

Supervisión de una clase de sistemas dinámicos de eventos discretos

Supervision of a class of discrete event dynamic systems

Quilumba, Eduardo^{1*}; Solé, Luz²

¹Instituto de posgrado, Universidad Técnica de Manabí, Ecuador.

²Facultad de Ciencias, Departamento de Matemáticas, Universidad de Los Andes, Venezuela.

*eduquigi@gmail.com

Resumen

En el contexto de los sistemas dinámicos de eventos discretos, el control juega un rol preponderante en el problema de cómo verificar el orden en que se presentan los eventos dadas algunas especificaciones o condiciones sobre esta clase de sistemas. Convencionalmente, para esto es usada la lógica, las teorías de lenguajes y autómatas, modelos Booleanos y lógica temporal. La variedad de proposiciones refleja la diversidad de áreas de aplicación. Nosotros determinamos las características estructurales cualitativas de los problemas de control básico, y estudiamos el problema de sintetizar un supervisor centralizado que asegure el comportamiento colectivo preestablecido desde un punto de vista algebraico.

Palabras clave: Supervisión, gramática, procesos, categorías, dinámica.

Abstract

In the context of dynamic discrete event systems, control plays a preponderant role in the problem of how to verify the order in which events are presented given some specifications or conditions on this class of systems. Conventionally, logic, theories of languages and automata, Boolean models and temporal logic are used for this. The variety of propositions reflects the diversity of application areas. We determine the qualitative structural characteristics of the basic control problems, and we study the problem of synthesizing a centralized supervisor that ensures the pre-established collective behavior from an algebraic point of view.

Keywords: Supervision, grammar, processes, categories, dynamics.

1 Introducción

Un sistema de eventos discretos (Branicky 1995, Mata y col., 2018) es definido en términos algebraicos y su comportamiento dinámico es dado por un lenguaje formal apropiado.

Un sistema complejo consiste en un conjunto de procesos asincrónicos que interactúan y operan concurrentemente. Un problema de control para esta clase de sistemas es la supervisión de sus componentes para asegurar su interacción armoniosa y un flujo ordenado de eventos resultantes (Ramadge y col. 1986). Cada proceso se comunica con un supervisor central que puede ser considerado como una estación de referencia a partir de la cual la interacción entre dichos procesos puede ser “observada”, esta operación toma la forma de un “Shuffling” (reorganizador) de las sucesiones de comunicación producidas. Nosotros examinamos el problema de sintetizar un supervisor centralizado que

asegure el comportamiento colectivo deseado desde un punto de vista algebraico, dando de esta manera rigurosidad matemática a los contenidos.

La organización de este manuscrito comienza con una sección preliminar que constituye los fundamentos para desarrollar las secciones posteriores. Esta incluye algunos conceptos de la teoría de lenguajes (Eilenberg 1974) y de algunas nociones de teoría de grafos, inmersas en la teoría de categorías (MacLane 1971). En la sección siguiente se incluyen algunas nociones tales como procesos de eventos controlables (Mata 2017, Ramadge y col., 1982, Ramadge y col., 1984), máquinas secuenciales, gramáticas regulares (también incluyendo a estas en la teoría de las categorías). Finalmente se introducen definiciones, propiedades y análisis de las estructuras citadas (con énfasis en los procesos bajo supervisión).

2. Preliminares

Dado $\Sigma = \{u_1, \dots, u_n\}$ un conjunto finito no vacío, el cual será llamado alfabeto, consideraremos el conjunto de todas las sucesiones finitas constituidas con elementos del conjunto Σ , incluyendo la sucesión que no tiene símbolos, denotada por ε . Este conjunto será denotado por Σ^* . Los elementos de Σ^* serán llamados palabras y cada elemento de Σ será llamado letra. Σ^* tiene estructura de monoide con la operación $\cdot$ llamada concatenación, la cual es definida como sigue: para $a = a_1 \dots a_n$ y $b = b_1 \dots b_m$ en Σ^* , $a \cdot b = a_1 \dots a_n \cdot b_1 \dots b_m$. En adelante denotaremos $a \cdot b$ por ab , además, diremos que una palabra $x \in \Sigma^*$ es llamada prefijo de $y \in \Sigma^*$ si existe una palabra $z \in \Sigma^*$ tal que $y = x \cdot z$, donde $\cdot$ es la operación de concatenación sobre Σ^* . Todo subconjunto $L \in \Sigma^*$ es llamado un lenguaje sobre Σ . El conjunto formado por todos los prefijos de palabras de un lenguaje $L \in \Sigma^*$ es llamado clausura de L y será denotado por $pre(L)$. De este modo

$$pre(L) = \{x \in \Sigma^* / \exists y \in \Sigma^*, x y \in L\}.$$

En adelante al subconjunto de números naturales $\{1, 2, \dots, n\}$ lo denotaremos por N_n^+ y consideraremos también la función

$$l: \Sigma^* \rightarrow \Sigma$$

$$\begin{cases} l(\varepsilon) = \varepsilon, \\ l(w\sigma) = \sigma \quad \forall w \in \Sigma^* \text{ y } \sigma \in \Sigma. \end{cases}$$

Todo $w \in \Sigma^*$ se puede escribir $w = w_1 \dots w_n$ con $w_i \in \Sigma \cup \{\varepsilon\}$, $1 \leq i \leq n$.

Sean Σ y Ω dos alfabetos tales que $\Omega \subset \Sigma$; definimos $h: \Sigma^* \rightarrow \Omega^*$ por

$$h(w_i) = \begin{cases} w_i, & \text{si } w_i \in \Omega \\ \varepsilon, & \text{si } w_i \notin \Omega. \end{cases}$$

A partir de h definimos la e -proyección de Σ^* en Ω^* por

$$H(w) = H(w_1 \dots w_n) = h(w_1) \dots h(w_n)$$

Esta proyección está bien definida y es un homomorfismo de monoides.

Un grafo dirigido G es un cuádruple $G = (N, E, \delta_0, \delta_1)$, donde N es un conjunto llamado conjunto de nodos, E es un conjunto llamado conjunto de arcos, junto a las funciones $\delta_0: E \rightarrow N$ y $\delta_1: E \rightarrow N$ especificando los nodos iniciales y finales respectivamente para cada arco.

Un camino (o trayectoria) c en G es una sucesión finita de arcos de E , $c = \alpha_1, \alpha_2, \dots, \alpha_n$, o simplemente $c = \alpha_1 \alpha_2 \dots \alpha_n$, que verifica la siguiente propiedad:

$$\begin{aligned} \delta_1(\alpha_1) &= \delta_0(\alpha_2), \delta_1(\alpha_2) \\ &= \delta_0(\alpha_3), \dots, \delta_1(\alpha_{n-1}) \\ &= \delta_0(\alpha_n). \end{aligned}$$

Aquí los nodos $q_1 = \delta_0(\alpha_1)$, y $q_n = \delta_1(\alpha_n)$ son llamados comienzo y final del camino c y el entero $n \geq 1$ es llamado la longitud del camino.

Un morfismo entre dos grafos $G = (N, E, \delta_0, \delta_1)$ y $H = (M, P, \tilde{\delta}_0, \tilde{\delta}_1)$ es un par de funciones $F = (F_n, F_e)$ tales que el siguiente diagrama conmuta.

$$\begin{array}{ccccc} N & \xleftarrow{\delta_0} & E & \xrightarrow{\delta_1} & N \\ F_n \downarrow & & F_e \downarrow & & F_n \downarrow \\ M & \xleftarrow{\tilde{\delta}_0} & P & \xrightarrow{\tilde{\delta}_1} & M \end{array}$$

Los grafos junto a sus morfismos forman una categoría que denotaremos por CG .

