Universidad de los Andes
Facultad de Ciencias
Departamento de Matemáticas

Control Supervisorio Centralizado

Arturo Barros R.
Trabajo Especial de Grado
Para Optar al Título de
Licenciado en Matemáticas
Tutor: Prof. Guelvis Mata

Mérida - Venezuela
2005

Índice general

Introducción

Un proceso de eventos discretos es definido en términos algebraicos y su comportamiento es dado por un lenguaje formal apropiado [VI].
Un sistema complejo puede consistir en un conjunto de procesos asincrónicos, interactuantes que operan concurrentemente. Un problema de control para tales sistemas es la supervisión de varias componentes para asegurar su interacción armoniosa y un flujo ordenado de eventos resultantes. En este trabajo nuestra atención se centra en la supervisión de un conjunto finito de tales procesos cada uno de los cuales opera asincrónicamente [III]. Cada proceso se comunica con un supervisor central el cual puede ser considerado como una estación de referencia a partir de la cual la interacción entre todos los 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. En líneas generales, nuestro resultado principal es que cada supervisor que resuelve el problema del supervisor centralizado es un cociente de un lenguaje para el comportamiento coordinado resultante.

Organización

El capítulo 1, (preliminares) constituye los fundamentos para desarrollar los capítulos posteriores. Este incluye algunos conceptos de la teoría de lenguajes y una formalización matemática de algunas nociones de teoría de grafos, inmersas en la teoría de categorías [VIII]. En el capítulo 2 se incluye una formalización matemática de algunas nociones tales como procesos de eventos controlables, máquinas secuenciales, gramáticas regulares (también incluyendo a estas en la teoría de las categorías). En el capítulo 3 se introducen definiciones, propiedades y análisis de las estructuras citadas (con énfasis en los procesos bajo supervisión). Finalmente los teoremas del capítulo 4 demuestran que un supervisor para un sistema $\mathcal{H}$ es esencialmente un cociente de un lenguaje para el comportamiento resultante.

Capítulo 1

Preliminares

Incluiremos en este capítulo algunas nociones que nos permitirán el desarrollo de nuestro propósito más adelante.

Lenguajes

Dado $\Sigma = \{u_1, \ldots, u_n\}$ 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 $a = a_1 \ldots a_n$ y $b = b_1 \ldots b_m$ en Σ^*, $a \cdot b = a_1 \ldots a_n \cdot b_1 \ldots b_m$. En adelante denotaremos $a \cdot b$ por ab, además, diremos que:

i.) 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^*.

ii.) Todo subconjunto $L \subset \Sigma^*$ es llamado un lenguaje sobre Σ.

iii.) El conjunto formado por todos los prefijos de palabras de un lenguaje $L \subset \Sigma^*$ es llamado **clausura de** L y será denotado por $pre(L)$. De este modo
$pre(L) = \{x \in \Sigma^* \ / \ \exists\ y \in \Sigma^* \ , \ x \cdot y \in L \ \}$.

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

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

Todo $w \in \Sigma^*$ se puede escribir $w = w_1 \ldots w_n$ con $w_i \in \Sigma \cup \{\varepsilon\}$, $1 \leq i \leq n$.
Sean Σ y Ω dos alfabetos tales que $\Omega \subset \Sigma$; definimos $h : \Sigma \longrightarrow \Omega$ 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 ε-proyección de Σ^* en Ω^* por

$$H(w) = H(w_1 \dots w_n) = h(w_1) \cdot h(w_2) \dots h(w_n).$$

Esta proyección está bien definida y es un homomorfismo de monoides. En efecto, sean $w, v \in \Sigma^*$, luego $w = w_1 \dots w_n$ y $v = v_1 \dots v_m$. Por lo tanto,

$$\begin{aligned} H(wv) = H(w_1 \dots w_n v_1 \dots v_m) &= h(w_1) \dots h(w_n) h(v_1) \dots h(v_m) \\ &= H(w)H(v). \end{aligned}$$

Además $H(\varepsilon) = \varepsilon$.

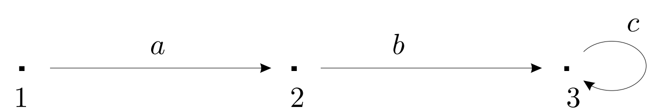
Grafos Dirigidos

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 $\delta_0 : E \longrightarrow N$ y $\delta_1 : E \longrightarrow N$ especificando los nodos iniciales y finales respectivamente para cada arco.

Ejemplo 1.1. Sea $G_1 = (N, E, \delta_0, \delta_1)$, donde $N = \{1, 2, 3\}$, $E = \{a, b, c\}$

$$\begin{array}{ccc} \delta_0(a) = 1 & & \delta_1(a) = 2 \\ \delta_0(b) = 2 & \text{y} & \delta_1(b) = 3 \\ \delta_0(c) = 3 & & \delta_1(c) = 3 \end{array}$$

Gráficamente podemos representar al grafo dirigido G_1 por



Sea $G = (N, E, \delta_0, \delta_1)$ un grafo dirigido. 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: $\delta_1(\alpha_1) = \delta_0(\alpha_2)$, $\delta_1(\alpha_2) = \delta_0(\alpha_3), \dots, \delta_1(\alpha_{n-1}) = \delta_0(\alpha_n)$. 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.

Observando el ejemplo anterior, $c_1 = ab$ y $c_2 = abc$ son dos ejemplos de caminos en el grafo dirigido G_1.

Morfismos entre Grafos Dirigidos

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

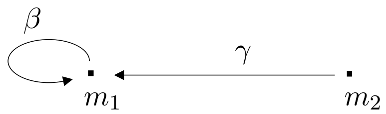
$$\begin{tikzcd} N \arrow[d, "F_n"'] & E \arrow[l, "\delta_0"'] \arrow[r, "\delta_1"] \arrow[d, "F_e"'] & N \arrow[d, "F_n"'] \\ M & P \arrow[l, "\widetilde{\delta_0}"'] \arrow[r, "\widetilde{\delta_1}"] & M \end{tikzcd}$$

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

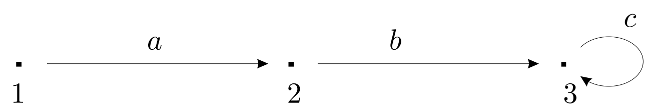
Ejemplo 1.2. Consideremos el grafo $H = (M, P, \delta'_0, \delta'_1)$, donde $M = \{m_1, m_2\}$, $P = \{\gamma, \beta\}$ y

$$\begin{array}{ll} \delta'_0(\gamma) = m_2 & \delta'_1(\gamma) = m_1 \\ \delta'_0(\beta) = m_1 & \delta'_1(\beta) = m_1 \end{array}$$

Este grafo tiene la siguiente representación gráfica



y el grafo G_1, del ejemplo anterior, tiene la representación gráfica



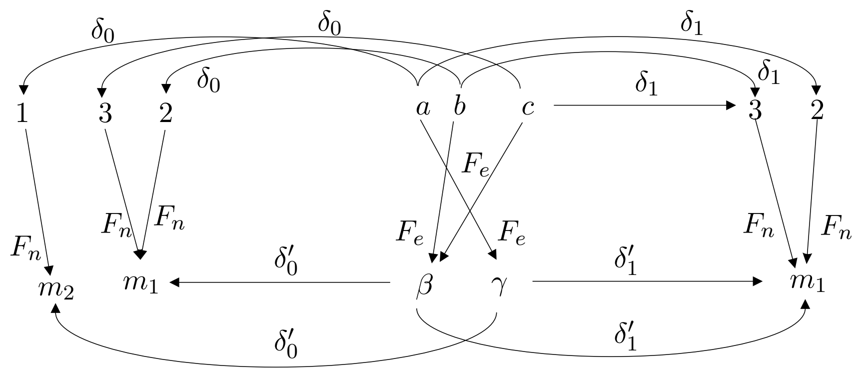
Consideremos ahora el siguiente par de funciones $F = (F_e, F_n)$ dadas por

$$\begin{array}{ll} F_e(a) = \gamma & F_n(1) = m_2 \\ F_e(b) = \beta & F_n(2) = m_1 \\ F_e(c) = \beta & F_n(3) = m_1 \end{array}$$

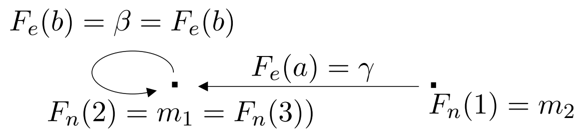
para verificar que F es un morfismo entre los grafos G_1 y H debemos mostrar que el diagrama siguiente conmuta

$$\begin{tikzcd} N \arrow[d, "F_n"'] & E \arrow[l, "\delta_0"'] \arrow[r, "\delta_1"] \arrow[d, "F_e"'] & N \arrow[d, "F_n"'] \\ M & P \arrow[l, "\delta'_0"'] \arrow[r, "\delta'_1"] & M \end{tikzcd}$$

El que este diagrama conmute quedara ilustrado en la siguiente figura



Luego, $F = (F_e, F_n)$ es un morfismo entre los grafos G_1 y H. Ilustraremos por medio de un grafo, en la siguiente figura, como el morfismo F asocia o identifica los nodos y arcos del grafo G_1 con los nodos y arcos del grafo H.



