son transición distinguible (concepto similar a la evento detectabilidad cuando se permite asignar etiquetas repetidas a las transiciones, es decir se hace uso de la información de entrada y salida para determinar la evento detectabilidad).

Este capítulo está dividido de la siguiente manera: en la sección 3.2 se presenta la definición de secuencia detectabilidad estructural y secuencia detectabilidad, así como las relaciones utilizadas en la caracterización de dichas propiedades. La caracterización de la secuencia detectabilidad estructural en $RPIs$ pertenecientes a la clase de ME , GM y LE es presentada en las secciones 3.3, 3.4 y 3.5, respectivamente, junto con algoritmos que permiten determinar si una RPI posee la propiedad de secuencia detectabilidad estructural. Posteriormente, en la sección 3.6 se presenta la caracterización de la secuencia detectabilidad. Por último, las conclusiones del capítulo son presentadas en la sección 3.7.

3.2. Definición de la Secuencia Detectabilidad Estructural y de la Secuencia Detectabilidad

En esta sección se introduce la noción de secuencia detectabilidad y secuencia detectabilidad estructural, así como la diferencia entre dichos conceptos. También se definen algunas relaciones que serán de utilidad en la caracterización de estas propiedades.

La secuencia detectabilidad se define de la siguiente manera:

Definición 38. Una $RPI (Q, M_0)$ se dice ser secuencia detectable (SD) si existe un entero $k < \infty$ y una función $\Psi_s : \Lambda^k(Q, M_0) \times \{(Q, M_0)\} \rightarrow \mathcal{L}(Q, M_0)$ tal que $\forall \omega \in \Lambda^k(Q, M_0)$ se satisface que $\Psi_s(\omega, (Q, M_0)) = \sigma_\omega$ y no existe $\sigma'_\omega \neq \sigma_\omega$ tal que $\sigma'_\omega \in S_{\sigma_\omega}$.

Dicho en otras palabras, la secuencia detectabilidad implica que a partir de un marcado inicial dado no existen dos diferentes secuencias de disparo de transiciones que generen los mismos símbolos de salida, y cualquier secuencia de disparo de transiciones debe poder distinguirse de las demás en un número finito de pasos.

En cambio la secuencia detectabilidad estructural es definida como:

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Definición 39. Una RPI (Q, μ_0) se dice ser estructuralmente secuencia detectable (SSD) si el disparo de cualquiera dos secuencias de disparo de transiciones $\sigma_1, \sigma_2 \in \mathcal{L}(Q, \mu_0)$ pueden ser distinguidas entre sí usando la información de salida de la RPI después de un número finito de disparo de transiciones.

Como se puede observar, la secuencia detectabilidad estructural implica que no hay dos marcados iniciales, vivos y binarios, a partir de los cuales se puedan disparar secuencias de disparo de transiciones que generen los mismos símbolos de salida, o bien no hay dos secuencias de disparo de transiciones a partir del mismo marcado, vivo y binario, que generen los mismos símbolos de salida. Además, cualquier secuencia de disparo de transiciones debe de poder distinguirse de las demás en número finito de pasos.

Note que una RPI que sea estructuralmente secuencia detectable implica que será secuencia detectable para un marcado vivo y binario dado, sin embargo la aseveración contraria es falsa, es decir, una RPI que sea secuencia detectable para un marcado dado no implica que sea estructuralmente secuencia detectable.

Ejemplo 40. Considera la RPI Q binaria, pura y fuertemente conexa mostrada

en la Figura 3.1, donde $\Phi = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$.

Suponga que el marcado inicial es desconocido. Observe que $\exists \sigma_1 = t_2\beta^+$, $\sigma_2 = t_7\beta^+ \in \mathcal{L}(Q, \mu_0)$, donde β es una secuencia de disparo de transiciones tal que $M_k \xrightarrow{\beta} M_k$ y el símbolo $+$ representa la cerradura positiva de Kleene (Es un lenguaje que contiene todas las posibles cadenas excepto la vacía). Entonces la RPI Q no es SSD debido a que σ_1 no es distinguible de σ_2 después del disparo de un número finito de transiciones, basado en los símbolos de salida de la RPI.

Note que para este ejemplo las transiciones t_2 y t_5 no pueden estar simultáneamente habilitadas, debido a que el marcado inicial $M_0^k = [0 \ 1 \ 0 \ 1 \ 0]^T$ no pertenece al conjunto μ_0 , es decir, el marcado inicial M_0^k no es factible porque con dicho marcado la RPI Q no es binaria.

El disparo de dos transiciones no puede ser distinguido entre sí cuando éstas generan el mismo cambio de símbolos de salida, es decir suponga que $M_i \xrightarrow{t_i} M_{i+1}$, $M_j \xrightarrow{t_j} M_{j+1}$, y el cambio de símbolos de salida $\Phi M_{i+1} - \Phi M_i$, cuando t_i es disparada, es igual al cambio de símbolos de salida $\Phi M_{j+1} - \Phi M_j$, cuando t_j es disparada. Por tal motivo, en el ejemplo de la Figura 3.1 no es posible distinguir el

CAPÍTULO 3. CARACTERIZACION DE LA SSD

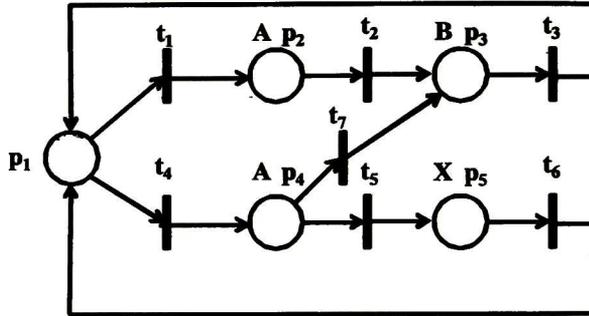


Figura 3.1: Una RPI que no es SSD.

disparo de las transiciones t_1 y t_4 cuando el lugar p_1 es marcado o si el símbolo de salida inicialmente observado es $\Phi M_0 = [1 \ 0 \ 0]^T$ y si el cambio de símbolos de salida es $\Phi M_1 - \Phi M_0 = [-1 \ 1 \ 0]^T$ tampoco se podría saber si la transición que se disparó fue t_2 o t_7 . La definición mostrada a continuación toma en consideración la observación anterior.

Proposición 41. Sean t_i, t_j dos transiciones. El disparo de t_i es indistinguible del disparo de t_j ($t_i \approx_I t_j$) si $\Phi C(\bullet, t_i) = \Phi C(\bullet, t_j)$.

Observe que esta es una relación de equivalencia, por lo que particiona al conjunto de transiciones. Las clases de equivalencia $[t_x]$ tales que $|[t_x]| = 1$, son clases de equivalencia que pueden ser distinguidas, de lo contrario son clases de equivalencia $[t_k]$ de transiciones indistinguibles (es decir, las transiciones que pertenecen a las clases $[t_k]$ son indistinguibles entre sí).

Esta definición puede ser extendida a dos secuencias, finitas o arbitrariamente largas, de disparo de transiciones $\sigma_1 = t_1 \dots t_k \dots$, $\sigma_2 = t'_1 \dots t'_k \dots$, $|\sigma_1| = |\sigma_2|$, donde σ_1 y σ_2 son indistinguibles entre sí ($\sigma_1 \approx_I \sigma_2$) si $t_1 \approx_I t'_1, \dots, t_k \approx_I t'_k, \dots$

Debido a que se tiene el interés en los pares ordenados de diferentes transiciones que son indistinguibles entre sí, se define el conjunto $Gr(\approx_I) = \{(t_i, t_j) \in T \times T \mid t_i \neq t_j \text{ y } t_i \approx_I t_j\}$.

Claramente, si $\sigma_1 \approx_I \sigma_2$ y éstas son secuencias de longitud arbitrariamente larga que pueden ser disparadas en una RPI viva y acotada, entonces son indistinguibles entre sí. Estas secuencias indistinguibles pudieron haber sido generadas

CAPÍTULO 3. CARACTERIZACION DE LA SSD

por el disparo de las transiciones de T – *semiflujos*. La siguiente definición introduce la noción de T – *componentes* indistinguibles entre sí.

Definición 42. Sean $\mathcal{T}_i, \mathcal{T}_j$ dos T –*componentes* inducidos por los T –*semiflujos* X_i, X_j , respectivamente. Los T –*componentes* $\mathcal{T}_i, \mathcal{T}_j$ (el subíndice i puede ser igual al j) son indistinguibles entre sí ($\mathcal{T}_i \approx_I \mathcal{T}_j$) si existen dos secuencias de disparo de transiciones $\sigma_i \neq \sigma_k, |\sigma_i| = |\sigma_k|, \sigma_i \approx_I \sigma_k$, tal que $\vec{\sigma}_i = X_i, \vec{\sigma}_k = X_j$.

Note que cuando un T – *componente* es indistinguible con respecto de sí mismo las secuencias de disparo de transiciones σ_i y σ_k generan al mismo T –*componente*; pero σ_k es una permutación de transiciones disparables de la secuencia σ_i .

Sin embargo, no todas las secuencias indistinguibles de longitud arbitrariamente larga son generadas por T – *componentes* indistinguibles entre sí, por lo cual es necesario caracterizar todas las estructuras que puedan generar este tipo de secuencias. Las siguientes relaciones de transiciones capturan otras estructuras en la *RPI* que pueden generar secuencias de disparo de transiciones de longitud arbitrariamente larga e indistinguibles entre sí.

Definición 43. *Relación hacia atrás* (\approx^-). Sea Q una *RPI*. Las transiciones $t_i, t_j \in T$ están relacionadas hacia atrás ($t_i \approx^- t_j$) si $t_i \approx_I t_j$ y $\bullet t_i = \bullet t_j$.

Explicado en palabras más simples, dos transiciones se relacionan hacia atrás si no podemos distinguir su disparo y su conjunto de lugares de entrada es el mismo.

Definición 44. *Relación hacia adelante* (\approx^+). Sea Q una *RPI*. Las transiciones $t_i, t_j \in T$ están relacionadas hacia adelante ($t_i \approx^+ t_j$) si $t_i \approx_I t_j$ y $t_i^\circ = t_j^\circ$.

Explicado en palabras más simples, dos transiciones se relacionan hacia adelante si son indistinguibles y su conjunto de lugares de salida es el mismo.

Ejemplo 45. En la *RPI* Q mostrada en el Figura 3.1, se puede apreciar que $t_2 \approx^+ t_7, t_1 \approx^- t_4$ y $t_4 t_7 t_3 \dots t_4 t_7 t_3 \approx_I t_1 t_2 t_3 \dots t_1 t_2 t_3$.

3.3. Secuencia Detectabilidad Estructural en *ME*

El siguiente resultado caracteriza la propiedad de la secuencia detectabilidad estructural en *ME*.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Teorema 46. Sea (Q, μ_0) una *ME* viva, binaria y pura. Q es estructuralmente secuenciable detectable si y sólo si las siguientes tres condiciones son satisfechas:

1. $\forall t_i, t_j \in T, i \neq j, t_i \not\approx^+ t t_j,$
2. $\forall t_i, t_j \in T, i \neq j, t_i \not\approx^- t_j,$
3. No existen T – componentes indistinguibles entre sí.

Demostración. En el desarrollo de esta demostración se procede usando la forma contrapositiva del enunciado del Teorema 46.

(\rightarrow) Si existen $t_i, t_j \in T, i \neq j, t_i \approx^+ t t_j$ en una *RPI*, entonces por la Definición 44 $t_i^* = t_j^*$. Debido a que Q es viva y binaria, entonces existen dos marcados iniciales diferentes $M_0, M'_0 \in \mu_0$ que permiten habilitar t_i, t_j respectivamente. Por lo tanto, $M_0 \xrightarrow{t_i} M_1 \xrightarrow{\sigma_1} M_k$ y $M'_0 \xrightarrow{t_j} M_1 \xrightarrow{\sigma_1} M_k$, donde σ_1 es una secuencia de longitud arbitrariamente larga. Las secuencias $t_i\sigma_1, t_j\sigma_1$ son indistinguibles entre sí, por lo tanto Q no es *SSD*.

Si $t_i \approx^- t_j$, entonces ${}^*t_i = {}^*t_j$. Debido a que Q es viva entonces existe una secuencia de transiciones disparable σ a partir de M_0 que permite alcanzar el marcado M_k donde el lugar ${}^*t_i = {}^*t_j$ se marca. Puesto que Q es viva, entonces existe otra secuencia de disparo β tal que $M_k \xrightarrow{\beta} M_k$. Entonces la secuencia $\sigma\beta^*t_i$ es indistinguible de $\sigma\beta^*t_j$. Por lo tanto Q no es *SSD*.

Si existen T – componentes indistinguibles, entonces existen dos secuencias de disparo de transiciones de longitud arbitrariamente larga que son indistinguibles, por lo tanto Q no es *SSD*.

(\leftarrow) Suponga que Q no es *SSD*, entonces existen dos secuencias de disparo de transiciones de longitud arbitrariamente larga $\sigma_1, \sigma_2, \sigma_1 \neq \sigma_2$, tales que $\sigma_1 \approx_I \sigma_2$. Podrían darse los siguientes tres casos 1) la secuencia σ_1 es completamente diferente de σ_2 , 2) σ_1, σ_2 tienen una subsecuencia común o 3) son iguales. Este último caso no es relevante para el estudio de la secuenciable detectabilidad estructural porque se necesitan secuencias de disparo de transiciones diferentes. Por lo tanto, hay que enfocarse en los primeros dos casos.

Si las secuencias son completamente diferentes entre sí, entonces la *RPI* presenta T – componentes indistinguibles.

Si $\sigma_1 = \dots t_a \alpha_1 \dots, \sigma_2 = \dots t_b \alpha_1 \dots$ (es decir, comparten subsecuencias en común), entonces, como t_a y t_b pertenecen al mismo P – componente (Por el hecho de que Q es una *ME*), entonces $t_a \approx^+ t t_b$ o hay T – componentes indistinguibles.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Si $\sigma_1 = \dots\alpha_1 t_a \dots$, $\sigma_2 = \dots\alpha_1 t_b \dots$ (es decir, comparten subsecuencias en común), entonces como t_a, t_b pertenecen al mismo P – *componente*, entonces $t_a \approx^- t_b$ o existen T – *componentes* indistinguibles. ■

Aunque la condición 3) del teorema anterior implica la búsqueda de T – *componentes* indistinguibles, lo cual es, computacionalmente hablando, una tarea compleja, este problema puede ser superado usando la siguiente definición y un algoritmo similar al algoritmo de Moore.

Definición 47. *Relación cíclica (\approx°).* Sea Q una ME viva y pura. Las transiciones $t_i, t_j \in T$ están cíclicamente relacionadas ($t_i \approx^\circ t_j$) si las siguientes condiciones son satisfechas:

1. $t_i \approx_I t_j$
2. existen secuencias de disparo de transiciones α_1, α_2 tales que $|\alpha_1| = |\alpha_2|$ y $t_i \alpha_1 t_i \approx_I t_j \alpha_2 t_j$.

La relación \approx° será usada para encontrar dos secuencias de disparo de transiciones arbitrariamente largas que sean indistinguibles entre sí. Note que si existen tales secuencias en una ME y no existen transiciones relacionadas \approx^+ ni transiciones relacionadas \approx^- , entonces estas secuencias pertenecen al soporte de T – *semiflujos*.

La definición anterior sugiere el siguiente algoritmo para determinar si existen dos T – *componentes* indistinguibles entre sí, el cual se asemeja al algoritmo de minimización de autómatas de Moore. Sin embargo, en el algoritmo de Moore, dos estados son equivalentes si todos los estados sucesores son equivalentes (es decir, se está implementado la relación de equivalencia de Nerode). En el algoritmo que aquí se propone, dos transiciones son equivalentes si al menos un par de las transiciones, en el conjunto de las transiciones subsecuentes, son también equivalentes.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Algoritmo 48. *Determinar si existen T – componentes indistinguibles entre sí en una ME, cuando no hay transiciones relacionadas \approx^- ni \approx^+ .*

Entrada: Una ME $Q = \{P, T, \Phi, F, W\}$

Salida: Si existe $(t_i, t_j) \in Gr(\approx_I)$ tal que $t_i \approx^o t_j$

1. Llenado del arreglo H

- a. Sea H un arreglo de dimensión $|T| \times |T|$, donde el renglón $H(r, \bullet)$ representa a la transición $t_r \in T$ y la columna $H(\bullet, s)$ a la transición $t_s \in T$.
- b. Sea $H(i, j) = \square$, $j \geq i$ (debido a que la relación cíclica es reflexiva y simétrica, entonces únicamente se requiere del triángulo inferior del arreglo H).
- c. Si $\exists t_i, t_j \in T$, $j < i$, $t_i \approx_I t_j$ entonces $H(i, j) = \square$, de lo contrario $H(i, j) = \times$.

2. Actualizando el arreglo H

- a. Si $H(r, s) = \square$, y $\exists t_a \in (t_r^*)^o$ y $\exists t_b \in (t_s^*)^o$ tales que $H(a, b) = \square$, entonces $H(r, s) = \square$. De lo contrario $H(r, s) = \times$.

3. Repetir el paso 2 hasta que el arreglo H permanezca sin cambios.

Proposición 49. *El algoritmo 48 permite determinar las relaciones cíclicamente relacionadas.*

Demostración. Note que el arreglo H se llena de la siguiente manera. Al comienzo (en el paso 1.c) $H(r, s) = \square$, $r \neq s$, significa que $t_r \approx_I t_s$. En el siguiente paso, 2.a, $H(r, s) = \square$ si y sólo si $\exists t_a \in (t_r^*)^o$, $\exists t_b \in (t_s^*)^o$ tal que $H(a, b) = \square$. Esto significa que el disparo de $t_r t_a$ no puede distinguirse del disparo de $t_s t_b$. De lo contrario, si $H(r, s) = \times$ entonces t_r puede ser distinguida de t_s después de un número finito de disparo de transiciones. Este paso es iterado para determinar si existen secuencias de disparo de transiciones que no pueden distinguirse entre sí.

Cuando el arreglo H permanece sin cambios y cada entrada tiene un símbolo \times , entonces en un número finito de pasos cualquier secuencia de disparo de transiciones puede distinguirse entre sí. Si existe una entrada $H(r, s) = \square$, entonces existe un sucesor de r , nombrado r_1 , y otro sucesor de s , nombrado s_1 ,

CAPÍTULO 3. CARACTERIZACION DE LA SSD

tales que son indistinguibles entre sí. Este procedimiento continúa con r_2 y s_2 , y así sucesivamente. Debido a que la red es finita y fuertemente conexa, entonces las transiciones t_r y t_s serán alcanzadas nuevamente. Como Q es una *ME* donde $\forall t \in T, |t^*| = |^*t| = 1$, entonces estas transiciones cíclicamente relacionadas forman T - *semiflujos* cuando no hay transiciones relacionadas \approx^+ . ■

Proposición 50. *El arreglo H permanece sin cambios con una complejidad de orden $O(m^3)$ donde $m = |T|$.*

Demostración. El arreglo H tiene una dimensión de $|T| \times |T|$. Suponga que $|T| = m$. En el peor de los casos solamente una entrada del arreglo H es igual a \times y el resto son \square . Suponga que no hay transiciones cíclicamente relacionadas, entonces cada iteración del algoritmo 48 agrega un símbolo \times en el peor de los casos, por lo tanto después de $m - 1$ iteraciones el arreglo H únicamente contendrá símbolos \times y hasta la iteración m el arreglo H permanecerá sin cambios. Por tal motivo las m^2 entradas del arreglo H son visitadas a lo más m veces. Por lo tanto, el arreglo H permanece sin cambios a lo más con una complejidad de $O(m^3)$. ■

Ejemplo 51. *Sea Q una *ME* binaria, viva y pura representada en la Figura 3.2 donde*

$$\Phi = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Observe que $\forall t_i, t_j, i \neq j, t_i \not\approx^+ t_j, t_i \not\approx^- t_j$ y el Algoritmo 48 determina que $\forall T_i, T_j$ se satisface que $T_i \not\approx T_j$, donde $\langle X_1 \rangle = \{t_1, t_2, t_3, t_4, t_5, t_6\}$, $\langle X_2 \rangle = \{t_7, t_8, t_9, t_{10}, t_{11}, t_{12}\}$. Entonces Q es *SSD*.

Ejemplo 52. *Considérese la *ME* mostrada en la Figura 3.3 donde*

$$\Phi = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

En este caso no hay transiciones \approx^+ relacionadas ni transiciones \approx^- relacionadas, sin embargo los T - *componentes* generados por el soporte de los T - *semiflujos*

CAPÍTULO 3. CARACTERIZACION DE LA SSD

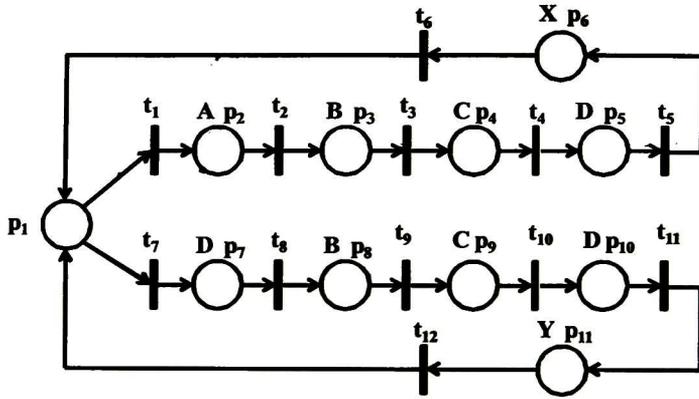


Figura 3.2: Una ME que es SSD.

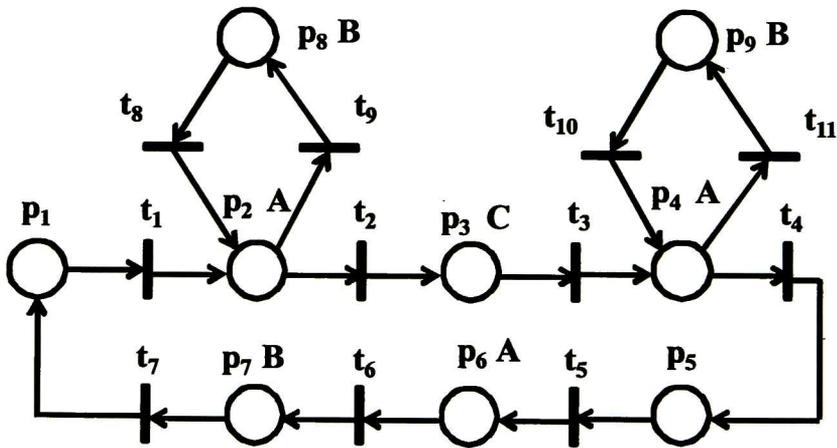


Figura 3.3: Una ME con T - componentes indistinguibles.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

$\langle X_1 \rangle = \{t_8, t_9\}$ y $\langle X_2 \rangle = \{t_{10}, t_{11}\}$ son indistinguibles entre sí, entonces la ME no es SSD. La existencia de tales T – componentes indistinguibles es determinada de aplicar el Algoritmo 48.

