

Capítol 3

Disseny d'un supervisor per a tasques de vigilància

En aquest capítol, es descriurà la metodologia a seguir per a l'obtenció d'un model a esdeveniments discrets del sistema que es desitja supervisar. Aquest model permetrà conèixer en tot moment l'estat del procés, i per tant, permetrà al supervisor desenvolupar les tasques de vigilància.

El punt de partida és el coneixement que l'expert té sobre el procés, que es concreta en els ordenaments d'esdeveniments significatius esperats. El model a esdeveniments discrets servirà com a vehicle per descriure aquests ordenaments. El modelatge del procés com a DES es farà en base a la teoria dels llenguatges i dels autòmats, com a generadors d'aquests.

Aquest capítol comença amb una introducció a aquestes teories, definint conceptes, propietats i operacions, adaptades a les necessitats que es plantegen en aquesta Tesi. A [55], es pot trobar una descripció general sobre els llenguatges formals i els autòmats. Si es vol tenir un punt de vista més proper als sistemes a esdeveniments discrets es pot consultar [33] i l'apartat 1.3.

Després, es presenta la metodologia de modelatge del DES, basada en l'operació de composició d'autòmats. A [90], Sampath descriu una metodologia similar: proposa la composició de components basats en actuadors i l'incorporació d'informació dels sensors mitjançant mapes de sensors. En canvi, aquí es proposa treballar directament amb components basats en sensors, actuadors i altres elements del procés real. D'altra banda, la metodologia que aquí es planteja, estableix una terminologia nova i una estructuració més detallada.

Finalment, un exemple d'aplicació permet mostrar com es pot utilitzar el model obtingut per a monitoritzar un procés real, identificant en línia l'estat de funcionament d'un sistema de dipòsits pulmó.

3.1 Teoria dels llenguatges formals

L'element bàsic de la teoria és el *símbol*. Un símbol pot representar una lletra, un dígit, una idea, etc. A partir d'aquí apareixen conceptes derivats com alfabet, cadena, prefix o llenguatge. La interpretació que aquí se'n fa és la de considerar com element bàsic l'*esdeveniment*. Un esdeveniment estarà representat per una lletra.

Sigui Σ un conjunt finit d'esdeveniments anomenat *alfabet*. Un alfabet representa un conjunt d'esdeveniments que es poden originar en un procés.

Una *cadena* o paraula és una seqüència finita d'esdeveniments i representa una trajectòria d'estat. Una cadena es simbolitza per la juxtaposició d'esdeveniments pertanyents a un mateix alfabet. Sovint, quan una cadena està constituïda per un sol esdeveniment, s'acostuma a representar-la amb la mateixa lletra que l'esdeveniment. La cadena buida ϵ està constituïda per zero esdeveniments.

El conjunt Σ^* agrupa totes les cadenes finites que es poden constituir a partir de l'alfabet Σ , incluint ϵ . La cardinalitat del conjunt Σ^* és infinita.

La operació de *concatenació* de dues cadenes u i v resulta en una nova cadena w formada per la seqüència d'esdeveniments de la primera cadena seguida per la seqüència d'esdeveniments de la segona. La concatenació es simbolitza per la juxtaposició de cadenes, $w = uv$ i permet definir noves cadenes a partir de cadenes existents.

La cadena $u \in \Sigma^*$ és un *prefix* de la cadena $w \in \Sigma^*$ si existeix alguna cadena $v \in \Sigma^*$ tal que w resulti de la concatenació uv . De manera similar, es pot definir el concepte de sufix.

Donat un alfabet d'esdeveniments Σ , és possible constituir múltiples cadenes $w \in \Sigma^*$. Tenint en compte que aquests esdeveniments s'originen en un procés, és lògic pensar que certs ordenaments d'esdeveniments siguin *admissibles* mentre que uns altres no. Ara per ara, s'entendrà com admissible aquell ordenament que permeti descriure un dels diversos comportaments esperats del procés. A la pàg. 112 es donarà una explicació més concisa sobre quins són els comportaments esperats.

Un *llenguatge formal*, o simplement llenguatge, L reuneix totes les cadenes admissibles. Es compleix que $L \subseteq \Sigma^*$. El cas més usual és l'estricta $L \subset \Sigma^*$, en que Σ^* inclou també cadenes que corresponen a trajectòries d'estat no admissibles, és a dir, no possibles segons el comportament esperat del procés. La cardinalitat de L pot ser tan finita com infinita. Sovint, quan un llenguatge només està constituït per una sola cadena s'acostuma a representar amb el mateix nom que la cadena.

Exemple 3.1.1: Sigui un dipòsit sense canalització d'evacuació i alimentat per una bomba. Suposem que inicialment el dipòsit està buit i la bomba aturada, i que hi ha la possibilitat de detectar únicament els següents esdeveniments:

- La bomba es posa en marxa, b , i
- el nivell del dipòsit augmenta, n .

Tot i que la bomba també es pot aturar, imaginem que només és possible detectar si s'ha posat en marxa. De forma anàloga, malgrat que el nivell també pot mantenir-se o disminuir, suposem que només és possible detectar quan augmenta.

En aquest cas l'alfabet és $\Sigma = \{b, n\}$. El conjunt de totes les cadenes definibles sobre l'alfabet Σ és $\Sigma^* = \{\epsilon, b, n, bb, bn, nb, nnb, bbb, \dots\}$. Està clar que aquest conjunt conté un número infinit de cadenes, contemplant totes les concatenacions possibles. De totes elles, tant les que tenen com a prefix n com les que presenten dos o més esdeveniments n successius no són admissibles (e.g., nb , $bnnb$, $bnbn$). La raó és que només és possible un augment de nivell si prèviament s'ha posat en marxa la bomba. Això vol dir que només són admissibles les que constitueixen el següent llenguatge:

$$L = \{\epsilon, b, bb, bbb, bn, bnb, bnbb, bnbbn, bnbbnb, bbbn, bbbnb, bbbnbb, \dots\}$$

En aquest cas, L també conté un número infinit de cadenes. \diamond

Un llenguatge és un conjunt i, per tant, es poden aplicar les operacions usuals sobre conjunts: unió, intersecció, diferència, etc. A més, es defineixen tres operacions addicionals:

Concatenació Siguin dos llenguatges L_1 i L_2 , llavors $L_1L_2 = \{w \mid (w = uv) \wedge (u \in L_1) \wedge (v \in L_2)\}$.