Grafos Etiquetados

Denotemos por 1 al conjunto formado solo por el elemento uno. Dado un conjunto Σ nosotros podemos construir el grafo especial $\widetilde{\Sigma}= (1, \Sigma, \delta'_0, \delta'_1)$, donde Σ es el conjunto de arcos y para cada $\sigma \in \Sigma$, $\quad \delta_0(\sigma) = \delta_1(\sigma) = 1$.

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 $\widetilde{\Sigma}$ tal que el siguiente diagrama conmuta.

$$\begin{tikzcd}
N \arrow[d, "F_n"'] & E \arrow[l, "\delta_0"'] \arrow[r, "\delta_1"] \arrow[d, "F_e"'] & N \arrow[d, "F_n"'] \\
1 & \Sigma \arrow[l, "\delta'_0"'] \arrow[r, "\delta'_1"] & 1
\end{tikzcd}$$

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

Ejemplo 1.3. Considere el grafo G_1 citado en el ejemplo anterior; el cual representamos gráficamente mediante la figura

$$\begin{tikzcd} 1 \arrow[r, "a"] & 2 \arrow[r, "b"] & 3 \arrow[loop right, "c"] \end{tikzcd}$$

Considere el grafo etiquetado $G_{1\Sigma} = (G_1, g, \Sigma,)$, donde $\Sigma = \{\alpha, \beta\}$ y $g = (g_e, g_n)$ es dada por

$$\begin{array}{ll} g_e(a) = \alpha & g_n(1) = 1 \\ g_e(b) = \beta & g_n(2) = 1 \\ g_e(c) = \beta & g_n(3) = 1 \end{array}$$

Ilustramos gráficamente a $G_{1\Sigma}$ en la siguiente figura

$$\begin{tikzcd} 1 \arrow[r, "\alpha"] & 2 \arrow[r, "\beta"] & 3 \arrow[loop right, "\beta"] \end{tikzcd}$$

Aquí interpretamos que el morfismo g marca los arcos de G_1 con los elementos de Σ.

Sea $G_\Sigma = (G, g, \Sigma)$ un grafo etiquetado. Una **Traza** c en G_Σ es una sucesión finita $\sigma_1\sigma_2 \ldots \sigma_n$ de elementos de Σ tales que $g_e(\alpha_1) = \sigma_1$, $g_e(\alpha_2) = \sigma_2, \ldots, g_e(\alpha_n) = \sigma_n$, para algún camino $\alpha_1\alpha_2 \ldots \alpha_n$ en G. Aquí los nodos $q_0 = \delta_0(\alpha_1)$ y $q_n = \delta_1(\alpha_n)$ 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.

Ejemplo 1.4. Son ejemplos de trazas en el grafo etiquetado $G_{1\Sigma}$ del ejemplo anterior:

α, con longitud uno, el nodo 1 como comienzo y el nodo 2 como final de la traza.
$\alpha\beta\beta$, con longitud tres, el nodo 1 como comienzo y el nodo 3 como final de la traza.
$\beta\beta\beta\beta$, con longitud cuatro y el nodo 3 como comienzo y final de la traza.

Morfismos entre Grafos Etiquetados

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

$$\begin{tikzcd} G \arrow[r, "g"] \arrow[d, "F_1"'] & \widetilde{\Sigma} \arrow[d, "F_2"] \\ H \arrow[r, "h"] & \widetilde{\Omega} \end{tikzcd}$$

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

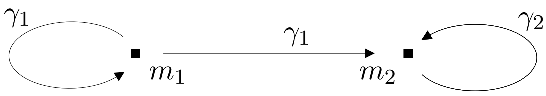
Ejemplo 1.5. Considere el grafo $H = (M, E, \delta'_0, \delta'_1)$, donde $M = (m_1, m_2)$, $E = \{a', b', c'\}$ y

$$\begin{array}{ll} \delta'_0(a') = m_1 & \delta'_1(a') = m_1 \\ \delta'_0(b') = m_1 & \delta'_1(b') = m_2 \\ \delta'_0(c') = m_2 & \delta'_1(c') = m_2 \end{array}$$

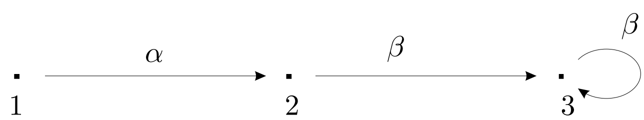
luego considere el grafo etiquetado $H_\Omega = (H, h, \widetilde{\Omega})$, donde $\Omega = \{\gamma_1, \gamma_2\}$ y $h(h_e, h_n)$ está definida como sigue

$$\begin{array}{ll} h_e(a') = \gamma_1 & h_n(m_1) = 1 \\ h_e(b') = \gamma_1 & h_n(m_2) = 1 \\ h_e(c') = \gamma_2 & \end{array}$$

Ilustramos gráficamente a H_Ω por medio de la siguiente figura



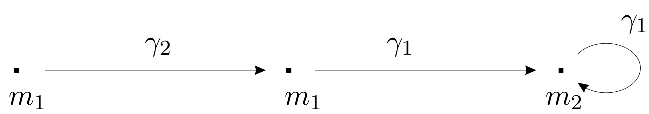
Ahora definiremos un morfismo f entre los grafos etiquetados $G_{1\Sigma}$ del ejemplo anterior y H_Ω, este constituirá nuestro ejemplo: recordemos que $G_{1\Sigma}$ es ilustrado gráficamente por la siguiente figura



Sea ahora $f = (f_1, f_2)$ definido por

$$\begin{array}{llll} f_{1n}(1) = m_1 & f_{1e}(a) = c' & f_{2e}(\alpha) = \gamma_2 & f_{2n}(1) = 1 \\ f_{1n}(2) = m_1 & f_{1e}(b) = b' & f_{2e}(\beta) = \gamma_1 & \\ f_{1n}(3) = m_2 & f_{1e}(c) = a' & & \end{array}$$

Ilustramos mediante la siguiente figura como actúa f sobre los arcos y nodos de $G_{1\Sigma}$.



Ahora consideremos al grafo $G_\Sigma = (G, g, \Sigma)$, donde $G = (N, E, \delta_0, \delta_1)$, $g = (g_n, g_e)$, $\widetilde{\Sigma} = (1, \Sigma, \delta'_0, \delta'_1)$ y consideremos el siguiente diagrama conmutativo.

$$
\begin{tikzcd}
 & 1 & \\
N \arrow[ur, "g_n"] & E \arrow[l, "\delta_0"'] \arrow[r, "g_e"] \arrow[d, "K"] & \Sigma \arrow[ul, "\delta_1"'] \\
 & \Sigma \times N \arrow[ul, "\Pi_2"] \arrow[ur, "\Pi_1"'] &
\end{tikzcd}
$$

Aquí, Π_1 y Π_2 son las proyecciones canónicas y $K : E \longrightarrow \Sigma \times N$ está dada por

$$K(e) := (g_e(e), \delta_0(e)), \text{ para todo } e \in E.$$

Esta función está definida de forma única. En efecto, para todo $e \in E$, supongamos que F es una función tal que $e \overset{F}{\mapsto} (F_1(e), F_2(e)) = (\alpha, n)$ para algún $(\alpha, n) \in \Sigma \times N$. Como el diagrama conmuta tenemos

$$
\begin{aligned}
\delta_0(e) = \Pi_2(K(e)) &= \Pi_2(F_1(e), F_2(e)) \\
&= F_2(e) \\
g_e(e) = \Pi_1(K(e)) &= \Pi_1(F_1(e), F_2(e)) \\
&= F_1(e).
\end{aligned}
$$

Por lo tanto, K es la única función tal que el diagrama conmuta.
Luego, diremos que G_Σ es un grafo etiquetado determinístico si K es inyectiva.

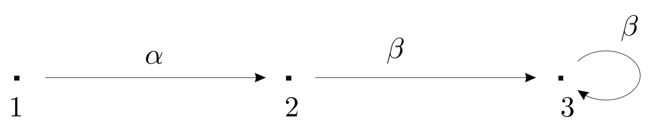
Ejemplo 1.6. Examinemos al grafo $G_{1\Sigma} = (G_1, g, \Sigma)$ del ejemplo anterior; donde $N = \{1, 2, 3\}$ y $E = \{a, b, c\}$ son los nodos y arcos respectivamente de G_1 y donde $\Sigma = \{\alpha, \beta\}$ y $g = (g_e, g_n)$ es dada por

$$
\begin{array}{ll}
g_e(a) = \alpha & g_n(1) = 1 \\
g_e(b) = \beta & g_n(2) = 1 \\
g_e(c) = \beta & g_n(3) = 1
\end{array}
$$

para este grafo la función $K : E \longrightarrow \Sigma \times N$ está dada por

$$
\begin{aligned}
K(a) &= (\alpha, 1) \\
K(b) &= (\beta, 2) \\
K(c) &= (\beta, 3)
\end{aligned}
$$

Esta función es inyectiva y por lo tanto el grafo $G_{1\Sigma}$ que ilustramos en la siguiente figura

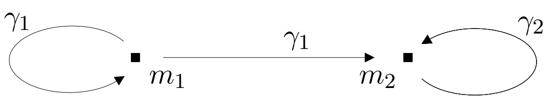


es determinístico.

