

Sistema en lazo cerrado para el diagnóstico de fallos de sistemas de eventos discretos utilizando redes de Petri interpretadas

Closed loop system for discrete event system failure diagnosis using interpreted Petri nets

Klever S. Arrechea-Castillo ^{1a}, Óscar J. Muriel-Velazques ^{1b}, Mariela Muñoz-Añasco ^{2a}, Wilber Acuña-Bravo ^{2b}

¹ Universidad del Cauca, Colombia. Correo electrónico: cearrechea@unicauca.edu.co ^a, oscarjmuriel@unicauca.edu.co ^b.

² Grupo de investigación en Automática, Departamento de Electrónica Instrumentación y Control, Universidad del Cauca, Colombia. Correo electrónico: mamunoz@unicauca.edu.co ^a, wacuna@unicauca.edu.co ^b
Orcid: [0000-0003-4392-947X](https://orcid.org/0000-0003-4392-947X) ^a, [0000-0002-6388-3897](https://orcid.org/0000-0002-6388-3897) ^b

Recibido: 20 enero, 2022. Aceptado: 21 mayo, 2022. Versión final: 24 junio, 2022.

Resumen

Este trabajo presenta la implementación de un diagnosticador de fallos en línea, para la detección de fallos operacionales presentes en la planta piloto de procesamiento de aguas, en el marco del proyecto REAGRITECH de la cátedra Unesco de sostenibilidad, modelada como un sistema de eventos discretos (SED), haciendo uso de redes de Petri interpretadas (IPN). Se realiza un análisis de funcionamiento del sistema real, del cual se identifican algunas escenas de funcionamiento, que permiten la construcción de un controlador encargado de representar, de forma simulada, el comportamiento real de la planta. A partir de este, se obtiene una matriz de información de entradas-salidas (E/S), donde las entradas corresponden a las señales de mando de control, y las salidas son las señales de sensores. Estos datos se ingresan como parámetros a un algoritmo de identificación, que entrega como resultado el modelo de la IPN identificado a partir de los datos, es decir, información de una IPN: matriz de incidencia A , función de entrada (F_e) con las etiquetas asociadas a las transiciones, la función de salida (F_s) y una matriz de salida φ que relaciona los sensores con cada plaza del sistema. Con base en el modelo identificado se realiza un análisis de detectabilidad para saber si el sistema es detectable o no por eventos, y se construye un diagnosticador a partir de fallos predefinidos. Se concluye que el diagnosticador implementado es capaz de detectar fallos presentes en la planta piloto de procesamiento de agua, en el marco del proyecto REAGRITECH, haciendo un análisis del marcado en los lugares de diagnóstico y en los lugares *Post* de riesgos del sistema.

Palabras clave: sistemas de eventos discretos (SED); redes de Petri interpretadas (IPN); matriz de incidencia; diagnosticador de fallos; control de nivel; humedal artificial; identificación de sistemas; detectabilidad de eventos; automatización de procesos; observador de estados.

Abstract

This paper presents the implementation of an online fault diagnostics, for the detection of operational faults present in the pilot water processing plant, within the framework of the REAGRITECH project of the UNESCO chair of sustainability, modeled as a system of discrete events (SED), making use of interpreted Petri nets (IPN). An analysis of the operation of the real system is carried out, from which some operating scenes are identified, which allow the

ISSN impreso: 1657 - 4583. ISSN en línea: 2145 – 8456.

Este es un artículo de acceso abierto bajo la licencia [CC BY-ND 4.0](https://creativecommons.org/licenses/by-nd/4.0/)



Como citar: K. S. Arrechea-Castillo, O. J. Muriel-Velázquez, M. Muñoz-Añasco, W. Acuña-Bravo, “Sistema en lazo cerrado para el diagnóstico de fallos de sistemas de eventos discretos utilizando redes de Petri interpretadas,” *Rev. UIS Ing.*, vol. 21, no. 3, pp. 55-70, 2022, doi: <https://doi.org/10.18273/revuin.v21n3-2022005>.

construction of a controller in charge of representing, in a simulated way, the real behavior of the plant. From this, an input-output (I/O) information matrix is obtained, where the inputs correspond to the control command signals, and the outputs are the sensor signals. These data are entered as parameters to an identification algorithm, which delivers as a result the model of the IPN identified from the data, that is, information of an IPN: incidence matrix A , input function (F_e) with the labels associated with the transitions, the output function (F_s) and an output matrix φ that relates the sensors to each slot in the system. Based on the identified model, a detectability analysis is carried out to find out if the system is detectable or not by events, building a diagnostic from predefined faults. It is concluded that the diagnostic tool implemented is capable of detecting faults present in the pilot water processing plant, within the framework of the REAGRITTECH project, making an analysis of the marking in the diagnostic places, and in the Post risk places of the system.

Keywords: Discrete Event Systems (DES); Interpreted Petri Nets (IPN); matrix of incidence; fault diagnosis; level control; constructed wetland; system identification; detectability of events; automation of processes; observer of states.

1. Introducción

En la actualidad, se usan herramientas flexibles capaces de percibir las dinámicas del entorno; un ejemplo es el uso y aplicación de la inteligencia artificial (IA) para solucionar problemas de detectabilidad y diagnóstico de fallos presentes en sistemas de eventos discretos [1], [2] como plantas químicas, reactores nucleares, aeronaves o la aplicabilidad en sistemas de vehículos inteligentes. Los sistemas de diagnóstico y aislamiento de fallos han sido estudiados desde hace algunos años, debido a problemas de seguridad y confiabilidad presentes en máquinas, procesos o elementos de control [3].

Un fallo es una alteración del comportamiento normal de un sistema. Esta desviación afecta la calidad final de un producto o la terminación de un ciclo previamente definido. El diagnóstico de fallos es la técnica de revelar, identificar y separar las desviaciones presentes, determinando las causas que las producen, a partir de la información disponible de las variables del sistema [4]. La detección es una función que decide las condiciones de trabajo del sistema; esta detecta si se encuentra en modo normal o en modo de fallo [5]. En el campo industrial, la naturaleza compleja de los sistemas no solo aumenta el potencial de la ocurrencia de fallos, sino que también hace que el problema de diagnóstico sea más difícil y desafiante [6].

Los sistemas de detección de fallos juegan un papel importante en la protección de la vida y de las propiedades físicas de un dispositivo, al igual que en el aumento de la productividad y del tiempo operativo de un dispositivo o sistema [7]. Solucionar los problemas de diagnóstico de sistemas de gran escala es una tarea compleja que requiere de un enfoque sistemático [7], que lleve a métodos que resuelvan el problema.

Un factor importante en el diagnóstico de fallos es su detección, lo cual se puede hacer por medio de un operario encargado de analizar los reportes de alarmas

generados por la activación de sensores, o de forma automática por medio de la implementación de algoritmos que permitan monitorizar y analizar los datos de forma oportuna. La monitorización realizada por operarios en ocasiones es deficiente debido a las limitaciones del personal encargado de analizar y tomar decisiones, según la información brindada por el sistema. Las limitaciones de los operarios en cuanto a la capacidad de procesar la información emitida hace que los mecanismos automatizados para el diagnóstico de fallos sean esenciales [6].

En el diagnóstico de fallos se han propuesto variedades de esquemas que difieren tanto en su marco teórico como en su metodología de diseño e implementación [8]. Los procesos de diagnósticos pueden ser abordados con base en expertos, útiles en situaciones donde es complejo conseguir un modelo de la planta, o con base en modelos [9]. Gran parte de los procesos de diagnóstico basados en modelos se realizan con variables continuas, como las ecuaciones diferenciales. Para el diagnóstico de fallos drásticos que se aborda en este trabajo, los modelos detallados de variables continuas son innecesarios. Los modelos de diagnóstico mencionados a continuación son utilizados para el diagnóstico en SED, los cuales se pueden clasificar como (1) enfoque basado en experto y (2) enfoque basado en modelos (enfoque basado en árbol de fallo, enfoque basado en estados, enfoque basado en eventos, etc.) [9], [6], [10], [11].