Tancament de Kleene (Kleene closure) Sigui un llenguatge L , llavors $L^* = \bigcup_{n=0}^{\infty} L^n$, on $L^0 = \{\epsilon\}$ i $L^n = L^{n-1}L$.

Tancament de prefixes (prefix-closure) Sigui un llenguatge L , llavors $\bar{L} = \{u \mid (w = uv) \wedge (w \in L) \wedge (u, v \in \Sigma^*)\}$. És a dir, \bar{L} és el llenguatge constituït per tots els prefixes de L .

Abans s'ha dit que L està constituït per totes les cadenes admissibles. Si una determinada cadena és admissible, és lògic que també ho siguin tots els seus prefixes. Per tant, un llenguatge que representi tots els comportaments admissibles d'un determinat procés ha de complir la següent propietat.

Definició 3.1.1. Un llenguatge L és *tancat amb prefixes* (prefix-closed) si $L = \bar{L}$. \blacklozenge

És a dir, un llenguatge tancat de prefixes conté tots els seus propis prefixes.

Exemple 3.1.2: Siguin $L_1 = \{a, cba\}$ i $L_2 = \{\epsilon, a, ab\}$ llavors,

$$\begin{aligned} L_1^* &= \{\epsilon, a, cba, aa, cbacba, acba, cbaa, aacba, cbaacba, \dots\} \\ \bar{L}_1 &= \{\epsilon, a, c, cb, cba\} \\ L_2^* &= \{\epsilon, a, ab, aa, abab, aab, aba, aaab, abaab, \dots\} \\ \bar{L}_2 &= \{\epsilon, a, ab\} \end{aligned}$$

Per tant, L_1 no és tancat de prefixes, però L_2 sí. ◇

Exemple 3.1.3: El llenguatge de l'ex. 3.1.1 representa un procés, i efectivament, és tancat de prefixes. ◇

Els elements que constitueixen un llenguatge es poden expressar de forma molt compacta mitjançant l'ús d'*expressions regulars*. Aquestes expressions es basen en l'aplicació dels operadors concatenació, unió (*e.g.*, $a + b \equiv a \cup b$) i el tancament de Kleene, un número finit de vegades sobre els llenguatges¹ \emptyset , ϵ i qualsevol $\sigma \in \Sigma$.

Exemple 3.1.4: Siguin $L_1 = \{a, b\}$, $L_2 = \{ab, abab, ababab, abababab, \dots\}$ i $L_3 = (ab + c)^*d$. L_1 es pot representar per l'expressió regular $a + b$. Més interessant resulta per L_2 , que es pot representar fàcilment per $ab(ab)^*$. Finalment, la tercera expressió regular representa el següent llenguatge:

$$L_3 = \{d, abd, cd, abcd, ccabd, abcabd, ccabd, \dots\} \quad \diamond$$

Exemple 3.1.5: El llenguatge de l'ex. 3.1.1 es pot representar mitjançant l'expressió regular següent:

$$(bb^*n)^* b^* \quad \diamond$$

Observi's que una expressió regular permet representar llenguatges finits i infinits. De fet, pot expressar qualsevol llenguatge finit, però només certs llenguatges infinits. Un *llenguatge regular* és aquell que es pot representar per una expressió regular.

En resum, l'aspecte més interessant de la teoria dels llenguatges és que aquells llenguatges que són tancats de prefixes permeten definir el comportament admissible d'un procés en base a l'ordenació dels esdeveniments que s'hi originen. A continuació, veurem que, si aquest llenguatge és també regular, llavors és possible definir una màquina d'estats finits com a model a esdeveniments discrets del procés.

¹Convé recordar la convenció segons la que un llenguatge constituït per un únic esdeveniment pren el nom d'aquest.

3.2 Teoria dels autòmats

3.2.1 Autòmats i generadors

Un autòmat és un dispositiu capaç d'acceptar un llenguatge d'acord amb unes regles específiques.

Definició 3.2.1. Un *autòmat determinista*, o simplement autòmat, A és una tupla,

$$A = (Q, \Sigma, \eta, q_0, Q_f)$$

on:

- Q – és un conjunt d'estats,
- Σ – és un alfabet finit d'entrada,
- η – és la funció de transicions d'estat: $\eta : Q \times \Sigma \rightarrow Q$,
- q_0 – és l'estat inicial, $q_0 \in Q$, i
- Q_f – és un conjunt d'estats finals, $Q_f \subseteq Q$. ♦

La funció η és completa. És a dir, proporciona, per a cada estat origen i cada esdeveniment, un estat destí. Es tracta d'un autòmat determinista, ja que aquest estat destí és únic per cada estat origen i esdeveniment. Existeixen també els autòmats no deterministes en els que hi ha la possibilitat de que, per un estat origen i un esdeveniment, es defineixin múltiples estats destí. En aquesta tesi només mereix interès l'autòmat determinista, al que s'anomenarà a partir d'ara simplement *autòmat* o *màquina d'estats*.

D'altra banda, aquest autòmat està constituït per un número indeterminat d'estats, que podrien arribar a ser infinits. Un cas particularment interessant és aquell en que el número d'estats és finit. Quan interressi referir-se a aquest cas en particular es parlarà d'un *autòmat finit* o *màquina d'estats finits*.

D'altra banda, ara per ara, de Q_f únicament es dirà que representa un subconjunt distingit d'estats als que convindrà tenir en compte a l'hora de garantir certes propietats desitjables de l'autòmat.