Denotemos por 1 al conjunto formado solo por el elemento uno. Dado un conjunto Σ nosotros podemos construir el grafo especial $\tilde{\Sigma} = (1, \Sigma, \delta_0, \delta_1)$, donde Σ es el conjunto de arcos y para cada $\sigma \in \Sigma$, $\delta_0(\sigma) = \delta_1(\sigma) = 1$.

Un grafo etiquetado G_Σ es un tripe (G, g, Σ) , donde $G = (N, E, \delta_0, \delta_1)$ es un grafo, Σ es un alfabeto y g un morfismo de G en $\tilde{\Sigma}$ tal que el siguiente diagrama conmuta.

$$\begin{array}{ccccc} N & \xleftarrow{\delta_0} & E & \xrightarrow{\delta_1} & N \\ F_n \downarrow & & F_e \downarrow & & F_n \downarrow \\ 1 & \xleftarrow{\delta'_0} & \Sigma & \xrightarrow{\delta'_1} & 1 \end{array}$$

El morfismo g marca los arcos de G con los elementos del conjunto Σ .

Sea $G_\Sigma = (G, g, \Sigma)$, un grafo etiquetado. **Una Trazas c** en G_Σ es una sucesión finita $\sigma_1 \sigma_2 \dots \sigma_n$ de elementos de Σ tales que $g_e(\alpha_1) = \sigma_1, g_e(\alpha_2) = \sigma_2, \dots, g_e(\alpha_n) = \sigma_n$, para algún camino $\alpha_1 \alpha_2 \dots \alpha_n$ en G . Aquí los nodos $q_0 = \delta_0(\alpha_1)$ y $q_n = \delta_0(\alpha_n)$ serán llamados nodo asociado al comienzo de la traza c y nodo asociado al final de la traza c respectivamente, y el entero $n \geq 1$, será llamado la longitud de la traza c . También, consideramos la traza ε , con longitud cero.

Un morfismo entre dos grafos etiquetados $G_\Sigma = (G, g, \Sigma)$ y $H_\Omega = (H, h, \Omega)$ es un par $F = (F_1, F_2)$ de morfismos dados tales que el siguiente diagrama conmuta.

$$\begin{array}{ccc}
 G & \xrightarrow{g} & \tilde{\Sigma} \\
 F_1 \downarrow & & \downarrow F_2 \\
 H & \xrightarrow{h} & \tilde{\Omega}
 \end{array}$$

Los grafos etiquetados y sus morfismos forman una categoría que denotaremos por CGE .

3. Procesos Secuenciales

Un **proceso secuencial (PS)** P es un triple $P = (G_x, S, T)$, donde G_x es un grafo etiquetado, S es un subconjunto del conjunto de nodos de G_x que especifica nodos iniciales de P y T es un subconjunto del conjunto de nodos de G_x que especifican nodos finales de P .

En lo que sigue, los nodos de G_x serán llamados estados; luego, P será llamado un proceso secuencial de estado finito si el grafo G_x es finito.

Llamaremos a una sucesión finita $c = \sigma_1 \sigma_2 \dots \sigma_n$ de elementos de Σ una traza en el proceso $P = (G_x, S, T)$, si c es una traza en el correspondiente grafo etiquetado G_x de P . Con abuso de lenguaje llamaremos a los nodos q_0 comienzo y q_n final de la traza c en P si estos son los correspondientes comienzo y final de la traza c en G_x . De igual modo el número $n \geq 1$ será llamado la longitud de la traza.

Una traza será llamada exitosa en P si el comienzo de ésta es desde algún elemento de S y el final de ésta es en algún elemento de T .

Definimos el comportamiento de P como el conjunto de todas las trazas exitosas en P , denotaremos este conjunto por $|P|$.

Ahora podemos interpretar a un proceso secuencial como un grafo etiquetado donde se especifican sus estados iniciales y finales.

Un PS P será llamado accesible si para todo estado $x \in P$ existe una traza con comienzo en algún $x_0 \in S$ y final en x .

Un PS P es llamado coaccesible si para todo estado $x \in P$ existe una traza con comienzo en $x_0 \in S$ y final en algún $t \in T$.

Un PS P será llamado Limpio si es simultáneamente accesible y coaccesible.

Sean $S = (G_x, S, T)$ y $P = (H_\Omega, U, V)$, procesos secuenciales.

Un **morfismo $F : S \rightarrow P$ de procesos secuenciales** consiste en:

1. Un morfismo de grafos etiquetados $F = (F_1, F_2)$ de G_x en H_Ω
2. Un par de funciones $a : S \rightarrow U, b : T \rightarrow V$ dadas de manera que el siguiente diagrama conmute.

$$\begin{array}{ccccc}
 S & \xrightarrow{\Pi_1} & N & \xleftarrow{\Pi_2} & T \\
 a \downarrow & & \downarrow F_n & & \downarrow b \\
 U & \xrightarrow{\Pi_3} & M & \xleftarrow{\Pi_4} & V
 \end{array}$$

Donde los Π_i son proyecciones canónicas, para $i = 1, 2, 3, 4$

Los procesos secuenciales y sus morfismos forman una categoría la cual nosotros llamaremos CPS .

Una **máquina secuencial determinística (MS)** es un quintuple $A = (\Sigma, Q, q_0, T, \delta)$, donde Σ es un alfabeto, Q es un conjunto llamado conjunto de estados, q_0 es el estado inicial de A , T es el conjunto de estados finales de A y $\delta : \Sigma \cup \{\varepsilon\} \times Q \rightarrow Q$ es una función parcial llamada función de transición de estados de A , que está definida como sigue:

$$\delta(\sigma, q) = \begin{cases} q, & \text{si } \sigma = \varepsilon \\ q_2, & \text{si alcanzamos a } q_2 \\ & \text{desde } q \text{ con etiqueta } \sigma. \end{cases}$$

Extendemos esta función parcial δ de acuerdo a

$$\delta^*(u\sigma, q) = \delta(\sigma, \delta^*(u, q)), \text{ para } u \in \Sigma^* \text{ y } \sigma \in \Sigma.$$

Note que δ^* está definida siempre que $\delta^*(u, q)$ y $\delta(\sigma, \delta^*(u, q))$ estén definidas. En lo que sigue escribiremos $\delta(u, q)$ en lugar de $\delta^*(u, q)$, pues δ^* es una extensión de δ .

Sean $A = (\Sigma, Q, q_0, T, \delta)$ y $B = (\Omega, X, x_0, U, \beta)$ dos (MS). Un **morfismo $h : A \rightarrow B$** , es un par de funciones $h_1 : \Sigma \rightarrow \Omega$ y $h_2 : Q \rightarrow X$, tales que $h_2(q_0) = x_0$, $h_2(T) \subset U$ y $h_2(\delta(\Sigma \times Q)) \subset \beta(h_1(\Sigma) \times h_2(Q))$.

Las (MS), junto a sus morfismos forman una categoría la cual llamaremos CMS . Más aún, existe un functor $F : CMS \rightarrow CPS$ que mapea cada (MS) en un (PS), y además

$$|A| = |F(A)|.$$

Una **máquina secuencial generalizada (MSG)** es una Máquina secuencial Determinística

$A = (Q, \Sigma, q_0, T, \delta)$ junto con una función $\tau : \Sigma \cup \{\varepsilon\} \times Q \rightarrow \Gamma^*$, donde Γ es un alfabeto, dada por

$$\tau(\sigma, q) = \begin{cases} \varepsilon, & \text{si } \sigma = \varepsilon \\ w, & \text{para alguna } w \in \Gamma. \end{cases}$$

Esta función será llamada función de salida.