En la primera iteración del Algoritmo 48, se obtiene el arreglo H mostrado en la Figura 3.4. Note que $H(i, i) = \square$, esto es debido a que la relación cíclica es reflexiva, es decir la transición t_i está relacionada cíclicamente consigo misma. Cabe señalar que el Algoritmo 48 únicamente trabaja con los elementos que aparecen por debajo de la diagonal principal porque debido a que la relación cíclica es simétrica, es decir si $t_i \approx^\circ t_j$ entonces $t_j \approx^\circ t_i$, por tal motivo no es necesario el llenado de los elementos que aparecen por encima de la diagonal principal puesto que su valor es igual al de su recíproco.

Note que $H(9, 6) = \square$. Sin embargo, en la siguiente iteración, debido a que $H(8, 7) = \times$ entonces $H(9, 6) = \times$. Este procedimiento es iterado hasta que ninguna \times sea agregada al arreglo H . Entonces el arreglo representado en la Figura 3.5 es obtenido. Debido a que existen entradas en blanco por debajo de la diagonal principal de dicho arreglo, entonces existen T – componentes indistinguibles y por lo tanto la red no es SSD.

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}
t_1											
t_2	x										
t_3	x	x									
t_4	x	x	x								
t_5		x	x	x							
t_6	x	x	x	x	x						
t_7	x	x	x	x	x	x					
t_8	x	x	x	x	x	x	x				
t_9	x	x	x	x	x		x	x			
t_{10}	x	x	x	x	x	x	x		x		
t_{11}	x	x	x	x	x		x	x		x	

Figura 3.4: Arreglo H inicial.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}
t_1											
t_2	x										
t_3	x	x									
t_4	x	x	x								
t_5	x	x	x	x							
t_6	x	x	x	x	x						
t_7	x	x	x	x	x	x					
t_8	x	x	x	x	x	x	x				
t_9	x	x	x	x	x	x	x	x			
t_{10}	x	x	x	x	x	x	x		x		
t_{11}	x	x	x	x	x	x	x	x		x	

Figura 3.5: Arreglo H final. Este permanece sin cambios cuando el Algoritmo 48 es ejecutado.

3.4. Secuencia Detectabilidad Estructural en GM

Los grafos marcados binarios, a diferencia de las ME binarias, exhiben concurrencia. Cuando dos lugares $p_x \in \bullet t_i$, $p_k \in \bullet t_j$, tal que $t_i \approx_I t_j$, pertenecen a diferentes P -*semiflujos*, entonces existe un marcado donde dichas transiciones pueden estar simultáneamente habilitadas. Por lo tanto, si existen transiciones $t_i \approx_I t_j$ y no existe un P -*semiflujo* mínimo Y_K tal que $\bullet t_i, \bullet t_j, t_i^*, t_j^* \subseteq \langle Y_K \rangle$, es decir si ambas transiciones pueden ser disparadas concurrentemente, entonces las dos secuencias de disparo de transiciones $t_i t_j \sigma$, $t_j t_i \sigma$ son indistinguibles entre sí y el GM no es SSD . Esta sección formaliza y caracteriza este caso.

Ejemplo 53. *Considérese la red Q mostrada en la Figura 3.6 donde*

$$\Phi = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

CAPÍTULO 3. CARACTERIZACION DE LA SSD

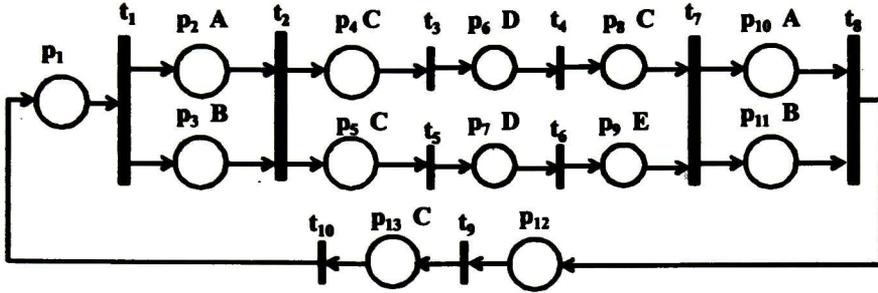


Figura 3.6: Una RPI binaria y viva que contiene paralelismo, la cual no es SSD.

Note que debido a que en un GM $|p_i| = |p_i^*| = 1$, las relaciones en las Definiciones 43 y 44 no son aplicables. Sin embargo, puesto que $t_3 \approx_I t_5$ y t_3, t_5 pertenecen a diferentes P -componentes, entonces las secuencias $t_3 t_5 \sigma$, $t_5 t_3 \sigma$ son indistinguibles entre sí. Por lo tanto Q no es SSD.

Con el fin de abordar el problema de la concurrencia, la siguiente relación es presentada.

Definición 54. Relación en concurrencia (\approx_p). Sea Q una RPI, y $t_i, t_j \in T$, $i \neq j$, tal que $t_i \approx_I t_j$. Si $\nexists P$ -semiflujo mínimo Y_k tal que ${}^*t_i, {}^*t_j, t_i^*, t_j^* \subseteq \langle Y_k \rangle$ entonces $t_i \approx_p t_j$.

En palabras más simples, dos transiciones indistinguibles están relacionadas en concurrencia si éstas pueden habilitarse simultáneamente.

Note que \approx_p , a diferencia de la relación \approx^+ o de la relación \approx^- , siempre relaciona transiciones que pertenecen a diferentes P -componentes mínimos. El siguiente teorema establece que la relación \approx_p es suficiente y necesaria para determinar la secuencia detectabilidad estructural en un GM.

Teorema 55. Sea (Q, μ_0) un GM vivo, binario y puro. Entonces Q es estructuralmente secuencia detectable si y sólo si $\forall t_i, t_j \in T, i \neq j, t_i \not\approx_p t_j$ y $T_i \not\approx_I T_j$, donde T_i representa el T -componente inducido por el único T -semiflujo X_i .

Demostración. (\rightarrow) En el desarrollo de esta demostración se procede usando la forma contrapositiva del enunciado del Teorema 55.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Suponga que $t_i \approx_p t_j$ o que el único T - componente es indistinguible con respecto de sí mismo. Entonces dos casos son posibles:

Caso (a) : Debido a que $t_i \approx_p t_j$ entonces no existe P - *semiflujo* mínimo Y_k tal que ${}^*t_i, {}^*t_j, t_i^*, t_j^* \subseteq \langle Y_k \rangle$. Además, debido a que Q es un GM , entonces $p_i \in {}^*t_i, p_j \in {}^*t_j$ pueden ser marcados simultáneamente en el mercado M_k . Como $\Phi C(\bullet, t_i) = \Phi C(\bullet, t_j)$ entonces no es posible conocer cual transición t_i o t_j fue disparada, de ahí que las secuencias de disparo de transiciones $t_i t_j \sigma$ o $t_j t_i \sigma$ pueden ocurrir, donde σ es una secuencia de longitud arbitrariamente larga y existe por el hecho de que Q es viva. Las secuencias $t_i t_j \sigma$ y $t_j t_i \sigma$ son indistinguibles entre sí, por lo tanto el GM no es SSD .

Caso (b) : Si el único T - componente es indistinguible con respecto de sí mismo, entonces hay dos secuencias de disparo de transiciones que son indistinguibles entre sí, por lo tanto Q no es SSD .

(\leftarrow) Debido a que Q es un GM entonces el soporte de su T - *semiflujo* mínimo es el conjunto de todas sus transiciones. Suponga que Q no es SSD (note que la relación \approx^+ ni la relación \approx^- existen en un GM , debido a que las relaciones \approx^+ y \approx^- requieren lugares con más de un arco de entrada o salida). Entonces existen dos secuencias indistinguibles $\sigma_i = \alpha_i t_i \beta_i, \sigma_j = \alpha_j t_j \beta_j$. Sin pérdida de generalidad, suponga que el sistema es analizado después del disparo de la secuencias α_i, α_j . Los siguientes dos casos son posibles.

a) t_i, t_j están habilitadas en el mismo marcado, entonces existen dos transiciones t_i, t_j indistinguibles $t_i \approx_p t_j$ relacionadas.

b) t_i, t_j no están habilitadas en el mismo marcado. Debido a que $t_i \beta_i$ es indistinguible de $t_j \beta_j$, entonces $t_i \approx_I t_j$, al igual que $t_{i+1} \approx_I t_{j+1}$ (La primera transición en β_i y la primera en β_j , respectivamente), y así sucesivamente. Puesto que el número de transiciones es finito y existe solamente un T - *semiflujo*, entonces $t_i \beta_i$ representa el mismo T - *semiflujo* que $t_j \beta_j$. Por ello, este T - componente es indistinguible con respecto de sí mismo, es decir, no hay transiciones distinguibles. Por lo tanto cualquier transición t_i es indistinguible de al menos otra transición t_j de este T - componente. ■

En un GM el número de P - *semiflujos* mínimos crece exponencialmente, por lo que las condiciones del teorema anterior parecen ser muy complejas de probar. La condición $\forall t_i, t_j \in T, i \neq j, t_i \not\approx_p t_j$ puede ser probada demostrando que no existe un par de transiciones $t_i \approx_p t_j$ relacionadas. El siguiente algoritmo, basado en un problema de programación lineal, prueba si existe un par de transiciones $t_i \approx_p t_j$ relacionadas.

Algoritmo 56. *Entrada:* Una RPI $Q = \{P, T, \Phi, F, W\}$

Salida: el conjunto $Gr(\approx_p)$

Calcular $Gr(\approx_p) = \emptyset$

Para todo $(t_i, t_j) \in Gr(\approx_I)$ hacer

$$\text{Min} \sum_{i=1}^{|P|} Y(i)$$

s.a.

$$Y^T C = \vec{0}$$

$Y \geq \vec{0}$ - Y es un P -semi flujo

$$Y(p_a) = 1, p_a \in \bullet t_i$$

$$Y(p_b) = 1, p_b \in \bullet t_j$$

$0 \leq Y(p_k) \leq 1$, donde p_k es cualquier lugar diferente a p_a y p_b .

Si este problema no tiene solución,

entonces incluir (t_i, t_j) en $Gr(\approx_p)$.

De lo contrario no incluir (t_i, t_j) en $Gr(\approx_p)$.

finalizar hacer.

Observe que el Algoritmo 56 calcula todo el conjunto transiciones relacionadas en concurrencia, resultado que se utilizará en el Capítulo 5. Para el caso de verificar la SSD se le puede implementar un condicional al algoritmo para que en cuanto el conjunto $Gr(\approx_p) \neq \emptyset$ ya no continúe e indique que sí hay transiciones relacionadas en concurrencia, por lo que la red no es SSD.

Cabe señalar que la complejidad teórica del algoritmo anterior es $|Gr(\approx_I)|^3$ y la complejidad de calcular el conjunto $Gr(\approx_I)$ es $|T|^2$.

Proposición 57. *El Algoritmo 56 determina la existencia de transiciones relacionadas en concurrencia.*

Demostración. Observe que cada P -semi flujo mínimo Y de un GM satisface que $\forall p_i \in \langle Y \rangle, |\bullet p_i| = 1$. Además cada P -componente inducida por Y es una ME. Por lo tanto $\forall t_j$ de esta ME $|t_j^\bullet| = 1$.

Las condiciones $Y^T C = \vec{0}$, $Y \geq \vec{0}$ en el algoritmo indican que Y es un P -semi flujo. Las condiciones $Y(p_a) = 1, p_a \in \bullet t_i, Y(p_b) = 1, p_b \in \bullet t_j$ garantizan que los lugares p_a y p_b que se encuentran en la entrada de las transiciones indistinguibles pertenecen al mismo P -semi flujo. La condición $0 \leq Y(p_k) \leq 1$ asegura que el resto de los lugares p_k aparecerán una sola vez en el soporte del P -semi flujo. Por lo tanto el P -semi flujo es mínimo. ■

CAPÍTULO 3. CARACTERIZACION DE LA SSD

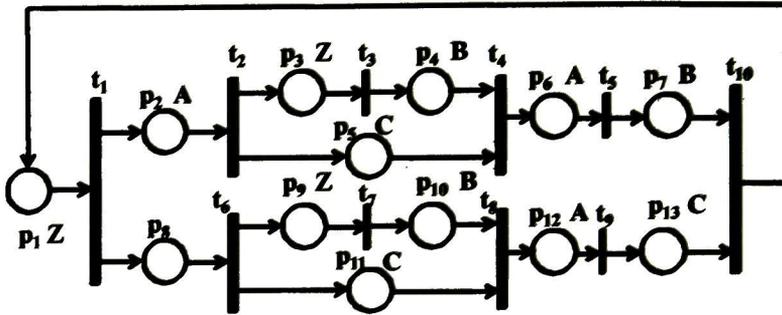


Figura 3.7: Un GM que no es estructuralmente secuencial detectable.

Ejemplo 58. Sea Q un GM vivo, binario y puro, dibujado en la Figura 3.7, donde

$$\Phi = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

A través de este ejemplo se aplicará el Algoritmo 56. Primeramente se requiere del conjunto de las transiciones indistinguibles $Gr(\approx_I) = \{(t_2, t_6), (t_3, t_7), (t_4, t_8), (t_6, t_2), (t_7, t_3), (t_8, t_4)\}$. Del conjunto $Gr(\approx_I)$ se selecciona el par ordenado (t_2, t_6) y se tratará de resolver el problema de programación lineal propuesto, cuya solución tratará de encontrar un P -semi flujo donde $Y(p_2) = 1, Y(p_6) = 1$ y $Y(p_k)$ donde k representa un número entero diferente de 2 y de 6 para este caso. Puesto que el problema de programación lineal no tiene solución entonces para las transiciones indistinguibles t_2 y t_6 , no existe P -semi flujo mínimo Y_k tal que ${}^*t_2, {}^*t_6, t_2^o, t_6^o \subseteq \langle Y_k \rangle$, por lo cual $Gr(\approx_p) = \{(t_2, t_6)\}$. Tras ejecutar el problema de programación lineal para el resto de los pares ordenados de $Gr(\approx_I)$ se concluye que $Gr(\approx_p) = Gr(\approx_I)$. Como el conjunto $Gr(\approx_p) \neq \emptyset$ entonces existen transiciones \approx_p relacionadas. De hecho, con el primer par ordenado de transiciones se concluye que $t_2 \approx_p t_6$ con lo cual ya no era necesario finalizar el cálculo de $Gr(\approx_p)$. Por lo tanto, como existen transiciones relacionadas en concurrencia Q no SSD.

Ejemplo 59. Sea Q un GM binario, vivo y puro, representada en la Figura 3.8,

CAPÍTULO 3. CARACTERIZACION DE LA SSD

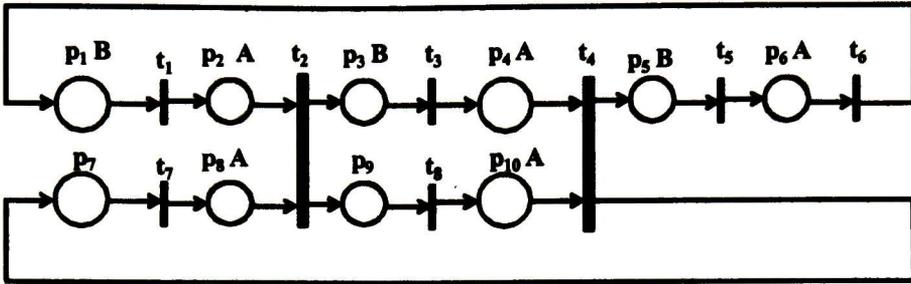


Figura 3.8: Un GM que es SSD.

donde

$$\Phi = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, y$$

$$\Phi C = \begin{bmatrix} 1 & -2 & 1 & -2 & 1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 & -1 & 1 & 0 & 0 \end{bmatrix}$$

Ejecutar el Algoritmo 56 implica buscar solución al problema de programación lineal propuesto para cada par ordenado del conjunto $Gr(\approx_I) = \{(t_1, t_3), (t_1, t_5), (t_3, t_1), (t_3, t_5), (t_5, t_1), (t_5, t_3), (t_2, t_4), (t_4, t_2), (t_7, t_8), (t_8, t_7)\}$. Para las transiciones t_1 y t_3 se concluye que existe el P -semi flujo $Y_1^T = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0]$ tal que $Y(p_3) = 1$ y $Y(p_1) = 1$. El P -semi flujo Y_1^T también es solución para los pares $(t_1, t_5), (t_3, t_5)$ (y sus recíprocos) por lo que $\bullet t_1, \bullet t_3, \bullet t_5, \bullet t_1^*, \bullet t_3^*, \bullet t_5^* \subseteq \langle Y_1 \rangle$. También, para las transiciones indistinguibles t_7 y t_8 existe el P -semi flujo $Y_2^T = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$ tal que $\bullet t_8, \bullet t_9, \bullet t_8^*, \bullet t_9^* \subseteq \langle Y_2 \rangle$. Lo mismo se puede concluir acerca de las transiciones indistinguibles t_2 y t_4 que existe el P -semi flujo Y_2 , el cual satisface que $\bullet t_2, \bullet t_4, \bullet t_2^*, \bullet t_4^* \subseteq \langle Y_2 \rangle$. Debido a que $\forall t_i, t_j \in T, i \neq j$, tal que $t_i \approx_I t_j$, existe P -semi flujo mínimo Y_k , tal que $\bullet t_i, \bullet t_j, \bullet t_i^*, \bullet t_j^* \subseteq \langle Y_k \rangle$, entonces Por lo tanto $Gr(\approx_p) = \emptyset$, es decir $t_i \not\approx_p t_j$ y debido a que la transición t_6 es distinguible del resto, entonces el único T -componente es distinguible respecto de sí mismo. Por lo tanto Q es SSD.

3.5. Secuencia Detectabilidad Estructural en LE

En la sección 3.3, la secuencia detectabilidad estructural en ME fuertemente conexas, vivas y binarias fue abordada. Debido a que las máquinas de estado no exhiben concurrencia, entonces las relaciones \approx^+ , \approx^- y el análisis de T – *componentes* son necesarios y suficientes para caracterizar la secuencia detectabilidad estructural. En GM fuertemente conexas, vivas y binarios, sin embargo, existen P – *semiflujos* evolucionando en concurrencia, entonces la relación \approx_p fue introducida para caracterizar la secuencia detectabilidad estructural en dicha clase de RPI . En lo que respecta a las LE fuertemente conexas y binarias, existen P – *semiflujos* y T – *semiflujos* evolucionando simultáneamente. Por lo tanto, con el fin de caracterizar la secuencia detectabilidad estructural en este tipo de redes algunas consideraciones adicionales deben ser tomadas en cuenta.

Ejemplo 60. Considere la LE fuertemente conexas, binaria y pura dibujada en la Figura 3.9, donde

$$\Phi = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Note que la LE mostrada en la Figura 3.9 no tiene transiciones relacionadas de la forma \approx^+ , \approx^- o \approx_p . Sin embargo, la LE no es SSD . La razón de esto es que existe un T – *componente*, formado por el conjunto de lugares $\{p_3, p_{13}\}$ y el conjunto de transiciones $\{t_{12}, t_{13}\}$, cuyas transiciones pueden dispararse en paralelo con algunas de las transiciones del P – *componente* formado por el conjunto de lugares $\{p_1, p_2, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}, p_{11}, p_{12}\}$, y el conjunto de transiciones $\{t_1, \dots, t_{11}, t_{15}\}$. Debido a que $t_6 \approx_I t_7$ y puesto que éstas pertenecen

CAPÍTULO 3. CARACTERIZACION DE LA SSD

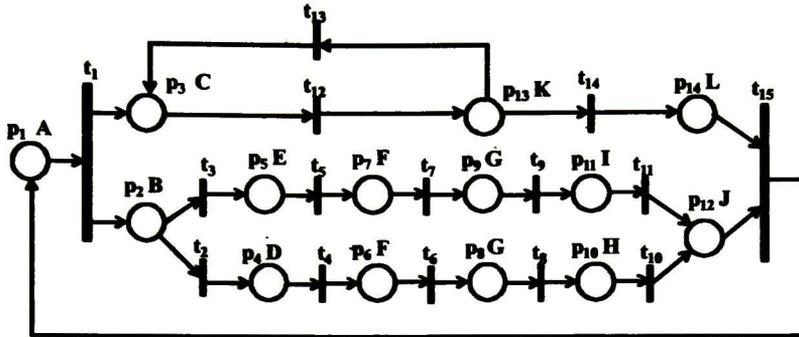


Figura 3.9: Una RPI de libre elección que no es SSD.

al P - componente anterior, entonces las secuencias de disparo de transiciones $\sigma_1 = t_6 t_{12} t_{13} t_{12} t_{13} \dots$, disparable desde el marcado $M_0 = [0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$, y $\sigma_2 = t_7 t_{12} t_{13} t_{12} t_{13} \dots$, disparable desde el marcado $M'_0 = [0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$, son indistinguibles entre sí.

Por lo tanto, las LE que son SSD no exhiben:

- Transiciones relacionadas de la forma \approx^+ , \approx^- y \approx_p ,
- T - componentes cuyas transiciones pueden habilitarse en concurrencia con transiciones indistinguibles pertenecientes al mismo P - semiflujo ni
- T - componentes indistinguibles entre sí.

Cabe señalar que las cinco condiciones previamente establecidas son independientes entre sí. Las RPIs expuestas en la Figura 3.10 muestran este hecho. La RPI de la Figura 3.10.a) captura la relación hacia adelante, en este caso $t_3 \approx^+ t_6$. La RPI de la Figura 3.10.b) captura la relación hacia atrás, en este caso $t_1 \approx^- t_4$. La RPI de la Figura 3.10.c) captura la relación en concurrencia, en este caso $t_3 \approx_p t_6$. La RPI de la Figura 3.10.d) cuenta con transiciones indistinguibles t_2, t_4 que pueden estar habilitadas en concurrencia con transiciones del T - semiflujo $X = \overrightarrow{t_8 t_9}$ (es decir, hay suficientes marcas para disparar simultáneamente las transiciones t_2 o t_4 y la secuencia de transiciones $t_8 t_9$). En las RPIs de la Figura 3.10 e) y f) existen T - componentes indistinguibles. En la Figura 3.10.e) $X_1 = \overrightarrow{t_5 t_6}$

CAPÍTULO 3. CARACTERIZACION DE LA SSD

y $X_2 = \overrightarrow{t_7 t_8}$ son los T - *semiflujos* que generan a estos T - *componentes*. En la Figura 3.10.f) el único T - *componente* es indistinguible con respecto de sí mismo, por ejemplo las permutaciones $\sigma_1 = (t_1 t_2 t_3 t_4)^*$ y $\sigma_2 = (t_3 t_4 t_1 t_2)^*$ son indistinguibles entre sí. En total 2^5 combinaciones son posibles. El siguiente teorema caracteriza la propiedad de secuencia detectabilidad estructural en $RPIs$ de la clase de las LE .