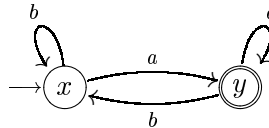
És possible representar gràficament un autòmat mitjançant un *diagrama de transicions*. Un diagrama de transicions és un graf dirigit en que els nodes representen els estats i les fletxes les transicions d'estat. Hi ha una fletxa etiquetada $\sigma \in \Sigma$ que va de l'estat origen $q \in Q$ a l'estat destí $q' \in Q$ sempre i quan es compleixi $\eta(q, \sigma) = q'$. Cada node s'etiqueta amb el nom d'un estat. L'estat inicial s'identifica amb una petita fletxa d'entrada no etiquetada, i els estats finals es simbolitzen amb un doble cercle concèntric.

Exemple 3.2.1: Sigui l'autòmat següent:

- $Q = \{x, y\}$
- $\Sigma = \{a, b\}$

- $\eta(x, a) = y, \eta(x, b) = x, \eta(y, b) = x, \eta(y, a) = y$
- $q_0 = x$
- $Q_f = \{y\}$

El diagrama de transicions corresponent s'illustra a la fig. 3.1.



◇

Figura 3.1: Autòmat corresponent a l'ex. 3.2.1.

La informació més important que permet representar un diagrama de transicions és l'encadenament d'esdeveniments. Per això, convé que un diagrama representi totes les transicions, acompanyades dels esdeveniments corresponents. Tanmateix, l'etiqueta associada a cada estat no és rellevant, i moltes vegades s'obviarà.

És usual estendre la funció de transició d'estat η al domini $Q \times \Sigma^*$ de la següent manera:

$$\begin{aligned}\eta(q, \epsilon) &= q \\ \eta(q, u\sigma) &= \eta(\eta(q, u), \sigma) \\ q &\in Q, u \in \Sigma^*, \sigma \in \Sigma\end{aligned}$$

D'aquesta manera, és possible determinar quin és l'estat destí q' quan, partint d'un estat origen q , es recorre el diagrama transitant d'un estat al següent en l'ordre establert per una cadena d'esdeveniments w . Només cal aplicar recursivament les relacions anteriors i determinar $\eta(q, w)$.

S'ha iniciat aquest apartat indicant que un autòmat és un dispositiu capaç d'acceptar un llenguatge.

Definició 3.2.2. El llenguatge $L(A)$ acceptat per l'autòmat A és el conjunt:

$$L(A) = \{w \in \Sigma^* \mid \eta(q_0, w) \in Q_f\}$$

◆

És a dir, aquest llenguatge reuneix totes les cadenes que permeten recórrer l'autòmat des de l'estat inicial a qualsevol estat final. Aquesta definició permet establir la següent proposició.

Proposició 3.2.3: Sigui l'autòmat $A = (Q, \Sigma, \eta, q_0, Q_f)$, amb $Q_f = Q$. Llavors, el llenguatge acceptat per aquest autòmat és $L(A) = \Sigma^*$. ■

DEMOSTRACIÓ: La funció η és completa i, per tant, l'autòmat A representa tots els encadenaments possibles. Si tots els estats són finals, llavors L agrupa tots aquests encadenaments. \square

S'està ara en condicions d'enunciar el següent teorema, demostrat per S.C. Kleene en la dècada dels 50s:

Teorema 3.2.4: Tot llenguatge regular pot ésser acceptat per un autòmat finit, i tot llenguatge acceptat per un autòmat finit és regular. \blacksquare

Això implica que si el comportament d'un procés (*i.e.*, el llenguatge) es pot representar amb una expressió regular, llavors és possible definir un autòmat finit com a model d'aquest procés. S'estableix així l'equivalència de representació entre un llenguatge i un autòmat finit.

Aquesta equivalència és molt important. A l'hora de representar un procés es presenten dues opcions: expressar el seu comportament mitjançant un llenguatge o mitjançant un autòmat. Normalment, el llenguatge associat a un procés és un llenguatge infinit. Això implica que, de voler emmagatzemar-ho en un ordinador, les necessitats de memòria són *a priori* infinites. No obstant, si aquest llenguatge és regular, hi haurà la possibilitat de descriure-ho mitjançant un autòmat finit. Amb la diferència que en aquest cas les necessitats de memòria esdevindran finites.

Tot i la possible existència d'aquest autòmat, resultarà més útil representar un procés mitjançant un *generador*.

Definició 3.2.5. Un *generador*, \mathcal{G} és una tupla,

$$\mathcal{G} = (X, \Sigma, \gamma, \Gamma, x_0, X_f)$$

on:

X – és un conjunt d'estats,

Σ – és un alfabet finit d'entrada,

γ – és la funció de transicions d'estat, podent estar parcialment definida:
 $\gamma : X \times \Sigma \rightarrow X$,

Γ – és la funció d'esdeveniments actius $\Gamma : X \times \Sigma \rightarrow 2^\Sigma$, que indica per cada estat $x \in X$ el conjunt d'esdeveniments $\Sigma(x) \subseteq \Sigma$ que són *admissibles*. És a dir²,

$$\Sigma(x) = \{\sigma \in \Sigma \mid \gamma(x, \sigma)!\}$$

x_0 – és l'estat inicial, $x_0 \in X$, i

X_f – és un conjunt d'estats finals, $X_f \subseteq X$. \blacklozenge

²S'utilitza la notació $\gamma(x, \sigma)!$ per indicar que la funció γ està definida per l'estat $x \in X$ i l'esdeveniment $\sigma \in \Sigma$

Aquesta definició és bastant similar a la de l'autòmat (veure def. 3.2.1). La diferència és que es permet que la funció γ estigui parcialment definida. Això implica que, en el diagrama de transicions, per a cada estat només apareguin aquelles transicions de sortida associades a esdeveniments admissibles. La idea és que un generador només representi el comportament admissible del procés.

D'altra banda, potser també serà interessant que el generador sigui capaç de representar cert comportament distingit; entenent per distingit aquell que porti a un estat final. La definició de X_f dependrà de les propietats que siguin desitjables garantir en \mathcal{G} (per exemple, no bloqueig, estabilitat, imparcialitat, seguretat, etc.)