Examinemos ahora al grafo $H_\Omega = (H, h, \widetilde{\Omega})$ del ejemplo anterior, donde $M = (m_1, m_2)$, $E = \{a', b', c'\}$ son los nodos y arcos de H, $\Omega = \{\gamma_1, \gamma_2\}$ y h es el morfismo entre H y $\widetilde{\Omega}$. La función $K : E \longrightarrow N \times \Omega$ para este grafo está dada por

$$
\begin{aligned}
K(a') &= (\gamma_1, m_1)\\
K(b') &= (\gamma_1, m_1)\\
K(c') &= (\gamma_2, m_2)
\end{aligned}
$$

Esta función no es inyectiva por lo tanto el grafo H_Ω, ilustrado en la siguiente figura



no es determinístico.

Capítulo 2

Procesos Secuenciales

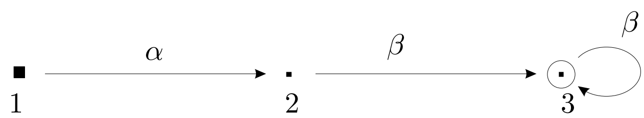
Procesos Secuenciales de Estado Finito

Un proceso secuencial (PS) P es un triple $P = (G_\Sigma, S, T)$, donde G_Σ es un grafo etiquetado, S es un subconjunto del conjunto de nodos de G_Σ que especifica nodos iniciales de P y T es un subconjunto del conjunto de nodos de G_Σ que especifican nodos finales de P.
En lo que sigue, los nodos de G_Σ serán llamados estados; luego, P sera llamado un proceso secuencial de estado finito si el grafo G_Σ es finito.

Llamaremos a una sucesión finita $c = \sigma_1\sigma_2\ldots\sigma_n$ de elementos de Σ una **traza** en el proceso $P = (G_\Sigma, S, T)$, si c es una traza en el correspondiente grafo etiquetado G_Σ 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_Σ. 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 $|P|$

Ejemplo 2.1. Sea $P = (G_{1\Sigma}, S, T)$ donde $G_{1\Sigma}$ es el grafo etiquetado del ejemplo anterior, $S = \{1\}$ y $T = \{3\}$; ilustramos gráficamente a P en la siguiente figura



señalamos en la figura a los estados de S con un punto oscuro y a los estados de T con un círculo. Cualquier camino en P puede representarse por una palabra en Σ^*, como por ejemplo $\alpha\beta, \alpha\beta\beta, \alpha\beta\beta\beta\beta$ entre otras, luego el comportamiento de P queda representado por $|P| = \alpha\beta\beta^*$.
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.

Ejemplo 2.2. El proceso definido en el ejemplo anterior (2.1) es accesible y coaccesible, por lo tanto limpio.

Morfismos entre Procesos Secuenciales

Sean $S = (G_\Sigma, S, T)$ y $P = (H_\Omega, U, V)$ procesos secuenciales.
Un morfismo $F : S \longrightarrow P$ de procesos secuenciales consiste en:

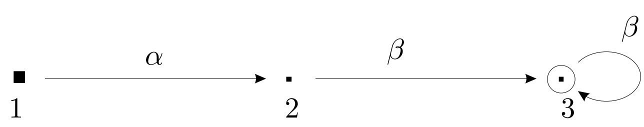
1. Un morfismo de grafos etiquetados $F = (F_1, F_2)$ de G_Σ en H_Ω
2. Un par de funciones $a : S \longrightarrow U$, $b : T \longrightarrow V$ dadas de manera que el siguiente diagrama conmute.

$$\begin{tikzcd} S \arrow[r, "\Pi_1"] \arrow[d, "a"'] & N \arrow[d, "F_n"'] & T \arrow[l, "\Pi_2"'] \arrow[d, "b"] \\ U \arrow[r, "\Pi_3"] & M & V \arrow[l, "\Pi_4"'] \end{tikzcd}$$

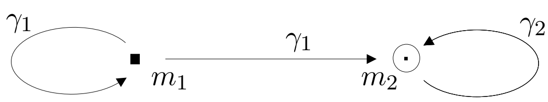
Donde los Π_i son proyecciones canónicas, para $i = 1, 2, 3, 4$

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

Ejemplo 2.3. Consideremos al proceso secuencial $S = (G_{1\Sigma}, s, T)$, donde $G_{1\Sigma}$ es el grafo etiquetado del ejemplo 1.3, $s = \{1\}$ y $T = \{3\}$, ilustrado en la figura siguiente:



y al proceso secuencial $P = (H_\Omega, U, V)$, donde H_Ω es el grafo etiquetado dado en el ejemplo 1.5, $U = \{m_1\}$ y $V = \{m_2\}$, ilustrado en la figura siguiente



El morfismo $\mathbf{F}$ entre los procesos secuenciales S y P que constituirá nuestro ejemplo constara del morfismo $F = (F_1, F_2)$, entre los grafos $G_{1\Sigma}$ y H_Ω dado en el ejemplo 1.5, junto al par de funciones

$$\begin{array}{ccc} a : s \to U & y & b : T \to V \\ a(1) = m_1 & & b(3) = m_2 \end{array}$$

dadas de manera tal que el siguiente diagrama conmute

$$\begin{tikzcd} S \arrow[r, "\Pi_1"] \arrow[d, "a"'] & N \arrow[d, "F_n"'] & T \arrow[l, "\Pi_2"'] \arrow[d, "b"'] \\ U \arrow[r, "\Pi_3"] & M & V \arrow[l, "\Pi_4"'] \end{tikzcd}$$

Y en efecto, que este diagrama conmute queda ilustrado en la siguiente figura

$$\begin{tikzcd} 1 \arrow[r, "\Pi_1"] \arrow[d, "a"] & 1 \arrow[d, "F_{1n}"'] & 2 \arrow[dl, "F_{1n}"] & 3 \arrow[d, "F_{1n}"] & 3 \arrow[l, "\Pi_2"'] \arrow[d, "b"] \\ m_1 \arrow[r, "\Pi_3"] & m_1 & & m_2 & m_2 \arrow[l, "\Pi_4"'] \end{tikzcd}$$

de este modo $\mathbf{F} = (F, a, b)$, es un morfismo entre los procesos secuenciales S y P.

Máquinas Secuenciales Determinísticas

Una Máquina Secuencial Determinística (MS) es un quíntuple $A = (\Sigma, Q, q_0, T, \delta)$, 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 $\delta : \Sigma \bigcup \{\epsilon\} \times Q \longrightarrow Q$ es una función parcial llamada función de transición de estados de A, que está definida como sigue:

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

Extendemos esta función parcial δ de acuerdo a

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

Note que δ^* esta 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.

Ejemplo 2.4. Sea $A = (\Sigma, Q, q_0, T, \delta)$, donde $\Sigma = \{\alpha, \beta\}$, $Q = \{1, 2, 3\}$, $q_0 = \{1\}$, $T = \{3\}$, y la función parcial δ dada por

$$\delta(\alpha, 1) = 2$$
$$\delta(\beta, 2) = 3$$
$$\delta(\beta, 3) = 3$$

.

El comportamiento de una (MS) A, denotado por $|A|$, es el conjunto formado por todas las palabras $w \in \Sigma^*$ tal que $\delta(w, q_0) \in T$.
En el ejemplo (2.4) el comportamiento de A, es dado por $|A| = \alpha\beta\beta^*$.

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

Ejemplo 2.5. Consideremos dos (MS), $A = (\Sigma, Q, q_0, T, \delta)$, dada en el ejemplo anterior y $B = (\Omega, X, x_0, U, \beta)$, donde $\Omega = \{\gamma_1, \gamma_2, \gamma_3\}$, $X = \{m_1, m_2\}$, $x_0 = \{m_1\}$, $U = \{m_2\}$ y la función parcial β dada por

$$\beta(\gamma_1, m_1) = m_2$$
$$\beta(\gamma_2, m_2) = m_2$$
$$\beta(\gamma_3, m_1) = m_1$$

Sea $\mathbf{h} = (h_1, h_2)$, donde h_1 y h_2 son funciones dadas por

$$\begin{array}{lcl} h_1 : \Sigma \to \Omega & y & h_2 : Q \to X \\ h_1(\alpha) = \gamma_2 & & h_2(1) = m_1 \\ h_1(\beta) = \gamma_2 & & h_2(2) = m_2 \\ & & h_2(3) = m_2 \end{array}$$

$\mathbf{h}$ así definida constituirá nuestro ejemplo. En efecto, $h_2(1) = m_1$, $h_2(3) = m_2$ y además
$h_2[\delta(\Sigma \times Q)] = \{h_2[\delta(\alpha, 1)], h_2[\delta(\alpha, 2)], h_2[\delta(\alpha, 3)], h_2[\delta(\beta, 1)], h_2[\delta(\beta, 2)], h_2[\delta(\beta, 3)]\}$
$h_2[\delta(\Sigma \times Q)] = \{h_2(2), h_2(3)\} = \{m_2\}$
y
$\beta[h_1(\Sigma) \times h_2(Q)] = \{\beta[\gamma_2 \times m_1], \beta[\gamma_2 \times m_2]\} = \{m_2\}$

De manera que es claro que $h_2[\delta(\Sigma \times Q)] \subset \beta[h_1(\Sigma) \times h_2(Q)]$. Así $\mathbf{h}$ es un morfismo entre las (MS) A y B.