Sean $A = (\Sigma, Q, P, q_0, T, \delta, \tau)$ y $B = (\Omega, X, \phi, x_0, U, \beta, V)$ dos (MSG) . Un morfismo

$h : A \rightarrow B$ es un triple de funciones $h_1 : \Sigma \rightarrow \Omega$, $h_2 : Q \rightarrow X$, $h_3 : P \rightarrow \phi$ tales que $h_2(q_0) = x_0$, $h_2(T) \subset U$, $h_2\delta \subset \beta(h_1 \times h_2)$ y $h_3\tau \subset V(h_1 \times h_2)$.

Observación, cada (MSG) A , tiene una correspondiente (MS) , la cual es obtenida a partir de A por la eliminación de su función de salida.

Definimos el comportamiento de A , denotado por $|A|$ como el comportamiento de su correspondiente (MS) .

Las (MSG) y sus morfismos forman una categoría la cual llamaremos $(CMSG)$.

Una gramática regular determinística (Grm) G sobre Σ es un (PS) de estado finito y limpio con un solo estado inicial y con su grafo etiquetado determinístico.

Llamaremos a un lenguaje L sobre Σ , lenguaje regular si existe una (Grm) G sobre Σ tal que $|G| = L$.

La categoría $(CGrm)$ de gramáticas regulares es una subcategoría de (CPS) .

Las categorías $(CGrm)$ y (CMS) son isomorfas; por ende, en adelante consideraremos una (Grm) como un (MS) y una (MS) como una (Grm) siempre que sea conveniente.

Sean $G = (G_\Sigma, h_0, T_1)$ y $H = (H_\Sigma, m_0, T_2)$ dos (Grm) sobre el alfabeto Σ .

Considere ahora $G \cap H$ y el diagrama conmutativo siguiente

$$\begin{array}{ccc} K & \xrightarrow{b} & H \\ a \downarrow & & \downarrow h \\ G & \xrightarrow{g} & \tilde{\Sigma} \end{array}$$

donde $G = (Q_1, E_G, d_0, d_1)$, $H = (Q_2, E_H, d_0, d_1)$ y $K = (Q_1 \times Q_2, E_K, d_0, d_1)$ son grafos dirigidos y a, b, h y g son morfismos de grafos.

Luego $b \circ h$ y $a \circ g$ marcan los arcos de K con etiquetas de $\tilde{\Sigma}$.

De este modo definimos a $G \cap H$ como la componente limpia del proceso secuencial

$(K_\Sigma, (n_0, m_0), T_1 \times T_2)$. Luego estamos interesados en ver el comportamiento $|G \cap H|$ que resulta ser $|G \cap H| = |G| \cap |H| \subset \Sigma^*$. En efecto,

$$\begin{aligned} |G \cap H| &= \{\sigma \in \Sigma^* / (h_0, m_0)(\sigma) \in (T_1, T_2)\} \\ &= \{\sigma \in \Sigma^* / n_0(\sigma) \in T_1 \text{ y } m_0(\sigma) \in T_2\} \\ &= \{\sigma \in \Sigma^* / n_0(\sigma) \in T_1\} \cap \{\sigma \in \Sigma^* / m_0(\sigma) \in T_2\} \\ &= |G| \cap |H| \end{aligned}$$

Así tenemos un procedimiento finito para definir el comportamiento $|G \cap H|$ dados los comportamientos de G y H .

Dados Ω y Σ dos alfabetos disjuntos; es decir $\Omega \cap \Sigma = \emptyset$ Definimos la **proyección "Shuffle"**

$P : (\Sigma \cup \Omega)^* \rightarrow \Sigma^* \times \Omega^*$ de la manera siguiente.

Considere

$$P_1(\sigma) = \begin{cases} (\sigma, \varepsilon), & \text{si } \sigma \in \Sigma \\ (\varepsilon, \sigma), & \text{si } \sigma \in \Omega \end{cases}$$

Y considere $\Pi_1 : \Sigma \times \Omega \rightarrow \Sigma$ y $\Pi_2 : \Sigma \times \Omega \rightarrow \Omega$, proyecciones canónicas.

Para cada $\gamma \in (\Sigma \cup \Omega)^*$ se tiene que $\gamma = \sigma_1 \cdots \sigma_n$, donde $\sigma_i \in \Sigma \cup \Omega$, $1 \leq i \leq n$

luego:

$$P(\gamma) = P(\sigma_1 \cdots \sigma_n)$$

$$= (\Pi_1(P_1(\sigma_1)) \cdots \Pi_1(P_1(\sigma_n)), \Pi_2(P_1(\sigma_1)) \cdots \Pi_2(P_1(\sigma_n)))$$

donde para $\beta, \alpha \in \Sigma \times \Omega$ consideramos β concatenando a α , denotado por $\beta\alpha$ de la siguiente

manera: si $\beta = (\beta_1, \beta_2)$ y $\alpha = (\alpha_1, \alpha_2)$ entonces $\beta\alpha = (\beta_1\alpha_1, \beta_2\alpha_2)$ La proyección "Shuffle" definida así es un homomorfismo.

Sean $L_1 \in \Sigma^*$ y $L_2 \in \Omega^*$, lenguajes regulares donde $\Sigma \cap \Omega = \emptyset$, el **producto Shuffle** de L_1 y L_2 denotado por $L_1 \otimes L_2$ está definido por

$$L_1 \otimes L_2 := P^{-1}(L_1 \times L_2)$$

Sean L_1 y L_2 lenguajes regulares, y sean $G = (G_\Sigma, q_0, T_1)$ y $H = (H_\Omega, x_0, T_2)$ dos gramáticas regulares tales que $|G| = L_1$ y $|H| = L_2$, donde Q es el conjunto de nodos de G y X es el conjunto de nodos de H . Supondremos también que $\Sigma \cap \Omega = \emptyset$.

Definimos la **gramática producto Shuffle** como sigue: consideremos el grafo dirigido $G_{\Sigma \cup \Omega}$ donde $Q \times X$ es el

conjunto de nodos. $\sigma \in \Sigma \cup \Omega$ es un arco en $G_{\Sigma \cup \Omega}$ si $\sigma \in \Sigma$ y existe un arco $q_1 \xrightarrow{\sigma} q'$ en G denotando en este caso el arco σ de $G_{\Sigma \cup \Omega}$ por $(q_1, x) \xrightarrow{\sigma} (q', x)$, o si $\sigma \in \Omega$ y existe el arco $x \xrightarrow{\sigma} x'$ en H , denotando en este caso el arco σ de $G_{\Sigma \cup \Omega}$ por $(q, x) \xrightarrow{\sigma} (q, x')$.

Luego $G \otimes H$ es la componente Limpia del proceso secuencial $(G_{\Sigma \cup \Omega}, (q_0, x_0), T_1 \times T_2)$ y su comportamiento es $L_1 \otimes L_2$.

Como consecuencia inmediata de esto, el producto shuffle de dos lenguajes regulares es un lenguaje regular.

4. Procesos de Eventos Discretos Controlables

Un proceso de eventos discreto controlable (PEDC) P consiste en:

1. Un $PS \ P = (G_{\Sigma \cup \{\varepsilon\}}, S, T)$;
2. Un conjunto de eventos de entrada $\Gamma = \Gamma_e \cup \Gamma_\delta$, donde $\Gamma_e \cap \Gamma_\delta = \emptyset$ y $\Gamma \cap \Sigma = \emptyset$;
3. Un par de funciones parciales $f_e: E \rightarrow \Gamma_e$ y $f_\delta: E \rightarrow \Gamma_\delta$ con el mismo dominio, donde E es el conjunto de arcos de P , f_e especifica los eventos habilitados y f_δ los eventos inhabilitados para cada arco controlable.