En este documento se realiza la implementación de un diagnosticador de fallos para la planta piloto de procesamiento de aguas, en el marco del proyecto REAGRITTECH de la cátedra Unesco de sostenibilidad. El documento está organizado de la siguiente manera: en la sección 2 se hace un repaso de algunos conceptos teóricos, en la sección 3 se explica la metodología y materiales utilizados para implementar el diagnosticador de fallos, en la sección 4 se realiza la implementación del diagnosticador para la detección de fallos en la planta de procesamiento de aguas del proyecto REAGRITTECH de

la cátedra Unesco de sostenibilidad, en la sección 5 se realiza la validación del diagnosticador implementado y en la sección 6 se exponen las principales conclusiones.

2. Antecedentes de las redes Petri

2.1. Red de Petri

Las redes de Petri (PN) permiten representar los SED mediante un formalismo matemático que describe el comportamiento de los sistemas dinámicos conducidos por eventos. En la actualidad, las PN constituyen una herramienta importante para el modelado de sistemas dinámicos de eventos discretos [12], [13], [14] por su facilidad para modelar comportamientos asíncronos y capturar el paralelismo en los estados de un sistema.

Definición 1. Formalmente, una PN se define como una quintupla.

$$PN = (P, T, Pre, Post, M_0), \quad (1)$$

Donde $P = P_1; P_2; P_3; \dots; P_{np}$ es un conjunto finito de lugares; $T = t_1; t_2; t_3; \dots; t_{nt}$ es un conjunto finito de transiciones; $Pre: P \times TR \rightarrow N$, $Post: TR \times P \rightarrow N$ son las matrices que contienen los pesos de los arcos que conectan lugares y transiciones y transiciones y lugares, respectivamente. La matriz de incidencia $I = Post - Pre$ es una matriz de $np \times nt$ y la función de marcado $M: P \rightarrow N$ representa el número de marcas en cada lugar. M_0 es el marcado inicial la PN, el cual es un vector con las marcas que posee cada lugar de la red en su momento de inicio. La matriz de incidencia $I = Post - Pre$ es una matriz de $np \times nt$, donde np es el número de lugares y nt es el número de transiciones.

2.2. Red de Petri interpretada

Las redes de Petri interpretadas (IPN) son una extensión de la estructura básica de una PN [15]. Las IPN añaden señales periféricas de entrada o salida de sensores y/o actuadores a los nodos de la red, lo cual permite dar una interpretación física al modelo del sistema estudiado [16], [17].

Definición 2. Una IPN se define formalmente como una 5 - tuple [18]

$$IPN = (N, \Sigma, \phi, \lambda, \varphi) \quad (2)$$

Donde,

- $N = (Q, M_0)$ es una PN.
- $\Sigma = \alpha_1, \alpha_2, \alpha_3 \dots \alpha_r$ es el alfabeto de entrada de la red, donde α_i es un símbolo de entrada.
- ϕ es el conjunto de símbolos de salida.

- $\lambda: TR \rightarrow \Sigma \cup \mathcal{E}$ es una función de etiquetado que asigna un símbolo de entrada a cada transición.
- $\varphi: R(N) \rightarrow \phi$ es una función de salida que asigna un símbolo de salida a cada marcado alcanzado, de modo que $y_k: R(N) \rightarrow \varphi M_k$ está mapeando la marca M_k en el vector de observación q - dimensional.

Observe los lugares p_i , de modo que $\varphi(j, i) \neq 0$ para algunos j se denominan medibles u observables; las transiciones que han asociado un símbolo de entrada del sistema se denominan manipulables o controlables.

2.3. Detectabilidad de una IPN

La detectabilidad prueba si un sistema es detectable o no por eventos, en un número determinado de sucesos estudiados. Al respecto, [4] plantea que un evento de fallo es detectable cuando, luego de haberse generado, existe al menos una señal que no corresponde al comportamiento normal del sistema. Lo anterior implica que, una vez presentado el fallo, este es detectable si, luego de un número finito de eventos, al menos una representación de la entrada o de la salida no es la esperada (ver teorema 1).

En [19] se expresa el concepto de detectabilidad como la multiplicación de la matriz φ de salida, que corresponde a la relación existente entre los sensores y los lugares de la IPN, y la matriz de incidencia I , que contiene la información de la dinámica del sistema.

Definición 3. Una IPN $(Q, M_0) = (N, \Sigma, \phi, \lambda, \varphi)$ es un evento detectable si el disparo de cualquier par de transiciones t_i, t_j de (Q, M_0) se puede distinguir entre sí mediante la observación de la señal de salida dada por la función φ .

En el teorema 1 se indica cuando una IPN es evento detectable.

Teorema 1. Sea $(Q, M_0) = (N, \Sigma, \lambda, \varphi)$, es un evento detectable si todas las columnas de φA son no nulas y diferentes entre sí [20].

Prueba (suficiencia): todas las columnas φA no son nulas y diferentes entre sí. Suponga, sin pérdida de generalidad, que se alcanza el marcado M_j a través del disparo de la transición t_j desde el marcado M_i , es decir, $M_i \xrightarrow{t_j} M_j$. Si todas las columnas, al multiplicar las matrices φA , son no nulas y diferentes entre sí, entonces se puede detectar el disparo de cualquier transición, es decir, (Q, M_0) es evento detectable. De la misma forma, la ausencia de cambios en el marcado evita la detección de cambios presentados en una transición t_i . Por lo tanto, no es

evidente la secuencia de t_i , lo cual implica que el evento ocurrido en la IPN no es un evento detectable [18].

3. Materiales y métodos

Considere el diagrama de flujo de procesos (PFD por su sigla en inglés) mostrado en la [figura 1](#), correspondiente a la planta piloto de procesamiento de aguas, del marco del proyecto REAGRITTECH de la cátedra Unesco de sostenibilidad. En el diagrama se puede observar que el sistema está compuesto principalmente por ocho depósitos con etiquetas ST para depósito de precipitación, MM para depósito de sonda, In_{VF} para el depósito de entrada al humedal vertical, VF para el humedal vertical, Out_{VF} para el depósito de salida humedal vertical, Out_{HF} para el depósito de salida humedal horizontal, HF para el humedal horizontal y In_{HF} para el depósito de entrada al humedal horizontal. Estos depósitos junto con la instrumentación del sistema, en la cual se encuentran ubicadas ocho válvulas y seis motobombas, permiten realizar diferentes configuraciones al sistema, según las características del agua deseada por los investigadores, luego del tratamiento o ciclo de recorrido aplicado.

Además de esto, se cuenta con dos sensores (sensor por alta y sensor por baja) de efecto *hall* ubicados en cada tanque para conocer el nivel del líquido en cada uno de estos y evitar reboce o daño de las motobombas por cavitación. El ciclo de recorrido se define por medio de pulsadores ubicados en la planta, los ciclos que se pueden realizar se encuentran definidos por trece escenas: (1) llenar depósito de precipitación, (2) tomar muestra depósito de precipitación, (3) vaciar depósito de precipitación a la calle, (4) tomar muestra depósito Out_{HF} , (5) vaciar depósito Out_{HF} , (6) llenar depósito In_{HF} desde Out_{VF} , (7) recircular el líquido entre los depósitos In_{HF} y Out_{HF} , (8) vaciar depósito HF , (9) tomar muestra del depósito Out_{VF} , (10) vaciar el depósito Out_{VF} , (11) llenar el depósito In_{VF} desde ST , (12) recircular fluido del depósito In_{VF} al depósito Out_{VF} y (13) vaciado del depósito VF . Estas escenas brindan información de la dinámica del sistema, y, con base en estas, se construye un controlador con el fin de representar de forma simulada el comportamiento de la planta real. El sistema se divide en tres subsistemas con el fin de facilitar el diagnóstico de fallos: el subsistema uno está compuesto por las escenas 1 a la 4, el subsistema dos lo conforman las escenas 4 a la 9 y el subsistema tres está compuesto por las escenas 9 a la 13; todas estas escenas se pueden realizar en el controlador por medio de la manipulación de los pulsadores.