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

Máquinas Secuenciales Generalizadas

Una Máquina Secuencial Generalizada (MSG) es una Máquina secuencial Determinística $A = (Q, \Sigma, q_0, T, \delta)$ junto con una función $\tau : \Sigma \bigcup \{\epsilon\} \times Q \longrightarrow \Gamma^*$, donde Γ es un alfabeto, dada por:

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

Esta función será llamada función de salida.

Ejemplo 2.6. Considere la (MS) $A = (\Sigma, Q, q_0, T, \delta)$, donde $\Sigma = \{\alpha, \beta\}$, $Q = \{1, 2, 3\}$, $q_0 = \{1\}$, $T = \{3\}$ y la función parcial δ dada por

$$\delta(\alpha, 1) = 2$$
$$\delta(\beta, 2) = 3$$
$$\delta(\beta, 3) = 3$$

Sea $\Omega = \{\omega_1, \omega_2\}$ un alfabeto y considere a la función $\tau : \Sigma \bigcup \{\epsilon\} \times Q \longrightarrow \Omega^*$, dada por:

$$\tau(\alpha, 1) = \omega_1$$
$$\tau(\beta, 2) = \omega_2$$
$$\tau(\beta, 3) = \omega_2$$

Finalmente, $\mathbf{A} = (\Sigma, Q, \Omega, q_0, T, \delta, \tau)$ es una (MSG).

Morfismos de Máquinas Secuenciales Generalizadas

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

Ejemplo 2.7. Consideraremos dos (MSG), $\mathbf{A} = (\Sigma, Q, P, q_0, T, \delta, \tau)$ definida por la (MS) A dada en el ejemplo 2.5, junto a la función de salida $\tau : \Sigma \times Q \longrightarrow P^*$,
donde $P = \{p_1, p_2\}$ es un alfabeto y τ es dada por:

$$\tau(\alpha, 1) = p_1$$
$$\tau(\beta, 2) = p_2$$
$$\tau(\beta, 3) = p_2$$

y $\mathbf{B} = (\Omega, X, \phi, x_0, U, \beta, V)$, definida por la (MS) B dada en el ejemplo 2.5, junto a la función de salida
$V : \Omega \times X \longrightarrow \phi^*$, donde $\phi = \{\phi_1, \phi_2, \phi_3\}$ es un alfabeto y V es dada por:

$$V(\gamma_1, m_1) = \phi_1$$
$$V(\gamma_2, m_2) = \phi_2$$
$$V(\gamma_3, m_1) = \phi_2$$

Como las (MSG) **A** y **B** están definidas a partir de las (MS) A y B del ejemplo (2.5), junto a sus correspondientes funciones de salida τ y V. Sea $\mathbf{h} = (h_1, h_2, h_3)$, donde h_1 y h_2 son las funciones que definen el morfismo entre las (MS) A y B del ejemplo (2.5), entonces solo debemos mostrar que $h_3 : P \longrightarrow \phi$ dada por:

$$h_3(p_1) = \phi_1$$
$$h_3(p_2) = \phi_2$$

verifica $h_3\tau \subset V(h_1 \times h_2)$ y en efecto

$V[h_1(\Sigma)\times h_2(Q)] = \{V[(h_1(\alpha), h_2(1))], V[(h_1(\alpha), h_2(2))], V[(h_1(\alpha), h_2(3))], V[(h_1(\beta), h_2(1))], V[(h_1(\beta), h_2(2))], V[(h_1(\beta), h_2(3))]\} = \{V(\gamma_2, m_1), V(\gamma_2, m_2)\} = \{\phi_2\}$

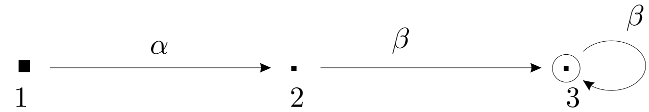
$h_3[\tau(\Sigma \times Q)] = \{h_3(p_1), h_3(p_2)\} = \{\phi_2\}$. Asi $\mathbf{h} = (h_1, h_2, h_3)$ constituye nuestro ejemplo de morfismo entre (MSG).

De aquí resaltamos la siguiente 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).

Gramática Regular Determinística

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.

Ejemplo 2.8. El (PS) P definido en el ejemplo (2.1), ilustrado en la siguiente figura:



es de estado finito, es limpio (como se hizo la observación en el ejemplo 2.2), tiene un solo estado inicial y con su grafo etiquetado determinístico como queda ilustrado en el ejemplo 1.6, asi este constituye nuestro ejemplo de (Grm).

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.

Operaciones con gramáticas regulares determinísticas

Sea $G = (G_\Sigma, h_0, T)$ y $H = (H_\Sigma, m_0, T_2)$ dos (Grm) sobre el alfabeto Σ.
Considere ahora $G \cap H$ y el diagrama conmutativo siguiente

$$\begin{tikzcd} K \arrow[r, "b"] \arrow[d, "a"'] & H \arrow[d, "h"] \\ G \arrow[r, "g"] & \widetilde{\Sigma} \end{tikzcd}$$

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 $\widetilde{\Sigma}$.
De este modo definimos a $G \cap H$ como la componente limpia del proceso secuencial $(K_\Sigma, (n_0, m_0), T \times T_2)$. 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 comportamiento de G y H.

Proyección Shuffle

Dados Ω y Σ dos alfabetos disjuntos; es decir, $\Omega \cap \Sigma = \emptyset$. Definimos la proyección "Shuffle" $P : (\Sigma \cup \Omega)^* \longrightarrow \Sigma^* \times \Omega^*$ 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 $\Pi_1 : \Sigma \times \Omega \longrightarrow \Sigma$ y $\Pi_2 : \Sigma \times \Omega \longrightarrow \Omega$, proyecciones canónicas.
Para cada $\gamma \in (\Sigma \cup \omega)^*$ se tiene que $\gamma = \sigma_1 \ldots \sigma_n$, donde $\sigma_i \in \Sigma \cup \Omega$, $1 \leq i \leq n$ luego:

$$P(\gamma) := P(\sigma_1 \ldots \sigma_n) = \big(\Pi_1(P_1(\sigma_1)) \ldots \Pi_1(P_1(\sigma_1)), \Pi_2(P_1(\sigma_1)) \ldots \Pi_2(P_1(\sigma_1))\big)$$

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.

Producto Shuffle

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

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

Gramática Producto Shuffle

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 ésto, el producto shuffle de dos lenguajes regulares es un lenguaje regular.

Capítulo 3

Procesos de Eventos Discretos Controlables

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

1. Un PS $P = (G_{\Sigma \cup \{\epsilon\}}, S, T)$;
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 $f_e : E \longrightarrow \Gamma_e$ y $f_\delta : E \longrightarrow \Gamma_\delta$ 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 $g : E \longrightarrow \Sigma \cup \{\epsilon\}$ la función etiqueta de P.
Sea $i : \Sigma \longrightarrow \Sigma \cup \{\epsilon\}$ la función inclusión de P. Así, $g' : E \longrightarrow \Sigma$ 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 $h_e : \Sigma \longrightarrow \Gamma_e$ y $h_\delta : \Sigma \longrightarrow \Gamma$ tales que $f_e = h_e(g')$ y $f_\delta = h_\delta(g')$.
En este trabajo asumiremos también 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.

Supervisión de sistemas de eventos discretos

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

$$\mathcal{H} = \{P_1, \ldots, P_n \quad / \quad \text{cada } P_i \text{ es un (PEDC)}\}$$

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

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

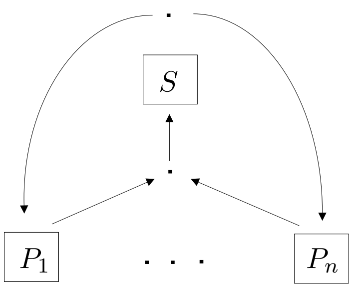
del comportamiento de los procesos constituyentes. Nosotros interpretaremos a $\mathcal{H}$ como un conjunto de procesos asincrónicos independientes y que los procesos tienen efectos interactuantes sobre su medio compartido. Esta interacción da comienzo a la necesidad de supervisión del comportamiento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

El Supervisor Secuencial

En el resto del trabajo sea $\mathcal{H} = \{P_1 \ldots P_n \quad / \quad \text{Cada } P_i \text{ es un (PEDC)}\}$ 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, $i = 1, \ldots, n$.
Además $\Sigma = \bigcup_{i=1}^{n} \Sigma_i$, $\Gamma = \bigcup_{i=1}^{n} \Gamma_i$ y $\Lambda = \bigcup_{i=1}^{n} \Lambda_i$.