En aquest sentit, es defineixen els següents dos llenguatges:

Definició 3.2.6. El llenguatge $L(\mathcal{G})$ generat pel generador \mathcal{G} és el conjunt:

$$L(\mathcal{G}) = \{w \in \Sigma^* \mid \gamma(x_0, w)!\}$$

El llenguatge $L_f(\mathcal{G})$ marcat pel generador \mathcal{G} és el conjunt:

$$L_f(\mathcal{G}) = \{w \in L(\mathcal{G}) \mid \gamma(x_0, w) \in X_f\} \quad \blacklozenge$$

En definitiva, el llenguatge generat correspon a aquelles cadenes que és possible recórrer a partir de l'estat inicial, i el llenguatge marcat correspon a aquelles cadenes que, partint de l'estat inicial, porten a un estat final. Clarament, $L_f(\mathcal{G}) \subseteq L(\mathcal{G})$.

Partint del teorema de Kleene 3.2.4 és possible enunciar la següent proposició:

Proposició 3.2.7: Tot llenguatge regular L que representi el comportament esperat d'un procés, pot ésser representat mitjançant un generador finit \mathcal{G} , complint-se que $L(\mathcal{G}) = L$. ■

Observi's que s'està referint a un generador amb un número finit d'estats.

DEMOSTRACIÓ: Segons el teorema de Kleene 3.2.4, si L és regular, existeix un autòmat finit $A = (Q, \Sigma, \eta, q_0, Q_f)$ que accepta aquest llenguatge, $L(A) = L$. D'altra banda, segons s'ha vist en l'apartat 3.1, el llenguatge L es caracteritza perquè:

- recull únicament les cadenes admissibles, $L \subseteq \Sigma^*$, i
- és tancat de prefixes.

La primera afirmació implica que l'autòmat A compleix $Q_f \subseteq Q$. La segona implica que, partint de l'estat origen, qualsevol recorregut associat a les cadenes contemplades a L només passa per estats finals $q \in Q_f$. En definitiva, si el procés segueix el comportament esperat, certs estats de l'autòmat A

mai seran acceditis; concretament els estats $q \in Q \setminus Q_f$. Per tant, és possible obtenir un generador finit $\mathcal{G} = (X, \Sigma, \gamma, \Gamma, x_0, X_f)$ a partir de l'autòmat A de la següent manera:

- $X = Q_f$,
- $\forall q \in Q_f \wedge x = q \wedge \sigma \in \Sigma, \quad \gamma(x, \sigma) = \eta(q, \sigma) \Leftrightarrow \eta(q, \sigma) \in Q_f$,
- Γ es defineix a partir de γ ,
- $x_0 = q_0$, i
- X_f es defineix tenint en compte les seqüències admissibles que es vulguin distingir. □

En definitiva, si el procés segueix el comportament esperat, un generador només conté aquells estats que és factible visitar.

Exemple 3.2.2: El procés de l'ex. 3.1.1 té un comportament admissible descrit pel llenguatge regular $L = (bb^*n)^*b^*$. Per tant, aquest procés es pot representar amb un autòmat finit A , tal que $L(A) = L$. El diagrama de transicions corresponent a A es mostra a la fig. 3.2(a).

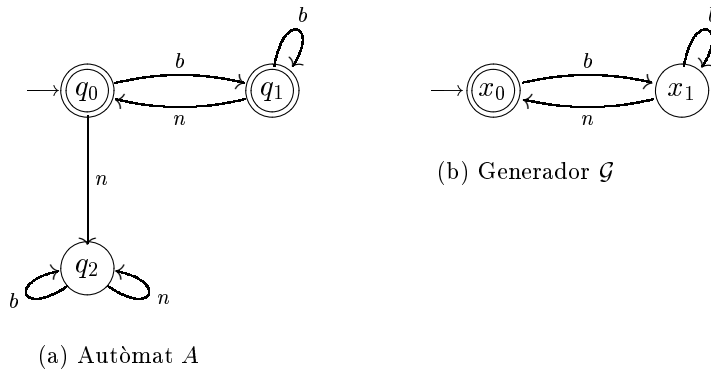


Figura 3.2: Diagrames de transicions corresponents a l'ex.3.2.2.

Observi's que els estats $Q \setminus Q_f = \{q_2\}$ mai seran assolits, ja que les seqüències que hi porten no són admissibles sota el comportament esperat del procés. Per tant, és possible representar el procés per un generador finit \mathcal{G} , tal que $L(\mathcal{G}) = L$. El diagrama de transicions corresponent a \mathcal{G} es mostra a la fig. 3.2(b).

En aquest cas, la funció γ no està definida per l'estat x_0 i l'esdeveniment n . D'altra banda, observi's que s'han distingit aquelles seqüències que condueixen a l'estat inicial, fent $X_f = \{x_0\}$. ◇

3.2.2 Operacions sobre generadors

Es defineixen a continuació un conjunt d'operacions vàlides per manipular generadors, igualment adequades per manipular autòmats.

Operacions unàries

Part accessible, $Ac(\mathcal{G})$ Consisteix a eliminar tots els estats, i transicions associades, que no es puguin assolir partint de l'estat inicial. Aquesta operació no afecta ni al llenguatge generat $L(Ac(\mathcal{G})) = L(\mathcal{G})$, ni al marcat $L_f(Ac(\mathcal{G})) = L_f(\mathcal{G})$. Per tant, a partir d'ara es suposarà que tot generador és *accessible*, és a dir, $Ac(\mathcal{G}) = \mathcal{G}$.

Part coaccessible respecte a X_f , $CoAc(\mathcal{G})$ Consisteix en eliminar tots els estats, i transicions associades, des dels que no existeixi una cadena que pugui conduir a un estat final. Aquesta operació pot afectar al llenguatge generat $L(CoAc(\mathcal{G})) \neq L(\mathcal{G})$, però no altera al marcat $L_f(CoAc(\mathcal{G})) = L_f(\mathcal{G})$. Un generador és *coaccessible* si $CoAc(\mathcal{G}) = \mathcal{G}$. Es pot demostrar que un generador és coaccessible si, i només si, $\bar{L}_f(\mathcal{G}) = L(\mathcal{G})$.