En este documento, se realiza el diagnosticador para el subsistema uno. Para la implementación del diagnosticador, se usan varias herramientas *software*,

entre las cuales se encuentra el *software* de diseño de flujo de control de redes Petri Woped, que permite construir, en un principio, una aproximación gráfica (red de Petri interpretada, IPN) del comportamiento del sistema, basados en las escenas de funcionamiento definidas por los investigadores del proyecto. Las herramientas Stateflow y Simulink del *software* Matlab se usaron para la construcción del controlador, teniendo en cuenta para esto la modelización del sistema que permitió obtener la dinámica de los elementos, como tanques, válvulas y motobombas, representados por una función de transferencia, y el *software* Graphviz que permite finalmente construir la IPN que representa de manera gráfica y aproximada la dinámica del sistema, luego del proceso de identificación realizado, utilizando el modelo de caja negra.

La metodología para la implementación del diagnosticador de fallo para la planta piloto de procesamiento de agua, una vez implementado el controlador, se muestra a continuación:

1. Obtención de la matriz de datos
2. Clasificación de datos en entrada-salida (E/S)
3. Identificación del sistema
4. Análisis de detectabilidad de eventos

Esta metodología parte del modelo dinámico de la planta. En la [figura 2](#) se muestra una porción del sistema, representado por un tanque, cuyo proceso es aplicable en los otros depósitos, pues cuando el flujo de entrada al depósito está activado, el de salida está desactivado, y viceversa. Con base en esta información, se procede a encontrar la función de transferencia que modela el sistema; en la sección 4 se muestra en detalle el procedimiento.

4. Implementación del diagnosticador

El modelo dinámico del sistema mostrado en la [figura 1](#) tiene en cuenta las características físicas de dimensión de los tanques, el flujo de entrada y salida que circula a través de las motobombas y válvulas *on=off*.

La ecuación del balance de masa para el sistema mostrado en la [figura 2](#) es la siguiente:

$$A \times \rho \times \frac{dh(t)}{dt} = \rho \times Q_{in} - \rho \times Q_{out}, \quad (3)$$

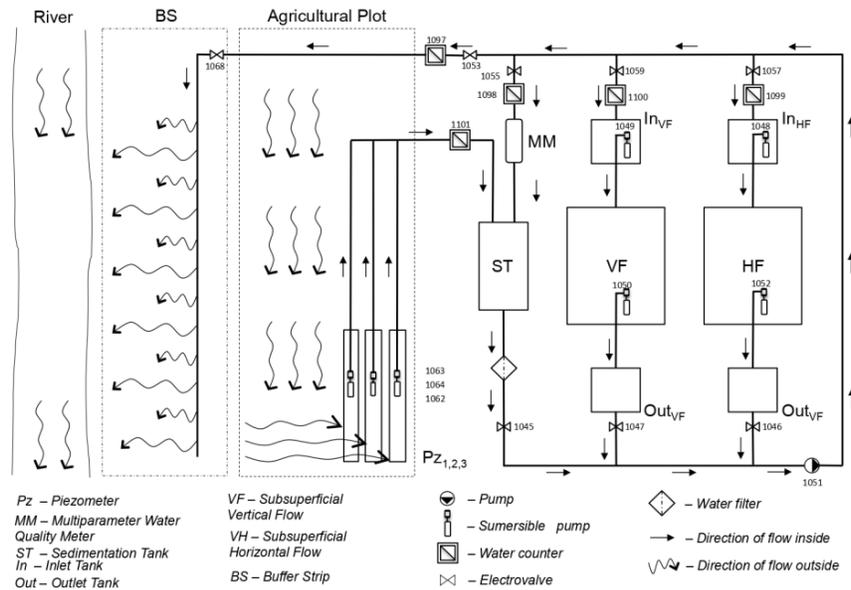


Figura 1. Diagrama de flujo planta de procesamiento de aguas residuales proyecto REAGRITECH. Fuente: [21].

Donde:

A : es el área del tanque en m^2 .

ρ : es la densidad del fluido.

$h(t)$: es el nivel del tanque.

Q_{in} : es el caudal de entrada al tanque.

Q_{out} : es el caudal de salida del tanque.

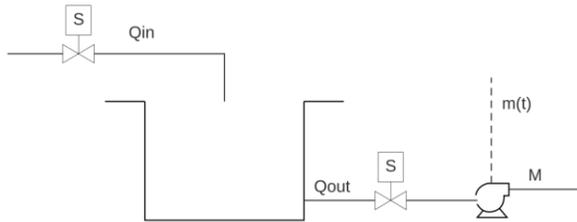


Figura 2. Sistema de tanques.

Para efectos de este trabajo, la densidad del líquido es despreciable, ya que, para el análisis temporal, esta no causa un efecto relevante en la dinámica del sistema, en relación con los tiempos de llenado y vaciado de cada depósito. Descartando la densidad de la ecuación 3, se tiene que:

$$A \times \frac{dh(t)}{dt} = Q_{in}(t) - Q_{out}(t) \quad (4)$$

Las funciones de transferencia (5) y (6) corresponden al llenado del depósito cuando $Q_{out} = 0$ y al vaciado de depósito cuando $Q_{in} = 0$, respectivamente.

$$H(s) = \frac{1}{A \times s} \times Q_{in}(s) \quad (5)$$

$$H(s) = \frac{1}{A \times s} \times Q_{out}(s) \quad (6)$$

Es de anotar que las ecuaciones 5 y 6 representan el comportamiento de una acción integral pura. A partir del conocimiento de los elementos de la planta (actuadores) y del comportamiento de las trayectorias del sistema almacenadas, en el proceso de modelización, se determinó que era una descripción suficiente para el objetivo del modelo que se construye. La implementación del controlador se realiza con base en las escenas de funcionamiento ya mencionadas, de donde se obtiene como resultado la gráfica mostrada en la figura 3, que corresponde a la evolución del subsistema uno en un tiempo determinado t .

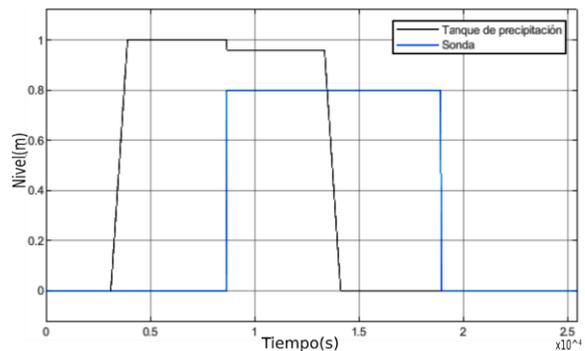


Figura 3. Evolución del sistema, depósito de precipitación y depósito de sonda.

Construido el controlador y realizado el ciclo de funcionamiento para el primer subsistema, se procede a realizar la simulación del subsistema en lazo cerrado conjunto planta-controlador para obtener los datos que serán posteriormente procesados en un algoritmo de identificación tipo caja negra. La información de entrada al algoritmo son datos de (E/S) , como se puede apreciar en la imagen de la [figura 4](#). El algoritmo de identificación genera como resultado una matriz W de eventos e_k .

Esta matriz está compuesta por las funciones de entrada F_e y las funciones de salida F_s , como se aprecia en la imagen. Este algoritmo, además de la matriz de datos, se encarga de crear gráficamente una IPN que representa el comportamiento interno del sistema a partir de la relación directa entre las funciones F_e/F_s [4], teniendo en cuenta que durante el proceso de identificación pueden suceder tres situaciones:

1. $F_e \neq 0$ y $F_s \neq 0$. Existen variaciones en la entrada y variaciones en la salida, lo cual hace que el sistema evolucione.
2. $F_e \neq 0$ y $F_s = 0$. Existe variación en la entrada, pero estas no se reflejan en la salida del sistema, lo cual se puede interpretar como un evento no observable.
3. $F_e = 0$ y $F_s \neq 0$. Existe variación en la salida, sin haberse presentado variación en la entrada, lo cual puede ser interpretado como un fallo en el sistema.