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. En este trabajo 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 $< \sigma >$ denotando los eventos habilitados y $< \overline{\sigma} >$ denotando los eventos inhabilitados por σ.
Sea $\Gamma_\sigma = \{< \sigma >, < \overline{\sigma} >\}$ y $P_\sigma : \Gamma^* \longrightarrow \Gamma_\sigma^*$ la ϵ-proyección de Γ^* sobre Γ_σ^* dada por

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

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

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

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

i.) $\Phi(\epsilon, \sigma) := \alpha(\sigma)$

ii.) para $u \in |S|, \Phi(u, \sigma) = 1$ si $\alpha(\sigma) = 1$ y $P_\sigma(\tau(u)) = \epsilon$ o si $lP_\sigma(\tau(u)) =< \sigma >$

iii.) en otro caso $\Phi(u, \sigma) = 0$.

Si $\Phi(u, \sigma) = 1$ o $\Phi(u, \sigma) = 0$ entonces decimos 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 $L_{\mathcal{H}}$.

Proposición 3.1. *El comportamiento controlado de $(\mathcal{H}, S)$ es un lenguaje regular.*

Demostración. 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 $P_1 = (G_{\Sigma \cup \{\epsilon\}}, s_0, T)$ 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 $L_c = L_p \cap L_{\mathcal{H}}$ sería 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 $M_{1c} = (Q, \Sigma, T, q_0, \delta_c)$, donde $\delta_c : \Sigma \times Q \longrightarrow Q$

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

Aquí hemos supuesto que $\delta(w, q_0) = q$ y $w = w_1 \dots w_n$ con $w_i \in \Sigma$ para cada i.
Considere ahora $A_c = (Q \times Q_2, \Sigma, T_\times T_2, (q_0, q_2), \delta_2 \times \delta_c)$, donde $\delta_2 \times \delta_c : \Sigma \times Q_2 \times Q \longrightarrow Q_2 \times Q$ 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 $\delta_c(\sigma, q)$ estén definidas; además, como para cada $wu\sigma \in pre(|A_c|)$ se tiene que $\Phi(u, \sigma) = 1$, se concluye finalmente 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 es habilitada desde de 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 $\delta(u, x_o) = x$. Abreviaremos funcionalmente limpio por **F-Limpio**.

Asumiremos en adelante que todo supervisor es F-completo y F-Limpio. Dada esta elección, S no tendrá 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 decimos 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 $\Pi_\sigma : X \longrightarrow \{0, 1\}$ 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 $\Pi = \wedge\{\Pi_\sigma \ / \ \sigma \in \Lambda\}$.
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 $L_p(S) = L_p(R)$.

Lema 3.1. *Para cada supervisor S de $\mathcal{H}$ existe un supervisor S^* y un epimorfismo $F : S^* \longrightarrow S$ 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 $\bigwedge' \subseteq \bigwedge$ el subconjunto del conjunto de salidas controlables de $\mathcal{H}$, para el cual existen estados de S los cuales no son consistentes. Sea $\sigma \in \bigwedge'$ y $X_\sigma = \{0, 1\}$. Definamos $\mathcal{X}_\sigma = (\Gamma, X_\sigma, x_0, \delta_\sigma)$, donde $x_0 = \Phi(\sigma)$, $\delta_\sigma(\gamma, x) = 0$ si $\gamma =< \overline{\sigma} >$ y $\delta_\sigma(\gamma, x) = 1$ si $\gamma =< \sigma >$, en otro caso $\delta_\sigma(\gamma, x) = x$. Luego $\mathcal{X}_\sigma$ es una (MS) que especifica el estatus de σ. Sea m la cardinalidad de $\bigwedge'$ y sea $Y = \{0, 1\}^m$. Definamos la (MS) $\mathcal{Y} = (\Gamma, Y, y_0, \alpha)$, donde $y_0 = (x_0^1, \ldots, x_0^m)$ y $\alpha(\gamma, (x^1, \ldots, x^m)) = ((x^1)\gamma, \ldots, (x^m)\gamma)$ [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, $\mathcal{S}_e = (\Sigma, X \times Y, (x_0, y_o), \delta_e, \tau_e)$, donde $\delta_e(\sigma, x, y) := (\delta(\sigma, x), \alpha^*(\tau(\sigma, x), y))$ si $\delta(\sigma, x)$ esta 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 $\mathcal{S}_e$ hace un control equivalente al de S y todo estado de $\mathcal{S}_e$ es $\sigma - consistente$ para cada $\sigma \in \bigwedge$ y S es un cosiente de S_e en (Gsm). Finalmente consideramos $S^* = S$. ■

Lema 3.2. *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 $w \in L_p \cap L_{\mathcal{H}} = L_c$.
Sí $w \in L_c$ 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 $L_c = |S| \cap L_{\mathcal{H}}$.

■

Inhabilitación y Paro de Eventos

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.

$D_\sigma = \{x \in X \ / \ \delta(\sigma, x)$ está definido$\}$
$E_\sigma = \{x \in X \ / \ x$ habilita a $\sigma\}$
$\overline{E_\sigma} = \{x \in X \ / \ x$ inhabilita a $\sigma\}$
$\overline{D_\sigma} = \{x \in X \ / \ x$ inhabilita a $\sigma, \exists u \in pre(L_c)$ donde $\delta(u, x_0) = x \quad y \quad u\sigma \in pre(L_{\mathcal{H}})\}$

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

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

Proposición 3.2. *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
$\exists\sigma \in \Sigma$ / $(\overline{D_\sigma})\Sigma^{-1} \not\subseteq \overline{E_\sigma}$, 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:
$A_\sigma := \{q \in Q \ / \ \delta(\sigma, q) \text{ está definido}\}$
$\overline{A_\sigma} := \{q \in Q \ / \ \delta(\sigma, q) \text{ no está definido y } \exists u \in pre(L), \text{ donde } \delta(u, q_0) = q \text{ y } u\sigma \in pre(L_{\mathcal{H}})\}$
$\overline{\overline{A_\sigma}} := \{q \in Q \ / \ \delta(\sigma, q) \text{ no está definido y } \forall u \in pre(L), \text{ donde } \delta(u, q_0) = q \text{ y } u\sigma \notin pre(L_{\mathcal{H}})\}$.

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

Proposición 3.3. *$L \subset L_{\mathcal{H}}$ 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 aboquemonos a la demostración de la proposición.
(si)
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 $pre(L)$, $uw\sigma \in pre(L_{\mathcal{H}})$
y $uw\sigma \notin pre(L)$. Como $uw\sigma \in pre(L_{\mathcal{H}})$, 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 $uw \in pre(L)$, $uw\sigma \notin pre(L)$ 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 $uw \in pre(L)$ 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.
(sólo si)
Supongamos que existe $\sigma \in \Sigma$ tal que $(\overline{A_\sigma})\Sigma^{-1} \nsubseteq \overline{A_\sigma} \cup \overline{\overline{A_\sigma}}$ y probemos que $L \subset L_{\mathcal{H}}$ no es parcialmente invariante. Sea entonces $\sigma \in \Sigma$ tal que $(\overline{A_\sigma})\Sigma^{-1} \nsubseteq \overline{A_\sigma} \cup \overline{\overline{A_\sigma}}$ 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 3.4. *$L \subset L_{\mathcal{H}}$ es parcialmente invariante si, y sólo si, para cada Σ_i, $i = 1, \ldots, n$ y para cada $\sigma \in \Sigma_i$ se verifica:*

1.) $(A_\sigma)(\Sigma - \Sigma_i) \subset A_\sigma$; es decir, si $q \in A_\sigma$ entonces para cada $w \in \Sigma - \Sigma_i$, $\delta(w, q) \in A_\sigma$ siempre que $\delta(w, q)$ este definido; y

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

Demostración. (Si)
Sean $\sigma \in \Sigma_i$ 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 $w \notin \Sigma_i$ entonces $w \in \Sigma - \Sigma_i$, luego como $u\sigma \in pre(L)$ entonces existe $q \in Q$ tal que $\delta(w,q)$ esta definido en A y por lo tanto $q \in A_\sigma$, luego $\delta(w,q)$ esta definido pues $uw \in pre(L)$ y por lo tanto $\delta(w,q) \in A_\sigma$. Así, $uw\sigma \in pre(L)$. Si $w \in \Sigma_i$, como $uw \in pre(L)$ entonces existe $q \in Q$ tal que $\delta(\sigma,q)$ esta definida en A y entonces $q \in A_\sigma$; además, como $uw \in pre(L)$ entonces $\delta(w,q)$ esta definida en A; luego, $q \in (A_\sigma)\Sigma_i$ 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 $\delta(\sigma,\delta(w,q))$ esta indefinida en A, entonces existiría $uw \in pre(L)$ donde $\delta(uw,q_0) = \delta(w,q)$, por lo tanto $u\sigma \notin preL_{\mathcal{H}}$ 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.