Poda, $Poda(\mathcal{G})$ Consisteix en fer un generador accessible i coaccessible: $Poda(\mathcal{G}) = CoAc(Ac(\mathcal{G})) = Ac(CoAc(\mathcal{G}))$.

Exemple 3.2.3: En l'ex. 3.2.2, el generador \mathcal{G} s'ha obtingut fent coaccessible l'autòmat A . També, s'ha canviat el conjunt d'estats finals. Però aquest fet no té res a veure amb l'operació per obtenir la part coaccessible de A . \diamond

Una de les propietats que pot ser interessant considerar a l'hora de definir X_f és la de *no blocatge*. Consideri's un procés en el que interessa garantir que en tot moment certs estats són assolibles des de l'estat actual. Només caldrà definir aquests estats com a finals, i comprovar si el generador és coaccessible.

Exemple 3.2.4: En l'ex. 3.2.2 interessa sempre poder retornar a l'estat inicial en que el dipòsit està buit. Per això s'ha definit $X_f = \{x_0\}$. En aquest cas resulta que \mathcal{G} és coaccessible, i per tant, no blocant. \diamond

Una altra de les propietats que pot interessar és la de *seguretat*. Consideri's un procés en el que es vol garantir la impossibilitat d'assolir un determinat estat indesitjable x' partint d'un altre estat x . Només caldrà redefinir l'estat inicial $x_0 = x$, obtenir la part accessible del nou generador i comprovar que l'estat x' no hi apareixi.

Operacions de composició

Es defineixen dues operacions de composició que permeten obtenir nous generadors (o autòmats) a partir de dos o més generadors (o autòmats) donats. Siguin dos generadors $\mathcal{G}_1 = (X_1, \Sigma_1, \gamma_1, \Gamma_1, x_{01}, X_{f1})$ i $\mathcal{G}_2 = (X_2, \Sigma_2, \gamma_2, \Gamma_2, x_{02}, X_{f2})$, les operacions de composició són:

Producte, $\mathcal{G}_1 \times \mathcal{G}_2$ També conegut com *producte completament síncron* o *intersecció*. Resulta en l'autòmat,

$$\begin{aligned} \mathcal{G}_1 \times \mathcal{G}_2 &= (X_1 \times X_2, \Sigma_1 \cap \Sigma_2, \gamma, \Gamma_{1 \times 2}, (x_{01}, x_{02}), X_{f1} \times X_{f2}) \\ \gamma((x_1, x_2), \sigma) &= \begin{cases} (\gamma_1(x_1, \sigma), \gamma_2(x_2, \sigma)) & \text{si } \sigma \in \Gamma_1(x_1) \cap \Gamma_2(x_2), \\ \text{no definit} & \text{en qualsevol altre cas.} \end{cases} \\ \Gamma_{1 \times 2} &= \Gamma_1(x_1) \cap \Gamma_2(x_2) \end{aligned}$$

Només provoquen una transició aquells esdeveniments que són generats simultàniament pels dos generadors. De no ser així no es produeix cap transició. Es pot comprovar que $L(\mathcal{G}_1 \times \mathcal{G}_2) = L(\mathcal{G}_1) \cap L(\mathcal{G}_2)$ i que $L_f(\mathcal{G}_1 \times \mathcal{G}_2) = L_f(\mathcal{G}_1) \cap L_f(\mathcal{G}_2)$.

Composició paral·lela, $\mathcal{G}_1 \parallel \mathcal{G}_2$ També coneguda per *producte síncron*. Resulta en l'autòmat,

$$\begin{aligned} \mathcal{G}_1 \parallel \mathcal{G}_2 &= (X_1 \times X_2, \Sigma_1 \cup \Sigma_2, \gamma, \Gamma_{1 \parallel 2}, (x_{01}, x_{02}), X_{f1} \times X_{f2}) \\ \gamma((x_1, x_2), \sigma) &= \begin{cases} (\gamma_1(x_1, \sigma), \gamma_2(x_2, \sigma)) & \text{si } \sigma \in \Gamma_1(x_1) \cap \Gamma_2(x_2), \\ (\gamma_1(x_1, \sigma), x_2) & \text{si } \sigma \in \Gamma_1(x_1) \setminus \Sigma_2, \\ (x_1, \gamma_2(x_2, \sigma)) & \text{si } \sigma \in \Gamma_2(x_2) \setminus \Sigma_1, \\ \text{no definit} & \text{en qualsevol altre cas.} \end{cases} \\ \Gamma_{1 \parallel 2} &= [\Gamma_1(x_1) \cap \Gamma_2(x_2)] \cup [\Gamma_1(x_1) \setminus \Sigma_2] \cup [\Gamma_2(x_2) \setminus \Sigma_1] \end{aligned}$$

Les transicions associades a esdeveniments comuns només es produeixen de forma sincronitzada, quan es produeixen simultàniament. Les transicions associades a esdeveniments no comuns es produeixen normalment, sense cap mena de sincronisme.

Ambdues operacions de composició compleixen les propietats commutativa i associativa. A més, es pot demostrar que $L(\mathcal{G}_1 \times \mathcal{G}_2) \subseteq L(\mathcal{G}_1 \parallel \mathcal{G}_2)$.

Un cas particular es produeix quan $\Sigma_1 = \Sigma_2$. Llavors, ambdues operacions coincideixen. D'altra banda, si $\Sigma_1 \cap \Sigma_2 = \emptyset$, la composició paral·lela s'anomena *producte barellat* (shuffle product).

Exemple 3.2.5: La fig. 3.3 mostra dos generadors \mathcal{G}_1 i \mathcal{G}_2 , així com el resultat de fer el seu producte $\mathcal{G}_1 \times \mathcal{G}_2$ i el de fer la seva composició paral·lela $\mathcal{G}_1 \parallel \mathcal{G}_2$.