La matriz de datos W_k se representa así:

$$W_k = \left(\begin{array}{c|c} F_e(1) & F_s(1) \\ F_e(2) & F_s(2) \\ F_e(3) & F_s(3) \\ \vdots & \vdots \\ F_e(k) & F_s(k) \end{array} \right) \quad (7)$$

Y la matriz de eventos que relaciona a cada evento de entrada con un evento de salida se representa según la ecuación 8.

La matriz de datos obtenida luego de la identificación del sistema se muestra en (9), compuesta por trece eventos de entrada; por cada evento de entrada, hay asociado un evento de salida.

Cada evento de entrada está compuesto por un vector de ocho posiciones, asociado a las entradas del sistema (pulsadores, válvulas y motobombas), correspondientes a señales de mando de control, y un vector de salida compuesto por cuatro posiciones asociadas a cada evento de entrada, pertenecientes a señales de los sensores.

$$e_k = \left(\begin{array}{c|c} e_1 & F_e(1) \\ e_2 & F_e(2) \\ e_3 & F_e(3) \\ \vdots & \vdots \\ e_k & F_e(k) \end{array} \middle| \begin{array}{c} F_s(1) \\ F_s(2) \\ F_s(3) \\ \vdots \\ F_s(k) \end{array} \right) \quad (8)$$

$$W_k = \left(\begin{array}{c|c} 10010000 & 0000 \\ 00010000 & 0100 \\ 00000000 & 1100 \\ 00000000 & 0000 \\ 01001110 & 0011 \\ 01001110 & 0001 \\ 01001110 & 0000 \\ 01000000 & 0011 \\ 00100000 & 0000 \\ 00101011 & 0001 \\ 00100000 & 0000 \\ 00000000 & 0100 \\ 00000000 & 0010 \end{array} \right) \quad (9)$$

Durante el proceso de construcción de la matriz de datos W , el algoritmo de identificación va construyendo en paralelo la IPN, a partir de la relación $F_e = F_s$ de los datos, teniendo en cuenta las tres situaciones que pueden presentarse.

Para esto, el algoritmo empieza construyendo un lugar P_i con etiqueta $[0 \ 0 \ 0 \ 0]$, que corresponde a la observación de las señales en el momento de inicio del proceso de identificación; este evento de salida tiene asociado un evento de entrada con etiqueta $[1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0]$.

A partir de los datos de inicio, el algoritmo realiza un proceso de evaluación por filas de la matriz de datos, verificando si en esta se han presentado cambios, ya sea en los datos de entrada, en los datos de salida o en ambos.

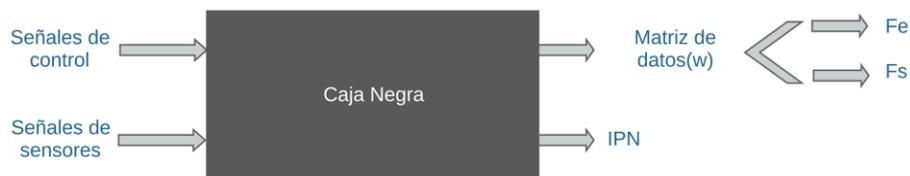


Figura 4. Representación gráfica del funcionamiento del algoritmo de identificación.

En la **figura 5** se observa una porción de la red, correspondiente a la evolución del sistema por cambios presentes en la segunda fila de la matriz de datos, lo cual hace que el sistema evolucione.

La IPN mostrada en la **figura 6** corresponde al comportamiento interno del sistema a partir de la relación directa de la función F_e y la función F_s .

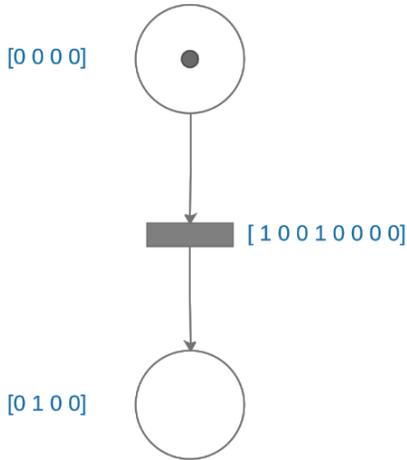


Figura 5. Porción de IPN para mostrar evolución del sistema.

Una vez construida la IPN, se obtiene la matriz Pre y la matriz $Post$, y se halla la matriz de incidencia como $Post-Pre$. La matriz φ mostrada en (13) corresponde a una función de salida, que asocia los sensores a los lugares de la red.

$$Pre = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (10)$$

$$Post = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad (11)$$

$$A = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \end{pmatrix} \quad (12)$$

$$\varphi = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (13)$$

Con la IPN obtenida del proceso de identificación, de donde se obtiene la matriz I de incidencia y la matriz φ , se realiza el análisis de detectabilidad que determina si el sistema detecta o no eventos de fallo, teniendo en cuenta lo expresado en el teorema 1. En la matriz resultado de la multiplicación de las matrices φA , obtenidas en el proceso de identificación, se puede observar que no cumplen con el teorema 1. Por esto, para que el sistema sea evento-detectable, se hace necesario la adición de un número mínimo de sensores, con el fin de garantizar la detectabilidad. Esto se realiza con base en la propuesta de [22].

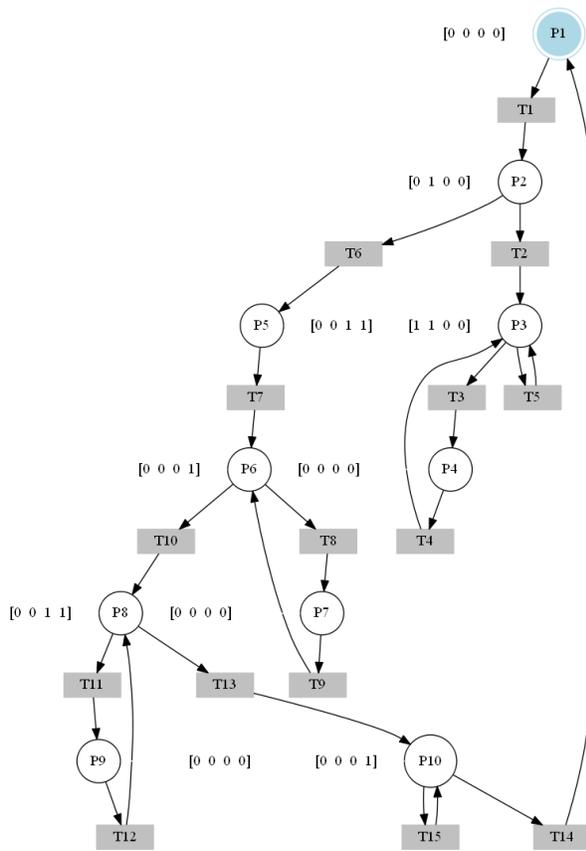


Figura 6. IPN del sistema construida a partir de la relación directa entre F_e / F_s .

$$\varphi \times A = \begin{pmatrix} 0 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 1 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 1 & 0 & -1 & 1 & -1 \end{pmatrix} \quad (14)$$

Como resultado, se adicionaron tres sensores: un sensor de nivel por baja, un sensor de nivel por alta y un sensor de flujo, ubicados en un nuevo depósito llamado depósito general. Realizando nuevamente el proceso mencionado, se obtiene como resultado una nueva IPN, mostrada en la figura 7, de donde se adquieren las matrices Pre , $Post$, I y φ . Se analiza la detectabilidad, y se obtiene como resultado (17), donde se puede apreciar que se cumple lo expuesto en el teorema 1 y en la definición 3. La IPN obtenida presenta autobucles en los lugares $P6$, $P11$ y $P12$, que pudieron haber sido generados por cambios en la función de entrada, y que no fueron detectados en la función de salida. Se propone adicionar a cada autobucle un lugar no observable, con el objeto de ser más precisos con el modelo identificado, como se aprecia en la figura 8.