Un proceso (PEDC) P tiene la siguiente interpretación: Los nodos de G representan los estados de P y los arcos de G (llamados eventos) representan transiciones de estados que son permitidas. Los estados en S son llamados estados iniciales permitidos y los estados en T son llamados estados finales permitidos. Cada transición de estado de P es controlable o no controlable. Si una transición de estado es controlable entonces ésta puede ser permitida o inhabilitada; en otro caso, la transición siempre será permitida. La "ocurrencia" de un evento permitido por una transición controlable habilita la transición, mientras la "ocurrencia" de los eventos no permitidos inhabilita la transición. En el momento en el cual estamos en el estado q en P , el proceso determina la ejecución de alguna transición del estado q hacia algún estado q' en P ; si la transición es permitida entonces P ejecuta la transición instantáneamente. Si la transición es inhabilitada entonces P espera en el estado q hasta elegir hacer transiciones permitidas; así P ejecuta las transiciones seleccionadas.

A Σ lo llamaremos conjunto de etiquetas de salida del (PEDC) P ; es decir, si $\sigma \in \Sigma$ es una etiqueta de un evento e de P entonces cuando e es ejecutado la etiqueta ocurre simultáneamente.

El comportamiento de P denotado por $|P|$, es de este modo el conjunto formado por todas las posibles

sucesiones de etiquetas de salida, que puedan ser generadas en la evolución del sistema.

En adelante asumiremos que cada proceso P es un proceso secuencial de estado finito (PSEF), limpio, con un estado inicial y con su grafo etiquetado determinístico, ya que con esta elección el comportamiento de P , $|P|$, es un lenguaje regular.

Sea $g: E \rightarrow \Sigma \cup \{\varepsilon\}$ la función etiqueta de P .

Sea $i: \Sigma \rightarrow \Sigma \cup \{\varepsilon\}$ la función inclusión de P . Así, $g': E \rightarrow \Sigma$ es la única función parcial "maximal" que verifica $i(g'(E)) \subset g(E)$.

P será llamado de salida controlable si existe un par de funciones parciales inyectivas

$$h_e: \Sigma \rightarrow \Gamma_e \text{ y } h_\delta: \Sigma \rightarrow \Gamma_\delta \text{ tales que } f_e = h_e(g') \text{ y } f_\delta = h_\delta(g').$$

Asumiremos que todo (PEDC) es de salida controlable. De este modo para precisar una condición inicial para P nosotros tenemos que dar un estado inicial para cada salida controlable de P .

Un sistema de eventos discreto (SED) es un conjunto finito

$$\mathcal{H} = \{P_1, \dots, P_n \mid \text{cada } P_i \text{ es un (PEDC)}\}$$

con alfabetos de entrada disjuntos y alfabetos de salida disjuntos. Definimos el comportamiento de $\mathcal{H}$ como el producto shuffle

$$L_{\mathcal{H}} = \bigotimes_{i=1}^n |P_i|$$

del comportamiento de los procesos constituyentes. Nosotros interpretaremos a $\mathcal{H}$ como un conjunto de procesos asincrónicos independientes que interactúan sobre su medio compartido. Esta interacción da comienzo a la necesidad de supervisión del comportamiento colectivo. El objeto de la supervisión es asegurar que los procesos interactúen alcanzando una coexistencia armoniosa o concluyan tareas colectivas.

En el resto del trabajo sea

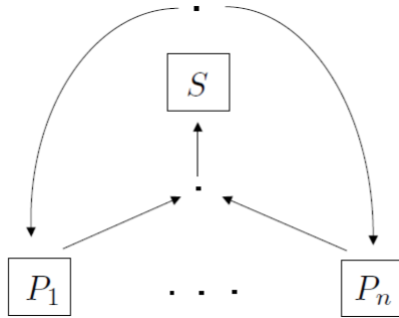
$\mathcal{H} = \{P_1, \dots, P_n \mid \text{cada } P_i \text{ es un (PEDC)}\}$ un (SED) con Σ_i alfabetos de salida, $\Gamma_i = \Gamma_{ei} \cup \Gamma_{\delta i}$ alfabetos de entrada y $\Lambda_i \subset \Sigma_i$ los conjuntos de salida controlable de P_i , $i = 1, \dots, n$.

Además

$$\Sigma = \bigcup_{i=1}^n \Sigma_i, \quad \Gamma = \bigcup_{i=1}^n \Gamma_i \quad \text{y} \quad \Lambda = \bigcup_{i=1}^n \Lambda_i.$$

Un supervisor Secuencial S para $\mathcal{H}$ es una (MSG) determinística de estado finito y limpia, con Σ como conjunto de entrada y Γ como conjunto de salida; asumiremos también que todo estado de S finaliza una tarea; es decir, $pre\ |S| = |S|$.

La figura siguiente representa la supervisión central de un (SED) por un supervisor secuencial



S es secuencial y la observación de la actividad de los procesos es modelada mediante un shuffling de la comunicación entrante en una secuencia. La secuencia de salida de S es un shuffling de sucesiones de comandos para los procesos de $\mathcal{H}$.

El control es separado por una proyección shuffle y transmitido a los correspondientes procesos. Se asume que la demora de comunicación entre $\mathcal{H}$ y S es insignificante.

Sea S un supervisor para un (SED) $\mathcal{H}$. Para cada $\sigma \in \Lambda$ sea $\langle \sigma \rangle$ denotando los eventos habilitados y $\langle \bar{\sigma} \rangle$ denotando los eventos inhabilitados por σ .

Sea $\Gamma_\sigma = \{ \langle \sigma \rangle, \langle \bar{\sigma} \rangle \}$ y $P_\sigma: \Gamma^* \rightarrow \Gamma_\sigma^*$ la ε -proyección de Γ^* sobre Γ_σ^* dada por

$$P_{\sigma 0}(w) = \begin{cases} w, & \text{si } w \in \Gamma_\sigma \\ \varepsilon, & \text{en otro caso} \end{cases}$$

Luego si $z \in \Gamma^*$ y $z = z_1 \cdots z_n$ con $z_i \in \Gamma \cup \{\varepsilon\}$, $i = 1, \dots, n$ entonces

$$P_\sigma(z) = P_{\sigma 0}(z_1) \cdots P_{\sigma 0}(z_n).$$

Sea $\alpha: \Lambda \rightarrow \{0, 1\}$ la función que especifica el status inicial de cada salida controlable. Se extiende la función α mediante una función $\Phi: |S| \times \Lambda \rightarrow \{0, 1\}$ como sigue:

- i.) $\Phi(\varepsilon, \sigma) = \alpha(\sigma)$
- ii.) para $u \in |S|$, $\Phi(u, \sigma) = 1$ si $\alpha(\sigma) = 1$ y $P_\sigma(\tau(u)) = \varepsilon$ o $P_\sigma(\tau(u)) = \langle \sigma \rangle$
- iii.) en otro caso $\Phi(u, \sigma) = 0$.

Si $\Phi(u, \sigma) = 1$ o $\Phi(u, \sigma) = 0$ entonces diremos que σ es habilitado o inhabilitado desde u .

El conjunto L_p contenido en Σ^* de sucesiones de salidas controladas de $(\mathcal{H}, S)$ es definido por: $w \in L_p$ si $w \in |S| \cap pre(L_{\mathcal{H}})$ y para cada factorización $u\sigma v$ de w con $\sigma \in \Lambda$, $\Phi(u, \sigma) = 1$.

Definimos el comportamiento controlado de $(\mathcal{H}, S)$ por $L_c := L_p \cap L_{\mathcal{H}}$.

Claramente $pre(L_p) = L_p$ y $pre(L_c) \subseteq L_p$.

Si $w \in L_p - pre(L_c)$ entonces w es una sucesión de salida de $(\mathcal{H}, S)$ la cual posiblemente pueda completarse legalmente para formar una palabra de $L_{\mathcal{H}}$.

Teorema 1. El comportamiento controlado de $(\mathcal{H}, S)$ es un lenguaje regular.