◇

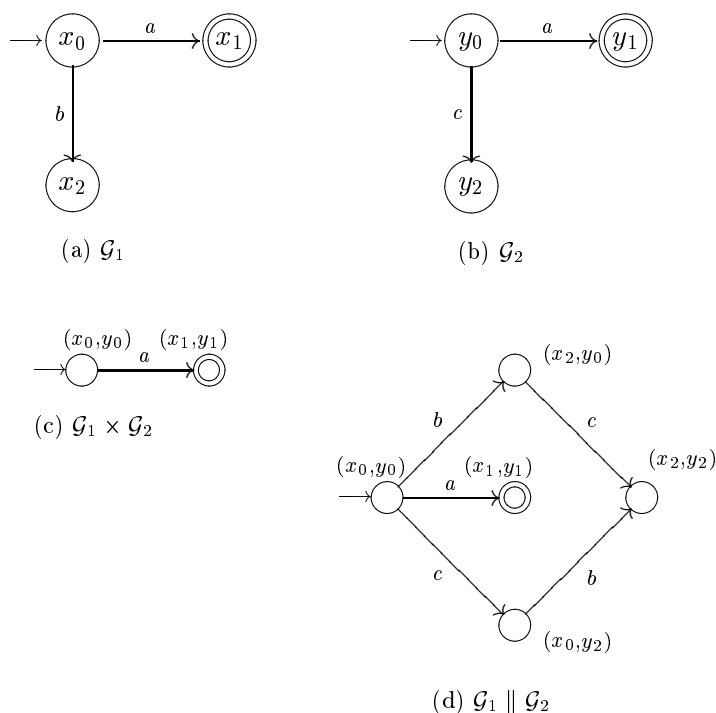


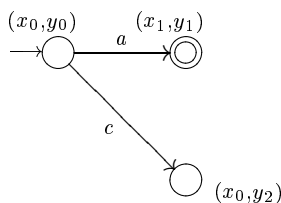
Figura 3.3: Diagrames de transicions corresponents a l'ex. 3.2.5.

En l'operació de composició paral·lela, és molt important tenir presents no només els diagrames de transicions dels generadors originals sinó també els alfabetes associats a cadascun d'ells. Podria donar-se el cas que un determinat esdeveniment formés part d'un alfabet però que no apareixés expressament en el diagrama de transicions. Això afectaria directament el resultat de la operació. A continuació en veiem un exemple.

Exemple 3.2.6: En l'ex. 3.2.5 s'ha suposat uns alfabetes segons els esdeveniments presents en els diagrames de transicions. És a dir, $\Sigma_1 = \{a, b\}$ i $\Sigma_2 = \{a, c\}$. Amb aquesta suposició les operacions de composició s'han realitzat correctament.

No obstant això, si s'haguessin tingut en compte els alfabetes $\Sigma_1 = \{a, b\}$ i $\Sigma_2 = \{a, b, c\}$, llavors l'operació producte no es veuria afectada, però l'operació de composició paral·lela resultaria en el generador mostrat en la fig. 3.4. A diferència d'abans, ara no es permet la transició associada a l'esdeveniment b , ja que s'hauria de produir sincronitzadament en els dos generadors \mathcal{G}_1 i \mathcal{G}_2 . Com b no apareix en el diagrama de transicions de \mathcal{G}_2 , aquest fet no es produirà mai.

En conclusió, no fer aparèixer esdeveniments d'un alfabet propi en un diagrama de transicions d'un generador donat, és una manera d'impedir que



◇

Figura 3.4: Diagrama de transicions corresponent a l'ex. 3.2.6.

apareixin aquests esdeveniments en el diagrama de transicions corresponent a tot generador resultant de l'operació de composició paral·lela que en pugui derivar. Aquest fet serà important tenir-lo en compte en el desenvolupament de la metodologia que es presentarà en l'apartat següent.

S'estableix el següent conveni, per tal d'evitar possibles confusions quan es realitzin operacions de composició. A menys que s'indiqui el contrari, quan es descriu un generador mitjançant un diagrama de transicions, es considerarà l'alfabet constituït implícitament pels esdeveniments associats a les transicions. Si en algun moment es desitja que s'inclogui algun esdeveniment no associat a cap transició, el diagrama de transicions s'haurà d'acompanyar de l'alfabet expressat explícitament.

Una de les aplicacions de l'operació producte és en l'anàlisi de determinades propietats dels generadors. En aquesta Tesi, no se'n farà ús. En canvi, la composició paral·lela té una aplicació clara en l'obtenció del generador d'un procés a partir dels generadors corresponents als diferents components del mateix. Per tant, és una eina que permet modelar processos complexos a partir del model de processos més simples.

La composició paral·lela és fàcil de calcular i es pot implementar fàcilment en un ordinador. No obstant això, presenta un problema greu de complexitat, a mida que el número de components o d'estats d'aquests va creixent. Observi's que l'espai d'estat del generador resultant de la composició paral·lela és el producte cartesià dels espais d'estat dels generadors inicials. Això vol dir que, en el pitjor dels casos, essent el número d'estats dels generadors originals $|X_1|$ i $|X_2|$, llavors el número d'estats del generador resultant pot arribar a ser $|X_1| \cdot |X_2|$. És a dir, el número d'estats creix exponencialment amb el número de components. Tractar un model amb molts estats comporta problemes de comprensió i claretat. A més, si s'ha de manipular i analitzar amb un ordinador pot crear dificultats i lentitud de processament.

En el següent apartat, es proposa una metodologia d'identificació d'un model a esdeveniments discrets basada en la composició paral·lela de generadors. De cara a reduir la complexitat que es pugui originar, es proposa representar el model final de la planta mitjançant *Statecharts*. Tal com s'ha argumentat a l'apartat 1.3, els *Statecharts* permeten representar DES eco-

nomitzant estats i transicions. L'única crítica a fer als Statecharts és que no contemplin la semàntica de la composició paral·lela (veure l'Apèndix D).