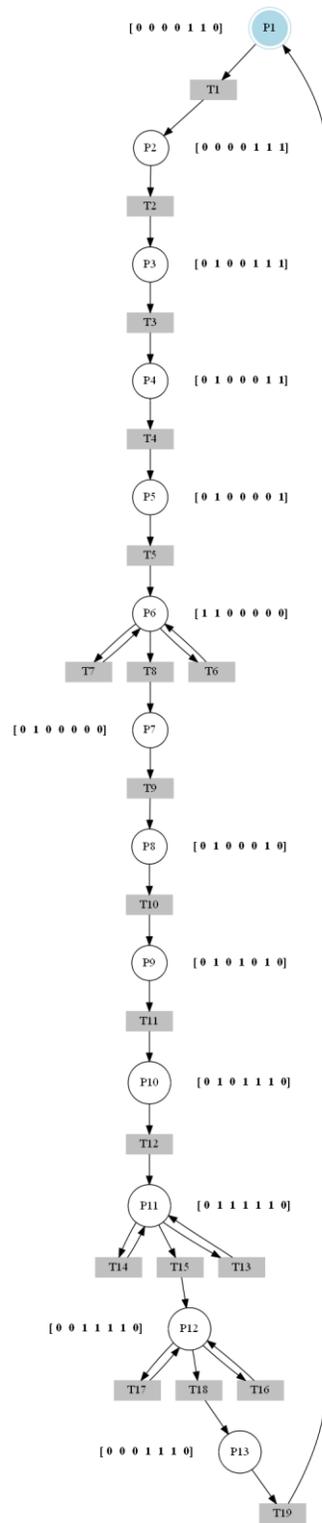


Figura 7. IPN construida a partir de la relación F_e / F_s , con la adición de sensores.

Dado que el subsistema es evento detectable, se procede a la implementación del diagnosticador de fallos.

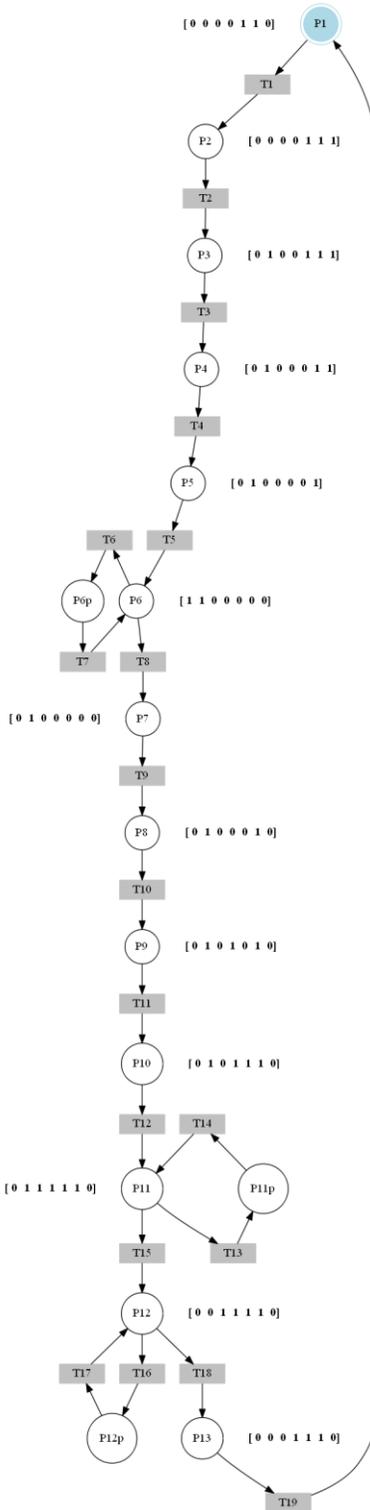


Figura 8. IPN construida a partir de la relación F_e / F_s con la adición de sensores y plazas no observables.

La implementación del diagnosticador se realiza con base en el método propuesto por Elvia Ruiz Beltrán en su tesis de doctorado y validado en diferentes estudios ([23], [18]). Para el diseño del diagnosticador, primero, se construye un vector base B^T , que calcula todas las posibles convenciones de eventos asociados a cada lugar. El vector B es un vector de tamaño $q \times 1$ construido únicamente con los lugares medibles del sistema [23], como se muestra en (15), luego del proceso de adición de sensores.

$$B = [b^0 \ b^1 \ \dots \ b^{q-1}] \quad (15)$$

Donde q son los lugares medibles de la matriz de incidencia. Para encontrar la base del vector, se hace uso de la ecuación 16.

$$B = 2\max(\text{abs}(a_{ij})) + 1, \quad (16)$$

Donde a_{ij} es un elemento de la matriz de incidencia. En este trabajo, se divide la matriz de incidencia en tres submatrices (M_{D1} , M_{D2} y M_{D3}), de tamaños 5×19 , 5×19 y 3×19 , respectivamente, con el fin de facilitar el cálculo de las constantes de diagnósticos mostradas en la última fila de cada submatriz. Igualmente, el vector B se divide en tres subvectores (B_{D1} , B_{D2} y B_{D3}), de tamaños 5×1 , 5×1 y 3×1 , respectivamente. En cada subvector se encuentra un parámetro adicional de “-1”, que corresponde a un puerto de entrada del PLC, el cual se encarga de analizar el número de marcas del sistema y diagnosticar, con base en estas, si ha ocurrido o no un fallo.

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & -1 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 1 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & 1 & 0 & -1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (17)$$

En las ecuaciones 18, 19 y 20 se muestran los subvectores, en las ecuaciones 21, 22 y 23 se muestran las submatrices. El diagnosticador se calcula multiplicando el vector B por la matriz de incidencia I , es decir, $D=B \times I$; en este caso, el diagnosticador fue dividido en tres lugares de diagnóstico, $D1$, $D2$ y $D3$. Realizando la operación pertinente, se obtienen las constantes de diagnóstico para cada lugar, mostradas en las tablas 1, 2 y 3, respectivamente.