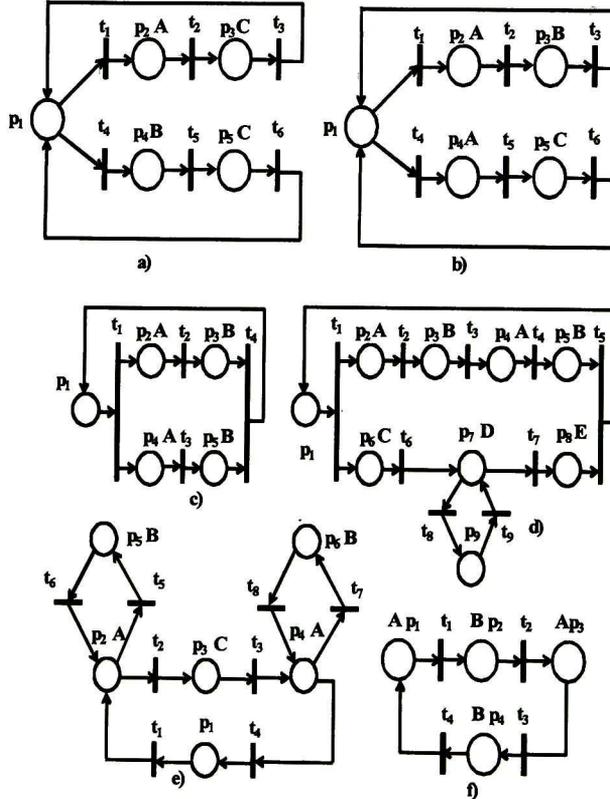


Figura 3.10: Esta figura presenta ejemplos para analizar la independencia de las condiciones que caracterizan a la SDD .

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Teorema 61. Sea (Q, μ_0) una RPI viva, binaria y pura. La RPI Q es SSD si y sólo si las siguientes condiciones son satisfechas:

1. $\forall t_i, t_j \in T, i \neq j, t_i \not\approx^+ t t_j,$
2. $\forall t_i, t_j \in T, i \neq j, t_i \not\approx^- t_j,$
3. $\forall t_i, t_j \in T, i \neq j, t_i \not\approx_p t_j,$
4. No existen $(t_i, t_j) \in Gr(\approx_I)$ y t_i, t_j están habilitadas en concurrencia con transiciones de $T - semi$ flujos.
5. No existen $T - componentes$ indistinguibles.

Demostración. En el desarrollo de esta demostración se procede usando la forma contrapositiva del enunciado del Teorema 61.

(\rightarrow) Si existen $t_i, t_j \in T, i \neq j, t_i \approx^+ t t_j$ en una RPI, entonces por la Definición 44 $t_i^* = t_j^*$. Debido a que Q es viva y binaria, entonces existen dos marcados iniciales diferentes $M_0, M'_0 \in \mu_0$ que permiten habilitar t_i, t_j respectivamente. En M_0 los lugares de entrada a t_i están marcados. M'_0 es muy similar a M_0 , la diferencia es que las marcas en los lugares de entrada a t_i son removidas y se colocan en los lugares de entrada a t_j . Por lo tanto, $M_0 \xrightarrow{t_i} M_1 \xrightarrow{\sigma_1} M_k$ y $M'_0 \xrightarrow{t_j} M_1 \xrightarrow{\sigma_1} M_k$. Las secuencias $t_i\sigma_1, t_j\sigma_1$ son indistinguibles entre sí, por lo tanto Q no es SSD.

Si $t_i \approx^- t_j$, entonces ${}^*t_i = {}^*t_j$. Debido a que Q es viva entonces existe una secuencia de transiciones disparable σ a partir de M_0 que permite alcanzar el marcado M_k donde los lugares ${}^*t_i = {}^*t_j$ se marcan. Puesto que Q es viva, entonces existe otra secuencia de disparo arbitrariamente larga β tal que $M_k \xrightarrow{\beta} M_q$, donde en M_q se encuentran nuevamente habilitadas las transiciones t_i y t_j . Entonces la secuencia $\sigma\beta t_i$ es indistinguible de $\sigma\beta t_j$. Por lo tanto Q no es SSD.

Si $t_i \approx_p t_j$ y como Q es una red viva, entonces existe un marcado tal que $M_0 \xrightarrow{t_i t_j} M_1 \xrightarrow{\sigma_1}$ y $M_0 \xrightarrow{t_j t_i} M_1 \xrightarrow{\sigma_1}$, es decir, $t_i t_j \sigma_1$ es indistinguible de la secuencia $t_j t_i \sigma_1$, por lo tanto Q no es SSD.

Si existen $T - componentes$ indistinguibles, entonces existen dos secuencias de disparo de transiciones de longitud arbitrariamente larga que son indistinguibles, por lo tanto Q no es SSD.

Si existe $(t_i, t_j) \in Gr(\approx_I)$ y t_i, t_j pueden estar habilitadas en concurrencia con las transiciones de un $T - semi$ flujo, entonces la secuencia de disparo de

CAPÍTULO 3. CARACTERIZACION DE LA SSD

transiciones $t_i\sigma$ es indistinguible de $t_j\sigma$ (donde σ es una secuencia generada por el disparo de las transiciones del T – *semiflujo*), por lo que Q no es *SSD*.

(←) Suponga que Q no es *SSD*, entonces existen dos secuencias de disparo de transiciones de longitud arbitrariamente larga $\sigma_1, \sigma_2, \sigma_1 \neq \sigma_2$, tales que $\sigma_1 \approx_I \sigma_2$. Podrían darse los siguientes tres casos 1) la secuencia σ_1 es completamente diferente de σ_2 , 2) σ_1, σ_2 tienen una subsecuencia común o 3) son iguales. Éste último caso no es relevante para el estudio de la secuencia detectabilidad estructural porque se necesitan secuencias de disparo de transiciones diferentes. Por lo tanto, hay que enfocarse en los primeros dos casos.

Si las secuencias son completamente diferentes entre sí, entonces la *RPI* presenta T – *componentes* indistinguibles.

Si $\sigma_1 = \dots t_a \alpha_1 \dots, \sigma_2 = \dots t_b \alpha_1 \dots$ (es decir, comparten subsecuencias en común), entonces, a) t_a y t_b pertenecen al mismo P – *componente*, entonces $t_a \approx^+ t_b$ o hay T – *componentes* indistinguibles; b) si t_a, t_b pertenecen a diferentes P – *componentes* entonces $t_a \approx_p t_b$, o bien, existe $(t_i, t_j) \in Gr(\approx_I)$ y t_i, t_j pueden estar habilitadas en concurrencia con las transiciones de un T – *semiflujo*, si α_1 es arbitrariamente larga.

Si $\sigma_1 = \dots \alpha_1 t_a \dots, \sigma_2 = \dots \alpha_1 t_b \dots$ (es decir, comparte subsecuencias en común), entonces como t_a, t_b pertenecen al mismo P – *componente*, entonces $t_a \approx^- t_b$ o existen T – *componentes* indistinguibles; b) si t_a, t_b pertenecen a diferentes P – *componentes* entonces $t_a \approx_p t_b$, o bien, existe $(t_i, t_j) \in Gr(\approx_I)$ y t_i, t_j pueden estar habilitadas en concurrencia con las transiciones de un T – *semiflujo*, si α_1 es arbitrariamente larga. ■

La condición 4) del teorema anterior podría sugerir (en primer instancia) que se requiere del cómputo de todos los T – *componentes* y P – *componentes* dando lugar a algoritmos muy complejos. Afortunadamente, la condición puede ser probada en tiempo polinomial si, en lugar de buscar que no existe un T – *semiflujo* cuyas transiciones pueden estar habilitadas en concurrencia con las transiciones de un P – *componente*, se busca que sí existe un T – *semiflujo* cuyas transiciones pueden habilitarse en paralelo con las transiciones de un P – *Componente*, labor que realiza el siguiente algoritmo.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Algoritmo 62. *Calcula si existe $(t_a, t_b) \in Gr(\approx_I)$ y t_a, t_b pueden habilitarse en concurrencia con algún T – semi flujo*

Entrada: Una RPI de Libre elección $Q = \{P, T, \Phi, F, W\}$

Salida: Si existe $(t_a, t_b) \in Gr(\approx_I)$ tal que t_a, t_b pueden habilitarse en concurrencia con algún T – semi flujo

Comienzo

1. *Calcular el conjunto $P_{FJ} = \{(p_i, t_F, t_J) \mid p_i \text{ está residendo dentro de un par de transiciones fork-join } t_F, t_J \text{ y } |p_i^\circ| > 1\}$. Si $P_{FJ} = \emptyset$ entonces no continuar con el algoritmo e indicar que las transiciones indistinguibles t_a, t_b no están habilitadas en paralelo con las transiciones de un T – semi flujo X_a , de lo contrario continuar con el siguiente paso.*
2. *Calcular el conjunto $P_{FJ}^{X_a} = \{(p_i, t_F, t_J, X_a) \mid (p_i, t_F, t_J) \in P_{FJ} \text{ y existe un } T \text{ – semi flujo } X_a \text{ tal que } X_a(t_F) = X_a(t_J) = 0 \text{ y } X_a(t_k) = 1, \text{ donde } t_k \in \bullet p_i\}$, es decir, si existe un T – semi flujo X_a que también resida dentro del par de transiciones fork-join t_F, t_J . Si el conjunto $P_{FJ}^{X_a}$ no puede ser calculado, entonces detener el algoritmo e indicar que las transiciones indistinguibles t_a, t_b no están evolucionando en paralelo con T – semi flujos, de lo contrario continuar con el siguiente paso.*
3. *Calcular si existen dos transiciones indistinguibles que puedan ser habilitadas en concurrencia con un T – semi flujo. Debido a que $(p_i, t_F, t_J, X_a) \in P_{FJ}^{X_a}$ contiene el T – semi flujo X_a que reside dentro de un par de transiciones fork-join, entonces las transiciones indistinguibles t_a y t_b , $(t_a, t_b) \in Gr(\approx_I)$, pueden estar habilitadas en concurrencia con las transiciones de X_a si los lugares de entrada de t_a, t_b pueden ser marcados simultáneamente con el lugar p_i (es decir, el lugar p_i tiene que pertenecer a un P – semi flujo mínimo diferente que los lugares $p_a \in \bullet t_a$ y $p_b \in \bullet t_b$ porque la RPI es segura). El siguiente problema de programación lineal determina si el lugar de selección p_i está en un P – componente diferente de aquellos que contienen a t_a y t_b . Note que el cálculo de un P – semi flujo que contenga el lugar p_i implica que $Y^T C = 0, Y \geq 0, Y(p_i) \geq 1, Y(p_a) = Y(p_b) = 0$ para aquellos lugares de entrada a t_a y t_b y el valor de Y puede ser un número real. El cálculo de este P – semi flujo, si existe, es efectuado en tres pasos. Primero un P – semi flujo Y_1 que contenga el lugar de selección p_i y los lugares de entrada a t_a es calculado. Después, un P – semi flujo Y_2 que contenga el lugar de selección p_i y los lugares de entrada a t_b es calculado. Si tanto Y_1*

CAPÍTULO 3. CARACTERIZACION DE LA SSD

como Y_2 existen, entonces el P -semiflujo $Y_G = Y_1 + Y_2$ es calculado. Note que el P -semiflujo Y_G es una combinación lineal de P -semiflujos y Y_G tiene la característica de que contiene tanto a p_i como a los lugares de entrada de t_a y t_b . Entonces un nuevo P -semiflujo Y_3 , incluido en Y_G , conteniendo los lugares de entrada a t_a y t_b donde $Y(p_i) = 0$ es calculado. Si tal P -semiflujo Y_3 es encontrado entonces las transiciones t_a y t_b pueden estar habilitadas en concurrencia con las transiciones del T -semiflujo X_a . Estos hechos se detallan en las siguientes líneas.

Para cada $(p_i, t_F, t_J, X_a) \in P_{FJ}^{X_a}$ hacer

seleccionar el lugar de selección p_i ,

Para cada $(t_a, t_b) \in Gr(\approx_I)$ hacer

seleccionar t_a, t_b

$\text{Min } \sum_{i=1}^{ P } Y_1(i)$	$\text{Min } \sum_{i=1}^{ P } Y_2(i)$
s.a.	s.a.
$Y_1^T C = 0$	$Y_2^T C = 0$
$Y_1(p_i) \geq 1$	$Y_2(p_i) \geq 1$
$Y_1(p_k) \geq 0, p_k \in P - \{p_i\}$	$Y_2(p_k) \geq 0, p_k \in P - \{p_i\}$
$\sum Y_1(p_x) = 0, p_x \in {}^*t_a$	$\sum Y_2(p_x) = 0, p_x \in {}^*t_b$

finalizar ciclo

finalizar ciclo /* Si Y_1 o Y_2 no son vacíos, entonces existen P -semiflujos mínimos que contienen los lugares $p_i, {}^*t_a, {}^*t_b$ o éstos están en dos P -semiflujos mínimos diferentes que no comparten lugares, el siguiente problema de programación lineal determina cuál de éstos es el caso. Si Y_1 y Y_2 son vacíos (ambos problemas no tienen solución), entonces no hay P -semiflujos que contienen los lugares $p_i, {}^*t_a, {}^*t_b$ */

Si ambos problemas tienen solución, entonces calcular $Y_G = Y_1 + Y_2$ y

CAPÍTULO 3. CARACTERIZACION DE LA SSD

$\text{Min } \sum_{i=1}^{ P } Y_3(i)$
s.a.
$Y_3^T C = 0$
$Y_3(p_i) = 0$, p_i es el lugar de selección es elegido
$Y_3(p_k) \geq 0$, p_k es cualquier lugar diferente de p_i
$\sum Y_3(p_x) \geq 1$, donde p_x son los lugares de entrada a t_a
$\sum Y_3(p_z) \geq 1$, donde p_z son los lugares de entrada a t_b
$\{t_a, t_b\} \subseteq {}^* \langle Y_3 \rangle \cap \langle Y_3 \rangle^*$
$Y_3(i) \leq Y_G(i)$

/ si Y_3 no es vacío, entonces los lugares p_i , *t_a , *t_b están en diferentes P – semi flujos que no comparten lugares*/*

- 4) Si Y_1 y Y_2 son vacíos o si Y_3 no es vacío entonces existe un $(t_i, t_j) \in Gr(\approx_I)$ tal que t_i, t_j pueden estar habilitadas en concurrencia con las transiciones de T – semi flujos

Fin

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Proposición 63. *Dada una RPI de la clase de las LE, la existencia de $(t_i, t_j) \in Gr(\approx_I)$ tal que t_i, t_j pueden estar habilitadas en concurrencia con las transiciones de $T - semi$ flujos puede ser determinado usando el Algoritmo 62.*

Demostración. En una RPI de libre elección viva y acotada (binaria en este caso), dos lugares pueden estar marcados simultáneamente en el mismo marcado si y sólo si pertenecen a diferentes $P - semi$ flujos [12]. De acuerdo con los teoremas de las $S - coberturas$ y $T - coberturas$ [12], los $P - semi$ flujos diferentes son generados por transiciones t_i tales que $|t_i^*| > 1$ (es decir, transiciones del tipo fork). Por lo tanto, si dos lugares son marcados simultáneamente, éstos deben de pertenecer a diferentes caminos generados a partir de la transición del tipo fork. Los teoremas de las $T - coberturas$ garantizan que los caminos generados a partir de la transición tipo fork se unirán por una transición t_j tal que $|t_j^*| > 1$ con lo cual el par de transiciones fork-join es formado. Por lo tanto, el algoritmo calcula los lugares de selección $|p_i^*| > 1$ que residen dentro de un par de transiciones fork-join. Los lugares de selección tienen más de una transición de salida, motivo por el cual éstos son los únicos candidatos para poder generar $T - semi$ flujos dentro de un par de transiciones fork-join (el resto de los lugares únicamente tienen una transición de salida, las cuales junto con la transición del tipo join pertenecen al mismo $T - semi$ flujo). Por lo tanto, el algoritmo hace una búsqueda de los lugares p_i tales que $|p_i^*| > 1$ y al menos una de sus transiciones de salida de este lugar esté incluida en un $T - semi$ flujo que no contenga a la transición del tipo join. El conjunto de lugares p_i es calculado como el conjunto $P_{FJ}^{X_a}$ en el paso 2). Ahora bien, los $T - semi$ flujos X_a pueden ser disparados en concurrencia con las transiciones que pertenecen a los diferentes caminos generados por la transición t_F del tipo fork. Por lo cual, las transiciones en diferentes caminos generados a partir de la transición t_F del tipo fork tienen que incluir transiciones indistinguibles t_a, t_b para garantizar que la RPI de libre elección no es SSD, de lo contrario será SSD si también se satisfacen el resto de las condiciones del Teorema 61. Por ello que, el algoritmo calcula (usando dos problemas de programación lineal) si existe un $P - semi$ flujo mínimo Y_G que contenga a p_i, t_a, t_b o la suma de algunos $P - semi$ flujos mínimos disjuntos (porque las entradas de Y_G son cero o uno) conteniendo a p_i, t_a, t_b . Si dichos $P - semi$ flujos no son encontrados, entonces existe una $(t_i, t_j) \in Gr(\approx_I)$ tal que t_i, t_j está evolucionando en concurrencia con $T - semi$ flujos. Si tales $P - semi$ flujos son encontrados, entonces un tercer problema de programación lineal usa a Y_G para determinar si los lugares p_i, t_a, t_b están en diferentes $P - semi$ flujos mínimos y no comparten lugares, si dicho $P - semi$ flujo existe entonces los lugares p_i, t_a, t_b están en $P - semi$ flujos

CAPÍTULO 3. CARACTERIZACION DE LA SSD

mínimos diferentes y no comparten lugares. ■

Proposición 64. *Dada una RPI de la clase de las LE, la existencia de un T – semi flujo evolucionando en concurrencia con un P – componente puede ser determinado en tiempo polinomial.*

Demostración. Usando el Algorithm 62, el cálculo de P_{FJ} es efectuado en $|P| \times |P + T|$ pasos. El cálculo del conjunto $P_{FJ}^{X_0}$ es efectuado en $|P_{FJ}| \times |T|^3$ pasos. El cálculo de $Gr(\approx_I)$ llevado a cabo en $|T|^2$ pasos. El paso 3) del algoritmo es calculado en $|P_{FJ}^{X_0}| \times |Gr(\approx_I)| \times 3|P|^3$. Por lo tanto, el algoritmo corre en $|P| \times |P + T| + |P_{FJ}| \times |T|^3 + |T|^2 + |P_{FJ}^{X_0}| \times |Gr(\approx_I)| \times 3|P|^3$ ■

Un algoritmo de tiempo polinomial más detallado para efectuar el cómputo del paso 1) del algoritmo anterior es presentado en el Apéndice A.

Cabe señalar que la complejidad de los problemas de programación lineal fue tomada en cuenta en la demostración de la Proposición 64. Para conocer más al respecto de la complejidad de los problemas de programación lineal un lector interesado puede consultar [24].

En el ejemplo presentado a continuación se aplica el algoritmo anteriormente expuesto.

Ejemplo 65. *Retomando la RPI Q de la Figura 3.9 se aplicará paso a paso el Algoritmo 62 para corroborar la respuesta presentada en el Ejemplo 60.*

En el primer paso del Algoritmo 62, se selecciona el lugar de selección p_{13} y utilizando el Algoritmo 94, propuesto en el Apéndice A, se determina si p_{13} se encuentra dentro de un par de transiciones fork-join. Como p_{13} si está dentro de un par de transiciones fork-join, se construye el conjunto $P_{FJ} = \{(p_{13}, t_1, t_{15})\}$ siendo las transiciones t_1, t_{15} el par de transiciones fork-join. En el paso 2) se tiene que determinar si dentro del par de transiciones fork-join t_1, t_{15} existe un T – semi flujo. Utilizando el Algoritmo 97, propuesto en el Apéndice A, se calcula que $P_{FJ}^{X_0} = \{(p_{13}, t_1, t_{15}, X_1)\}$, donde $X_1 = t_{12}t_{13}$. Dentro del par de transiciones fork-join t_1, t_{15} se encuentran las transiciones indistinguibles t_6 y t_7 ; pero ejecutando los problemas de programación lineal, presentados en el paso 3), no se encuentra ninguna solución, es decir, como no existen P – componentes que contengan simultáneamente al lugar de selección p_{13} y a las transiciones t_6 y t_7 entonces éstas últimas pueden estar habilitadas en concurrencia con las transiciones del T – semi flujo X_1 , por lo que Q no es SSD, lo cual concuerda con lo expuesto en el Ejemplo 60.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

En lo que respecta a la condición 5) del Teorema 61, se podría tratar de la siguiente manera. Primeramente verificar si se satisfacen las condiciones 1 a la 4 del Teorema 61, si es así se tiene que hacer una búsqueda de T -*semiflujos* mínimos en el conjunto de las transiciones indistinguibles. El conjunto de transiciones se puede particionar en $T = T_{\approx_I} \cup T_d$, donde T_{\approx_I} es el dominio de la relación \approx_I o conjunto de transiciones indistinguibles y T_d es el conjunto de las transiciones distinguibles o evento detectables, ambos conjuntos se pueden obtener a partir de las columnas de la matriz ΦC (las transiciones asociadas a las columnas repetidas en ΦC forman el conjunto T_{\approx_I} y las transiciones asociadas a las columnas que no se repiten en ΦC forman el conjunto T_d). En el caso de que no existan T -*semiflujos* mínimos en el conjunto de las transiciones indistinguibles, entonces la RPI es SSD; pero en el caso de que existan hay que realizar una prueba más minuciosa para determinar si los símbolos de salida que genere un sólo T -*componente* mínimo hacen que se confunda consigo mismo. En caso de que no exista un dicho T -*componente* que se confunda consigo mismo habría determinar los T -*componentes* que puedan generar los mismos símbolos de salida (lo cual se puede hacer con un problema de programación lineal) y finalmente para los pares de T -*componentes* que generen la misma salida corroborar que ésta se produce en el mismo orden. Dicho algoritmo es expuesto a continuación.

Algoritmo 66. *Determinar a manera de suficiencia la no existencia de T -componentes indistinguibles.*

Entrada: Una RPI $Q = \{P, T, \Phi, F, W\}$

Salida: Un T -*semiflujo* cuyo soporte está formado únicamente por transiciones indistinguibles.

Calcular el conjunto T_{\approx_I} .

$$\text{Min } \sum_{i=1}^{|T|} X(i)$$

s.t.

$$CX = 0$$

$$X \geq \vec{0}$$

$$\sum X(t_k) \geq 1, \text{ donde } t_k \in T_I$$

$$\sum X(t_q) = 0, \text{ donde } t_q \in T_d$$

Si el T -*semiflujo* X es vacío, entonces con seguridad no hay T -*componentes* indistinguibles entre sí. En caso contrario hay que realizar un estudio más exhaustivo para determinar si hay o no T -*componentes* indistinguibles.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Los algoritmos que se presentan a continuación permiten probar la suficiencia y necesidad para la condición 5) del Teorema 61.