Demostación. Debemos mostrar la existencia de una gramática regular que genere a L_c :

Para esto recordemos primero que al discutir la gramática producto shuffle concluimos que el producto shuffle de dos lenguajes regulares L_1 y L_2 es un lenguaje regular.

De aquí $L_{\mathcal{H}} = \bigotimes_{i=1}^n |P_i|$ es un lenguaje regular y por lo tanto $pre(L_{\mathcal{H}})$ es un lenguaje regular, pues la gramática que genera a $L_{\mathcal{H}}$ también producirá cualquier palabra de $pre(L_{\mathcal{H}})$.

Sea entonces $P_1 = (G_{\Sigma \cup \{\varepsilon\}}, s_0, T)$ la gramática que genera a $pre(L_{\mathcal{H}})$. Como la categoría ($CGrm$) de gramáticas regulares y categoría (CMS) de máquinas Determinísticas son isomorfas entonces nuestra prueba consistirá en mostrar la existencia de una máquina determinística que genere a L_p (ya que como $L_{\mathcal{H}}$ es regular, entonces $L_c = L_p \cap L_{\mathcal{H}}$ será regular).

Consideremos ahora la máquina determinística $M_1 = (Q, \Sigma, q_0, T, \delta)$ correspondiente con P_1 por isomorfismo, luego el comportamiento de M_1 es $L_{\mathcal{H}}$.

Sea $M_2 = (Q_2, \Sigma, q_2, T_2, \delta_2)$ la máquina determinística en CMS que es obtenida a partir de S por la eliminación de la función de salida. Es claro que con esta elección el comportamiento de M_2 es $|S|$.

Consideremos ahora la máquina $M_{1c} = (Q, \Sigma, T, q_0, \delta)$, donde $\delta_c: \Sigma \times Q \rightarrow Q$

está dada por

$$\delta_c(\sigma, q) = \begin{cases} \delta(\sigma, q), & \text{si } (\sigma, q) \text{ esta definida y } \Phi(w_n, \sigma) = 1 \\ \text{indefinida en otro caso} \end{cases}$$

Aquí hemos supuesto que $\delta(w, q_0) = q$ y $w = w_1 \cdots w_n$ con $w_i \in \Sigma$ para cada i .

Considere hora

$A_c = (Q \times Q_2, \Sigma, T \times T_2, (q_0, q_2), \delta_2 \times \delta_c)$, donde

$$\delta_2 \times \delta_c: \Sigma \times Q_2 \times Q \rightarrow Q_2 \times Q$$

es dada por

$$\delta_2 \times \delta_c(\sigma, x, q) = (\delta_c(\sigma, q), \delta_2(\sigma, x))$$

siempre que $\delta_2(\sigma, x)$ y $\delta_c(\sigma, x)$ estén definidas; además, como para cada $wu\sigma \in pre(|A_c|)$ se tiene que $\Phi(u, \sigma) = 1$, se concluye que esta máquina genera a L_p . ■

Un supervisor S es funcionalmente completo si para cada $u \in L_p$, $u\sigma \in pre(L_{\mathcal{H}})$ con $\sigma \in \Sigma$, y σ habilitada desde u implica que $u\sigma \in |S|$.

Abreviaremos funcionalmente completo por F -completo.

Un supervisor S es funcionalmente limpio si para cada estado $x \in X$ en S y para cada $\sigma \in \Sigma$ tal que $\delta(\sigma, x)$ esta definido, existe una palabra $u\sigma v$ perteneciente a L_c tal que $\delta(u, x_0) = x$. Abreviaremos funcionalmente limpio por F -Limpio.

Asumiremos en adelante que todo supervisor es F -completo y F -Limpio. Dada esta elección, S no tendría transiciones de estado que no jueguen un papel importante en la supervisión de $\mathcal{H}$.

Sea X el conjunto de estado de S . Para cada $\sigma \in \Lambda$ nosotros diremos que un estado $x \in X$ es σ -consistente si para cada par $u, v \in |S|$ se tiene que $\delta(u, x_0) = x = \delta(v, x_0)$ y $\Phi(u, \sigma) = \Phi(v, \sigma)$.

Diremos que un supervisor S es control-consistente si todo estado de S es σ -consistente, para cada $\sigma \in \Lambda$.

Cada estado de un supervisor control-consistente determina de forma única el estatus de cada salida controlable.

Sea S un supervisor control-consistente. El estatus que da x a σ es dado por la sobreyección

$\Pi_\sigma: X \rightarrow \{0, 1\}$ con $\Pi_\sigma(x) = 1$ si $\Phi(u, \sigma) = 1$ para algún $u \in |S|$, donde $\delta(u, x_0) = x$.

Denotaremos también por Π_σ al núcleo de esta sobreyección.

Definimos la control-partición de X por $\Pi = \Lambda\{\Pi_\sigma/\sigma \in \Lambda\}$.

Dos supervisores S y R de $\mathcal{H}$ hacen un control equivalente si ambos generan el mismo conjunto de salidas controladas para $\mathcal{H}$; es decir si $L_p(S) = L_p(R)$.

Teorema 2. Para cada supervisor S de $\mathcal{H}$ existe un supervisor S^* y un epimorfismo $F: S^* \rightarrow S$ en (Gsm) con S^* un supervisor control-consistente para $\mathcal{H}$ el cual es equivalentemente controlado por S .

Demostración. Sea $S = (\Sigma, X, \Gamma, x_0, T, \delta, \tau)$ un supervisor para $\mathcal{H}$, si S es control-consistente, entonces basta tomar $S^* = S$. Supongamos entonces que S no es control-consistente; sea $\Lambda' \subseteq \Lambda$ el subconjunto del conjunto de salidas controlables de $\mathcal{H}$, para el cual existen estados de S los cuales no son consistentes. Sea $\sigma \in \Lambda'$ y $X_\sigma = \{0, 1\}$. Definamos $X_\sigma = (\Gamma, X_\sigma, x_0, \delta_\sigma)$, donde $x_0 = \Phi(\sigma)$, $\delta_\sigma(\gamma, x) = 0$ si $\gamma = \langle \bar{\sigma} \rangle$ y $\delta_\sigma(\gamma, x) = 1$ si $\gamma = \langle \sigma \rangle$, en otro caso $\delta_\sigma(\gamma, x) = x$. Luego X_σ es una (MS) que especifica el estatus de σ . Sea m la cardinalidad Λ' y sea $Y = \{0, 1\}^m$. Definamos la (MS) $\mathcal{Y} = (\Gamma, Y, y_0, \alpha)$, donde $y_0 = (x_0^1, \dots, x_0^m)$ y $\alpha(\gamma, (x^1, \dots, x^m)) = ((x^1)\gamma, \dots, (x^m)\gamma)$. [Denotando en este caso por $(x^i)\gamma$ al estado alcanzado por γ desde x^i]. Luego la siguiente (MS) es la extensión dinámica que consideramos para S , $S_e = (\Sigma, X \times Y, (x_0, y_0), \delta_e, \tau_e)$, donde $\delta_e(\sigma, x, y) := (\delta(\sigma, x), \alpha^*(\tau(\sigma, x), y))$ si $\delta(\sigma, x)$ está definida y es habilitada en el estado y de $\mathcal{Y}$; en otro caso $\delta_e(\sigma, x, y)$ es indefinida, $\tau_e(\sigma, x, y) := \tau(\sigma, x)$. Por construcción S_e hace un control equivalente al de S y todo de estado S_e es σ -consistente para cada $\sigma \in \Lambda$ y S es un cociente de S_e en (Gsm) . Finalmente consideramos $S^* = S$. ■

Teorema 3. Si S es un supervisor control-consistente para $\mathcal{H}$ entonces $L_p = |S| \cap pre(L_{\mathcal{H}})$ y $L_c = |S| \cap L_{\mathcal{H}}$

Demostración. Sea $w \in |S| \cap pre(L_{\mathcal{H}})$ y sea $u\sigma v$ una factorización de w , con $\sigma \in \Lambda$. Como $w \in |S|$ entonces σ es una entrada permitida desde $\delta(u, x_0)$ en S y como S es F -Limpio entonces existe una palabra $t\sigma$, donde $\Phi(t, \sigma) = 1$, pero S es control-consistente, por lo tanto $\Phi(u, \sigma) = \Phi(t, \sigma) = 1$ de donde $\delta(u, x_0) = \delta(t, x_0)$, así $w \in L_p$.