$$B_{D1} = [3^0 \ 3^1 \ 3^2 \ 3^3 \ 3^4 \ -1] \quad (18)$$

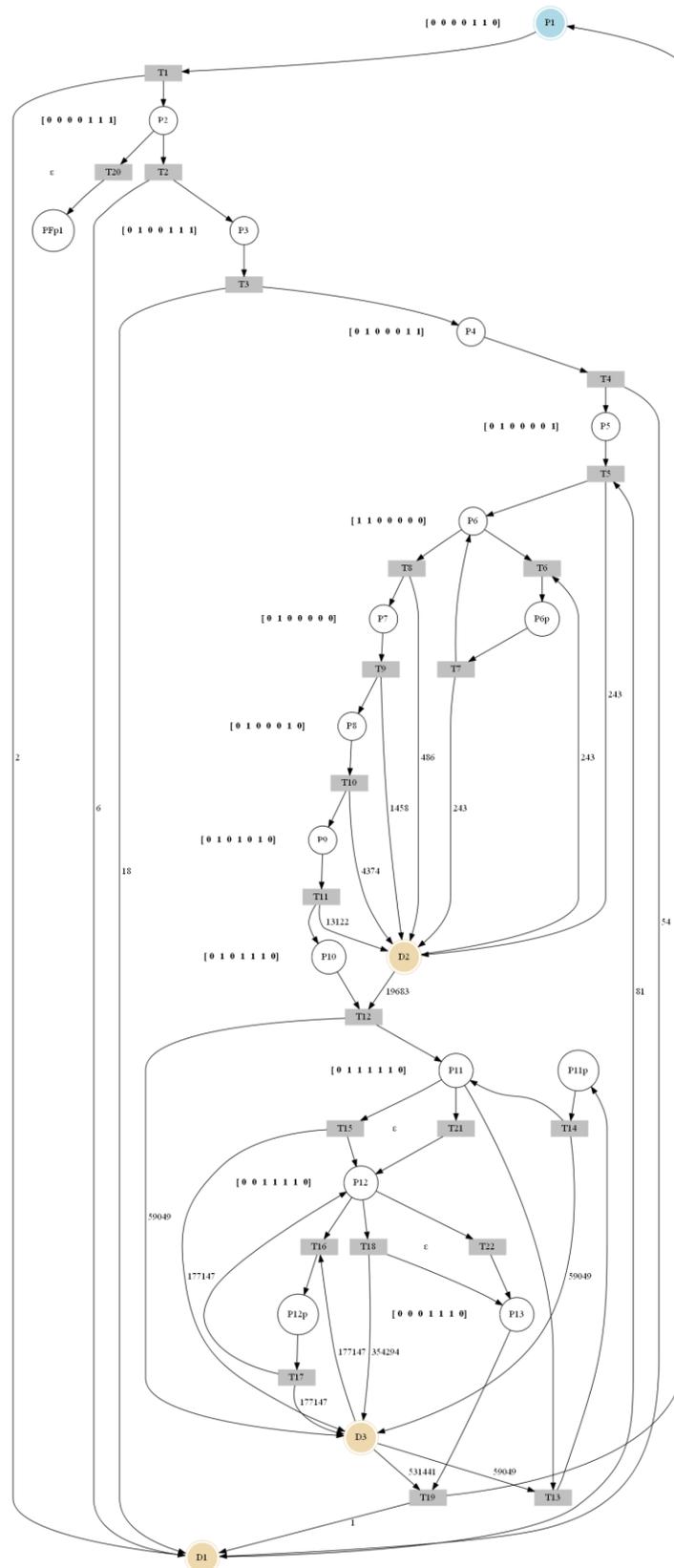


Figura 9. IPN con lugares de diagnóstico de fallo.

5. Validación del diagnosticador

Una vez implementados los lugares de diagnóstico, se procede a realizar la validación para saber si el diagnosticador es capaz de detectar o no la ocurrencia de fallos operacionales, haciendo uso del teorema del residuo, donde se tiene que:

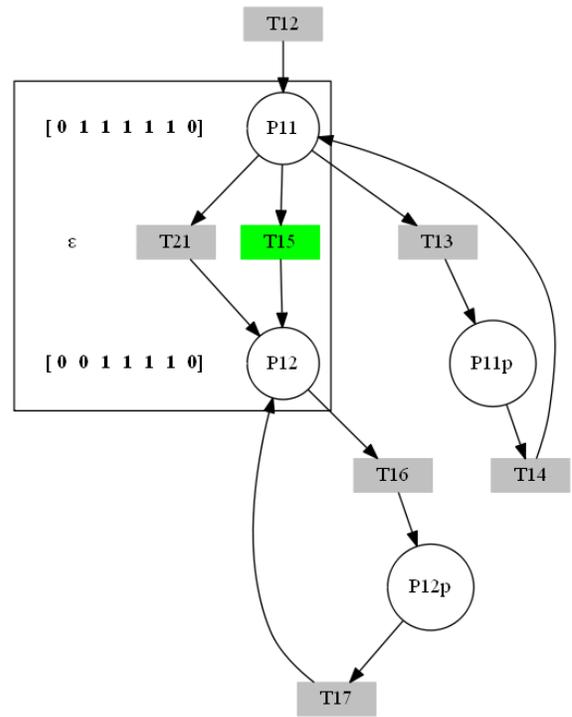
$$E_k = M_k^{af} - M_k^{rf}, \tag{24}$$

M_k^{af} es el modelo de funcionamiento actual y M_k^{rf} es el modelo de funcionamiento ideal o requerido. Si $E_k = 0$, significa que no hay evidencia de fallo, por tanto, el sistema funciona de forma correcta; si $E_k \neq 0$, implica que ha sucedido un fallo en el sistema.

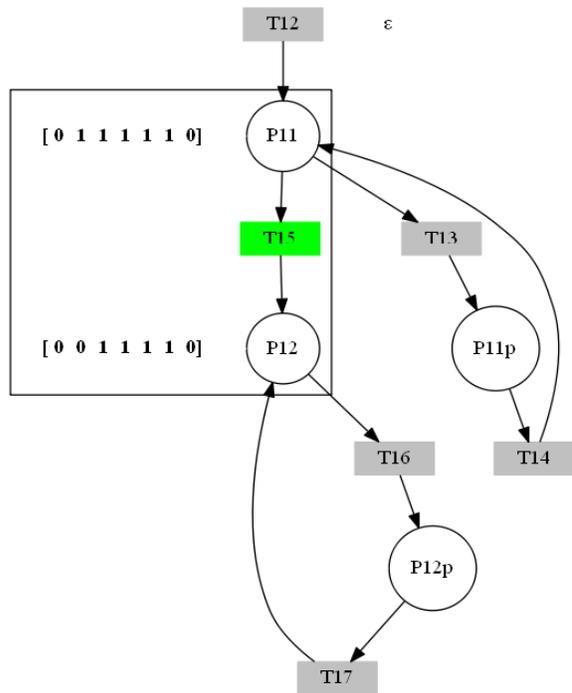
Para la validación se simula un fallo operacional correspondiente a ruptura en la superficie del tanque de precipitación, al dispararse la transición $T21$. En la figura 10 se muestra una porción de la IPN mostrada en la figura 9, donde ocurre el fallo. En la figura 10(a) se muestra el modelo de funcionamiento actual, en la figura 10(b) se muestra el modelo requerido; en este caso, en ambos modelos se dispara la transición $T15$, correspondiente al funcionamiento correcto, la cual, ante su disparo, suma una marca en el lugar $P12$, por tanto, la diferencia entre $M_k^{af} - M_k^{rf}$ da como resultado $E_k = 0$, es decir, el sistema evoluciona de forma normal sin evidencia de fallo.

Ahora, se realiza el mismo proceso explicado, pero, a diferencia del caso anterior, en el modelo de funcionamiento actual, se dispara la transición $T21$, y se aplica el teorema del residuo, por lo que se obtiene como resultado $E_k \neq 0$; debido a que no existe el lugar de diagnóstico, la transición $T21$, al igual que la transición $T15$, suma una marca al lugar $P12$, como se muestra en la figura 11. Este resultado es erróneo, debido a que en el sistema ha ocurrido un fallo, el cual no fue detectado; para resolver esto, se procede a realizar el mismo proceso, teniendo en cuenta el lugar de diagnóstico $D3$, el cual tiene en cuenta el peso de los arcos calculados anteriormente.

Cuando se dispara la transición $T15$ en ambos modelos (real y requerido), el marcado depositado en $D3$ es de 708.588 marcas; como se puede ver en la figura 12, al realizar el teorema del residuo, aplicado al lugar de diagnóstico, se tiene que $E_k = 0$. Esta respuesta es correcta, debido a que, en ambos modelos, se dispara la transición $T15$, perteneciente al funcionamiento apropiado.