Algoritmo 67. *Determinar la existencia de un T – componente indistinguible con respecto de sí mismo.*

Entrada: Una RPI $Q = \{P, T, \Phi, W\}$ de la clase de las FC.

Salida: Si existe un T – componente indistinguible con respecto de sí mismo.

1. Calcular el conjunto T_{\approx_I} .
 2. Calcular el T – componente $\mathcal{T}_i = \{(P_i, T_i, W_i)\}$, donde $T_i \subseteq T_{\approx_I}$.
 3. Utilizar la relación de equivalencia de Nerode para verificar que cada transición del T – componente \mathcal{T}_i puede bisimular a otra del mismo T – componente, es decir que los T – componentes se relacionan si pueden generar los mismos símbolos de salida y en el mismo orden. Para más información de bisimulación revisar [11].
 4. Si cada transición de \mathcal{T}_i es bisimulable entonces el T – componente \mathcal{T}_i es indistinguible con respecto de sí mismo, de lo contrario el T – componente \mathcal{T}_i no es indistinguible con respecto de sí mismo.
-

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Algoritmo 68. *Determinar la existencia de T – componentes indistinguibles entre sí.*

Entrada: Una RPI $Q = \{P, T, \Phi, W\}$ de la clase de las FC.

Salida: Si existen T – componentes indistinguibles entre sí.

1. Calcular el conjunto T_{\approx_1} .
 2. Calcular un T – componente $T_i = \{(P_i, T_i, W_i)\}$, tal que $T_i \subseteq T_{\approx_1}$.
 3. Calcular un T – componente $T_j = \{(P_j, T_j, W_j)\}$, tal que $T_j \subseteq T_{\approx_1}$ y $X_i \neq X_j$ (es decir, los T – semiflujos que generaron a estos T – componentes tienen que ser diferentes).
 4. Utilizar la relación de equivalencia de Nerode para verificar si cada transición del T – componente T_i puede bisimilar otra transición del T – componente T_j .
 5. Si cada transición del T – componente T_i puede bisimilar otra transición del T – componente T_j entonces $T_i \approx_1 T_j$, de lo contrario $T_i \not\approx_1 T_j$.
-

La complejidad teórica de ambos algoritmos, para probar el cumplimiento de la condición 5) del Teorema 61, es de tiempo no polinomial. Sin embargo, el desempeño de estos algoritmos puede ser optimizado de la siguiente manera. En el Algoritmo 67 el número de T – semiflujos a ser probados es reducido si se agrega la restricción de que el máximo común divisor de las entradas del vector $\Phi Post X_i$ sea mayor a uno, donde $Post(i, j) = W(t_i, p_j)$. En el Algoritmo 68 el número de T – semiflujos a ser probados se reduce agregando la restricción de que los T – semiflujos X_i y X_j tienen que generar los mismos símbolos de salida (es decir, los T – semiflujos tienen la misma proyección natural).

3.6. Secuencia Detectabilidad en las RPI

Como ya se ha analizado al inicio de este capítulo, la secuencia detectabilidad es una propiedad más relajada que la secuencia detectabilidad estructural, puesto que el hecho de conocer el marcado inicial de la red facilita la verificación de tal propiedad.

Por ejemplo, para el caso de las ME no importa que presenten la relación \approx^+ ya que como se conoce el marcado inicial se sabe que transición indistinguible

CAPÍTULO 3. CARACTERIZACION DE LA SSD

es la que está habilitada. Tampoco afecta la existencia de T - componentes indistinguibles, pues nuevamente gracias a que se conoce el marcado inicial se puede saber cuál de ellos es el que está marcado. Por tal motivo la relación \approx^- es la única que ocasiona en las ME que no se puedan distinguir las secuencias de disparo de transiciones. El siguiente teorema captura lo anteriormente expuesto.

Teorema 69. *Sea (Q, M_0) una ME viva, binaria y pura. Entonces Q es SD si y sólo si $\forall t_i, t_j \in T, i \neq j, t_i \not\approx^- t_j$.*

Demostración. (\rightarrow) Suponga que $t_i \approx^- t_j$, entonces ${}^*t_i = {}^*t_j$. Debido a que Q es viva entonces existe una secuencia disparable σ a partir de marcado inicial M_0 que permite alcanzar el marcado M_k donde el lugar ${}^*t_i = {}^*t_j$ es marcado. Debido a que Q es viva existe otra secuencia de disparo de transiciones arbitrariamente larga β tal que $M_k \xrightarrow{\beta} M_q$, donde en M_q nuevamente están habilitadas t_i y t_j . Entonces la secuencia $\sigma\beta t_i$ es indistinguible de $\sigma\beta t_j$. Por lo tanto Q no es SD .

(\leftarrow) Suponga que Q no es SD , entonces existen dos secuencias de disparo de transiciones de longitud arbitrariamente larga $\sigma_1, \sigma_2 \in \mathcal{L}(Q, M_0)$, $\sigma_1 \neq \sigma_2$, tal que $\sigma_1 \approx_I \sigma_2$. Debido a que Q es una ME viva y binaria, entonces cuenta únicamente con una marca en su marcado inicial, por lo que cualquier secuencia de disparo de transiciones surge de dicho lugar marcado. Como la secuencia no puede ser la misma, entonces debe de existir un lugar de selección $|p_i^*| > 1$ para que puedan ser secuencias diferentes; pero debido a que hay dos secuencias de disparo de transiciones indistinguibles entonces se tiene que al menos existe un par de transiciones $t_i, t_j \in p_i^*$ tal que $t_i \approx_I t_j$ y como Q es una ME entonces ${}^*t_i = {}^*t_j$. Por lo tanto, $t_i \approx^- t_j$. ■

También en lo que respecta a los GM sólo se requiere de una condición para caracterizar la secuencia detectabilidad, porque el conocimiento del marcado inicial evita que el único T - *semi*flujo se pueda confundir consigo mismo, lo cual se enuncia en el siguiente teorema.

Teorema 70. *Sea (Q, M_0) un GM vivo, binario y puro. Entonces Q es SD si y sólo si $\forall t_i, t_j \in T, i \neq j, t_i \not\approx_p t_j$.*

Demostración. Similar a la demostración del Teorema 55. ■

En lo que respecta a las RPI del tipo de las LE , el hecho de que existan transiciones indistinguibles que evolucionen en concurrencia con un T - *semi*flujo no afecta a la secuencia detectabilidad, esto debido a que el marcado inicial de la red permite conocer cuál de las transiciones indistinguibles es la que está habilitada

CAPÍTULO 3. CARACTERIZACION DE LA SSD

y cuál de ellas se disparará, aún cuando se de un ciclo infinito producido por el T – *semiflujo*. En el siguiente teorema se capturan las relaciones involucradas en la secuencia detectabilidad de las LE .

Teorema 71. *Sea (Q, M_0) una RPI de la clase de las LE viva, binaria y pura. Entonces Q es SD si y sólo si las siguientes condiciones son satisfechas:*

1. $\forall t_i, t_j \in T, i \neq j, t_i \not\approx^- t_j,$
2. $\forall t_i, t_j \in T, i \neq j, t_i \not\approx_p t_j,$

Demostración. (\rightarrow) Suponga que $t_i \approx^- t_j$, entonces $\bullet t_i = \bullet t_j$. Debido a que Q es viva, entonces existe una secuencia σ disparable a partir de M_0 que permita alcanzar el marcado M_k donde el lugar $\bullet t_i = \bullet t_j$ es marcado. Debido a que Q es viva, Debido a que Q es viva existe otra secuencia de disparo de transiciones β tal que $M_k \xrightarrow{\beta} M_k$. Entonces la secuencia $\sigma\beta^*t_i$ es indistinguible de $\sigma\beta^*t_j$. Por lo tanto Q no es SD .

Ahora, suponiendo que $t_i \approx_p t_j$ y debido a que Q es viva, entonces existe una secuencia σ_1 disparable a a partir de M_0 que permite alcanzar el marcado M_k donde las transiciones t_i y t_j están simultáneamente habilitadas. Desde el marcado M_k existen dos secuencias de disparo de transiciones tales que $M_k \xrightarrow{t_i t_j} M_g \xrightarrow{\sigma_2} M_h$ and $M_k \xrightarrow{t_j t_i} M_g \xrightarrow{\sigma_2} M_h$, es decir la secuencia de disparo de transiciones $\sigma_1 t_i t_j \sigma_2$ es indistinguible de la secuencia $\sigma_1 t_j t_i \sigma_2$. Por lo tanto Q no es SD .

(\leftarrow) Suponga que Q no es SD , entonces existen dos secuencias de disparo de transiciones de longitud arbitrariamente larga $\sigma_1, \sigma_2 \in \mathcal{L}(Q, M_0)$, $\sigma_1 \neq \sigma_2$, tales que $\sigma_1 \approx_I \sigma_2$. Como se ha expuesto en demostraciones anteriores la indistinguibilidad entre σ_1 y σ_2 puede deberse principalmente a dos casos a) σ_1 es completamente diferente de σ_2 y b) tienen subsecuencias en común.

a) Si las secuencias de disparo de transiciones $\sigma_1, \sigma_2 \in \mathcal{L}(Q, M_0)$ son completamente diferentes entre sí, entonces existen T – *semiflujos* que pertenecen a diferentes P – *componentes* cuyas transiciones son indistinguibles, por lo que estas transiciones están \approx_p relacionadas.

b) Si $\sigma_1 = \dots t_a \alpha_1 \dots, \sigma_2 = \dots t_b \alpha_1 \dots$ (es decir, comparten un sufijo en común), entonces, i) si t_a, t_b pertenecen al mismo P – *componente* entonces $\exists M_k \in R(Q, M_0)$ tal que $M_k \xrightarrow{t_k \beta_1} M_l \xrightarrow{t_a}$ y $M_k \xrightarrow{t_i \beta_2} M_l' \xrightarrow{t_b}$ (una decisión que puede ser realizada en el marcado M_k), por lo tanto $t_k \approx^- t_l$, ii) si t_a, t_b pertenecen a diferentes P – *componentes* mínimos entonces $t_a \approx_p t_b$.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

Si $\sigma_1 = \dots\alpha_1 t_a \dots$, $\sigma_2 = \dots\alpha_1 t_b \dots$ (es decir, comparten un prefijo en común), entonces, *i*) si t_a, t_b pertenecen al mismo P -componente entonces $t_a \approx^- t_b$; *ii*) si t_a, t_b pertenecen a diferentes P -componentes entonces $t_i \approx_p t_j$. ■

3.7. Conclusiones

En este capítulo se presentó la caracterización de la secuencia detectabilidad estructural y de la secuencia detectabilidad desde la clase de las ME hasta las LE haciendo uso de diversas relaciones de transiciones, propuestas en este mismo capítulo.

Asimismo, se presentaron una serie de algoritmos de tiempo polinomial que permiten satisfacer algunas de las condiciones expuestas en los teoremas presentados a lo largo de este capítulo, así como la demostración de la correctitud de los mismos (es decir, que realmente llevan a cabo la tarea para la cual fueron diseñados) y un breve análisis de su complejidad.

Aunque la complejidad teórica de los algoritmos que permiten encontrar los T -componentes indistinguibles entre sí es de tiempo no polinomial, en la práctica podría no llegarse a dicho extremo porque se requeriría que los sistemas fueran altamente redundantes con respecto a los símbolos de salida que generen.

CAPÍTULO 3. CARACTERIZACION DE LA SSD

4. OBSERVADOR DE LA SECUENCIA DE TRANSICIONES DISPARADA

Resumen: en este capítulo se aborda el problema de determinar la secuencia de disparo de transiciones que ha sido disparada en una RPI que es SSD. Primeramente se define una entidad nombrada como el Observador de Vector de Parikh (PVO) el cual, como su nombre lo indica, permite conocer el vector de Parikh asociado a la secuencia de disparo de transiciones actual de una RPI. En este capítulo también se indican las reglas de evolución del PVO y la manera en que converge al vector de Parikh para la secuencia de disparo de transiciones actual, para lo cual se define la noción de error y la manera en que éste es medido.

Tomando como base el PVO, también se propone un observador para estimar la secuencia de disparo de transiciones. Dicho observador de secuencias es construido a base de RPI coloreadas, donde los colores permiten almacenar la información sobre el orden en el que fueron disparadas las transiciones, así como evitar crear una familia de redes (que es uno de los inconvenientes del PVO). Asimismo, se muestra a través de un ejemplo, las reglas de evolución y convergencia del Observador coloreado de secuencias de disparo de transiciones (CFTSO) y su comparativa con el PVO.

4.1. Introducción

En el capítulo anterior la propiedad de secuencia detectabilidad estructural ha sido caracterizada en RPI para las clases de ME, GM y LE. En este capítulo se aborda el problema de estimar la secuencia de disparo de transiciones usando la información a la salida de una RPI. Este problema es diferente del cálculo del marcado de una RPI. Por ejemplo, considere la RPI mostrada en la Figura 4.1, donde

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

t_7 , que pertenecen a otro P – componente distinto de t_{12} y t_{13} , para decidir en qué lugar había marcas si en el lugar p_6 o en el lugar p_7 .

En la literatura existen diversos trabajos enfocados diseñar en observadores para inferir el estado o marcado actual de la red, [15],[3], [16], [42]; pero existen muy pocos trabajos que han puesto sus esfuerzos en diseñar observadores que permitan conocer la secuencia de transiciones disparadas, además del marcado, de la red [44]. Sin embargo, el observador de secuencias propuesto en [44] está limitado a la clase de ME fuertemente conexas.

Con la finalidad de explicar la secuencia de disparo de transiciones observada será introducirá una entidad llamada el Observador de Vector de Parikh (PVO). Como su nombre lo indica, el PVO permite calcular el vector de Parikh de la secuencia de disparo de transiciones actual usando la información de entrada y salida de la RPI . El PVO está constituido por una familia de $RPIs$, cada una con su respectivo marcado inicial, representando un posible marcado inicial vivo y binario de la RPI . Los elementos de la familia del PVO son removidos cuando sus explicaciones de la secuencia de disparo de transiciones son inconsistentes con la salida observada. Si la RPI es SSD , entonces solamente un elemento en la familia del PVO prevalecerá y éste contendrá el vector de Parikh de la secuencia disparada.

El PVO está limitado a calcular un vector de Parikh legal (un vector compuesto únicamente por transiciones disparables) que explique los símbolos de salida observados de la secuencia de transiciones disparada, no obstante el problema de calcular la secuencia de transiciones también es abordado en este capítulo. Con el objetivo de almacenar la secuencia de transiciones disparada, el PVO es extendido usando RPI coloreadas. Dicha extensión es nombrada como el Observador Coloreado de Secuencia de Disparo de Transiciones ($CFTSO$). Con este nuevo enfoque y usando un macro marcado virtual, el uso de un gran número de $RPIs$ que pertenecen a la familia del PVO es evitado. Todas las marcas coloreadas que compartan el mismo color son removidas del $CFTSO$ cuando son inconsistentes con la secuencia de símbolos de salida.

Nota 72. En este trabajo, se usará la notación Q_{SSD} para denotar el conjunto de $RPIs$ que son binarias, vivas, fuertemente conexas, puras y SSD .

Este capítulo está ordenado de la siguiente manera: En la sección 4.2 se define formalmente el PVO y se demuestra que ha convergido al vector de Parikh de la secuencia de disparo actual cuando el error es igual a cero. En la sección 4.3 se define formalmente el $CFTSO$ y se proporcionan sus reglas de evolución y

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

eliminación de marcas en exceso para alcanzar su convergencia, lo cual es mostrado a través de un ejemplo. Por último, en la sección 4.4 se presentan las conclusiones de este capítulo.

4.2. Observador de Vector de Parikh

En esta sección se presenta el *PVO* para determinar el vector de Parikh de la secuencia actual de disparo de transiciones de una *RPI* dada. Aunque el *PVO* no permite obtener la secuencia de disparo de transiciones, permite sentar las bases para construir el observador de secuencias propuesto en la siguiente sección. El *PVO* se construye como una copia de la *RPI* con algunos lugares adicionales. A la salida de cada transición t_i se le tiene que agregar un lugar etiquetado como \tilde{p}_i . De esta forma, cada vez que la transición t_i sea disparada, el marcado del lugar \tilde{p}_i será incrementado en uno, es decir, el vector de Parikh de la secuencia de disparo de transiciones será representado por el marcado en los lugares adicionales de la *RPI*. Debido a que el marcado inicial de la *RPI* es desconocido, entonces existen diversos marcados iniciales que son posibles, por lo que el *PVO* tiene que capturarlos a todos, motivo por el cual el *PVO* contiene una copia de la red por cada marcado inicial admisible de la *RPI*, lo cual se describe formalmente en las siguientes líneas.

Definición 73. Sea $(N, \Phi) \in Q_{SSD}$. El Observador de Vector de Parikh, representado como $PVO^N(\Phi M_0)$, de una *RPI* N es definida como el conjunto de redes $Q^N(\Phi M_0) = \{N_1, N_2, \dots, N_q\}$ tal que M_0 concuerda con M_0^i , es decir, $\Phi M_0 = \Phi M_0^i$ y $Y^T M_0^i = 1$, asimismo $M_0^1 \neq M_0^2 \neq \dots \neq M_0^q$, y

$N_i = (P_O^i, T_O^i, F^i, \Phi^i, M_0^i)$, donde:

- $T_O^i = T^i$, donde $T^i = \{t_1^i, \dots, t_m^i\}$,
- $P_O^i = P^i \cup \tilde{P}^i$, donde $P^i = \{p_1^i, \dots, p_n^i\}$ y $\tilde{P}^i = \{\tilde{p}_1^i, \dots, \tilde{p}_m^i\}$ son los lugares adicionales agregados a cada N_i (uno por cada transición), $|\tilde{P}^i| = |T_O^i|$ y $\forall t_j^i \in T_O^i, \{\tilde{p}_j^i\} \subset t_j^{i\bullet}$. El marcado inicial de cada lugar adicional agregado es cero.
- $F^i = F \cup \{(t_j^i, \tilde{p}_j^i)\}$,
- $\Phi^i = \begin{bmatrix} \Phi & 0 \\ 0 & I_d \end{bmatrix}$

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

Definición 74. Sea $(N, \Phi,) \in Q_{SSD}$ y sea $Q^N(\Phi M_0)$ su Observador de Vector de Parikh $PVO^N(\Phi M_0)$. La evolución de una red $N_i \in Q^N(\Phi M_k)$ está dividida en los siguientes tres pasos:

1. **Determinando cuál transición fue disparada en el sistema.** Esta tarea es efectuada usando el cambio a la salida de la siguiente manera: Debido a que N es SSD , entonces el disparo de una transición t_j conduce a un nuevo marcado M_{k+1} donde $\Phi M_{k+1} \neq \Phi M_k$. Por lo tanto, usando este cambio a la salida el disparo de la transición t_j puede ser detectado. Note que podría darse el caso en el que más de una transición satisfaga el cambio a la salida que es $\Phi M_{k+1} - \Phi M_k$.

2. **Disparando una Transición en $N_i \in Q^N(\Phi M_k)$.**

Cuando el disparo de una transición t_j en N es detectado, entonces

$$\gamma_k^i(r) = \begin{cases} 1 & \text{si } t_r^i \text{ está habilitada en } M_k^i, \\ M_{k+1}^i = M_k^i + \begin{bmatrix} C \\ I_d \end{bmatrix} \begin{matrix} \vec{t}_r^i \\ t_r^i \end{matrix} \\ \text{y } \Phi M_{k+1} = \Phi(\hat{M}_k^i + C t_r^i) \\ 0 & \text{en cualquier otro caso.} \end{cases}$$

donde I_d es la matriz identidad, cuya dimensión es igual al número de lugares adicionales y $\gamma_k^i(r)$ representa a las transiciones a ser disparadas en el PVO.

3. **Actualización de $Q^N(\Phi M_k)$.**

Si ninguna de las transiciones de N_i fue disparada en el paso anterior, entonces $Q^N(\Phi M_{k+1}) = Q^N(\Phi M_k) - \{N_i\}$.

Nota 75. Note que la ecuación de estados de $N_i \in Q^N(\Phi M_k)$ es:

$$M_{k+1}^i = M_k^i + \begin{bmatrix} C \\ I_d \end{bmatrix} [\gamma_k^i] \\ y_k^i = \begin{bmatrix} \hat{y}_k^i \\ \tilde{y}_k^i \end{bmatrix} = \begin{bmatrix} \Phi & 0 \\ 0 & I_d \end{bmatrix} \begin{bmatrix} \hat{M}_k^i \\ \tilde{M}_k^i \end{bmatrix}$$

Asimismo advierta que el marcado en los lugares adicionales \tilde{P}^i representan el vector de Parikh de la secuencia de disparo de transiciones actual en la red N_i .

Proposición 76. Sea $(N, \Phi) \in Q_{SSD}$ y sea $Q^N(\Phi M_0)$ su Observador de Vector de Parikh $PVO^N(\Phi M_0)$. Entonces existe $N_i \in Q^N(\Phi M_k)$, $\forall M_k$ alcanzado en N

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

tal que $\vec{\sigma}_k = M_k^i|_{\tilde{P}^i}$, donde $\vec{\sigma}_k$ es el vector de Parikh de la secuencia de disparo de transiciones en N y $M_k^i|_{\tilde{P}^i}$ es el marcado en N_i restringido a los lugares de \tilde{P}^i (los lugares adicionales).

Demostración. Debido a que $\forall N_k \in Q^N(\Phi M_0)$, entonces el M_0^k concuerda con M_0 , es decir ambos marcados iniciales generan los mismos símbolos de salida. Además, como M_0^i concuerda con M_0 , entonces existe $N_i \in Q^N(\Phi M_0)$ tal que $M_0^i = M_0$ sobre los lugares $p_j \in P^i - \tilde{P}^i$. Se afirma que $\vec{\sigma}_k = M_k^i|_{\tilde{P}^i}$ y $M_k = M_k^i|_{P^i - \tilde{P}^i}$, donde $M_k^i|_{P^i - \tilde{P}^i}$ representa el marcado restringido a los lugares $P^i - \tilde{P}^i$ (los lugares originales de la red Q). Entonces, procediendo por inducción matemática sobre la longitud de la secuencia de disparo de transiciones σ_k en N para probar esta afirmación.

Si $|\sigma_k| = 0$ entonces $\vec{\sigma}_0 = \vec{0}$. Debido a que el marcado inicial sobre los lugares adicionales en \tilde{P}^i es cero, entonces $M_0^i|_{\tilde{P}^i} = \vec{0}$, por lo tanto $\vec{\sigma}_0 = M_0^i|_{\tilde{P}^i}$ y $M_0 = M_0^i|_{P^i - \tilde{P}^i}$.

Suponga que es cierto que para $|\sigma_k| = n$, es decir $\vec{\sigma}_n = M_n^i|_{\tilde{P}^i}$ y $M_n = M_n^i|_{P^i - \tilde{P}^i}$.