(solo si)

Supongamos que L es parcialmente invariante y sean $\sigma \in \Sigma_i$ y $\delta(u,q_0) = q \in A_\sigma$ para algún prefijo $u \in pre(L)$. Suponga que $\delta(w,q)$ esta definido con $w \notin \Sigma_i$, 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 tanto, $\delta(w,q) \in A_\sigma$. Si $w \notin \Sigma_i$ entonces $w \in \Sigma - \Sigma_i$, luego $\delta(w,q) \in A$. Suponga que $\delta(w,q)$ esta definido con $w \in \Sigma_i$, 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 $\delta(w,q) \in A_\sigma \cup \overline{\overline{A_\sigma}}$. ■

Proposición 3.5. *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 el teorema 3.2, $L_c = L_{\mathcal{H}} \cap |S|$, 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 numero 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 $x = \delta(u,x_o)$ en S desde el cual, $\delta(\sigma,x)$ y $\delta(w,x)$ están definidos. Luego como $\sigma \in \Sigma$, $w \in \Sigma$,$uw\sigma$ es un prefijo en $L_{\mathcal{H}}$ y uw es un 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 como $u\sigma$ 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 $\Sigma_i^u \subset \Sigma_i$ 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 $j \in \{1, \ldots, n\}$, existe $i \in \{1, \ldots, n\}$ 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 3.6. *Si $L \subset L_{\mathcal{H}}$ es parcialmente bloqueado, entonces $pre(L) \bigcap L_{\mathcal{H}} = L$*

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 $L \subset pre(L) \bigcap L_{\mathcal{H}}$.
De este modo solo falta probar que $pre(L) \bigcap L_{\mathcal{H}} \subset L$. Sea entonces $w \in pre(L) \bigcap L_{\mathcal{H}}$, como $L \subset pre(L)$ y es claro que si $w \in L \bigcap L_{\mathcal{H}}$ no hay nada que probar, supongamos entonces que $w \notin L$. Luego como L es parcialmente bloqueado existe $i \in \{1, \ldots, n\}$ 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 $w \in pre(L) \bigcap L_{\mathcal{H}}$ 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, $w \in L$. ■

Coordinación

Sea $\mathcal{H}$ un SED tal que su comportamiento $L_{\mathcal{H}} \subset \Sigma^*$.
Una tarea será coordinada para $\mathcal{H}$ por medio de un comportamiento admisible $L_a \subset L_{\mathcal{H}}$. En este trabajo no discutiremos como tal comportamiento admisible es determinado, siendo este un problema, el cual depende de la forma específica de la labor a ser coordinada.

Sea G_i una (Grm) tal que $|G_i| = |P_i|$, $i \in N_n^+ = \{1, \ldots, n\}$, entonces $G_{\mathcal{H}} := \bigotimes_{i=1}^{n} |G_i|$ es una gramática para $L_{\mathcal{H}}$.

Un lenguaje regular $L \subset L_{\mathcal{H}}$ es llamado **"Factible"** si ambos lenguajes L y $L_{\mathcal{H}}$ son parcialmente invariantes y parcialmente bloqueados.

Sea $\mathcal{F}_G$ la familia formada por todos los lenguajes regulares factibles de $L_{\mathcal{H}}$; es decir, $\mathcal{F}_G = \{L \subset L_{\mathcal{H}}/\text{L es un lenguaje regular factible}\}$. $\mathcal{F}_G$ no es cerrado bajo la unión de conjuntos.

Capítulo 4

Problema Del Supervisor Central

Sea $\mathcal{H}$ un SED y sea $L_a \subset L_{\mathcal{H}}$ un comportamiento admisible para $\mathcal{H}$. El problema del supervisor central (PSC), consiste en sintetizar (si es posible) un supervisor restrictivo, F-completo y F-Limpio para $\mathcal{H}$ tal que $L_c \subset L_a$ y $(\mathcal{H}, S)$ sea libre de estancamiento.

Sea $L \subset L_{\mathcal{H}}$ un lenguaje regular parcialmente invariante y sea A una (MS) tal que $|A| = pre(L)$. Para cada $\sigma \in \Sigma$ sean A_σ, $\overline{A_\sigma}$ y $\overline{\overline{A_\sigma}}$ los subconjuntos del conjunto de estados Q de A, tales como fueron definidos anteriormente.
Diremos que σ es una salida bloqueada para L si $\overline{A_\sigma} \neq \emptyset$ y denotaremos por $\triangle_L$ al conjunto formado por todas las salidas bloqueadas de L; es decir, $\triangle_L := \{\sigma \in \Sigma / \overline{A_\sigma} \neq \emptyset\}$.
Para cada $\sigma \in \Sigma$, definimos una partición Π_σ de Q como "$\sigma - favorable$" si Π_σ tiene dos elementos C_σ y $\overline{C_\sigma}$ que verifican $A_\sigma \subseteq C_\sigma$ y $\overline{A_\sigma}\bigcup(\overline{A_\sigma})\Sigma^{-1} \subseteq \overline{C_\sigma}$.
Sea Π^*_σ la partición de Q que verifica $A_\sigma = C_\sigma$ y $\overline{A_\sigma}\bigcup\overline{\overline{A_\sigma}} = \overline{C_\sigma}$. Como hemos supuesto que $L \subset L_{\mathcal{H}}$ es un lenguaje regular parcialmente invariante, por la proposición 3.3 tenemos que $(\overline{A_\sigma})\Sigma^{-1} \subseteq \overline{A_\sigma}\bigcup\overline{\overline{A_\sigma}}$, luego la partición $\Pi^*_\sigma = C_\sigma \bigcup \overline{C_\sigma}$ tiene dos elementos, $A_\sigma = C_\sigma$ y $\overline{A_\sigma}\bigcup\overline{\overline{A_\sigma}}\Sigma^{-1} \subseteq \overline{A_\sigma}\bigcup\overline{\overline{A_\sigma}} = \overline{C_\sigma}$; por lo tanto, Π^*_σ es una partición $\sigma - favorable$ de Q, y podemos concluir que existe por lo menos una partición $\sigma - favorable$ para Q.

La familia de particiones favorables de Q no es cerrada bajo uniones.
Nosotros diremos que una partición Π de Q es "**favorable**"si Π es la reunión de la familia $\{\Pi_\sigma/\sigma \in \Sigma$ y Π_σ es una partición $\sigma - favorable$ de $Q\}$. Notemos que siempre existe al menos una partición favorable para Q, ya que $\Pi^* = \bigcup\{\Pi^*_\sigma/\sigma \in \Sigma\}$ es siempre favorable.

Teorema 4.1. *El problema del supervisor central (PSC) es soluble si existe un lenguaje $L \subset \mathcal{F}_G$ tal que $L \subset L_a$ y $\triangle_L \subset \Lambda$.*

Demostración. (Si) Supongamos que existe un lenguaje $L \neq \emptyset$, $L \subseteq \mathcal{F}_G$ tal que $L \subseteq L_a \subseteq L_{\mathcal{H}}$ y $\triangle_L \subseteq \Lambda$. Sea $A = (Q, \Sigma, q_0, T, \delta)$ una (MS) tal que $|A| = pre(L)$ y sea Π una partición favorable para A; es decir,
$\Pi = \bigcup\{\Pi_\sigma/\sigma \subseteq \Sigma$ y Π_σ es una partición $\sigma - favorable$ de $Q\}$
$\Pi = \bigcup\{\Pi_\sigma/\sigma \subseteq \Sigma$ y $\Pi_\sigma = C_\sigma \bigcup \overline{C_\sigma}$, donde $A_\sigma = C_\sigma$ y $\overline{A_\sigma}\bigcup(\overline{A_\sigma})\Sigma^{-1} \subseteq \overline{C_\sigma}\}$.
Por hipótesis $\triangle_L \subset \Lambda$, por lo tanto Π es una control-partición de Q donde podemos poner $C_\sigma = \{q \in Q$ / q habilita a $\sigma\} = E_\sigma$
y $\overline{D_\sigma} = \overline{A_\sigma}$,
$\overline{D_\sigma} = \{q \in Q$ / q inhabilita a σ, existe $u \in pre(L_c)$, con $\delta(u, x_0) = x$ y $u\sigma \in pre(L_{\mathcal{H}})\}$;
$\overline{A_\sigma} = \{q \in Q$ /$\delta(\sigma, q)$ es indefinida y existe $u \in pre(L)$, con $\delta(u, q_0) = q$, $u\sigma \in pre(L_{\mathcal{H}})\}$.