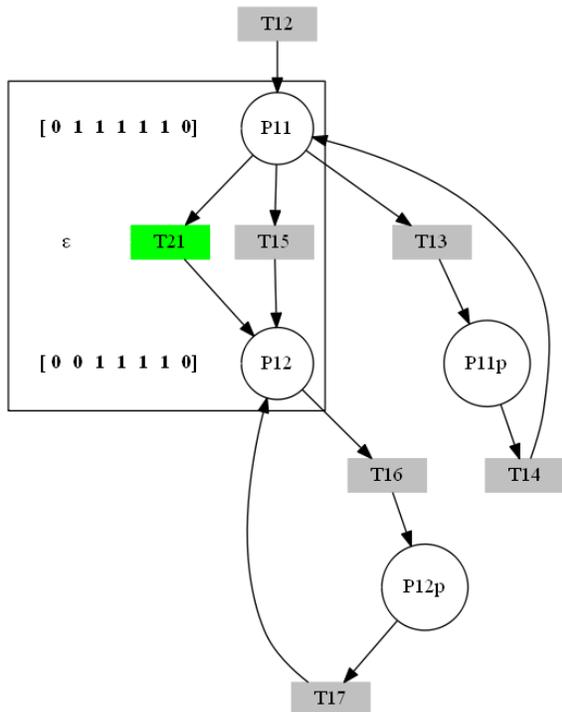


(a) Modelo de funcionamiento actual

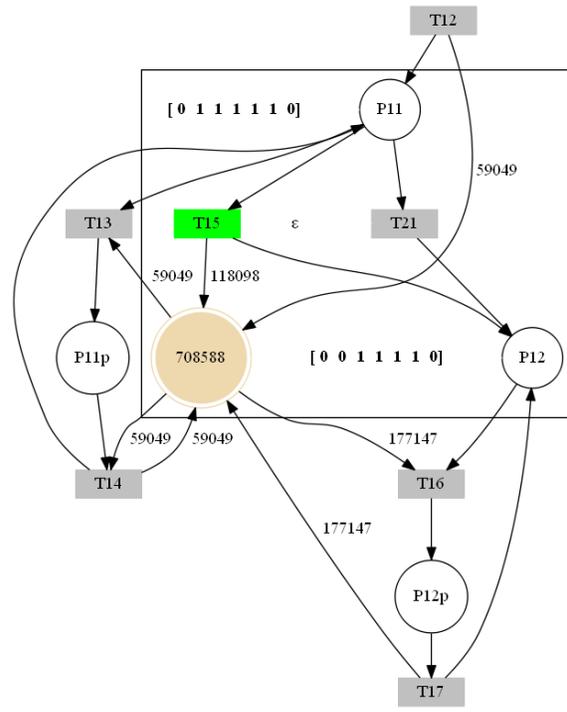


(b) Modelo de funcionamiento requerido

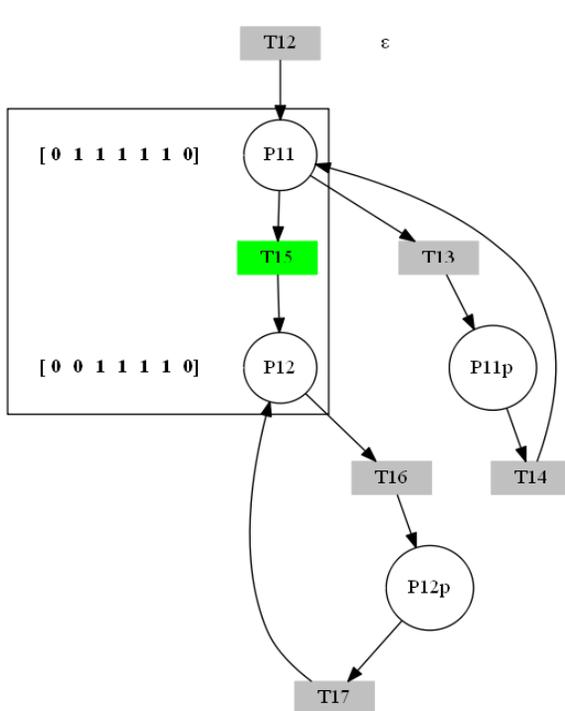
Figura 10. Simulación del sistema, sin accionamiento de la transición de fallo $T21$ ni plaza de diagnóstico.



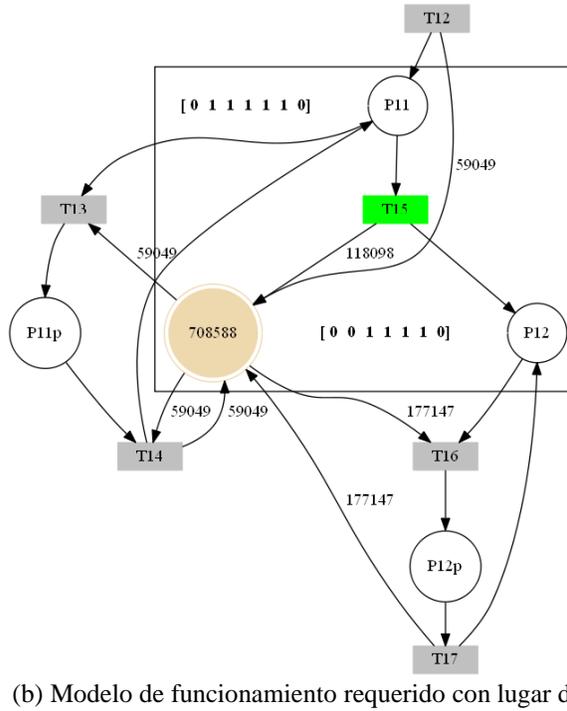
(a) Modelo de funcionamiento actual, con transición de fallo T21



(a) Modelo de funcionamiento actual con lugar de diagnóstico sin transición de fallo



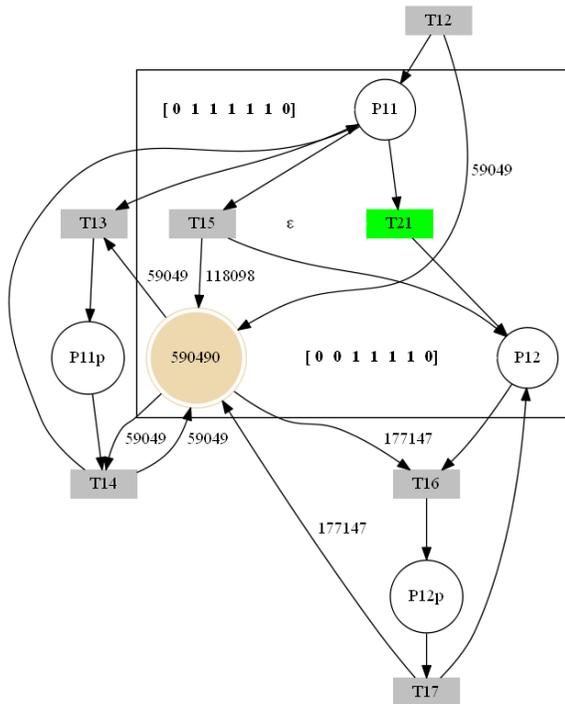
(b) Modelo de funcionamiento requerido



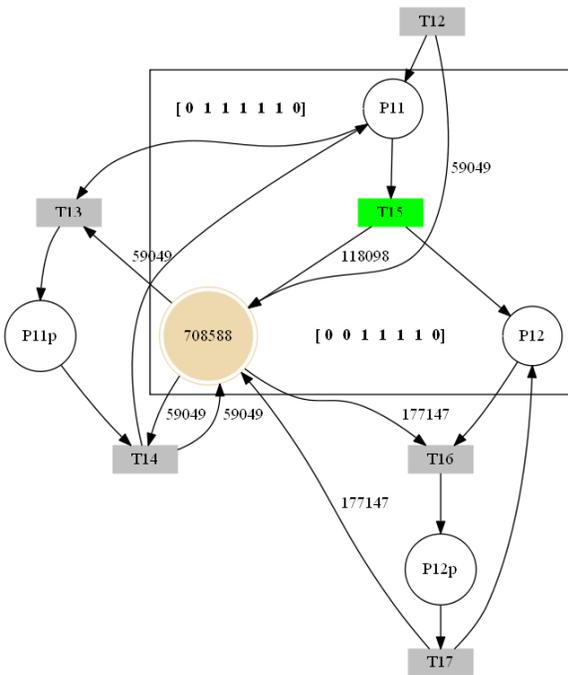
(b) Modelo de funcionamiento requerido con lugar de diagnóstico sin transición de fallo

Figura 11. Simulación del sistema, con accionamiento de transición de fallo T21, sin plaza de diagnóstico.

Figura 12. Simulación del sistema, con lugar de diagnóstico D3, sin accionamiento de la transición de fallo T21.



(a) Modelo de funcionamiento actual con lugar de diagnóstico y transición de fallo T21



(b) Modelo de funcionamiento requerido con lugar de diagnóstico sin transición de fallo

Figura 13. Simulación del sistema, con lugar de diagnóstico D3, y accionamiento de la transición de fallo T21.