Si $|\sigma_k| = n + 1$ entonces la transición t_q fue disparada en N en el marcado M_n . Entonces $M_{n+1} = M_n + C \vec{t}_q$. Como $M_n = M_n^i|_{P^i - \tilde{P}^i}$, la copia de la transición t_q en N_i puede ser disparada en N_i en el marcado M_n^i alcanzando el marcado $M_{n+1}^i = M_n^i + \begin{bmatrix} C \\ I_d \end{bmatrix} \vec{t}_q$. Por lo tanto, $M_{n+1}^i|_{\tilde{P}^i} = M_n^i|_{\tilde{P}^i} + \vec{t}_q$, por inducción matemática $M_{n+1}^i|_{\tilde{P}^i} = \vec{\sigma}_n + \vec{t}_q = \vec{\sigma}_{n+1}$. Más aún, debido a que $M_{n+1} = M_n + C \vec{t}_q = M_n^i|_{P^i - \tilde{P}^i} + C \vec{t}_q = M_{n+1}^i|_{P^i - \tilde{P}^i}$, entonces $\Phi M_{n+1} = \Phi(M_{n+1}^i|_{P^i - \tilde{P}^i})$. ■

En palabras más simples, lo que se ha demostrado en la proposición anterior es que existe al menos una de las redes que componen al PVO en la cual el marcado de sus lugares adicionales representa el vector de Parikh de la secuencia de disparo de transiciones de la $RPI N$. En el siguiente resultado se formaliza la idea de que si en el PVO prevalecen muchas redes en las cuales el marcado de sus lugares adicionales es el mismo, entonces el vector de Parikh de la secuencia de disparo de transiciones de la $RPI N$ es el marcado de los lugares adicionales de cualquiera de ellas.

Proposición 77. Sea $(N, \Phi) \in Q_{SSD}$ y sea $Q^N(\Phi M_0)$ su Observador de Vector de Parikh $PVO^N(\Phi M_0)$. Si $Q^N(\Phi M_k) = \{N_a, N_b, \dots, N_z\}$ tal que $M_k^a|_{\tilde{P}^i} = M_k^b|_{\tilde{P}^i} = \dots = M_k^z|_{\tilde{P}^i}$, entonces $\vec{\sigma}_k = M_k^a|_{\tilde{P}^i} = M_k^b|_{\tilde{P}^i} = \dots = M_k^z|_{\tilde{P}^i}$, donde $\vec{\sigma}_k$ es el vector

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

de Parikh vector de la secuencia de disparo de transiciones en N tal que $M_k = M_0 + C\vec{\sigma}_k$.

Demostración. De la Proposición 76 $\exists N_i \in Q^N(\Phi M_k)$, $\forall M_k$ alcanzado en N tal que $\vec{\sigma}_k = M_k^i|_{\bar{P}_i}$. Debido a que $N_i \in Q^N(\Phi M_k)$ y $M_k^a|_{\bar{P}_i} = M_k^b|_{\bar{P}_i} = \dots = M_k^z|_{\bar{P}_i} = M_k^i|_{\bar{P}_i}$, entonces $\vec{\sigma}_k = M_k^i|_{\bar{P}_i} = M_k^a|_{\bar{P}_i} = M_k^b|_{\bar{P}_i} = \dots = M_k^z|_{\bar{P}_i}$. ■

A continuación se define formalmente la ecuación del error del PVO.

Definición 78. Sea $(N, \Phi) \in Q_{SSD}$ y sea $Q^N(\Phi M_0)$ su Observador de Vector de Parikh $PVO^N(\Phi M_0)$. Sea $Q^N(\Phi M_k) = \{N_a, N_b, \dots, N_z\}$ y $N_f \in Q^N(\Phi M_k)$. La ecuación del error del Observador de Vector de Parikh $PVO^N(\Phi M_k)$ es definida por:

$$e_k = \sum_{N_j \in Q^N(\Phi M_k)} \left\| M_k^j|_{\bar{P}_i} - M_k^f|_{\bar{P}_i} \right\| \quad (4.1)$$

Cabe señalar que si el error calculado en la ecuación anterior es igual a cero, entonces se puede decir que el PVO a convergido al vector de Parikh de la secuencia de disparo de transiciones, lo cual se formaliza en las siguientes líneas.

Proposición 79. Sea $(N, \Phi) \in Q_{SSD}$ y sea $Q^N(\Phi M_0)$ su Observador de Vector de Parikh $PVO^N(\Phi M_0)$. Si $e_k = 0$, entonces el vector de Parikh $\vec{\sigma}_k$ de la secuencia de disparo de transiciones en N ($M_k = M_0 + C\vec{\sigma}_k$) puede ser calculada como $\vec{\sigma}_k = M_k^i|_{\bar{P}_i}$, $N^i \in Q^N(\Phi M_0)$.

Demostración. Si el error en el marcado M_k es cero, $e_k = 0$, entonces dos casos son posibles:

(a) Suponga que en el marcado M_k , $Q^N(\Phi M_k) = \{N_i\}$, (por la Proposición 76 tal N_i existe). Debido a que $e_k = \sum_{N_j \in Q^N(\Phi M_k)} \left\| M_k^j|_{\bar{P}_i} - M_k^f|_{\bar{P}_i} \right\| = \left\| M_k^i|_{\bar{P}_i} - M_k^f|_{\bar{P}_i} \right\| =$

0, entonces todas las redes en $Q^N(\Phi M_k)$ tienen el mismo marcado, por lo tanto por la Proposición 77 $\vec{\sigma}_k = M_k^i|_{\bar{P}_i}$.

(b) Suponga que en el marcado M_k , $Q^N(\Phi M_k) = \{N_a, N_b, \dots, N_z\}$.

Debido a que $e_k = \sum_{N_j \in Q^N(\Phi M_k)} \left\| M_k^j|_{\bar{P}_i} - M_k^f|_{\bar{P}_i} \right\| = 0$, entonces $\forall j, M_k^j|_{\bar{P}_i} = M_k^f|_{\bar{P}_i}$,

por lo que la única posibilidad que hay para que $\forall j, M_k^j|_{\bar{P}_i} = M_k^f|_{\bar{P}_i}$ es que $M_k^a|_{\bar{P}_i} = M_k^b|_{\bar{P}_i} = \dots = M_k^z|_{\bar{P}_i}$. Por lo tanto, por la Proposición 77 $\vec{\sigma}_k = M_k^i|_{\bar{P}_i}$.

■

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

A continuación se presenta un ejemplo en el cual se explica detalladamente el funcionamiento del PVO para una RPI que es SSD.

Ejemplo 80. Sea Q una RPI, tal $(N, \Phi) \in Q_{SSD}$, mostrada en la Figura 4.2,

$$\text{donde } \Phi = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \text{ y}$$

$$\Phi C = \begin{bmatrix} -1 & 1 & -1 & 1 & -1 & 0 & 1 & -1 & 0 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}.$$

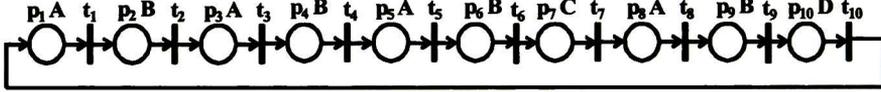


Figura 4.2: RPI Q que es SSD.

Suponga que en el marcado inicial de la RPI se observa el símbolo de salida $\Phi M_0 = [0 \ 1 \ 0 \ 0]^T$, entonces $Q^N(\Phi M_0) = \{N_1, N_2, N_3, N_4\}$ es su Observador de Vector de Parikh $PVO^N(\Phi M_0)$, mostrado en la Figura 4.3, donde:

$$M_0^1 = [0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T,$$

$$M_0^2 = [0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$$

$$M_0^3 = [0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0]^T \text{ y}$$

$$M_0^4 = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0]^T$$

El conjunto $Q^N(\Phi M_0) = \{N_1, N_2, N_3, N_4\}$ es el PVO porque $M_0^1, M_0^2, M_0^3, M_0^4 \in \mu_0$ y porque sus marcados iniciales concuerdan con el símbolo de salida observado inicialmente, es decir $\Phi M_0 = \Phi M_0^1 = \Phi M_0^2 = \Phi M_0^3 = \Phi M_0^4$.

Ahora suponga que una transición es disparada en la RPI Q y que el símbolo de salida observado es $\Phi M_1 = [1 \ 0 \ 0 \ 0]^T$ por lo que el cambio de símbolos de salida es $\Phi M_1 - \Phi M_0 = [1 \ -1 \ 0 \ 0]^T$. En cada una de las redes que conforman a $Q^N(\Phi M_0)$ se debe buscar una transición que pueda producir el mismo cambio de símbolos de salida, por lo en N_1 se tiene que disparar la transición t_2^1 , en

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

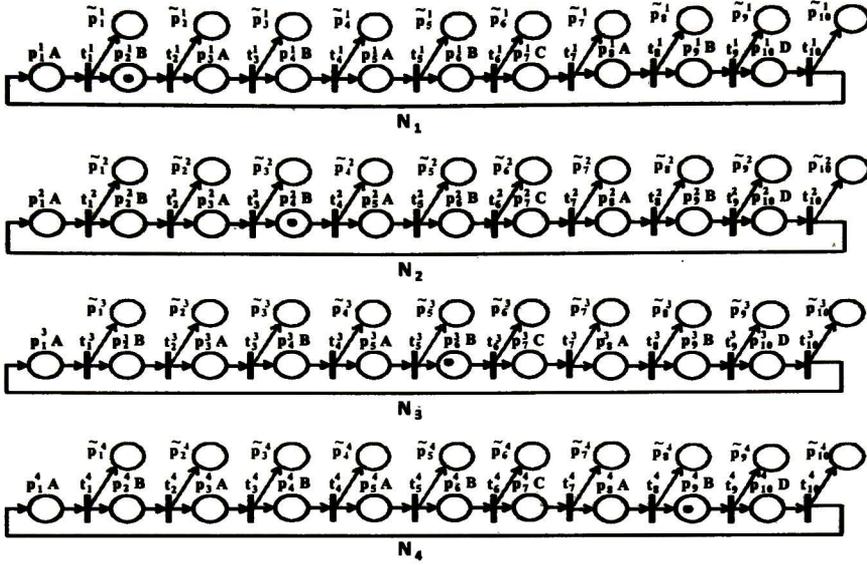


Figura 4.3: $PVO Q^N(\Phi M_0)$.

N_2 se tiene que disparar la transición t_4^2 , pero en las redes N_3 y N_4 no existen transiciones habilitadas que produzcan dicho cambio de símbolos de salida, por lo tanto actualizando el PVO se tiene que $Q^N(\Phi M_1) = Q^N(\Phi M_0) - \{N_3, N_4\}$, resultando las RPIs que se muestran en la Figura 4.4.

El PVO aún no ha convergido, esto se puede verificar a través de la Ecuación 4.1 que permite calcular el error, que en este caso:

$$e_1 = \|M_1^1|_{\bar{p}_i} - M_1^2|_{\bar{p}_i}\|$$

$$e_1 = \left\| \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T - \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \right\|$$

$$e_1 = \left\| \begin{bmatrix} 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \right\| = 1^2 + (-1)^2 = 2$$

y como $e_1 \neq 0$, entonces no se tiene aún la convergencia del PVO.

Suponga que en la RPI Q se dispara otra transición y que el símbolo de salida observado es $\Phi M_2 = [1 \ 0 \ 0 \ 0]^T$ por lo que el nuevo cambio de símbolos de salida es $\Phi M_2 - \Phi M_1 = [1 \ -1 \ 0 \ 0]^T$ Como en la red N_1 la transición habilitada t_3^1 y en la red N_2 la transición habilitada t_5^2 producen dicho cambio de

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

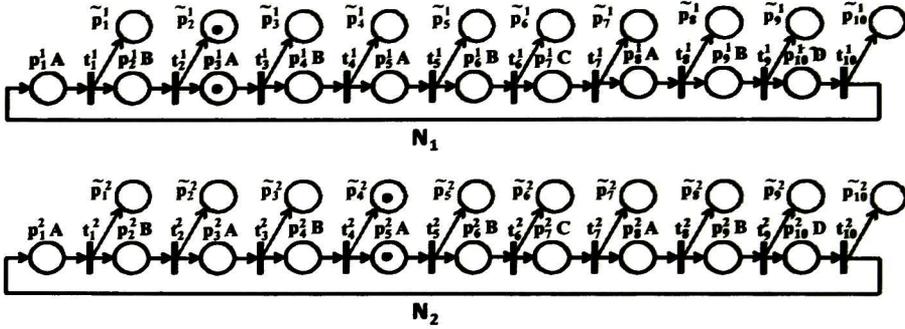


Figura 4.4: PVO \$Q^N(\Phi M_1)\$.

símbolos de salida entonces \$Q^N(\Phi M_2) = Q^N(\Phi M_1)\$. El nuevo estado alcanzado en el PVO \$Q^N(\Phi M_2)\$ es mostrado en la Figura 4.5. Nuevamente aún no se tiene la convergencia del PVO porque el error \$e_2 \neq 0\$, puesto que

$$e_2 = \left\| M_2^1 | \tilde{p}_i - M_2^2 | \tilde{p}_i \right\|$$

$$e_2 = \left\| \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T - \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \right\|$$

$$e_2 = \left\| \begin{bmatrix} 0 & 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \right\| = 1^2 + 1^2 + (-1)^2 + (-1)^2 = 4.$$

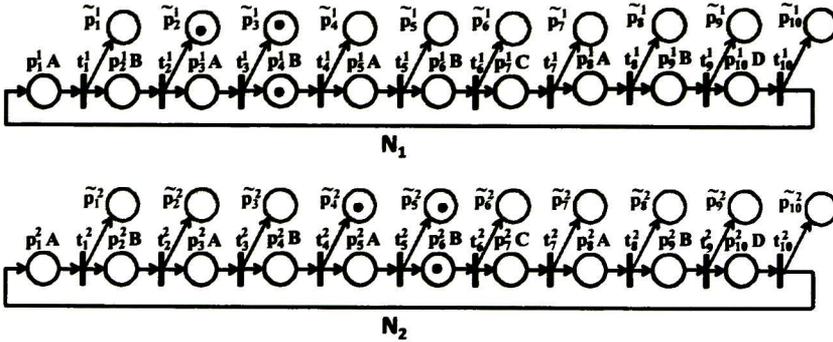


Figura 4.5: PVO \$Q^N(\Phi M_2)\$.

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

Ahora suponga que en la RPI Q se dispara otra transición y que el símbolo de salida observado es $\Phi M_3 = [0 \ 0 \ 1 \ 0]^T$ por lo que el cambio de símbolos de salida es $\Phi M_3 - \Phi M_2 = [-1 \ 0 \ 1 \ 0]^T$. En cada una de las redes que componen al PVO $Q^N(\Phi M_2)$ se tiene que buscar una transición habilitada que produzca el mismo cambio de símbolos. En el caso de la red N_2 se tiene a la transición t_6^2 y en N_1 no existe dicha transición, por tal motivo $Q^N(\Phi M_3) = Q^N(\Phi M_2) - \{N_1\}$, que es lo mismo que $Q^N(\Phi M_3) = \{N_2\}$.

En este caso el PVO $Q^N(\Phi M_3)$ ya convergió porque $|Q^N(\Phi M_3)| = 1$ y $e_3 = 0$, ya que $e_3 = \|M_3^2|_{\tilde{P}_i} - M_3^2|_{\tilde{P}_i}\| = 0$. Por lo tanto se concluye que el vector de Parikh de la secuencia disparada en la RPI Q es el marcado de los lugares adicionales de la RPI mostrada en la Figura 4.6, es decir es $\vec{\sigma}_3 = M_3^2|_{\tilde{P}_i} = [0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0]^T$

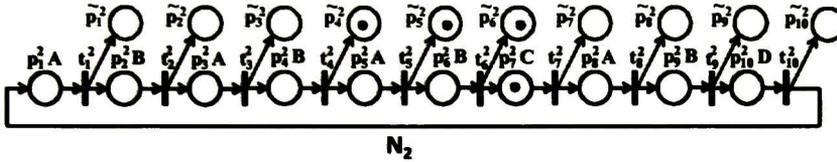


Figura 4.6: PVO $Q^N(\Phi M_3)$.

En RPIs más complejas la cantidad de redes con las que se compone el PVO puede aumentar exponencialmente. Por tal motivo, en la próxima sección se propone un observador coloreado para evitar este problema.

4.3. Observador Coloreado de la Secuencia de Transiciones Disparada

Esta sección aborda el problema del diseño de un observador para la secuencia de transiciones disparada. El enfoque que aquí se presenta usa RPI coloreadas para implementar cada red que pertenece al $PVO^N(\Phi M_0)$. Por lo que la construcción del conjunto $Q^N(\Phi M_0)$ que contiene una gran cantidad de RPIs es evitada. El enfoque propuesto usa un color diferente con el fin de representar cada marcado M_0^i de $N_i \in PVO^N(\Phi M_0)$. Sin embargo, debido al gran número de marcados iniciales factibles que concuerdan con M_0 , el cálculo de todos los M_0^i es una tarea

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

complicada. Afortunadamente, usando un macro marcado virtual, es decir un marcado mayor o igual a cualquiera de los marcados en $M_0^i \in PVO^N(\Phi M_0)$, permite determinar un marcado inicial coloreado que reproducirá el mismo cambio de símbolos a la salida que aquellos que ocurren en N .

En este caso el $PVO^N(\Phi M_0)$ es llamado $CFTSO^N$. El $CFTSO^N$ trabaja de la siguiente manera:

Debido a que N es SSD , entonces el disparo de cualquier transición produce un cambio a la salida en N . Por lo que el i -ésimo cambio a la salida produce un evento denominado como E_i . Por lo tanto, para cualquier evento E_i existe un conjunto potencial de explicaciones (conjunto $S_{\sigma\omega}$) formadas por las transiciones cuyo disparo producen el i -ésimo cambio a la salida. Este conjunto es llamado S_{E_i} .

La evolución del $CFTSO^N$ será explicada usando el siguiente ejemplo, donde la evolución y la validación del $CFTSO^N$ se introduce mediante su comparación con respecto al $PVO^N(\Phi M_0)$.

Ejemplo 81. Sea S una LE fuertemente conexa, pura, viva y binaria dibujada en la Figura 4.7, donde

$$\Phi = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Suponiendo que $\Phi M_0 = [1 \ 0 \ 0 \ 0 \ 0]^T$ entonces el PVO^N está compuesto por tres redes denotadas por N_1, N_2, N_3 , representadas en la Figura 4.8, donde cada red es una copia de la estructura de red de la LE original. En N_1 hay únicamente una marca inicialmente residiendo en p_1 ; en N_2 las marcas inicialmente marcan a los lugares p_2 y p_4 ; y en N_3 las marcas inicialmente marcan a los lugares p_6 y p_8 , los cuales representan los tres posibles marcados iniciales tales que $\Phi M_0 = [1 \ 0 \ 0 \ 0 \ 0]^T$ y $M_0 \in \mu_0$. El $CFTSO^N$ propuesto, mostrado en la Figura 4.8, tiene la misma estructura que cualquier red en el PVO^N , sin embargo, el marcado inicial no es explícitamente representado debido a que se hace la suposición de que el $CFTSO^N$ tiene un macro marcado virtual.

Suponga que una transición es disparada en la LE produciendo el símbolo de salida $\Phi M_1 = [0 \ 1 \ 0 \ 0 \ 0]^T$. Debido a que este vector elemental es el primer cambio a la salida que es detectado (de ΦM_0 a ΦM_1) entonces el primer evento E_1

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

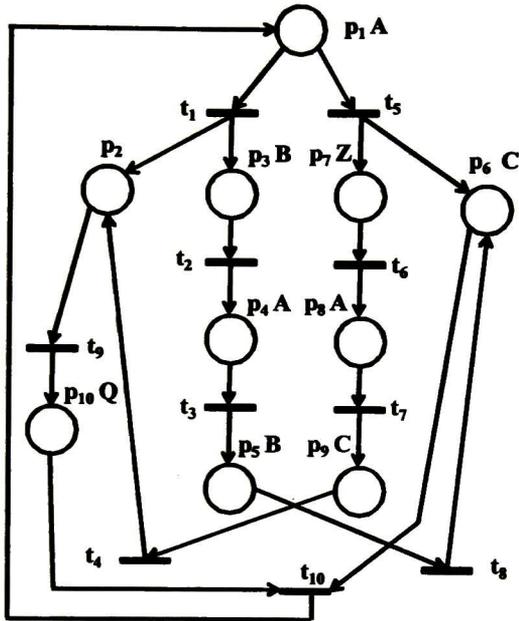


Figura 4.7: Una LE que es SSD y marcado detectable.

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

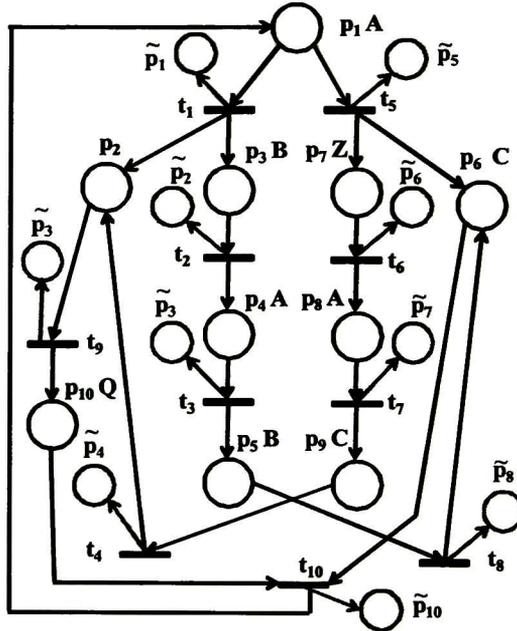


Figura 4.8: La estructura de RPI para una $N_i \in PVO^N(\Phi M_0)$ o para el $CPVO^N$ para la RPI mostrada en la Figura 4.7.

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

es registrado. Para el nuevo símbolo a la salida ΦM_1 existen transiciones habilitadas t_1^1, t_3^2 en N_1 y en N_2 respectivamente, cuyo disparo da origen a los marcados M_1^1, M_1^2 en N_1 y en N_2 respectivamente, y $\Phi M_1^1 = \Phi M_1^2 = [0 \ 1 \ 0 \ 0 \ 0]^T$. Debido a que la red N_3 no puede evolucionar, entonces tiene que ser removida del PVO^N . De manera similar, el $CFTSO^N$ registra el evento E_1 y el conjunto $S_{E_1} = \{\alpha_1(t_1), \alpha_3(t_3)\}$ es calculado, lo cual significa que el disparo de las transiciones t_1 y t_3 explican el cambio de símbolo a la salida producido de ΦM_0 a ΦM_1 , y α_1, α_3 son los colores asociados a las transiciones, es decir estos colores representan a las redes N_1, N_2 respectivamente.