3.3 Metodologia de modelatge

L'objectiu de la metodologia és sintetitzar un model a esdeveniments discrets del procés. No només s'entén per procés el propi procés físic, sinó també els actuadors i sensors que permeten el seu control. És a dir, un procés es compon de dipòsits, vàlvules, motors, termoparells, canalitzacions, tancs, reaccions químiques, fluids, tacodinamos, cabalímetres, pH-ímetres, etc.

D'altra banda, el model a esdeveniments discrets que es vol obtenir ha de representar la dinàmica en llaç tancat del procés. És a dir, no només s'ha de considerar el propi procés físic, sinó que s'han de tenir en compte el sistema que el controlen: reguladors PID, autòmats programables, DSP, etc. Tota aquesta tecnologia modificarà la pròpia dinàmica del procés d'acord amb unes especificacions de control i de producció preestablertes. En altres paraules, es vol modelar el procés controlat. Des del punt de vista de la metodologia de síntesi que es vol presentar, interessa fer la següent definició formal d'un *procés*.

Definició 3.3.1. Un *procés* \mathcal{P} és una tupla,

$$\mathcal{P} = (\mathbb{D}, \mathbb{S}, \mathbb{I})$$

on:

\mathbb{D} – representa la dinàmica pròpia del procés en llaç tancat,

\mathbb{S} – és un conjunt de n senyals mesurats del procés, $\mathbb{S} = \{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_n\}$,

i

\mathbb{I} – és una interfície detectora d'esdeveniments significatius del procés. ♦

És a dir, d'acord amb una dinàmica pròpia \mathbb{D} , el procés proporciona un conjunt de senyals mesurats \mathbb{S} . La dinàmica és en llaç tancat: contempla tant les pròpies lleis físiques, com l'intervenció del sistema de control.

Interessa veure el procés com un generador espontani d'esdeveniments significatius. La interfície \mathbb{I} proveeix els mecanismes necessaris per detectar aquests esdeveniments a partir de l'anàlisi del conjunt de senyals mesurats \mathbb{S} .

La metodologia emprada per detectar aquests esdeveniments no necessàriament ha d'estar basada en el paradigma de les finestres lliscants, desenvolupada al Capítol 2. Tot i així, aquí es suposarà que la interfície detectora d'esdeveniments implementa un pla de detecció capaç de proporcionar els esdeveniments desitjats (veure def. 2.5.8). És a dir, $\mathbb{I} = \mathfrak{P}\mathfrak{Q}$. L'esquema de la fig. 3.5 il·lustra tots aquests conceptes.

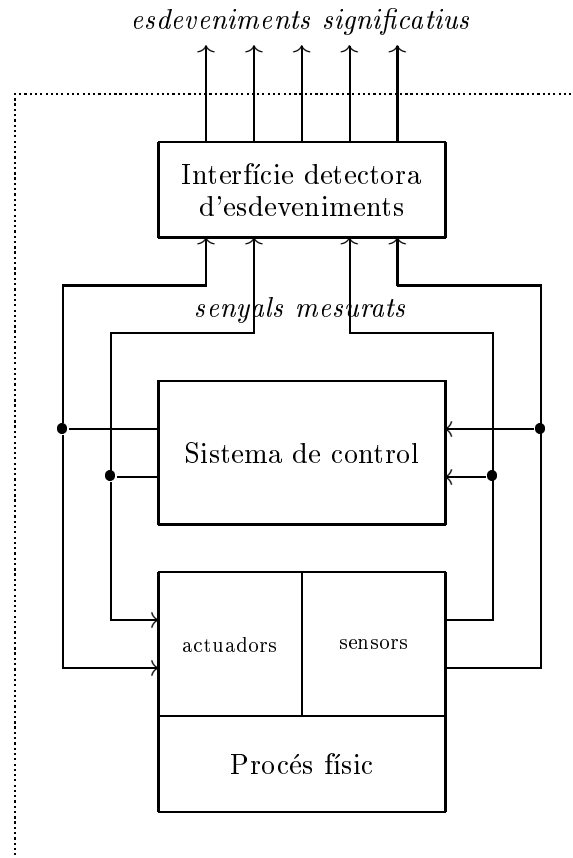


Figura 3.5: El procés a modelar.

Arribats a aquest punt convé fer dues puntualitzacions respecte a certs comentaris fets en l'apartat 2.5. D'una banda, s'ha remarcat el caràcter mostrejat de l'etapa de detecció d'esdeveniments. És a dir, els esdeveniments es detecten de forma sincronitzada a la freqüència de mostreig del senyal analitzat. Això no vol dir que es detecti un esdeveniment per a cada instant de mostreig, sinó que, de detectar-se algun esdeveniment, es farà coincidint amb algun instant de mostreig. Per tant, des del punt de vista del model a esdeveniments discrets que es vol implementar, els esdeveniments no perdaran el caràcter asíncron; continuaran apareixent de forma imprevista, però coincidint amb una granularitat temporal concreta. D'aquesta manera no es perd la coherència amb la definició de sistema a esdeveniments discrets de l'apartat 1.3.

D'altra banda, s'ha apuntat la possibilitat que un pla de detecció pugui detectar diversos esdeveniments simultàniament. Com es veurà més endavant, es proposa implementar el model a esdeveniments discrets del procés

en STATEFLOWTM. Segons es descriu a l'apartat 4.1.2, STATEFLOWTM segueix un criteri de prioritització pels esdeveniments simultanis. Aquest fet es tradueix en que, a l'hora de plantejar-se obtenir un model a esdeveniments discrets del procés, es podrà assumir que els esdeveniments mai arribaran simultàniament. En definitiva, l'entorn d'implementació resol la problemàtica de la simultaneïtat generada per la interfície detectora d'esdeveniments.

Com a punt de partida, es suposa que es té la possibilitat de detectar un conjunt d'esdeveniments significatius a partir dels senyals mesurats del procés. Aquest conjunt s'anomena alfabet del procés Σ_{PRO} . Es suposa igualment que tots els esdeveniments Σ_{PRO} són observables. És a dir, la interfície detectora d'esdeveniments permet detectar tots els esdeveniments significatius.