Además si $w \in |S| \cap L_{\mathcal{H}}$, como $L_{\mathcal{H}} \subset pre(L_{\mathcal{H}})$

entonces $w \in |S| \cap pre(L_{\mathcal{H}})$ de donde se tiene

que $w \in L_p$. Luego, $w \in L_p$ y $w \in L_{\mathcal{H}}$, así $w \in L_p \cap L_{\mathcal{H}} = L_c$.

Si $w \in L_c$ entonces $w \in L_p$ y $w \in L_{\mathcal{H}}$, luego $w \in |S| \cap pre(L_{\mathcal{H}}) \cap L_{\mathcal{H}} = |S| \cap L_{\mathcal{H}}$.

Así $L_c = |S| \cap L_{\mathcal{H}}$. ■

Un supervisor se llama restrictivo si verifica la siguiente propiedad:

Para cada $\sigma \in \Sigma$ y para cada prefijo $uw\sigma$ en $L_{\mathcal{H}}$, con $w \in \Sigma$.

Si uw es prefijo en L_c y $\Phi(uw, \sigma) = 0$ entonces $\Phi(u, \sigma) = 0$.

Sea S un supervisor control-consistente para $\mathcal{H}$, para cada $\sigma \in \Sigma$ definamos los siguientes subconjuntos de estados de X , donde X es el conjunto de estados de S .

$$D_\sigma = \{x \in X / \delta(\sigma, x) \text{ está definido} \}$$

$$E_\sigma = \{x \in X / x \text{ habilita a } \sigma \}$$

$$\overline{E}_\sigma = \{x \in X / x \text{ inhabilita a } \sigma \}$$

$$\overline{D}_\sigma = \{x \in X / x \text{ inhabilita a } \sigma, \exists u \in \text{pre}(L_c), \\ \text{donde } \delta(u, x_0) = x \text{ y } u\sigma \in \text{pre}(L_{\mathcal{H}})\}$$

Sea $K \subset X$, el subconjunto $(K)\Sigma^{-1} \subset X$ se define por

$$(K)\Sigma^{-1} = \{x \in X / \exists \sigma \in \Sigma, \delta(\sigma, x) \text{ está} \\ \text{definido}, \delta(\sigma, x) \in K\}$$

Proposición 1. Para un supervisor control-consistente S las siguientes condiciones son equivalentes:

- 1.) S es un supervisor restrictivo
- 2.) $\overline{D}_\sigma \Sigma^{-1} \subset \overline{E}_\sigma, \forall \sigma \in \Sigma$
- 3.) $(E_\sigma)\Sigma \cap \overline{D}_\sigma = \emptyset, \forall \sigma \in \Sigma$.

Demostración. 1.) $\Rightarrow$ 2.)

Si $\overline{D}_\sigma = \emptyset$ entonces $\overline{D}_\sigma \Sigma^{-1} = \emptyset \subset \overline{E}_\sigma, \forall \sigma \in \Sigma$.

Razonemos por el contra recíproco. Supongamos que $\overline{D}_\sigma \neq \emptyset$ y que $\exists \sigma \in \Sigma / (\overline{D}_\sigma)\Sigma^{-1} \not\subset \overline{E}_\sigma$, luego existe $y \in \overline{D}_\sigma \Sigma^{-1}$ y simultáneamente $y \notin \overline{E}_\sigma$.

Como $y \in \overline{D}_\sigma \Sigma^{-1}$ entonces existe $w \in \Sigma$ tal que $\delta(w, y)$ está definido y $\delta(w, y) \in \overline{D}_\sigma$, ahora

como $\delta(w, y) \in \overline{D}_\sigma$ entonces $\delta(w, y)$ inhabilita a σ , luego como $y \notin \overline{E}_\sigma$ entonces $y \in E_\sigma$; es decir, y habilita a σ lo que es una contradicción con el hecho de ser S restrictivo.

2.) $\Rightarrow$ 3.)

Razonemos por el absurdo. Asumamos que $\overline{D}_\sigma \Sigma^{-1} \subset \overline{E}_\sigma, \forall \sigma \in \Sigma$ y que $\exists \sigma \in \Sigma / (E_\sigma)\Sigma \cap \overline{D}_\sigma \neq \emptyset$.

Luego, si $x \in (E_\sigma)\Sigma \cap \overline{D}_\sigma$ entonces existe $y \in E_\sigma$ y $w \in \Sigma$ tal que $\delta(w, y) \in \overline{D}_\sigma$ de donde $y \in E_\sigma \cap \overline{D}_\sigma \Sigma^{-1}$ lo que es una contradicción a la hipótesis pues si $\overline{D}_\sigma \Sigma^{-1} \subset \overline{E}_\sigma \forall \sigma \in \Sigma$, entonces $\overline{D}_\sigma \Sigma^{-1} \cap E_\sigma = \emptyset$.

3.) $\Rightarrow$ 1.)

Suponga que $(E_\sigma)\Sigma \cap \overline{D}_\sigma = \emptyset \forall \sigma \in \Sigma$ entonces sea $uw\sigma$ un prefijo en $L_{\mathcal{H}}$ donde uw es un prefijo en L_c y sea

σ inhabilitado después de uw . Sea $\delta(uw, x_0) = x$ y $\delta(u, x_0) = y$, entonces $x \in \overline{D}_\sigma$ y por lo tanto $y \notin E_\sigma$, de aquí σ es inhabilitado después de u . Así S es restrictivo. ■

Un lenguaje regular $L \subset L_{\mathcal{H}}$ es llamado parcialmente invariante si para cada par $\sigma, w \in \Sigma$, donde $u\sigma$ y uw son prefijos en L y $uw\sigma$ es un prefijo en $L_{\mathcal{H}}$ se tiene que $uw\sigma$ es un prefijo en L .

Sean A una (MS) tal que $|A| = L$, Con Q el conjunto de estados de A y δ la función de transición de estados de A . Para cada $\sigma \in \Sigma$ definamos los siguientes subconjuntos de Q :

$$A_\sigma := \{q \in Q / \delta(\sigma, q) \text{ está definido} \}$$

$$\overline{A}_\sigma := \{q \in Q / \delta(\sigma, q) \text{ no está definido y } \exists u \\ \in \text{pre}(L), \text{ donde } \delta(u, q_0) = q \text{ y } u\sigma \in \text{pre}(L_{\mathcal{H}})\}$$

$$\overline{\overline{A}}_\sigma := \{q \in Q / \delta(\sigma, q) \text{ no está definido y } \forall u \\ \in \text{pre}(L), \text{ donde } \delta(u, q_0) = q \text{ y } u\sigma \notin \text{pre}(L_{\mathcal{H}})\}$$

Claramente $A_\sigma, \overline{A}_\sigma$ y $\overline{\overline{A}}_\sigma$ son dos a dos disjuntos y $A_\sigma \cup \overline{A}_\sigma \cup \overline{\overline{A}}_\sigma = Q$.

Proposición 2. $L \subset L_{\mathcal{H}}$ es parcialmente invariante si, y sólo si, para cada $\sigma \in \Sigma, (\overline{A}_\sigma)\Sigma^{-1} \subseteq (\overline{A}_\sigma) \cup \overline{\overline{A}}_\sigma$

Demostración. Sea $L \subset L_{\mathcal{H}}$ y sea A una (MS) tal que $|A| = L$, Con Q' el conjunto de estados de A , δ la función de transición de A y donde Q denotará al conjunto de nodos o estados de $\mathcal{H}$.