Ahora, suponga que la LE evoluciona nuevamente y el símbolo a la salida $\Phi M_2 = [0 \ 0 \ 1 \ 0 \ 0]^T$ es observado. Debido a que este vector elemental es el segundo símbolo a la salida que es detectado, entonces el evento E_2 es registrado. En el PVO^N , solamente la red N_2 puede evolucionar, por lo tanto la red N_1 es removida del PVO^N . En el $CFTSO^N$ también el evento e_2 es registrado y el conjunto $S_{E_2} = \{\alpha_3(t_8)\}$ es calculado. Puesto a que al color α_1 no les es posible continuar explicando la salida en la LE , entonces éste es removido del conjunto S_{E_1} (note que el color α_1 estaba representando a la red N_1).

Debido a que el PVO^N contiene solamente la red N_2 , entonces ya ha convergido al vector de Parikh legal de la secuencia de disparo de transiciones actual. Sin embargo, aunque el $CFTSO^N$ contiene sólo un color, éste aún no converge porque existen marcas virtuales que no están representados en el marcado visible. Si $\sum_{i=1}^k |S_{E_i}| = k$, donde k representa el número de eventos detectados (es decir, si cada evento registrado al evento actual solamente tiene una explicación) entonces la secuencia de disparo de transiciones puede ser parcialmente determinada. Sin embargo, debido a que puede haber marcas no coloreadas, entonces el marcado actual del $CFTSO^N$ no puede ser calculado.

De hecho, el $CFTSO^N$ converge al marcado actual cuando $\sum_{i=1}^k |S_{E_i}| = k$ y no existen marcas virtuales no coloreadas en la red. Esta condición puede ser fácilmente probada usando los P -semiflujos y la propiedad de binaridad.

La convergencia de la secuencia de disparo de transiciones será explicada usando el mismo ejemplo, permitiendo que se efectúen más eventos en el sistema. Suponiendo que la LE evoluciona al marcado M_3 y el símbolo de salida $\Phi M_3 = [0 \ 0 \ 1 \ 1 \ 0]^T$ es observado, entonces el evento E_3 es registrado. En la $CFTSO^N$ la transición t_9 tiene que ser disparada con el fin de obtener el símbolo de salida ΦM_3 (la transición t_9 está habilitada porque se está suponiendo un

CAPÍTULO 4. OBSERVADOR DE SECUENCIAS

macromarcado virtual). Entonces el conjunto $S_{E_3} = \{\alpha_9(t_9)\}$ es agregado. En este caso el color α_3 no tiene que ser removido de los conjuntos S_{E_1} y S_{E_2} porque éste aún continúa explicando los símbolos de salida observados.

Ahora, suponga que la LE evoluciona nuevamente generando el símbolo de salida $\Phi M_4 = [1 \ 0 \ 0 \ 0 \ 0]^T$. Entonces el evento E_4 es registrado. En la $CFTSO^N$ la transición t_{10} tiene que ser disparada. Debido a que la transición t_{10} consume a los colores α_3 y α_9 en su disparo, entonces los colores α_3 y α_9 son fusionados para generar un nuevo color, por ejemplo el color c_1 . Entonces todos los colores α_3 , α_9 en los conjuntos S_{E_1} , S_{E_2} y S_{E_3} tienen que ser transformados en el color c_1 . El conjunto $S_{E_4} = \{c_1(t_{10})\}$ es calculado. Debido a que existe solamente un color y por cada P -semiflujo Y_i satisface que $Y_i M_4 = 1$, es decir no existen marcas virtuales no coloreadas en el $CFTSO^N$, entonces la secuencia de disparo de transiciones $\sigma = t_3 t_8 t_9 t_{10}$ es calculada de los conjuntos $S_{E_1} = \{c_1(t_3)\}$, $S_{E_2} = \{c_1(t_8)\}$, $S_{E_3} = \{c_1(t_9)\}$, $S_{E_4} = \{c_1(t_{10})\}$, dicha secuencia corresponde al vector de Parikh determinado por el PVO^N como $\vec{\sigma} = M_4^2 |_{\vec{P}_i}$. Además, debido a que no existen marcas virtuales no coloreadas, entonces el marcado del $CFTSO^N$ es el marcado actual del sistema.

4.4. Conclusiones

En este capítulo se abordó el problema de determinar la secuencia de disparo de transiciones que ha sido disparada en una RPI que es SSD y se mostró que no es equivalente al problema de determinar el estado o marcado actual de la RPI.

En primer lugar, se presentó el PVO para estimar el vector de Parikh de una secuencia de disparo de transiciones. Se demostró que dicha entidad, compuesta por una familia de RPIs, converge al vector de Parikh cuando todas las redes que lo conforman tienen el mismo marcado en sus lugares adicionales, o bien, si en su evolución sólo prevalece una de las redes iniciales.

Posteriormente, haciendo uso de RPIs coloreadas, se definió el CFTSO para capturar la secuencia de disparo de transiciones. Asimismo, se presentaron las reglas de evolución y marcado del CFTSO, así como también sus criterios de convergencia en la secuencia de disparo de transiciones.

5. DIAGNOSTICABILIDAD EN LAS RPI DE LA CLASE DE LAS LE

Resumen: En este capítulo se demuestra que la secuencia detectabilidad es una condición suficiente para la diagnosticabilidad de las RPI pertenecientes a la clase de las FC . Asimismo se presenta una caracterización de la diagnosticabilidad en RPI s que no son SD . Dicha caracterización utiliza las mismas relaciones y estructuras de las RPI s que se usaron en la caracterización de la secuencia detectabilidad estructural. También se presenta un algoritmo de tiempo polinomial para satisfacer la condición más problemática en dicha caracterización.

5.1. Introducción

Los sistemas industriales modernos cada vez son más grandes y complejos para poder satisfacer las demandas del mercado. Con la finalidad de poder incrementar su eficiencia y productividad los sistemas tienen que ser confiables. Por tal motivo es de vital importancia poder hacer una detección oportuna de las faltas (comportamientos no deseados) con el objetivo de implementar las tareas necesarias en los algoritmos de control para evitarlas o tomar las medidas adecuadas para corregirlas, evitando de esta manera el fallo del sistema que podría poner en peligro vidas humanas, causar pérdidas económicas, dañar equipos e instalaciones, entre otras catástrofes. Sin embargo, dependiendo de las características del sistema, esta labor podría ser o no ser posible, motivo por el cual es importante establecer las condiciones bajo las cuales esta labor es posible.

Como fue mencionado en la revisión de la literatura, hay trabajos que han caracterizado la diagnosticabilidad en AF en tiempo polinomial, no obstante para trasladar dichos resultados a las RP es necesario determinar su grafo de alcanzabilidad, lo cual resulta en algoritmos de tiempo no polinomial. Asimismo, hay investigaciones enfocadas en la expansión del grafo de alcanzabilidad de las RPI o una versión reducida del mismo [17], [10] y [9].

CAPÍTULO 5. DIAGNOSTICABILIDAD EN IPN

En el área del diagnóstico, muchos investigadores han optado por usar las *RP* como herramienta formal de modelado, no obstante esto ha conllevado a que los resultados propuestos no sean de tiempo polinomial [19], [7], [8], o bien cuando se tienen algoritmos de tiempo polinomial [39], [18] únicamente se tienen condiciones de suficiencia, aunado a que sólo se pueden analizar clases muy limitadas de *RP* [18]. Han sido muy pocas las investigaciones que han presentado condiciones suficientes y necesarias [5]; sin embargo las hipótesis de trabajo como lo son la no repetición de etiquetas en las transiciones y subredes observables y no observables acíclicas, hacen muy restrictiva a la caracterización de la diagnosticabilidad.

Los resultados en el área de diagnóstico se pueden resumir en la siguiente figura:

	Este capítulo	[18]	[19]	[39]	[8]	[7]
Propiedades de la <i>RPI</i>	Vs^1	Vs	Lb^2	Vs	Lb	Lb
Bloqueos	×	×	×	✓ ³	×	×
Tiempo polinomial	✓	✓	×	✓	×	×
Ciclos no observable	×	✓	✓	✓	×	×
Repetición de símbolos de entrada y/o salida	✓	✓	✓	×	✓	✓
Eventos no observables	× ⁴	✓	✓	✓	✓	✓
Faltas observables	✓	✓	✓	✓	×	✓
Suficiencia y necesidad	✓	×	✓	×	✓	✓

Tabla 5.1: Cuadro comparativo de la literatura relevante en diagnóstico.

- 1: La abreviatura *Vs* significa vivas y seguras.
- 2: La abreviatura *Lb* significa Libre de bloqueos.
- 3: La *RPI* se bloquea con la ocurrencia de una falta.

4: Aunque en la teoría propuesta de la *SD* los eventos tienen que ser observables, los resultados propuestos en diagnóstico se pueden aplicar también al caso de transiciones no observables. Es importante señalar que sí es permitida la ocurrencia de estados no observables.

En este capítulo se presenta una caracterización de la diagnosticabilidad haciendo uso de las mismas relaciones con las que se caracterizó la secuencia detectabilidad.

Este capítulo se organiza de la siguiente manera: En la Sección 5.2 se presenta la caracterización de la diagnosticabilidad en *RPI* de la clase de las *LE* y el resultado que indica que la secuencia detectabilidad es una condición suficiente

CAPÍTULO 5. DIAGNOSTICABILIDAD EN IPN

para la diagnosticabilidad. La Sección 5.3 proporciona un algoritmo para verificar la condición más problemática en la caracterización de la diagnosticabilidad, asimismo su correctitud y un breve análisis de complejidad. Posteriormente, en la Sección 5.4 se desarrollan algunos ejemplos aplicando el algoritmo propuesto para determinar la diagnosticabilidad en RPI de la clase LE . Finalmente, en la Sección 5.5 se exponen las conclusiones de este capítulo.

5.2. Caracterización de la Diagnosticabilidad en RPI en la clase de las LE

En esta sección la propiedad de diagnosticabilidad en RPI vivas, binarias y puras es analizada en base a las relaciones propuestas en el Capítulo 3, las cuales fueron utilizadas en la caracterización de la secuencia detectabilidad estructural.

El conjunto de las transiciones de una RPI puede ser particionada en dos subconjuntos, llamados $T = T_{reg} \cup T_f$ donde el conjunto T_f incluye a todas las transiciones de falta (las cuales permiten modelar los comportamientos no deseados o comportamientos de falta), mientras que T_{reg} incluye todas las transiciones que representan un comportamiento regular o normal del sistema. Además, el conjunto T_f puede ser particionado en k diferentes subconjuntos T_f^i , donde $i = 1, 2, \dots, k$, que representa las diferentes clases de falta.

Sea $\sigma \in \mathcal{L}(Q, M_0)$, como notación se utilizará $t_j^i \in \sigma$ para enfatizar que la secuencia de disparo de transiciones σ contiene la transición de falta $t_j \in T_f^i$. De forma análoga, $t_j^i \notin \sigma$ será usado para representar que la transición de falta $t_j \in T_f^i$ no pertenece a la secuencia de disparo de transiciones σ .

La definición de diagnosticabilidad que se utilizará en esta tesis es la que se enuncia a continuación.

Definición 82. Una $RPI (Q, M_0)$ se dice ser *diagnosticable* si no existen dos secuencias de disparo de transiciones $\sigma_1, \sigma_2 \in \mathcal{L}(Q, M_0)$ arbitrariamente largas tales que:

- $\sigma_1 \approx_I \sigma_2$,
- si $t_j^i \in \sigma_1$ entonces $t_k^i \notin \sigma_2$ (es decir, existen dos secuencias de disparo de transiciones σ_1, σ_2 , donde si σ_1 contiene una transición de falta perteneciente a la i -ésima clase de falta, entonces la secuencia σ_2 no contiene una transición de falta perteneciente a la misma clase de falta).

CAPÍTULO 5. DIAGNOSTICABILIDAD EN IPN

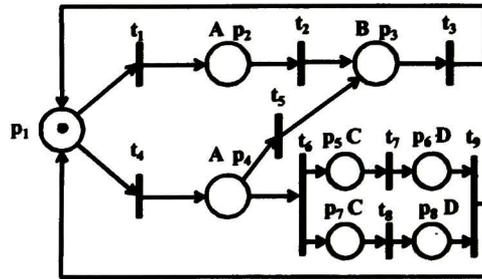


Figura 5.1: Una *LE* que no es *SD*; pero sí diagnosticable.

El siguiente resultado presenta un condición suficiente para la diagnosticabilidad en las *RPI*.

Teorema 83. *Sea (Q, M_0) una RPI viva, binaria y pura de la clase de las LE. Si Q es SD entonces Q es diagnosticable.*

Demostración. Debido a que Q es *SD* entonces no existen dos secuencias de disparo de transiciones cuyo disparo genere los mismos símbolos de salida, lo cual satisface la definición de diagnosticabilidad. Por lo tanto, Q es diagnosticable. ■

Es importante resaltar que la no secuencia detectabilidad no implica la no diagnosticabilidad, como se mostrará en el siguiente ejemplo.

Ejemplo 84. *Considere la RPI no secuencia detectable mostrada en la Figura*

5.1, donde $\Phi = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$

Suponga que $T_f^1 = \{t_7\}$. Claramente, debido a que las secuencias de disparo de longitud arbitrariamente larga e indistinguibles $\sigma_1 = t_4 t_6 t_7 t_8 t_9 \dots$ y $\sigma_2 = t_4 t_6 t_8 t_7 t_9 \dots$ contienen la transición de falta t_7 , entonces si es posible detectar su ocurrencia, por lo tanto la *RPI* es diagnosticable; pero no *SD*.

No obstante, también existen *RPI* que no son *SD* y que no son diagnosticables.

CAPÍTULO 5. DIAGNOSTICABILIDAD EN IPN

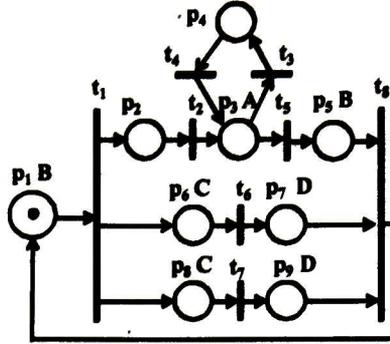


Figura 5.2: Una LE que no es SD ni tampoco diagnosticable.

Ejemplo 85. Sea Q la RPI mostrada en la Figura 5.2, donde

$$\Phi = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Note que Q no es SD debido a que $\exists t_6, t_7 \in T$ tales que $t_6 \approx_p t_7$. Suponga que $T_j^1 = \{t_7\}$ y $T_j^2 = \{t_6\}$. Observe que $\exists \sigma_1, \sigma_2 \in \mathcal{L}(Q, M_0)$, donde $\sigma_1 = t_1 t_2 t_6 (t_3 t_4)^*$ $\sigma_2 = t_1 t_2 t_7 (t_3 t_4)^*$ tal que $\sigma_1 \approx_I \sigma_2$, $t_7 \in \sigma_1$ y $t_7 \notin \sigma_2$. Por lo tanto, Q no es diagnosticable.

En el ejemplo anterior, el disparo de las transiciones relacionadas en concurrencia t_6 y t_7 no pueden ser detectadas porque existe un T -semi flujo ($t_3 t_4$) que evoluciona en paralelo con las transiciones t_6 y t_7 .

Además de lo expuesto en las líneas anteriores también es necesario hacer una extensión de la Definición 44. Esta definición puede ser extendida a dos secuencias de disparo de transiciones de longitud finita o arbitrariamente larga $\sigma_1 = t_1 \dots t_k \dots$, $\sigma_2 = t'_1 \dots t'_k \dots$, $|\sigma_1| = |\sigma_2|$, donde σ_1 y σ_2 se dicen estar relacionadas en concurrencia ($\sigma_1 \approx_p \sigma_2$) si $t_1 \approx_p t'_1, \dots, t_k \approx_p t'_k, \dots$

Definición 86. Dos T -semi flujos X_i y X_j están relacionados en concurrencia $X_i \approx_p X_j$ si $\exists \sigma_1, \sigma_2$ tales que $|\sigma_1| = |\sigma_2|$, $\sigma_1 \neq \sigma_2$, $\vec{\sigma}_1 = X_i$ y $\vec{\sigma}_2 = X_j$ y $\sigma_1 \approx_p \sigma_2$.

El siguiente teorema caracteriza la diagnosticabilidad usando las relaciones que permitieron caracterizar a la secuencia detectabilidad y tomando en consideración la información contenida en los párrafos anteriores.

CAPÍTULO 5. DIAGNOSTICABILIDAD EN IPN

Teorema 87. Sea (Q, M_0) una RPI viva, binaria y pura de la clase de las LE tal que $\forall t_i, t_j \in T, i \neq j, t_i \not\approx^- t_j$. Entonces Q es diagnosticable si y sólo si las siguientes condiciones son satisfechas:

1. No existen T -semiflujos evolucionando en paralelo con transiciones $t_i, t_j \in T$, tales que $t_i \approx_p t_j, t_i \in T_j^k$ y $t_j \notin T_j^k$ (es decir, que las transiciones t_i y t_j no pertenezcan a la misma clase de falta).
2. No existen T -semiflujos relacionados en concurrencia X_i y $X_j, i \neq j$, tales que $t_i^k \in \langle X_i \rangle, t_j^k \notin \langle X_j \rangle$ y éstos puedan producir las mismas secuencia de símbolos de salida y en el mismo orden una vez que han sido disparados.

Demostración. En el desarrollo de esta demostración se procede usando la forma contrapositiva del enunciado del Teorema 87.

(\rightarrow) Si existe un T -semiflujo X_i que evoluciona en paralelo con respecto a las transiciones $t_i, t_j \in T$, tal que $t_i \approx_p t_j, t_i \in T_j^k$ y $t_j \notin T_j^k$, entonces debido a que Q es una red viva existe un marcado $M_k \in R(Q, M_0), M_0 \xrightarrow{\sigma} M_k$, tal que las transiciones t_i, t_j están habilitadas simultáneamente y el T -componente T_i , inducido por X_i , está marcado. Por lo cual es posible disparar las secuencias de transiciones de longitud arbitrariamente larga $\sigma_1 = \sigma t_i \alpha^*$ y $\sigma_2 = \sigma t_j \alpha^*$ donde $\vec{\alpha} = X_i$. Como estas secuencias de disparo de transiciones no satisfacen las condiciones de la Definición 82, entonces Q no es diagnosticable.

Si existen dos T -semiflujos X_i, X_j tales que $X_i \approx_p X_j$ y $t_i^k \in \langle X_i \rangle, t_j^k \notin \langle X_j \rangle$ y éstos pueden producir los mismos símbolos de salida en el mismo orden, entonces debido a que Q es una red viva existe un marcado $M_k \in R(Q, M_0), M_0 \xrightarrow{\sigma} M_k$, tal que los T -componentes T_i, T_j inducidos por X_i y X_j , respectivamente, están simultáneamente marcados. Entonces existen dos secuencias de disparo de transiciones de longitud arbitrariamente larga $\sigma_1 = \sigma \alpha^*$ y $\sigma_2 = \sigma \beta^*$ donde $\vec{\alpha} = X_i$ y $\vec{\beta} = X_j$. Puesto que estas secuencias de disparo de transiciones no satisfacen las condiciones presentadas en la Definición 82, entonces Q no es diagnosticable.

(\leftarrow) Suponga que Q no es diagnosticable. Dado a que Q no es diagnosticable entonces $\exists \sigma_1, \sigma_2 \in \mathcal{L}(Q, M_0)$ de longitud arbitrariamente larga tales que $\sigma_1 \approx_l \sigma_2, t_j^k \in \sigma_1$ y $t_j^k \notin \sigma_2$. De la teoría de la secuencia detectabilidad en RPI, dos casos pueden generar secuencias de disparo de longitud arbitrariamente larga e indistinguibles entre sí, $\exists t_i, t_j \in T$ tales que $t_i \approx^- t_j$ o $t_i \approx_p t_j$. Por hipótesis no existen transiciones \approx^- relacionadas, entonces tienen que existir transiciones \approx_p relacionadas. Además, si $t_i \approx_p t_j$, entonces t_i y t_j tienen que pertenecer a diferentes tipos de clases de falta, de lo contrario el disparo de una transición de

falta podría ser detectado. En el peor de los casos el disparo de las transiciones \approx_p relacionadas puede ser detectado cuando la primera transición del tipo join t_k , que pertenece al mismo P -componente que estas transiciones \approx_p relacionadas, es disparada. Pero, debido a que se está suponiendo que Q no es diagnosticable esta transición del tipo join t_k no puede ser disparada. Entonces, después del disparo de una transición \approx_p relacionada las transiciones de un T -semiflujo tienen que ser infinitamente disparadas con la finalidad de evitar el disparo de la transición del tipo join t_k . Por lo tanto, tiene que existir un T -semiflujo evolucionando en paralelo con respecto a las transiciones relacionadas en concurrencia t_i y t_j para que se puedan producir secuencias de disparo de longitud arbitrariamente larga $\sigma_1 = \sigma t_i \alpha^*$ y $\sigma_2 = \sigma t_j \alpha^*$, o bien, t_i y t_j pertenecen a diferentes T -componentes que pueden ser simultáneamente marcados y el disparo de sus transiciones puede producir la misma secuencia de símbolos de salida. ■

La presencia de transiciones \approx^- relacionadas en una RPI de tipo de las LE también puede ser caracterizado, sin embargo dado a que pueden derivarse muchos casos diferentes y por no ser el tema principal en esta tesis, no será abordado.

5.3. Algoritmos

La sección anterior presentó condiciones suficientes y necesarias para la diagnosticabilidad de RPI pertenecientes a la clase de las LE . Ahora bien, en esta sección se presenta un algoritmo de tiempo polinomial que permite probar si la condición 1) del Teorema 87, la cual aparenta ser la más problemática, es satisfecha.

En el Teorema 87 como hipótesis se establece que una RPI no debe exhibir transiciones \approx^- relacionadas para probar su diagnosticabilidad. Verificar que una RPI cumple con esta condición es una labor bastante sencilla, la cual consiste en localizar los lugares de selección y corroborar que se cumplen las condiciones expuestas en las Definición 44. Por tal motivo, sólo se hace énfasis en las dos condiciones que se establecen en el Teorema 87, sobre todo en la primera condición.

La existencia o no existencia de T -semiflujos X_i y X_j relacionados en concurrencia tales que $t_i^k \in \langle X_i \rangle$ y $t_j^k \notin \langle X_i \rangle$ puede ser probado haciendo una búsqueda de T -semiflujos en el conjunto $Gr(\approx_p)$ calculado por el Algoritmo 56 presentado en el Capítulo 3, Sección 3.4, y posteriormente aplicar la relación de equivalencia de Nerode para corroborar que los símbolos de salida se produzcan en el mismo orden. Cabe señalar que dicha búsqueda no es complicada, puesto que comúnmente en las RPI la cardinalidad de $Gr(\approx_p)$ no es grande.

Con la finalidad de probar la condición de si no existen T -semiflujos evolu-

CAPÍTULO 5. DIAGNOSTICABILIDAD EN IPN

cionando en paralelo con respecto a las transiciones $t_i, t_j \in T$, tales que $t_i \approx_p t_j$, $t_i \in T_f^k$ y $t_j \notin T_f^k$ se propone el siguiente algoritmo de tiempo polinomial, el cual es una modificación del Algoritmo 62.