Luego, $(E_\sigma)\Sigma\bigcap\overline{D_\sigma}=(C_\sigma)\Sigma\bigcap\overline{A_\sigma}$. Veamos que $(C_\sigma)\Sigma\bigcap\overline{A_\sigma}=\emptyset$. En efecto, supongamos que $(C_\sigma)\Sigma\bigcap\overline{A_\sigma}\neq\emptyset$, entonces existe $y\in C_\sigma$ y existe $w\in\Sigma$ tales que $\delta(w,y)\in\overline{A_\sigma}$, de donde $y\in C_\sigma\bigcap(\overline{A_\sigma})\Sigma^{-1}$, lo que contradice el hecho de que Π_σ es $\sigma-favorable$; es decir, $\overline{A_\sigma}\bigcup(\overline{A_\sigma})\Sigma^{-1}\subseteq\overline{C_\sigma}$ implica que $(\overline{A_\sigma})\Sigma^{-1}\subseteq\overline{C_\sigma}$, de donde $C_\sigma\bigcap(\overline{A_\sigma})\Sigma^{-1}\subseteq\overline{C_\sigma}$ que claramente es una contradicción pues C_σ y $\overline{C_\sigma}$ conforman la partición Π_σ. Así, $(C_\sigma)\Sigma\bigcap\overline{A_\sigma}=\emptyset$ para todo $\sigma\in\Sigma$. Luego por la proposición 3.2 A es un supervisor **restrictivo**. Así, como A es control-consistente, por el lema 3.2, $L_c=|A|\bigcap L_{\mathcal{H}}$, pero $|A|=pre(L)$; por lo tanto, $L_c=pre(L)\bigcap L_{\mathcal{H}}$, pero $L\subseteq L_{\mathcal{H}}$ de donde $pre(L)\subseteq pre(L_{\mathcal{H}})$, luego $L_c=|A|\bigcap L_{\mathcal{H}}=L$. Sea $q\in Q$ y $\sigma\in\Sigma$ tales que $\delta(\sigma,q)$ esta definido. Como $|A|=pre(L)$ y $L_c=|A|\bigcap L_{\mathcal{H}}=L$, entonces existe $u\in pre(L_c)$ tal que $\delta(u,q_0)=q$ y $u\sigma\in pre(L_c)$; por lo tanto, A es **F-Limpio**. Luego, si $u\in L_p=|A|\bigcap pre(L_{\mathcal{H}})$ y $u\sigma\in pre(L_{\mathcal{H}})$, $\sigma\in\Sigma$ y $\Phi(u,\sigma)=1$, entonces $u\sigma\in pre(L_c)$ de donde $u\sigma\in pre|A|$. En consecuencia A es **F-Completo**. Luego, como $L\subseteq L_{\mathcal{H}}$, entonces $|A|=pre(L)\subseteq pre(L_{\mathcal{H}})$. Además, como $|A|=pre|A|$, entonces $L_c=|A|\bigcap L_{\mathcal{H}}$. Por lo tanto, $pre(L_c)=pre|A|\bigcap pre(L_{\mathcal{H}})=L_p$, luego por definición L es parcialmente bloqueado y así $(\mathcal{H},A)$ es **libre de estancamiento**.

(solo si)

Supongamos que el (PSC)es soluble, y sea S un supervisor restrictivo, F-completo y F-limpio para $\mathcal{H}$ tal que $L_c\subseteq L_a$, para algún lenguaje admisible $L_a\subseteq L_{\mathcal{H}}$ con $(\mathcal{H},A)$ libre de estancamiento. Sea además $L=L_c$, luego $\triangle_L\subset\Lambda$. En efecto, si $\triangle_L\nsubseteq\Lambda$ entonces existe $w\in\triangle_L$ y $w\notin\Lambda$, $\overline{A_w}\neq\emptyset$; así, existe $q\in Q$ tal que $\delta(w,q)$ no esta definida y existe $u\in pre(L)$, $\delta(u,q_0)=q$ y $uw\in pre(L_{\mathcal{H}})$; luego, $uw\in pre(L_{\mathcal{H}})$ y $\Phi(w,u)=1$ pero $uw\notin|A|$. Esto dice que A no es F-completo, lo que es una contradicción; por lo tanto, $\triangle_L\subset\Lambda$. Luego, por la proposición 3.5, L_c es parcialmente invariante y además por la definición de sistema libre de estancamiento L_c es parcialmente bloqueado. ■

Sea $Sup_{\mathcal{H}}$ la categoría cuyos objetos son pares (S,β), donde S es un supervisor para $\mathcal{H}$ y $\beta:\Lambda\longrightarrow\{0,1\}$ es una función que específica el estatus inicial de cada salida controlable de $\mathcal{H}$. Existe un morfismo $F=(f_1,f_2,f_3)$ definido de $(\mathcal{A},\alpha)$ en (S,β) en $Sup_{\mathcal{H}}$ siempre que $\alpha=\beta$ y $F=(f_1,f_2,f_3)$ sea un morfismo de (MSG) definido de $\mathcal{A}$ en β, con $f_1=id:\Sigma\longrightarrow\Sigma$ y $f_3=id:\Gamma\longrightarrow\Gamma$. Si un supervisor S es control-consistente entonces nosotros podemos reemplazar la función de salida por su control-partición asociada (definida anteriormente). Un supervisor control-consistente para $\mathcal{H}$ es entonces un par (S,Π), con S una (MS) y Π su control-partición asociada.

Un morfismo de supervisores control-consistentes definido desde $(\mathcal{A},\lambda)$ en (S,Π) es un morfismo de (MS) $(f_1,f_2):\mathcal{A}\longrightarrow S$, donde $f_1=id:\Sigma\longrightarrow\Sigma$ y $\lambda=\Pi f_2$. Denotaremos por $Sup^c_{\mathcal{H}}$ a la categoría formada por los supervisores control-consistentes y sus morfismos.

Presentaremos ahora los dos resultados principales de este trabajo

Teorema 4.2. *Un Supervisor control-consistente y F-Limpio* (S, Π) *resuelve el (PSC) con el comportamiento de* $(\mathcal{H}, S) = L$ *si y sólo si existe una (MS)A tal que* $|A| = pre(L)$*, con partición favorable* λ*, tal que* (A, λ) *resuelve el (PSC) con* $L_c = L$ *y existe un epimorfismo* $F : (A, \lambda) \longrightarrow (S, \Pi)$ *en* $Sup^c_{\mathcal{H}}$.

Demostración.
(si) Sea $\mathcal{H}$ un SED y sea $L \subseteq La \subseteq L_{\mathcal{H}}$
Sea $S = (\Sigma, X, x_0, T, \beta)$ una (MS) tal que (S,Π) es un supervisor control consistente y F-Limpio para $\mathcal{H}$. Supongamos que $A = (\Sigma, Q, q_0, T_2, \delta)$ es una (MS) tal que $|A| = pre(L)$ y que λ es una partición favorable de Q tal que (A, λ) resuelve el (PSC), con $L_c(A) = L$.
También, que existe un epimorfismo $f : (A, \lambda) \to (S, \Pi)$ en $Sup^c_{\mathcal{H}}$.
Debemos probar (I): que (S, Π) resuelve el (PSC) con $L_c(S) = L \subset L_a \subset L_{\mathcal{H}}$.
El lenguaje $L_c(S)$ buscado debe tener la posibilidad de realizar las tareas elegidas (esto se logra poniendo $L \subseteq L_c(S)$ y esto se puede suponer sin ningún problema).
Luego $preL \subseteq preL_c(S)$ pero $preL \bigcap preL_{\mathcal{H}} = preL \subseteq preL_c(S)$, pues $L \subset L_{\mathcal{H}}$; luego, $|A| \bigcap preL_{\mathcal{H}} = preL \bigcap preL_{\mathcal{H}} = preL \subseteq preL_c(S)$, por hipótesis.
Por lo tanto $L_p(A) = |A| \bigcap preL_{\mathcal{H}} \subseteq preL_c(S) = |S| \bigcap preL_{\mathcal{H}} = L_p(S)$ y $L_p(A) \subseteq L_p(S)$.
Sea ahora $\sigma \in \Sigma$ tal que $\sigma \epsilon L_p(S) = |S| \bigcap preL_{\mathcal{H}}$, entonces $\beta(\sigma, x_0)$ esta definida y σ es habilitada desde x_0, como $f_2 : Q \to x$ en nuestro morfismo es tal que $f_2(q_0) = x_0$, entonces σ es habilitada desde q_0 y como A es F-completo entonces $\delta(q_0, \sigma)$ esta definida y $\sigma \in L_p(A)$.
Asumamos ahora que cada palabra de longitud k en $L_p(S)$ es una palabra en $L_p(A)$ y sea $\omega = u\sigma \in L_p(S)$ una palabra de longitud $(k + 1)$, donde $\sigma \in \Sigma$.
Sean $x \in X$ y $q \in Q$ tales que $\beta(x_0, u) = x$ y $\delta(q_0, u) = q$, luego como $F_2 : Q \to X$ en nuestro morfismo es tal que $f_2(q) = x$, entonces σ es habilitada desde q y como A es F-completo, entonces $u\sigma \in L_p(A)$, de donde $L_p(S) \subseteq L_p(A)$.
Por lo tanto, $L_p(S) = L_p(A)$; luego, $L_c(A) = L_p(A) \bigcap L_{\mathcal{H}} = L_p(S) \bigcap L_{\mathcal{H}} = L_c(S)$.
Pero $L_c(A) = L \subseteq L_a \subseteq L_{\mathcal{H}}$. Así, $L_c(S) = L \subseteq L_a \subseteq L_{\mathcal{H}}$.
Para probar **(I)** debemos ver que S es un supervisor que verifica lo siguiente:

i.) Restrictivo

ii.) F-completo

iii.) F-Limpio

iv.) $L_c(S) = L \subset L_a \subset L_{\mathcal{H}}$

v.) $(\mathcal{H}, S)$ es libre de estancamiento

Luego iii) y iv) están probados.
Veamos i): S es restrictivo. En efecto supongamos que S no es restrictivo entonces existe $\sigma \in \Sigma$ y existe $u\omega\sigma \in L_{\mathcal{H}}$, $\omega \in \Sigma$, con $u\omega \in preL_c(S)$ y $\phi(u,\omega,\sigma) = 0$. Entonces, $\Phi(u,\sigma) = 1$ en S.
Como $u\omega \in preL_c(S) = L_p(S) = |S| \bigcap preL_{\mathcal{H}}$, entonces existen x_1 y x_2 en X tales que $\beta(x_0,u) = x_1$ y $\beta(x_1,\omega) = x_2$, además ω es habilitada después de u en S, σ es habilitada después de u en S y σ no es habilitada después de ω en S.
Luego $f_2(q_1) = x_1$ y $f_2(q_2) = x_2$, de aquí que ω es habilitada después de u en A, σ es habilitada después de u en A y σ no es habilitada después de ω en A.
Lo que es una contradicción con la hipótesis, pues $(\mathcal{H}, A)$ resuelve el (PSC). Así S es restrictivo.
ii): S es F-completo. En efecto, suponga que S no es F-completo, entonces existe una palabra $u \in L_p(S) = |S| \bigcap preL_{\mathcal{H}}$, $u\sigma \in preL_{\mathcal{H}}$, con $\sigma \in \Sigma$, σ es habilitada desde u en S y $u\sigma \notin |S|$.
Como $u \in L_p(S)$, entonces existen x_1, x_2 en X tales que $\beta(x_1,u) = x_2$; y como $f_2(q_1) = x_1$ y $f_2(q_2) = x_2$, entonces σ es habilitada desde u en A.
Como A es F-completo, entonces $u\sigma \in |A|$, luego $\delta(q_2,\sigma)$ esta definido y por lo tanto existe $x_3 = f_2(\delta(q_2,\sigma))$ *tal que* $\beta(x_2,\sigma) = x_3$ lo cual es una contradicción. Así, S es F-completo. V): El sistema bajo la supervision de S, (H,S), es libre de estancamiento. En efecto, considere al lenguaje controlado $L_c(S)$, por el lema 3.2 $L_c(S) = |S| \bigcap L_{\mathcal{H}}$, de donde $pre(L_c) = pre(|s|) \bigcap preL_{\mathcal{H}} = |S| \bigcap pre(L_{\mathcal{H}}) = L_p(S)$.
Veamos ahora que $L_c(S)$ es parcialmente bloqueado. Para esto supongamos que $L_c(S)$ no es parcialmente bloqueado, entonces existe una palabra u en $PreL_c(S)$ tal que para cada j en N_n^+, con $\Sigma_j^u \neq \emptyset$ se tiene que [existe $\sigma \in \Sigma_i^u, u\sigma \notin pre(L_c(S))$] ó [$u \in L_c(S)$ implica que $P_i(u) \in |P_i|P_i(u) \notin pre(|P_i|)$], pero $L_c(A) = L_c(S)$implica que hemos mostrado un elemento u en $pre(L_c(A))$ tal que para cada i en N_n^+, con $\Sigma_j^u \neq \emptyset$ se tiene que [existe $\sigma \in \Sigma_i^u, u\sigma \notin pre(L_c(A))$] ó [$u \in L_c(A)$ implica que $P_i(u) \in |P_i|P_i(u) \notin pre(|P_i|)$] . Por lo tanto, $L_c(S)$ no es parcialmente bloqueado, lo cual es una contradicción, pues (H,A) es libre de estancamiento por hipótesis. Así, queda demostrada esta implicación.

(solo si) Sea (S,Π) un supervisor control-consistente y F-Limpio que resuelve el (PSC) con el comportamiento $L_c(S) = L$. Luego, como S es control-consistente entonces por el lema 3.2 $L_c(S) = |S| \bigcap L_{\mathcal{H}}$ y por la proposición 3.1 $L_c(S)$ es regular. Sea entonces G_c una (Grm) sobre Σ tal que $|G_c| = L_c(S)$, donde $G_c = S \bigcap G_{\mathcal{H}}$, además S es F-limpio, luego como las categorías (CGrm) y (CMS) son isomorfas. Entonces, con abuso de lenguaje podemos considerar un epimorfismo $h : G_c \longrightarrow S$, donde $f_1 := id : \Sigma \longrightarrow \Sigma$.
Sea $A = (\Sigma, Q, q_0, T, \delta)$ un supervisor para $\mathcal{H}$ y sea $F = (f_1, f_2) : A \longrightarrow S$ donde $f_1 := id : \Sigma \longrightarrow \Sigma$ es el epimorfismo correspondiente con h. Definamos la partición λ de Q por $q \equiv q'$ en λ si $h_2(q) = h_2(q')$ en Π, donde $h_2 : Q \longrightarrow X$ es la función de estado del

morfismo h. Veamos que λ es una partición favorable.
Supongamos entonces que λ no es una partición favorable, entonces para algunos q y q' en Q y σ en Σ tal que $q \in A_\sigma$, $q' \in \overline{A_\sigma} \bigcup (\overline{A_\sigma})\Sigma^{-1}$ y $q \equiv q'$ en λ, luego $h_2(q) \in D_\sigma$, $h_2(q') \in \overline{D_\sigma} \bigcup (\overline{D_\sigma})\Sigma^{-1}$ y $h_2(q') \in \Sigma_\sigma$. Esto contradice el hecho de haber asumido que S resuelve el (PSC). Por lo tanto λ es una partición favorable, y por construcción $\lambda = \Pi h_2$. Finalmente $F : (A, \lambda) \longrightarrow (S, \Pi)$ es un epimorfismo en $Sup^c_{\mathcal{H}}$. Luego del teorema 4.1 (A, λ) resuelve el (PSC). ■

Teorema 4.3 (De la estructura cociente). *Un Supervisor F-Limpio* (S, α) *resuelve el (PSC) con el comportamiento de* $(\mathcal{H}, S) = L$ *si y sólo si existe una (MS)A tal que* $|A| = pre(L)$, *y una función de salida* $\tau : \Sigma \times \Phi \longrightarrow \Sigma^*$ *para A tal que* $(B = (A, \tau), \alpha)$ *resuelve el (PSC) con* $L_c = L$ *y existe un epimorfismo* $F : (B, \alpha) \longrightarrow (S, \alpha)$ *en* $Sup_{\mathcal{H}}$.

Demostración. La demostración se sigue al igual que la del teorema 4.2 ya que como en este trabajo hemos considerado a todo supervisor control consistente, entonces podemos reemplazar la función de salida del supervisor considerado por su control partición asociada. ■

Conclusión

El modelo $(S, \mathcal{H})$ verificando los argumentos establecidos en el teorema 4.3 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 (tarea que corresponde al un ingeniero de control), 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 $\mathcal{H}$.

Bibliografía

I. Ramadge P.J. and Wonham W.M., *Supervisory Control of Discrete Event Processes*,Lecture Notes in Control and Inform.,v.39, pp.202-214 (1982).

II. Ramadge P.J. and Wonham W.M., *Supervisory Control of a Class of Discrete Event Processes*,Lecture Notes in Control and Inform.,v.63, p.477-498 (1984).

III. Ramadge P.J. and Wonham W.M., *Modular Supervisory Control of Discrete Event Systems*,Lecture Notes in Control and Inform.,v.83, p.202-214 (1986).

IV. Dalta, Rodrigo, *Control supervisorio de una clase de sistemas de eventos discretos* tesis de licenciatura codigo en la biblioteca BIACI en la ULA QA402 D3

V. Méndez Urrieta, Arnaldo José, *Autómata Minimal* tesis de licenciatura codigo en la biblioteca BIACI en la ULA QA402.3 M4

VI. Camacho, Franklin J, *Dinámica de los autómatas* tesis de licenciatura codigo en la biblioteca BIACI en la ULA QA402.2 C3

VII. , Eilenberg. S. *Aútomata, Languages and Machines.*, Academic Press, New York. (1974).

VIII. , Maclane. S. *Categories for the Working Matematician.*Springer-Verlang, New York (1971).