Sabemos que si $\alpha \in L_{\mathcal{H}}$ entonces existen nodos q_1, q_2 en Q tales que el arco de q_1 hasta q_2 tiene la etiqueta α (denotando esto por $q_1 \xrightarrow{\alpha} q_2$). También, si $\alpha \in L$ entonces existen estados q'_1 y q'_2 en Q' tales que $\delta(\alpha, q'_1) = q'_2$. En este caso interpretaremos que $q'_1 = q_1$ y $q'_2 = q_2$, bajo esta consideración entonces interpretaremos a $Q' \subset Q$.

Ahora aboquémonos a la demostración de la proposición.

($\Rightarrow$)

Razonemos por el contrarecíproco y supongamos que L no es parcialmente invariante, entonces existen σ y w en Σ tales que $u\sigma, uw$ pertenecen a $\text{pre}(L)$, $uw\sigma \in \text{pre}(L_{\mathcal{H}})$ y $uw\sigma \notin \text{pre}(L)$. Como $uw\sigma \in \text{pre}(L_{\mathcal{H}})$ existen estados q_0, q_1, q_2, q_3 en Q tales que $q_0 \xrightarrow{u} q_1 \xrightarrow{w} q_2 \xrightarrow{\sigma} q_3$ y donde $uw \in \text{pre}(L)$, $uw\sigma \notin \text{pre}(L)$ y $uw\sigma \in L_{\mathcal{H}}$. Ahora para estos σ y q_1 , existe $w \in \Sigma$ tal que $\delta(w, q_1)$ está definido en A , $\delta(\sigma, \delta(w, q_1))$ no está definido en A y también existe $uw \in \text{pre}(L)$ tal que

$\delta(uw, q_0) = \delta(w, q_1)$. Por lo tanto, $uw\sigma \in L_{\mathcal{H}} \subset pre(L_{\mathcal{H}})$. Así $q_1 \in (\overline{A_\sigma} \Sigma^{-1})$. Pero $q_1 \notin \overline{A_\sigma} \cup \overline{\overline{A_\sigma}}$ pues q_1 debe ser tal que $\delta(\alpha, q_1)$ es definido en A para todo $\alpha \in \Sigma$ y sabemos que $uw \in pre(L)$. Luego, queda demostrada esta implicación.

($\Leftarrow$)

Supongamos que existe $\sigma \in \Sigma$ tal que $(\overline{A_\sigma}) \Sigma^{-1} \not\subset \overline{A_\sigma} \cup \overline{\overline{A_\sigma}}$ y probemos que $L \subset L_{\mathcal{H}}$ no es parcialmente invariante. Sea entonces $\sigma \in \Sigma$ tal que $(\overline{A_\sigma}) \Sigma^{-1} \not\subset \overline{A_\sigma} \cup \overline{\overline{A_\sigma}}$ y sea $q_1 \in (\overline{A_\sigma}) \Sigma^{-1}$ tal que $q_1 \notin \overline{A_\sigma} \cup \overline{\overline{A_\sigma}}$. Como $q_1 \in (\overline{A_\sigma}) \Sigma^{-1}$ entonces $\exists w \in \Sigma$ tal que $\delta(w, q_1)$ está definido y $\delta(w, q_1) \in \overline{A_\sigma}$, luego como $\delta(w, q_1) \in \overline{A_\sigma}$, entonces $\delta(\sigma, \delta(w, q_1))$ es indefinido, existe $uw \in L$ tal que $\delta(uw, q_0) = \delta(w, q_1)$ y $uw\sigma \in L_{\mathcal{H}}$; además, $q_1 \notin \overline{A_\sigma} \cup \overline{\overline{A_\sigma}}$ pues $\delta(w, q_1)$ está definida.

Luego, para estos $\sigma w \in \Sigma$ tenemos que $uw \in L \subset pre(L)$ y $uw\sigma \in L_{\mathcal{H}} \subset pre(L_{\mathcal{H}})$ y además $\delta(\sigma, \delta(w, q_1))$ está indefinido; es decir, $uw\sigma$ no es un prefijo en L . Así L es parcialmente invariante. ■

Proposición 3. $L \subset L_{\mathcal{H}}$ es parcialmente invariante si, y sólo si, para cada $\sum_i, i = 1, \dots, n$ y para cada $\sigma \in \sum_i$ se verifica:

1.) $(A_\sigma)(\Sigma - \sum_i) \subset A_\sigma$; es decir, si $q \in A_\sigma$ entonces para cada $w \in \Sigma - \sum_i, \delta(w, q) \in A_\sigma$

siempre que $\delta(w, q)$ este definido.

2.) $(A_\sigma)\sum_i \subset A_\sigma \cup \overline{\overline{A_\sigma}}$; es decir, si $q \in A_\sigma$ entonces para cada $\sigma' \in \sum_i, \delta(\sigma', q) \in A_\sigma \cup \overline{\overline{A_\sigma}}$

siempre que $\delta(\sigma', q)$ este definido.

Demostración. ($\Rightarrow$)

Sean $\sigma \in \Sigma$ y $w \in \Sigma$, suponga que $u\sigma$ y uw son prefijos en L y que $uw\sigma$ es un prefijo en $L_{\mathcal{H}}$.

Si $w \notin \sum_i$ entonces $w \in \Sigma - \sum_i$, luego como $u\sigma \in pre(L)$, entonces existe $q \in Q$ tal que $\delta(w, q)$ está definido en A y por lo tanto $q \in A_\sigma$, luego $\delta(w, q) \in A_\sigma$. Así, $uw\sigma \in pre(L)$. Si $w \in \sum_i$, como $uw \in pre(L)$ entonces existe $q \in Q$ tal que $\delta(\sigma, q)$ está definida en A y entonces $q \in A_\sigma$ además, como $uw \in pre(L)$ entonces $\delta(w, q)$ está definida en A ; luego $q \in A_\sigma \sum_i$ implica que $\delta(w, q) \in A_\sigma \cup \overline{\overline{A_\sigma}}$ pero $\delta(w, q) \notin \overline{\overline{A_\sigma}}$ (pues si suponemos que $\delta(\sigma, \delta(w, q))$ está indefinida en A , entonces existiría $uw \in pre(L)$ donde $\delta(uw, q_0) = \delta(w, q)$, luego $u\sigma \notin pre(L_{\mathcal{H}})$ lo que contradice la hipótesis). Por lo tanto, $\delta(w, q) \in A_\sigma$ y así

$uw\sigma \in pre(L)$. En consecuencia L es parcialmente invariante.

($\Leftarrow$)

Supongamos que L es parcialmente invariante y sean $\sigma \in \sum_i$ y $\delta(u, q_0) = q \in A_\sigma$ para algún prefijo $u \in pre(L)$. Suponga que $\delta(w, q)$ está definido con $w \notin \sum_i$, luego $u\sigma$ y uw son prefijos en L . Así, por la definición de $L_{\mathcal{H}}$, $uw\sigma \in pre(L_{\mathcal{H}})$, de donde $uw\sigma \in pre(L)$. Por lo tanto, $\delta(w, q) \in A_\sigma$. Si $w \in \sum_i$, entonces $w \in \Sigma - \sum_i$, luego $\delta(w, q) \in A$. Suponga que $\delta(w, q)$ está definido con $w \in \sum_i$, entonces $u\sigma$ y uw son palabras de $pre(L)$. Si $uw\sigma \in pre(L_{\mathcal{H}})$ entonces $uw\sigma \in pre(L)$, luego $\delta(w, q) \in A_\sigma$, de otra manera $\delta(w, q) \in \overline{\overline{A_\sigma}}$. Así $\delta(w, q) \in A \cup \overline{\overline{A_\sigma}}$. ■

Proposición 4.