Por último, se tiene la simulación del sistema, donde se acciona la transición de fallo T21, teniendo en cuenta que en este punto se encuentra el lugar de diagnóstico; la transición de fallo T21 es accionada en el modelo de funcionamiento actual, mientras que la transición T15 es disparada en el modelo de funcionamiento requerido, como se aprecia en la figura 13.

Al aplicar el teorema del residuo, se tiene que $E_k = M_k^{af} - M_k^{rf} = -118.098$, debido a que la transición de fallo al ser disparada no aporta marcas al lugar de diagnóstico, es decir, $D3=590.490$ marcas, cuando el valor correcto debería de ser de 708.588, luego de analizar el marcado en el lugar P12, donde se evidencia que esta cuenta con su respectiva marca; de esta forma, se concluye que el diagnosticador implementado es capaz de detectar la presencia de fallos existentes en el sistema, haciendo un análisis en el lugar de diagnóstico D3 y en el lugar P12 del sistema.

6. Conclusión

En este artículo se muestra el procedimiento utilizado para la implementación de un diagnosticador de fallos, en la planta piloto de procesamiento de aguas, en el marco del proyecto REAGRITTECH de la cátedra Unesco de sostenibilidad, modelada como un SED, haciendo uso de las IPN. En este documento, el diagnosticador es implementado teniendo en cuenta únicamente los lugares observables del sistema. Se desconoce si los lugares encontrados como no “observables”, durante el proceso de identificación, tienen afectación o no en el diagnosticador. Durante el análisis de detectabilidad del sistema, se logra identificar que el sistema no es detectable por eventos, por esta razón, se hace necesario realizar un proceso de adición de sensores, para garantizar el número mínimo de estos, que hagan que todas las entradas del sistema sean diferentes entre sí y cumplan con lo expuesto en el teorema 1. Se verifica que luego del proceso, donde se adiciona un sensor de nivel de baja, un sensor de nivel de alta y un sensor de flujo, el sistema se vuelva detectable, y se procede a construir el diagnosticador.

Referencias

[1] R. J. Patton, C. J. López-Toribio, F. J. Uppal, “Artificial intelligence approaches to fault diagnosis,” in *IEE Colloquium on Condition Monitoring: Machinery, External Structures and Health* (Ref. No. 1999/034), April 1999, pp. 5/1-5/18.

- [2] M. Ayoubi, "Neuro-fuzzy structure for rule generation and application in the fault diagnosis of technical processes," in *Proceedings of 1995 American Control Conference - ACC'95*, vol. 4, June 1995, pp. 2757-2761 vol. 4, doi: <https://doi.org/10.1109/ACC.1995.532351>
- [3] L. Urrego, M. Muñoz-Añasco, "Latent nestling method as an abstraction method in coloured petri nets for fault diagnosis in complex systems," *Revista Ontare*, vol. 3, pp. 97-123, 2015, doi: <https://doi.org/10.21158/23823399.v3.n1.2015.1252>
- [4] M. Muñoz-Añasco, "Identificación y diagnóstico de fallos en sistemas de eventos discretos estocásticos," tesis doctoral, Universitat Politècnica de València, 2015, doi: <https://doi.org/10.4995/Thesis/10251/53915>
- [5] J. Zaytoon, S. Lafortune, "Overview of fault diagnosis methods for discrete event systems," *Annual Reviews in Control*, vol. 37, no. 2, pp. 308-320, 2013, doi: <https://doi.org/10.1016/j.arcontrol.2013.09.009>
- [6] M. Sampath, "A Discrete Event Systems Approach to Failure Diagnosis", tesis doctoral, University of Michigan, 1995.
- [7] Y. Ru, C. N. Hadjicostis, "Fault diagnosis in discrete event systems modeled by partially observed petri nets," *Discrete Event Dynamic Systems*, vol. 19, no. 4, p. 551, 2009, doi: <https://doi.org/10.1007/s10626-009-0074-7>
- [8] R. C. de Vries, "An automated methodology for generating a fault tree," *IEEE Transactions on Reliability*, vol. 39, no. 1, pp. 76-86, 1990, doi: <https://doi.org/10.1109/24.52615>
- [9] S. Hashtrudi Zad, R. H. Kwong, W. M. Wonham, "Fault diagnosis in discrete-event systems: framework and model," *IEEE Transactions on Automatic Control*, vol. 48, no. 7, pp. 1199-1212, 2003, doi: <https://doi.org/10.1109/CDC.1998.761808>
- [10] A. S. Willsky, "A survey of design methods for failure detection in dynamic systems," *Automatica*, vol. 12, no. 6, pp. 601-611, 1976, doi: [https://doi.org/10.1016/0005-1098\(76\)90041-8](https://doi.org/10.1016/0005-1098(76)90041-8)
- [11] C. Angeli, "Diagnostic expert systems: From expert's knowledge to real-time systems," *Advanced Knowledge Based Systems: Model, Applications & Research*, vol. 1, 01 2010.
- [12] G. Cansever, I. Kucukdemiral, "A new approach to supervisor design with sequential control petri-net using minimization technique for discrete event system," *The International Journal of Advanced Manufacturing Technology*, vol. 29, no. 1108 2005.
- [13] G.-B. Lee, H. Zandong, J. Lee, "Automatic generation of ladder diagram with control petri net," *Journal of Intelligent Manufacturing*, vol. 15, pp. 245-252, 2004, doi: <https://doi.org/10.1023/B:JIMS.0000018036.84607.37>
- [14] J.-S. Lee, P.-L. Hsu, "An improved evaluation of ladder logic diagrams and petri nets for the sequence controller design in manufacturing systems," *The International Journal of Advanced Manufacturing Technology*, vol. 24, pp. 279-287, 2004, doi: <https://doi.org/10.1007/s00170-003-1722-y>
- [15] G. Frey, L. Litz, "Verification and validation of control algorithms by coupling of interpreted petri nets," vol. 1, pp. 7-12, 1998, doi: <https://doi.org/10.1109/ICSMC.1998.725375>
- [16] L. Rivera Cambero, L. Aguirre Salas, A. Sánchez, S. González-Palomares, "Sistema embebido para simular redes de petri interpretadas," *IX Semana Nacional de Ingeniería Electrónica*, vol. IX, pp. 532-539, 2013.
- [17] A. Ramírez-Trevino, I. Rivera-Rangel, 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, 2003, doi: <https://doi.org/10.1109/TRA.2003.814503>
- [18] I. Rivera Rangel, A. Ramírez Trevino, L. Aguirre Salas, J. Ruiz-León, "Geometrical characterization of observability in interpreted petri nets," *Kybernetika*, vol. 5, 2005.
- [19] A. Ramfrez Trevino, I. Rangel, E. Lope Mellado, "Observer design for discrete event systems modeled by interpreted petri nets," *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings*, vol. 3, 2000, pp. 2871-2876, doi: <https://doi.org/10.1109/ROBOT.2000.846463>

- [20] A. Ramírez-Trevino, E. Ruiz-Beltrán, J. Aramburo-Lizarraga, and E. López-Mellado, "Structural diagnosability of des and design of reduced petri net diagnosers," *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 42, no. 2, pp. 416-429, 2012, doi: <http://dx.doi.org/10.1109/TSMCA.2011.2169950>
- [21] A. Gallegos, L. Aguilar, I. Campos, P. Caro, S. Sahuquillo, C. Pérez, C. A. Arias, J. Montoya, J. Morató, "TIC para la determinación de los parámetros operacionales de humedales construidos diseñados para el tratamiento de aguas contaminadas por nitratos," *Revista Ingenierías Universidad de Medellín*, vol. 15, pp. 53-70, 06 2016.
- [22] D. F. Valencia Medina, "Verificación de la diagnosticabilidad de un sistema de eventos discretos, en función del número de señales de salida," trabajo fin de grado, Universidad del Cauca, 2018.
- [23] J. Arámburo-Lizárraga, A. Ramírez-Trevino, E. López-Mellado, and E. Ruiz Beltrán, "Fault Diagnosis in Discrete Event Systems Using Interpreted Petri Nets", in *Advances in Robotics, Automation and Control*, 2008, doi: <https://doi.org/10.5772/5534>