Algoritmo 88. Calcular si existen $(t_i, t_j) \in Gr(\approx_p)$ tal que $t_i \in T_f^k$, $t_j \notin T_f^k$ y t_i, t_j están evolucionando en paralelo con respecto a un T – *semiflujo*.

Entrada: Una RPI de la clase de las LE $Q = \{P, T, F, W\}$

Salida: La existencia de $(t_i, t_j) \in Gr(\approx_p)$ tal que $t_i \in T_f^k$, $t_j \notin T_f^k$ y t_i, t_j están evolucionando en paralelo con respecto a un T – *semiflujo*.

Comienzo

1. Calcular el conjunto $Gr(\approx_p)$ aplicando el Algoritmo 56. Si $Gr(\approx_p) = \emptyset$ entonces no continuar con el algoritmo e indicar que no hay transiciones relacionadas en concurrencia, de lo contrario continuar con el siguiente paso.
2. Calcular el conjunto $Safe_pairs = (T_{reg} \times T_{reg}) \cup (T_f^1 \times T_f^1) \cup \dots \cup (T_f^k \times T_f^k)$. /* Note que en este conjunto se encuentran los pares ordenados de transiciones que no son problemáticas entre sí desde el punto de vista de la diagnosticabilidad */
3. Calcular el conjunto $Gr_f(\approx_p) = Gr(\approx_p) - Safe_pairs$. /* el conjunto $Gr_f(\approx_p)$ contiene $(t_i, t_j) \in Gr(\approx_p)$ tal que $t_i \in T_f^k$, $t_j \notin T_f^k$, es decir, son los pares ordenados de transiciones indistinguibles que pueden conllevar a la no diagnosticabilidad. */
Si el conjunto $Gr_f(\approx_p) = \emptyset$ entonces no continuar con el algoritmo e indicar que no existen transiciones relacionadas en concurrencia que conlleven a la no diagnosticabilidad, en caso contrario proseguir con el siguiente paso.
4. Calcular el conjunto $P_{FJ} = \{(p_i, t_F, t_J) \mid p_i \text{ está residiendo dentro de un par de transiciones fork-join } t_F, t_J \text{ y } |p_i^*| > 1\}$. Si $P_{FJ} = \emptyset$ entonces no continuar con el algoritmo e indicar que las transiciones relacionadas en concurrencia t_a, t_b no están evolucionando en paralelo con un T – *semiflujo* X_a , de lo contrario continuar con el siguiente paso.
5. Calcular el conjunto $P_{FJ}^{X_a} = \{(p_i, t_F, t_J, X_a) \mid (p_i, t_F, t_J) \in P_{FJ} \text{ y existe un } T\text{-semiflujo } X_a \text{ tal que } X_a(t_F) = X_a(t_J) = 0 \text{ y } X_a(t_k) = 1, \text{ donde } t_k \in {}^*p_i\}$ es decir, existe un T – *semiflujo* X_a que también reside dentro del par de transiciones fork-join t_F, t_J . Si el conjunto $P_{FJ}^{X_a}$ no puede ser calculado,

CAPÍTULO 5. DIAGNOSTICABILIDAD EN IPN

entonces detener el algoritmo e indicar que las transiciones relacionadas en concurrencia t_a, t_b no están evolucionando en paralelo con T – *semiflujo*, de lo contrario continuar con el siguiente paso.

6. Calcular si existen dos transiciones indistinguibles que puedan ser disparadas en paralelo con algún T – *semiflujo*. Puesto que $(p_i, t_F, t_J, X_a) \in P_{FJ}^{X_a}$ contiene el T – *semiflujo* X_a el cual reside dentro de un par de transiciones fork-join, entonces las transiciones relacionadas en concurrencia t_a, t_b , $(t_a, t_b) \in Gr_f(\approx_p)$ pueden evolucionar en paralelo con respecto a X_a si los lugares de entrada de las transiciones t_a, t_b pueden ser simultáneamente marcados con el lugar p_i , es decir, el lugar p_i tiene que estar en un P – *semiflujo* mínimo diferente que los lugares $\bullet t_a, \bullet t_b$. Los siguientes problemas de programación lineal permiten determinar si este es el caso.

Para cada $(p_i, t_F, t_J, X_a) \in P_{FJ}^{X_a}$ hacer
seleccionar p_i ,

Para cada $(t_a, t_b) \in Gr_f(\approx_p)$ hacer
seleccionar t_a, t_b

$\text{Min} \sum_{i=1}^{ P } Y_1(i)$	$\text{Min} \sum_{i=1}^{ P } Y_2(i)$
s.a.	s.a.
$Y_1^T C = 0$	$Y_2^T C = 0$
$Y_1(p_i) \geq 1$	$Y_2(p_i) \geq 1$
$Y_1(p_k) \geq 0, p_k \in P - \{p_i\}$	$Y_2(p_k) \geq 0, p_k \in P - \{p_i\}$
$\sum Y_1(p_x) \geq 1, p_x \in \bullet t_a$	$\sum Y_2(p_x) \geq 1, p_x \in \bullet t_b$

Si Y_1 y Y_2 son vacíos, entonces las transiciones relacionadas en concurrencia t_a, t_b están evolucionando en paralelo con el T – *semiflujo* X_a y finalizar el algoritmo.

De lo contrario indicar que las transiciones relacionadas en concurrencia t_a, t_b no están evolucionando en concurrencia con el T – *semiflujo* X_a .

finalizar ciclo

finalizar ciclo

Fin.

CAPÍTULO 5. DIAGNOSTICABILIDAD EN IPN

Proposición 89. *Dada una RPI de la clase de las LE, la existencia de $(t_i, t_j) \in Gr(\approx_I)$ tal que t_i, t_j están evolucionando en paralelo con T - *semiflujos* puede ser determinado usando el Algoritmo 88.*

Demostración. En una RPI viva y acotada (binaria en este caso) de la clase de las LE dos lugares están marcados en el mismo marcado si y sólo si éstos pertenecen a diferentes P - *semiflujos* [12]. De acuerdo con los teoremas de las S-coberturas y las T-coberturas [12], los diferentes P - *semiflujos* son generados por transiciones t_i tales que $|t_i^*| > 1$ (es decir, transiciones del tipo fork). Por lo tanto, si dos lugares serán marcados simultáneamente, éstos tienen que pertenecer a diferentes caminos generados a partir de la transición del tipo fork. Los teoremas de las T-coberturas garantizan que los caminos generados a partir de la transición del tipo fork se unirán por una transición t_j tal que $|t_j^*| > 1$ y con lo cual el par de transiciones fork-join es formado. Por lo tanto, el algoritmo calcula los lugares de selección $|p_i^*| > 1$ que residen dentro de un par de transiciones fork-join. Los lugares de selección tienen más de una transición en su conjunto de salida, motivo por el cual éstos son los únicos candidatos para poder generar T - *semiflujos* dentro de un par de transiciones fork-join (el resto de los lugares únicamente tienen una transición de salida, las cuales junto con la transición del tipo join pertenecen al mismo T - *semiflujo*). Por lo tanto, el algoritmo hace una búsqueda de los lugares p_i tales que $|p_i^*| > 1$ y al menos una de sus transiciones de salida de este lugar esté incluida en un T - *semiflujo* que no contenga a la transición del tipo join. El conjunto de lugares p_i es calculado como el conjunto $P_{FJ}^{X_a}$ en el paso 5). Ahora bien, los T - *semiflujos* X_a pueden ser disparados en paralelo con alguna transición que pertenezca a uno de los diferentes caminos generados a partir de la transición del tipo fork t_F . Para asegurar la diagnosticabilidad de la RPI de la clase de las LE las transiciones en los diferentes caminos generados por la transición de tipo fork t_F no tienen que incluir transiciones relacionadas en concurrencia t_a, t_b , tales que $t_a \in T_f^k, t_b \notin T_f^k$, de lo contrario no es diagnosticable. Por lo tanto el algoritmo calcula (usando problemas de programación lineal) si existen P - *semiflujos* mínimos Y_1, Y_2 que contienen p_i, t_a , y p_i, t_b , respectivamente. Si dichos P - *semiflujos* mínimos no existen, entonces quiere decir que existe $(t_i, t_j) \in Gr_f(\approx_p)$ tal que t_i, t_j están evolucionando en paralelo con este T - *semiflujo*, por lo que la red Q no es diagnosticable. De lo contrario, el T - *semiflujo* X_a pertenece al mismo P - *semiflujo* mínimo que las transiciones t_a o t_b , por lo que éstas y las transiciones del T - *semiflujo* no pueden estar simultáneamente habilitadas y el disparo de las transiciones indistinguibles t_a y t_b pueden ser detectadas. ■

CAPÍTULO 5. DIAGNOSTICABILIDAD EN IPN

Proposición 90. *Dada una RPI de la clase de las LE, la existencia de un T – semi flujo evolucionando en paralelo con transiciones relacionadas en concurrencia puede ser determinado en tiempo polinomial.*

Demostración. Usando el Algoritmo 88, el cómputo del conjunto P_{FJ} es efectuado en $|P| \times |P + T|$ pasos. Con respecto al cómputo del conjunto $P_{FJ}^{X_0}$, éste es llevado a cabo en $|P_{FJ}| \times |T|^3$ pasos. El cómputo del conjunto $Gr(\approx_p)$ es realizado en $|Gr(\approx_p)|^3$ pasos. El paso 5) del algoritmo se puede calcular en $|P_{FJ}^{X_0}| \times |Gr_f(\approx_p)| \times 2|P|^3$ pasos. Por lo tanto, el algoritmo corre en $|P| \times |P + T| + |P_{FJ}| \times |T|^3 + |Gr(\approx_p)|^3 + |P_{FJ}^{X_0}| \times |Gr_f(\approx_p)| \times 2|P|^3$ pasos. ■

5.4. Ejemplos

Esta sección está dedicada a presentar algunos ejemplos en los cuales explicará el funcionamiento del Algoritmo 88 y se determinará si las RPI de la clase de las LE son diagnosticables de acuerdo a las condiciones propuestas en el Teorema 87.

Ejemplo 91. *Considere nuevamente la RPI mostrada en la Figura 5.2 y suponga que $T_f^1 = \{t_7\}$ y $T_f^2 = \{t_8\}$. A través de este ejemplo el Algoritmo 88 será seguido paso a paso. En el paso 1), haciendo uso del Algoritmo 56, el conjunto de transiciones relacionadas en concurrencia en pares ordenados es calculado, en este caso $t_6 \approx_p t_7$. Posteriormente, en el paso 3) el conjunto de pares ordenados de transiciones que pueden llevar a la no diagnosticabilidad $Gr_f(\approx_p) = \{(t_6, t_7)\}$ es calculado. En el paso 4), el conjunto de todos los lugares de selección p_i que residen dentro de un par de transiciones fork-join $P_{FJ} = \{p_3, t_1, t_8\}$ es calculado, donde p_3 es el lugar de selección y las transiciones t_1, t_8 son el par de transiciones fork-join. En el paso 5) se determina si existe un T – semi flujo que resida dentro del par de transiciones fork-join t_1, t_8 , por lo tanto el conjunto $P_{FJ}^{X_0} = \{(p_3, t_1, t_8, X_1)\}$, donde $X_1 = \overrightarrow{t_3 t_4}$ y $t_4 \in \bullet p_3$ es calculado. Debido a que no existen P – semi flujos mínimos Y_1 y Y_2 tales que contengan a $\bullet t_6$ y $\bullet t_7$, respectivamente, entonces estas transiciones relacionadas en concurrencia pueden evolucionar en paralelo con respecto al T – semi flujo X_1 . Por lo tanto Q no es diagnosticable.*

Tome en cuenta que el Algoritmo 88 únicamente permite determinar si la condición 1) del Teorema 87 es satisfecha. Sin embargo, con la finalidad de probar la diagnosticabilidad de una RPI también es necesario verificar que la condición 2) del Teorema 87 es satisfecha. En los siguientes ejemplos, se realiza dicha labor para completar la prueba de diagnosticabilidad.

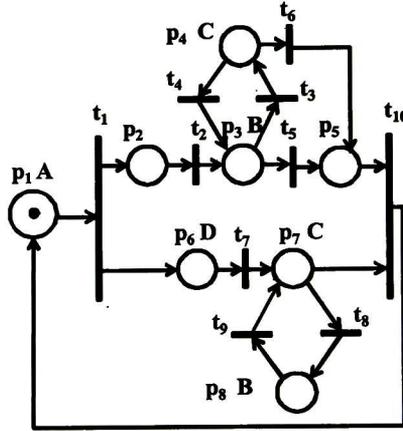


Figura 5.3: Una RPI de la clase de las LE que no diagnosticable.

Ejemplo 92. Sea (Q, M_0) una RPI viva, binaria y pura de la clase de las LE

mostrada en la Figura 5.3, donde $\Phi = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$. También

suponga que $T_f^1 = \{t_4\}$ y $T_f^2 = \{t_8\}$.

Primeramente, usando el Algoritmo 56 el conjunto de transiciones relacionadas en concurrencia $Gr(\approx_p) = \{(t_4, t_8), (t_3, t_9), (t_8, t_4), (t_9, t_3)\}$ es calculado y en el paso 3) del Algoritmo 88 el conjunto de transiciones conflictivas que podrían conllevar a la no diagnosticabilidad $Gr_f(\approx_p) = \{(t_4, t_8), (t_8, t_4)\}$ es calculado. En el paso 4), el conjunto de todos los lugares de selección p_i que residen dentro de un par de transiciones fork-join es calculado, en este caso el conjunto $P_{FJ} = \{(p_3, t_1, t_{10}), (p_7, t_1, t_{10})\}$ es calculado. En el paso 5), la existencia de un T -semiflujo X_a que también resida dentro del par de transiciones fork-join t_F, t_J es probada. Debido a que este caso es positivo, entonces el conjunto $P_{FJ}^{X_a} = \{(p_3, t_1, t_{10}, X_1), (p_7, t_1, t_{10}, X_2)\}$ es calculado, donde $X_1 = \overrightarrow{t_3 t_4}$, $X_2 = \overrightarrow{t_8 t_9}$, $t_4 \in \bullet p_3$, $t_9 \in \bullet p_7$, $t_4 \in \langle X_1 \rangle$ y $t_9 \in \langle X_2 \rangle$. Para (p_3, t_1, t_{10}, X_1) y (t_4, t_8) , se resuelven los problemas de programación lineal en el paso 6), donde existe el

CAPÍTULO 5. DIAGNOSTICABILIDAD EN IPN

P – *semiflujo* mínimo $Y_1^T = [1\ 1\ 1\ 1\ 1\ 0\ 0\ 0]$, tal que $p_3 \in \langle Y_1^T \rangle$ y $Y_1(p_4) = 1$, donde $p_4 \in {}^*t_4$. También, para (p_7, t_1, t_{10}, X_2) y (t_4, t_8) existe el P – *semiflujo* mínimo $Y_2^T = [1\ 0\ 0\ 0\ 0\ 1\ 1\ 1]$, donde $p_7 \in \langle Y_2^T \rangle$ y $Y_2(p_7) = 1$, donde $p_7 \in {}^*t_8$. Por lo tanto, debido a que ambos problemas de programación lineal tuvieron solución, entonces las transiciones relacionadas en concurrencia t_4, t_8 no pueden evolucionar en paralelo con el T – *semiflujo* X_1 ni con el T – *semiflujo* X_2 .

Para que la condición 2) del Teorema 87 sea probada se requiere calcular el conjunto de transiciones relacionadas en concurrencia o el dominio de \approx_p , es decir el conjunto $Dom(\approx_p)$. En este caso $Dom(\approx_p) = \{t_4, t_8, t_3, t_9\}$. A continuación se deben calcular dos T – *semiflujos* diferentes que puedan generar los mismos símbolos de salida usando las transiciones que pertenecen al conjunto $Dom(\approx_p)$ y un elemento de $Gr_f(\approx_p)$, este paso podría ser llevado a cabo resolviendo un problema de programación lineal. Por lo tanto, debido a que existen $X_1 = \overrightarrow{t_3 t_4}$ y $X_2 = \overrightarrow{t_8 t_9}$, los cuales pueden generar los mismo símbolos de salida, donde $\langle X_1 \rangle, \langle X_2 \rangle \subset Dom(\approx_p)$, $\langle X_1 \rangle \cap \langle X_2 \rangle = \emptyset$ y $(t_4, t_8) \in Gr_f(\approx_p)$. Posteriormente, es necesario verificar que los símbolos de salida de X_1 y X_2 pueden producirse en el mismo orden, usando la relación de Nerode y su respectiva tabla es fácil de verificar. Puesto que X_1 y X_2 efectivamente pueden producir los símbolos de salida en el mismo orden y $t_4^1 \in \langle X_1 \rangle, t_8^2 \in \langle X_2 \rangle$ (es decir, las transiciones t_4 y t_8 pertenecen a diferentes tipos de falta), entonces Q no es diagnosticable. De hecho, cuando el marcado $M_k^T = [0\ 0\ 0\ 1\ 0\ 0\ 1\ 0]$ es alcanzado, no es posible distinguir la secuencia de disparo de transiciones $\sigma_1 = (t_3 t_4)^*$ de la secuencia $\sigma_2 = (t_9 t_8)^*$, por lo que no se tendría el conocimiento si ocurrió la falta de la clase 1, la de clase 2 o ambas.

Ejemplo 93. Sea (Q, M_0) una RPI viva, binaria y pura perteneciente a la clase

de las LE mostrada en la Figura 5.4, donde $\Phi = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$ También

suponga que $T_f^1 = \{t_2\}$ y $T_f^2 = \{t_3\}$.

Haciendo uso del Algoritmo 88 los conjuntos $Gr(\approx_p) = \{(t_2, t_3), (t_3, t_2)\} = Gr_f(\approx_p)$ son calculados en el paso 1) y 3), respectivamente. En el paso 4), el conjunto $P_{FJ} = \emptyset$, por lo que no existen lugares de selección p_i residiendo dentro del par de transiciones fork-join. Por lo tanto, las transiciones relacionadas en concurrencia t_2, t_3 no están evolucionando en paralelo con algún T – *semiflujo* X_a .

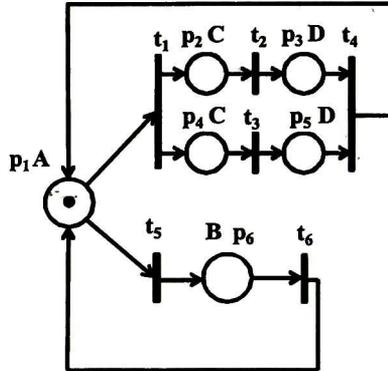


Figura 5.4: Una RPI diagnosticable.

Para este ejemplo se tiene que $Dom(\approx_p) = \{t_2, t_3\}$. No obstante, usando las transiciones pertenecientes al conjunto $Dom(\approx_p)$ no es posible construir ningún T -semi flujo. Por lo tanto, no existen T -semi flujos X_i y X_j , $i \neq j$ que estén relacionados en concurrencia. Por lo tanto, debido a que las condiciones 1) y 2) del Teorema 87 son satisfechas entonces la RPI Q es diagnosticable.

5.5. Conclusiones

En este capítulo se abordó el tema de la diagnosticabilidad en las RPI pertenecientes a la clase de las LE. Primeramente se mostró que la secuencia detectabilidad (propiedad que puede ser verificada en tiempo polinomial) es una condición suficiente para la diagnosticabilidad. Posteriormente, se caracterizó la diagnosticabilidad en las RPI que presentan la relación de tipo \approx^- ; pero que podrían ser o no SD, para lo cual se utilizaron algunas de las relaciones y estructuras de red que permitieron caracterizar a la secuencia detectabilidad.

Asimismo se presentó un algoritmo de tiempo polinomial que permite satisfacer la condición más problemática en la caracterización de la diagnosticabilidad de las RPI de la clase de las LE, como también su correctitud y su respectivo análisis de complejidad.

6. CONCLUSIONES Y TRABAJO FUTURO

6.1. Conclusiones

Esta tesis abordó el problema de la posibilidad de distinguir las secuencias de disparo de transiciones basándose en la información de salida que proporciona el modelo en *RPI* del sistema, ya sea a partir de un marcado inicial vivo y binario desconocido (secuencia detectabilidad estructural) o bien, de un marcado inicial conocido (secuencia detectabilidad). Para alcanzar dicho fin primeramente se definieron formalmente ambos conceptos y posteriormente se dio su caracterización.

Para llevar a cabo la caracterización de la secuencia detectabilidad estructural y la secuencia detectabilidad, en redes vivas, binarias y puras; se propusieron algunas relaciones sobre el conjunto de transiciones y se utilizó el análisis de ciertas estructuras como los *P-semiflujos*, los *T-semiflujos* y los lugares de selección. Por tal motivo, dicho análisis estructural permitió el desarrollo de algoritmos de tiempo polinomial para verificar en la *RPI* algunas de las condiciones para determinar la propiedad de *SSD*. Asimismo se presentaron algoritmos de tiempo polinomial para verificar la propiedad de secuencia detectabilidad, con lo cual se evitó el desarrollo del grafo de alcanzabilidad de la *RPI* que hubiese desencadenado algoritmos de tiempo no polinomial. Desafortunadamente, para una de las condiciones para analizar las *SSD* aún no se tienen algoritmos eficientes.

Como aplicación al área de la observabilidad, se propusieron el *PVO* y el *CFTSO*, que permiten determinar el vector de Parikh de la secuencia de transiciones disparada y la secuencia de disparo de transiciones, respectivamente, para redes que sean *SSD*. Asimismo, se establecieron las reglas de evolución y marcado para dichas entidades y la manera de medir el error para garantizar su convergencia. También, se hizo hincapié en que no es el mismo problema el determinar la secuencia de disparo de transiciones que determinar el marcado actual de la red, ya que son dos propiedades independientes.

Con respecto a la diagnosticabilidad de *RPI*, se demostró formalmente que

CAPÍTULO 7. CONCLUSIONES Y TRABAJO FUTURO

la secuencia detectabilidad es una condición suficiente para la diagnosticabilidad de *RPI* vivas, binarias y puras pertenecientes a la clase de las *LE*. También, se presentó una caracterización de la diagnosticabilidad para las redes no *SD*, la cual hace uso de las mismas relaciones y estructuras analizadas en la caracterización de la secuencia detectabilidad. Cabe señalar, que en los resultados obtenidos se presentaron condiciones suficientes y necesarias, además de cubrir una amplia gama de redes, aunado a que se propusieron algoritmos de tiempo polinomial para ello; pues en la literatura existente muchos algoritmos son de tiempo no polinomial o sus caracterizaciones son muy limitadas.