Sea S un supervisor para $\mathcal{H}$, F -completo y restrictivo, entonces el comportamiento controlado de $(\mathcal{H}, S)$ es parcialmente invariante.

Demostración.

Sea S un supervisor control-consistente para $\mathcal{H}$. Por la Teorema 1. $L_c = L_{\mathcal{H}} \cap |S|$, donde

$$L_{\mathcal{H}} = \bigotimes_{i=1}^n |P_i|.$$

Sean $w \in \Sigma$ y $\sigma \in \Sigma$ tales que $u\sigma$ y uw son prefijos en

L_c . Como $L_c = L_{\mathcal{H}} \cap |S|$, entonces $u\sigma$ y uw son prefijos en $L_{\mathcal{H}}$. Así existe un número entero $i \leq n$, tal que $\sigma \in |P_i|$; luego como uw es prefijo en $L_{\mathcal{H}}$, entonces por definición de $L_{\mathcal{H}}$ $uw\sigma$ es un prefijo en $L_{\mathcal{H}}$. También como $u\sigma$ y uw son prefijos en $L_c = L_{\mathcal{H}} \cap |S|$, entonces $u\sigma$ y uw son prefijos en $|S|$; por lo tanto, existe un estado $x = \delta(u, x_0)$ en S desde el cual, $\delta(\sigma, x)$ y $\delta(w, x)$ están definidos. Luego como $\sigma \in \Sigma$ y $w \in \Sigma$, $uw\sigma$ es un prefijo en $L_{\mathcal{H}}$ y uw es prefijo en L_c y como S es restrictivo, entonces si suponemos que $\phi(uw, \sigma) = 0$ concluimos que $\phi(u, \sigma) = 0$, lo cual contradice la Hipótesis $\phi(u, \sigma) = 1$, pues $u\sigma$ es un prefijo en L_c y por ende $\phi(uw, \sigma) = 1$. Además, como S es F -completo entonces $uw\sigma$ es un prefijo en $|S|$. Así, $uw\sigma$ es un prefijo en $L_{\mathcal{H}} \cap |S| = L_c$. ■

Sea S un supervisor para $\mathcal{H}$ y sea L_c el comportamiento controlado de $(\mathcal{H}, S)$. Diremos que S "Permite Detención" de $\mathcal{H}$ si existe una palabra u en L_p , después de la cual todo proceso que no se detenga estará bloqueado y no podrá proceder.

Sea $\sum_i^u \subset \sum_i$ el subconjunto del conjunto de salidas del proceso P_i , definido por:

$$\Sigma_i^u = \{\sigma \in |P_i|/u\sigma \in pre(L_{\mathcal{H}})\}.$$

Un lenguaje $L \subset L_{\mathcal{H}}$ es llamado “Parcialmente Bloqueado” si para cada $u \in pre(L)$, donde $\Sigma_j^u \neq \emptyset$, para algún $j \in \{1, \dots, n\}$, existe $i \in \{1, \dots, n\}$ tal que

- 1.) $u\sigma \in pre(L)$, para cada $\sigma \in \Sigma_i^u$
- 2.) Si $u \in pre(L)$ y $u \notin L$, entonces la proyección shuffle ($P_i(u)$) de u sobre el proceso P_i es tal que $P_i(u) \in pre|P_i|$ y $P_i(u) \notin |P_i|$.

Nosotros diremos que $(\mathcal{H}, S)$ es “Libre de Estancamiento” si $pre(L_c) = L_p$ y L_c es parcialmente bloqueado.

Proposición 5. Si $L \subset L_{\mathcal{H}}$ es parcialmente bloqueado, entonces $pre(L) \cap L_{\mathcal{H}} = L$.

Demostración. Sea $L \subset L_{\mathcal{H}}$ parcialmente bloqueado.

Es claro que $L \subset pre(L)$ y por hipótesis $L \subset L_{\mathcal{H}}$, de aquí $L \subset pre(L) \cap L_{\mathcal{H}}$.

De este modo sólo falta probar que $pre(L) \cap L_{\mathcal{H}} \subset L$.

Sea entonces $w \in pre(L) \cap L_{\mathcal{H}}$, como $L \subset pre(L)$ y es claro que si $w \in L \cap L_{\mathcal{H}}$ no hay nada que probar, supongamos entonces que $w \notin L$. Luego como L es parcialmente bloqueado existe $i \in \{1, \dots, n\}$ tal que la proyección shuffle ($P_i(w)$) de w en la componente P_i de $L_{\mathcal{H}}$ es tal que $P_i(w) \in pre(|P_i|)$ y $P_i(w) \notin |P_i|$.

Lo que es una contradicción con el hecho de que $w \in$

$pre(L) \cap L_{\mathcal{H}}$ y por ende $w \in L_{\mathcal{H}}$, con lo que, “ w debe completar una tarea en $\mathcal{H}$ ”; de aquí, la proyección $P_i(w)$ debe completar una tarea en el proceso P_i ; es decir, $P_i(w) \in |P_i|$. Así, $w \in L$. ■

Conclusión

El modelo $(S, \mathcal{H})$ verificando los argumentos establecidos en este manuscrito proporciona una estructura para estudiar un amplio rango de sistemas de eventos discretos, en los cuales los eventos controlables son conocidos. El resultado final es la construcción de un supervisor S que da solución al problema de control supervisorio centralizado.

Los dos ingredientes claves para aplicar estos resultados de síntesis son la disponibilidad del modelo $\mathcal{H}$ que

representa al *SED* de interés y la disponibilidad del lenguaje de requerimientos L_a . El modelo $\mathcal{H}$ del sistema se obtiene mediante la gramática Shuffle de los modelos individuales de las componentes del sistema, mientras que no hay recipiente mágico para la construcción de L_a . De hecho, en general, la construcción de L_a presenta mayor dificultad que la construcción de $\mathcal{H}$.

Referencias

- Branicky M, 1995, Studies in hibrid systems: Modeling, analysis and control. PhD thesis, Massachusetts inst technol, Cambridge, Dept. Elec. Eng. And computer Sci.
- Eilenberg S, 1974, Automata, Languages and Machines, Academic Press, New York.
- MacLane S, 1971, Categories for the Working Mathematician. Springer-Verlang, New York.
- Mata G, 2017, Supervisory control applications to solving optimal control problems for discrete event system. Revist. Ing. UC. Vol. 24. No1. pp 81-90.
- Mata G, Ruiz B, Camacho C, Méndez A, Muñoz S, Zambrano H, 2018, A planning algorithm in a class of discrete event system. DYNA. 85(206), 283-293.
- Ramadge PJ, Wonham WM, 1986, Modular Supervisory Control of Discrete Event Systems, Lecture Notes in Control and Inform., v.83, p.202-214
- Ramadge PJ, Wonham WM, 1984, Supervisory Control of a Class of Discrete Event Processes, Lecture Notes in Control and Inform., v.63, p.477-498.
- Ramadge PJ, Wonham WM, 1982, Supervisory Control of Discrete Event Processes, Lecture Notes in Control and Inform., v.39, pp.202-214.

Recibido: 23 de abril de 2021

Aceptado: 12 de julio de 2021

Quilumba Giraldo, Eduardo Alfonso: Ingeniero Zootecnista. Instituto de posgrado. Universidad Técnica de Manabí.

Solé Aguilera, Luz Estela: Licenciada en Matemáticas, Magister en Matemáticas, doctor en Matemáticas, Profesor Titular jubilada activa. Facultad de Ciencias (ULA). Departamento de Matemáticas. Línea de investigación: sistemas dinámicos de eventos discretos. Correo electrónico: luzsole@gmail.com