6.2. Trabajo Futuro

De acuerdo a los resultados que se han presentado a lo largo de esta tesis se podrían desarrollar las siguientes líneas de investigación:

- Extender la caracterización de la secuencia detectabilidad y secuencia detectabilidad estructural a redes más complejas que las *LE*. Cabe señalar, que para ello ya se tienen algunos resultados y demostraciones inconclusas, los cuales no se reportaron en esta tesis.
- Encontrar algoritmos de complejidad teórica polinomial en tiempo para verificar la existencia de *T – componentes* indistinguibles entre sí o demostrar que es un problema no resoluble en tiempo polinomial.
- Relajar las hipótesis de trabajo para determinar la secuencia detectabilidad estructural en más redes. Para tal efecto, se tendrían que caracterizar más casos que afectan a la secuencia detectabilidad estructural y proponer algoritmos de tiempo polinomial (en caso de existir) para validar las nuevas condiciones que surjan.
- Caracterizar el marcado detectable haciendo uso de las mismas relaciones con las que se caracterizó la secuencia detectabilidad estructural.
- Juntar los resultados en el área de diagnóstico presentados en [18] con los propuestos en esta tesis para cubrir una cantidad de redes más amplia.
- Formalizar la caracterización de la diagnosticabilidad en las *RPI* que presenten la relación \approx^- . Asimismo proponer algoritmos que permitan satisfacer las condiciones que se deriven de ello.

CAPÍTULO 7. CONCLUSIONES Y TRABAJO FUTURO

- Aplicar la teoría de secuencia detectabilidad estructural y secuencia detectabilidad al área de control para la implementación de nuevos controladores.
- Usar la teoría de secuencia detectabilidad estructural en el área de identificación.
- Proponer más aplicaciones, en el área de observabilidad, para la secuencia detectabilidad estructural.

CAPÍTULO 7. CONCLUSIONES Y TRABAJO FUTURO

BIBLIOGRAFÍA

- [1] L. Aguirre, A. Ramírez, and O. Begovich, "Design of asymptotic observers for discrete event systems," *Proc. IASTED International Conference on Intelligent Systems and Control*, pp. 188–193, October 1999.
- [2] L. Aguirre-Salas, O. Begovich, and A. Ramirez-Treviño, "Optimal sensor choice for observability in free-choice Petri nets," *Proceedings of the 2001 IEEE International Symposium on Intelligent Control*, pp. 270–275, September 2001.
- [3] —, "Observability in interpreted Petri nets using sequence invariants," *Proceedings of the 41st IEEE Conference on Decision and Control, 2002*, vol. 4, pp. 3602–3607, December 2002.
- [4] L. Aguirre-Salas and A. Santoyo-Sanchez, "Sequence-detectability analysis of interpreted Petri nets under partial state observations," *IEEE Conference on Emerging Technologies Factory Automation*, pp. 1 – 7, 2009.
- [5] F. Basile, P. Chiacchio, and G. D. Tommasi, "Sufficient conditions for diagnosability of Petri nets," *Proceedings of the 9th International Workshop on Discrete Event Systems*, pp. 370–375, 2008.
- [6] M. Cabasino, A. Giua, S. Lafortune, and C. Seatzu, "Diagnosability analysis of unbounded Petri nets," *IEEE Proceedings of the 48th Conference on Decision and Control, 2009, held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009.*, pp. 1267 – 1272, December 2009.
- [7] M. P. Cabasino, A. Giua, S. LaFortune, and C. Seatzu, "A new approach for diagnosability analysis of Petri nets using verifier nets." *IEEE Transactions on Automatic Control*, vol. 57, no. 12, pp. 3104 – 3117, March 2013.

CAPÍTULO 7. CONCLUSIONES Y TRABAJO FUTURO

- [8] M. P. Cabasino, A. Giua, and C. Seatzu, “Diagnosis using labeled Petri nets with silent or undistinguishable fault events,” *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 13, no. 2, pp. 345 – 355, December 2012.
- [9] M. Cabasino, A. Giua, and C. Seatzu, “Diagnosability of bounded Petri nets,” *Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference Shanghai, P.R. China*, pp. 1254 – 1260, December 2009.
- [10] —, “Diagnosis of discrete event systems using labeled Petri nets,” *Proc. 2nd IFAC Workshop on Dependable Control of Discrete Systems*, June 2009.
- [11] C. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*, 2nd ed. Springer, 2008.
- [12] J. Desel and J. Esparza, *Free Choice Petri Nets*. Cambridge University Press, 1995.
- [13] F. DiCesare, G. Harhalakis, J. M. Proth, M. Silva, and F. B. Vernadat, *Practice of Petri Nets in Manufacturing*. Chapman Hall, 1993.
- [14] M. Dotoli, M. Fanti, and A. Mangini., “Fault detection of discrete event systems using Petri nets and integer linear programming.” *Proc. of 17th IFAC World Congress, Seoul, Korea*, pp. 6554 – 6559, July 2008.
- [15] A. Giua and C. Seatzu, “Design of observers/controllers for discrete event systems using Petri nets,” *Synthesis and Control of Discrete Event Systems / B. Caillaud, X. Xie, Ph. Darondeau and L. Lavagno (Eds.)*, pp. 167–182, 2001.
- [16] —, “Observability of place/transition nets,” *IEEE Transactions on Automatic Control*, vol. 47, no. 9, pp. 1424–1437, September 2002.
- [17] —, “Fault detection for discrete event systems using Petri nets with unobservable transitions,” *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, pp. 6323–6328, December 2005.
- [18] E. Hernández-Flores, “Diagnóstico y predicción de fallas en redes de Petri interpretadas,” Ph.D. dissertation, Centro de Investigación y de Estudios Avanzados del I.P.N. Unidad Guadalajara, Noviembre 2012.

CAPÍTULO 7. CONCLUSIONES Y TRABAJO FUTURO

- [19] E. Hernández-Flores, E. López-Mellado, and A. Ramírez-Treviño, “Diagnosability analysis of partially observable deadlock-free Petri nets,” *3rd International Workshop on Dependable Control of Discrete Systems (DCDS)*, pp. 174 – 179, 2011.
- [20] J. Hopcroft and J. Ullman, *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.
- [21] K. M. Inan and P. P. Varaiya, “Algebras of discrete event models,” *Proceedings of the IEEE*, vol. 77, no. 1, pp. 24 – 38, January 1989.
- [22] S. Jiang, Z. Huang, V. Chandra, and R. Kumar, “A polynomial algorithm for testing diagnosability of discrete-event systems,” *IEEE Transactions on Automatic Control*, vol. 46, no. 8, pp. 1318 – 1321, August 2001.
- [23] G. Jiroveanu and R. Boel, “The diagnosability of Petri net models using minimal explanations,” *IEEE Transactions on Automatic Control*, vol. 5, no. 7, pp. 1663 – 1668, July 2010.
- [24] N. Karmarkar, “A new polynomial-time algorithm for linear programming,” *Combinatorica*, vol. 4, no. 4, pp. 373–395, December 1984.
- [25] D. Lefebvre and C. Delherm, “Diagnosis of DES with Petri net models.” *IEEE Trans. on Automation Science and Engineering*, vol. 4, no. 1, pp. 114–118, 2007.
- [26] L. Li and C.N.Hadjicostis, “Reconstruction of transition firing sequences based on asynchronous observations of place token changes,” *Proceedings of the 46th IEEE Conference on Decision and Control*, pp. 1898–1903, December 2007.
- [27] J. Lunze and J. Schroder, “Sensor and actuator fault diagnosis of systems with discrete inputs and outputs,” *IEEE Trans. Systems, Man and Cybernetics, Part B*, vol. 34, pp. 1096 – 1107, 2004.
- [28] M. E. Meda-Campaña, F. J. Lopez-Lopez, C. Lopez-Martin, and A. Chavoya, “Identification of Petri net models based on an asymptotic approach,” *ISDA '09 Proceedings of the 2009 Ninth International Conference on Intelligent Systems Design and Applications*, pp. 749–754, November 2009.

CAPÍTULO 7. CONCLUSIONES Y TRABAJO FUTURO

- [29] M. E. Meda-Campaña and S. Medina-Vazquez, "Synthesis of timed Petri net models for on-line identification of discrete event systems," *9th IEEE International Conference on Control and Automation ICCA*, pp. 1201–1206, 2011.
- [30] M. Meda-Campaña and E. Lopez-Mellado, "Identification of concurrent discrete event systems using Petri nets," *2005 IMACS: Mathematical Computer, Modelling and Simulation Conference*, July 2005.
- [31] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of IEEE*, pp. 541–580, April 1989.
- [32] C. Özveren and A. S. Willsky, "Invertibility of discrete-event dynamic systems," *Mathematics of Control, Signals, and Systems*, pp. 365–390, December 1992.
- [33] J. Peterson, "Petri nets," *Computing Surveys*, pp. 223–252, September 1977.
- [34] W. Qiu and R. Kumar, "Decentralized failure diagnosis of discrete event systems," *IEEE Transactions on Systems, Man and Cybernetics - PART A: Systems and Humans*, vol. 36, no. 2, pp. 384 – 395, 2006.
- [35] P. Ramadge and W. M. Wonham, "Modular supervisory control of discrete-event systems," *Mathematics of Control, Signals, and Systems*, vol. 1, pp. 13–30, 1988.
- [36] —, "The control of discrete event systems," *Proceedings of the IEEE*, vol. 77, no. 1, pp. 81–98, January 1989.
- [37] G. Ramirez-Prado, A. Santoyo, A. Ramirez-Treviño, and O. Begovich, "Regulation problem in discrete event systems using interpreted Petri nets," *IEEE International Conference on In Systems, Man, and Cybernetics, 2000*, vol. 3, pp. 2174–2179, October 2000.
- [38] A. Ramírez-Treviño, I. Rivera-Rangel, and E. López-Mellado, "Observer design for discrete event systems modeled by interpreted Petri nets," *IEEE International Conference on Robotic and Automation*, vol. 3, no. 4, pp. 2871–2876, 2000.

CAPÍTULO 7. CONCLUSIONES Y TRABAJO FUTURO

- [39] A. Ramírez-Treviño, E. Ruiz-Beltrán, J. Arámburo-Lizárraga, and E. López-Mellado., “Structural diagnosability of DES and design of reduced Petri net diagnosers.” *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 42, no. 2, March 2012.
- [40] A. Ramírez-Treviño, E. Ruiz-Beltrán, I. Rivera-Rangel, and E. López-Mellado, “Online fault diagnosis of discrete event systems. a Petri net based approach,” *IEEE Transactions on Automation Science and Engineering*, vol. 4, no. 1, pp. 31 – 39, 2007.
- [41] A. Ramírez-Treviño, I. Rivera-Rangel, and E. López-Mellado, “Observability of discrete event systems modeled by interpreted Petri nets,” *IEEE Transactions on Robotics and Automation.*, vol. 19, no. 4, pp. 557–565, August 2003.
- [42] J. Rangel, “Observability and modular synthesis of Petri net models of discrete event systems,” Ph.D. dissertation, Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, 2004.
- [43] I. Rivera-Rangel, A. Ramírez-Treviño, L. Aguirre-Salas, and J. Ruíz-León, “Geometrical characterization of observability in interpreted Petri nets,” *Kybernetika*, vol. 41, no. 5, pp. 553 – 574, 2005.
- [44] R. C. Rodríguez, “Algoritmos para el control de sistemas de eventos discretos bajo observación parcial del estado,” Ph.D. dissertation, Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, Junio 2007.
- [45] Y. Ru and C. Hadjicostis, “Sensor selection for structural observability in discrete event systems modeled by Petri nets,” *IEEE Transactions on Automatic Control*, vol. 55, no. 8, pp. 1751–1764, August 2010.
- [46] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, second edition ed. Pearson Education, Inc., 2003.
- [47] M. Sampath, S. Lafortune, and D. Teneketzis, “Active diagnosis of discrete-event systems,” *IEEE Trans. on Automatic Control*, vol. 43, no. 7, pp. 908 – 929, 1998.

CAPÍTULO 7. CONCLUSIONES Y TRABAJO FUTURO

- [48] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneket-zis, “Diagnosability of discrete event systems,” *IEEE Transactions on Auto-matic Control*, vol. 40, no. 9, pp. 1555–1575, 1995.
- [49] J. Sánchez-Blanco, A. Ramírez-Treviño, and A. Santoyo, “Multiple spec-ification regulation control in interpreted Petri nets,” *IEEE International Conference on Systems, Man and Cybernetics*, vol. 5, pp. 4989–4994, 2004.
- [50] —, “Regulation control in interpreted Petri nets using trace equivalence,” *IEEE Conference on Systems, Man and Cybernetics*, vol. 2, pp. 1843–1848, 2004.
- [51] S. Shu and F. Lin, “Generalized detectability for discrete event systems,” *Systems Control Letters*, vol. 60, no. 5, pp. 310 – 317, 2011.
- [52] S. Shu and F. Lin:, “I-detectability of discrete-event systems,” *IEEE Trans-actions on Automation Science and Engineering*, vol. 10, no. 1, pp. 187–196, 2013.
- [53] S. Shu, F. Lin, and H. Ying, “Detectability of discrete event systems,” *De-tectability of discrete event systems*, vol. 52, no. 12, pp. 2356–2359, December 2007.
- [54] M. Silva, *Las Redes de Petri En la Automática y la Informática*. AC, 1985.
- [55] W. W. Y. Lin, “Controllability and observability in the state-feedback control of discrete-event systems,” *Proc. 27th Conf. on Decision and Control*, p. 203Ú207, December 1988.
- [56] T. S. Yoo and H. Garcia, “Diagnosis of behaviors of interest in partially-observed discrete-event systems,” *Systems and Control Letters*, vol. 57, no. 12, pp. 1023 – 1029, 2008.

7. APÉNDICE A

En el paso 1) del Algoritmo 62 indica que es necesario calcular el conjunto $P_{FJ} = \{(p_i, t_F, t_J) \mid p_i \text{ está residiendo dentro de un par de transiciones fork-join } t_F, t_J \text{ y } |p_i \bullet| > 1\}$. En este apéndice se proporciona el algoritmo para llevar a cabo esta tarea.

Brevemente, el algoritmo funciona de la siguiente manera:

- Primeramente, un lugar de selección p_i es seleccionado y entonces es necesario realizar una búsqueda hacia atrás de éste para localizar la transición del tipo fork y una búsqueda hacia adelante para localizar la transición de tipo join.
- Estas búsquedas son efectuadas etiquetando el conjunto de transiciones de entrada y de salida del lugar de selección. Entonces, el lugar de selección es removido de la *RPI*, por lo que algunas transiciones se convierten en sumidero/fuente.
- Los conjuntos de transiciones de entrada y salida de las transiciones sumidero/fuente, respectivamente, son etiquetadas y entonces las transiciones sumidero/fuente son removidas junto con sus lugares de entrada y salida.
- El proceso de etiquetado y el paso remoción de transiciones sumidero/fuente es repetido hasta que no haya más transiciones sumidero/fuente. Entonces, las transiciones que no se hayan borrado y estén etiquetadas serán el par de transiciones fork-join.

En las siguientes líneas el algoritmo es presentado.

Algoritmo 94. *Entrada:* Una RPI de la clase de las LE $Q = \{P, T, \Phi, F, W\}$

Salida: Si un lugar de selección p_i reside dentro de un par de transiciones fork-join t_F, t_J

Para cada lugar de selección p_i hacer

Calcular $Q_2 = Q / *$ es decir, calcular $P_2 = P, T_2 = T, F_2 = F, W_2 = W *$ /

En el conjunto T_2 agregar una etiqueta I a cada transición que pertenezca al conjunto $\bullet(p_i)$ y una etiqueta O a cada transición que pertenezca al conjunto p_i^\bullet .

Calcular $P_{FJ} = \emptyset$

Calcular $P_2 = P_2 - \bullet(p_i)$

Calcular $T_s = \{t \in T_2 \mid t \text{ es una transición sumidero/fuente}\}; R = \emptyset$

Mientras que $T_s \neq \emptyset$ hacer

{

{Para cada $t \in T_s$ hacer

si $t \in T_2$ es una transición sumidero agregar una etiqueta I a cada transición de su conjunto de entrada $\bullet(t)$

si $t \in T_2$ es una transición fuente agregar una etiqueta O a cada transición de su conjunto de salida $(t)^\bullet$

Calcular $R = R \cup T_s$

Calcular $T_2 = T_2 - T_s$ y $P_2 = P_2 - (t^\bullet \cup \bullet t), \forall t \in T_s$

Calcular $T_s = \{t \in T_2 \mid t \text{ es una transición sumidero/fuente}\}$

}

Si $R = T$, entonces el lugar de selección p_i está fuera del par de transiciones fork-join.

De lo contrario calcular $P_{FJ} = P_{FJ} \cup \{(p_i, t_F, t_J)\}$, donde t_F son las transiciones que permanezcan en T_2 con la etiqueta I y t_J son las transiciones que permanezcan en T_2 con la etiqueta O .

Fin del ciclo hacer

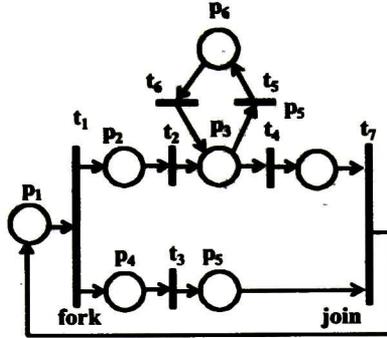


Figura 7.1: Una LE con un lugar de distribución p_i que reside dentro de un par de transiciones fork-join.

Claramente, el algoritmo presentado es de tiempo polinomial con una complejidad teórica de $(\text{número de lugares de distribución}) \times |T|$. Este algoritmo será aplicado en los siguientes ejemplos.

Ejemplo 95. Considere la RPI Q mostrada en la Figura 7.1.

Primeramente, el lugar de selección p_3 es seleccionado.

Entonces las transiciones t_2, t_6 serán etiquetadas con I y t_4, t_5 con O .

Cuando el lugar p_3 es removido de Q_2 , entonces $T_s = \{t_2, t_4, t_5, t_6\}$.

Posteriormente, las transiciones t_1, t_5 serán etiquetadas con I y t_6, t_7 con O .

Entonces, el conjunto $R = \{t_2, t_4, t_5, t_6\}$ es calculado y las t_2, t_4, t_5, t_6 junto con sus lugares de entrada y salida serán removidos de Q_2 . Entonces, el conjunto T_s tiene que actualizarse, lo que conduce a $T_s = \emptyset$.

Puesto que $R \neq T$, entonces se calcula $P_{FJ} = P_{FJ} \cup \{(p_i, t_F, t_J)\}$, en este caso $P_{FJ} = P_{FJ} \cup \{(p_3, t_1, t_7)\}$ porque $t_1, t_7 \in T_2$ y t_1 está etiquetada con una I y t_7 etiquetada con una O .

Ejemplo 96. Considere la RPI Q mostrada en la Figura 7.2. Claramente, siguiendo el algoritmo anterior todas las transiciones serán removidas de Q_2 . En este caso $R = T$, por lo tanto el lugar de selección p_1 está residiendo por fuera del par de transiciones fork-join.

En el paso 2) del Algoritmo 62 indica que es necesario calcular el conjunto $P_{FJ}^{X_a} = \{(p_i, t_F, t_J, X_a) | (p_i, t_F, t_J) \in P_{FJ} \text{ y existe un } T\text{-semiflujo } X_a \text{ tal que}$

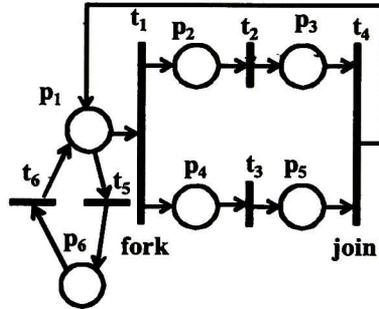


Figura 7.2: Una LE con un lugar de distribución p_i que reside por fuera de un par de transiciones fork-join.

$X_a(t_F) = X_a(t_J) = 0$ y $X_a(t_k) = 1$, donde $t_k \in \bullet p_i$. Esta tarea puede llevarse a cabo resolviendo el siguiente problema de programación lineal.

Algoritmo 97. Entrada: Una RPI de la clase de las LE $Q = \{P, T, \Phi, F, W\}$ y el conjunto P_{FJ}

Salida: el conjunto $P_{FJ}^{X_a}$

$P_{FJ}^{X_a} = \emptyset$

Para cada $(p_i, t_F, t_J) \in P_{FJ}$ hacer

$\text{Min } \sum_{i=1}^{ T } X_a(i)$
s.a.
$CX_a = 0$
$X_a > \vec{0}$
$X_a(t_F) = 0$
$X_a(t_J) = 0$
$X_a(t_k) = 1$ y $t_k \in \bullet(p_i)$

Si este problema tiene solución $P_{FJ}^{X_a} = P_{FJ}^{X_a} \cup (p_i, t_F, t_J, X_a)$

De lo contrario $P_{FJ}^{X_a} = P_{FJ}^{X_a}$

Finalizar ciclo

Ejemplo 98. Considere nuevamente la RPI Q mostrada en la Figura 7.1.

En el Ejemplo 95 se calculó que $P_{FJ} = \{(p_3, t_1, t_7)\}$, es decir el lugar de selección p_3 está dentro del par de transiciones fork-join t_1, t_7 . Haciendo uso del Algoritmo 97 se tiene que en un comienzo $P_{FJ}^{X_0} = \emptyset$. Posteriormente, se ejecuta el problema de programación lineal propuesto para el elemento (p_3, t_1, t_7) , donde se concluye que existe el T - *semiflujo* $X_1 = [0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0]^T$ tal que:

$$CX_1 = 0$$

$$X_1 > \vec{0}$$

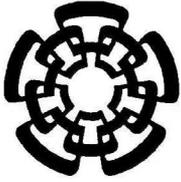
$$X_1(t_1) = 0$$

$$X_1(t_7) = 0$$

$$X_1(t_6) = 1 \text{ y } t_6 \in \bullet(p_3)$$

Por lo tanto $P_{FJ}^{X_0} = P_{FJ}^{X_0} \cup (p_3, t_1, t_7, X_1)$.

Como para este ejemplo $P_{FJ}^{X_0} \neq \emptyset$, entonces esto indica que existe un T - *semiflujo* dentro del un par de transiciones fork-join.



CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS DEL I.P.N. UNIDAD GUADALAJARA

El Jurado designado por la Unidad Guadalajara del Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional aprobó la tesis

Secuencia Detectabilidad Estructural en Redes de Petri Interpretadas

del (Ia) C.

Saúl Alonso NUÑO SÁNCHEZ

el día 23 de Agosto de 2013.

Dr. Luis Ernesto López Mellado
Investigador CINVESTAV 3C
CINVESTAV Unidad Guadalajara

Dra. Ofelia Begovich Mendoza
Investigador CINVESTAV 3B
CINVESTAV Unidad Guadalajara

Dr. José Javier Ruíz León
Investigador CINVESTAV 3B
CINVESTAV Unidad Guadalajara

Dr. Antonio Ramírez Treviño
Investigador CINVESTAV 3A
CINVESTAV Unidad Guadalajara

Dr. Carlos Renato Vazquez Topete
Profesor Investigador
Centro Universitario de los Valles,
UdG



CINVESTAV - IPN
Biblioteca Central



SSIT0012017