D'altra banda, es suposa que es coneixen els possibles ordenaments lògics esperats d'aquests esdeveniments. És d'esperar que en el procés es manifestin només certs ordenaments significatius. Significatius vol dir que siguin rellevants per l'expert, que coneix el procés. Per exemple, tot ordenament habitual en els diversos modes de funcionament normal, els ordenaments que permeten identificar una fallada en el procés, etc. En qualsevol cas, convindrà conèixer el comportament significatiu del procés controlat (*i.e.*, enllaç tancat). Aquests ordenaments es podran expressar mitjançant un llenguatge regular pel que es podrà trobar una representació equivalent mitjançant un generador finit. Donada aquesta equivalència, moltes vegades resultarà més fàcil obtenir directament el generador, sense considerar prèviament el llenguatge. Cada estat del generador identificarà un estat del procés real.

En definitiva, l'objectiu és trobar una representació com a generador finit del conjunt esquematitzat a la fig. 3.5.

En la fig. 3.5 es consideren dos tipus de senyals mesurats. Uns són originats en el procés i generats per sensors. Corresponen a mesures de variables físiques. Per exemple, temperatura, cabal, nivell, pressió, velocitat, posició, color, etc. Els altres són originats en els sistemes de control i destinats als actuadors. Per exemple, l'apertura d'una vàlvula, tensió d'alimentació d'un motor, tensió d'alimentació d'una bomba, etc.

La *Teoria del Control Supervisor*[86][108], iniciada per Wonham i Ramadge, es basa en la consideració de dos tipus d'esdeveniments: *esdeveniments controlables* i *esdeveniments no controlables*. L'objectiu és dissenyar el que anomenen un *supervisor*, a partir d'un model a esdeveniments discrets de la planta i un conjunt d'especificacions de control. El supervisor té la capacitat d'impedir o autoritzar l'aconteixement d'esdeveniments controlables sobre el procés. És a dir, el sistema de control només pot generar els esdeveniments controlables que siguin autoritzats pel supervisor. D'altra banda, no es té cap control directe sobre els esdeveniments no controlables. Usualment, aquests esdeveniments s'originen en els sensors del procés físic. En canvi, els controlables són aquells que s'originen en el propi sistema de control.

En la metodologia que aquí es presenta, el model a obtenir només ha de servir per disposar d'una representació abstracta del procés que permeti conèixer en tot moment l'estat del procés. No s'espera que aquest model permeti tenir algun control sobre el mateix. En aquest sentit, es defineixen els següents conceptes:

Esdeveniment no controlable, $\sigma \in \Sigma_{NC}$. És aquell que s'origina en els senyals mesurats provinents dels sensors del procés i destinats al sistema de control.

Esdeveniment controlable, $\sigma \in \Sigma_C$. És aquell que s'origina en els senyals mesurats provinents del sistema de control i destinats als actuadors del procés.

Tant per uns com pels altres, es suposarà que no es té cap tipus de control sobre ells. Per tant s'utilitza la mateixa terminologia que en el control superior per comoditat, encara que amb una caracterització diferent. S'haurà de tenir en compte aquesta terminologia en la metodologia d'identificació que es presenta més endavant.

Un esdeveniment, o bé és controlable o bé és no controlable. Per tant, Σ_{NC} i Σ_C constitueixen un partició de Σ_{PRO} .

La distinció entre esdeveniments controlables i no controlables també es manifesta en el diagrama de transicions d'un generador. Una transició barrada indica que l'esdeveniment corresponent és controlable, mentre que un esdeveniment no controlable està representat per una transició sense marca (veure la fig. 3.6).



Figura 3.6: Notació d'una transició controlable σ_C i d'una no controlable σ_{NC} .

A continuació, es descriuen diverses entitats que intervindran en la metodologia de síntesi del generador, representant del conjunt esquematitzat en la fig. 3.5.

Component. És un element que forma part del procés real. Són components tant els actuadors, com els sensors, com altres elements del propi procés físic. Per exemple, un termoparell, un motor o bé una reacció química. Un procés està constituït per un o més components. Es suposa conegut el comportament lliure de cada component, podent-se

expressar mitjançant un generador. Així doncs, el component i està descrit pel generador $\mathcal{G}_{COM,i}$. El comportament lliure vol dir sense tenir en compte la interacció amb els altres components del procés. Components diferents s'associen a alfabetos diferents $\Sigma_{COM,i} \subseteq \Sigma_{PRO}$, no necessàriament disjunts. No s'estableix cap limitació sobre el número màxim de components que constitueixen un procés. No obstant això, es requereix que el conjunt d'alfabetos associats a tots ells cobreixin l'alfabet del procés. És a dir, $\bigcup_i \Sigma_{COM,i} = \Sigma_{PRO}$.

Restricció física. Vincula el comportament entre components degut a la física del procés. La convivència de diversos components en un mateix procés comporta un cert grau d'interacció entre ells. Aquesta interacció es tradueix en una alteració del comportament lliure dels components individuals. Per exemple, una tuberia estableix un vincle entre el cabal i la pèrdua de pressió. Una restricció física estableix les interaccions entre dos o més components. Es suposen conegudes totes les interaccions entre components, podent-se expressar mitjançant un o varis generadors. Així doncs, la restricció física j està descrita pel generador $\mathcal{G}_{RF,j}$. Es requereix que l'alfabet de la restricció física j cobreixi tots els esdeveniments associats als components $i \in [n, m] \subset \mathbb{N}^*$ que relaciona. És a dir, $\Sigma_{RF,j} = \bigcup_{i=n}^m \Sigma_{COM,i}$.

Restricció de control. Vincula el comportament entre components degut a les especificacions de control desitjades. La tasca del sistema de control és limitar el comportament del procés en llaç obert d'acord amb unes especificacions de control preestablertes. Per exemple, es desitja aturar un motor si s'arriba a una velocitat de rotació màxima. Per tant, el controlador acotará la interacció entre l'alimentació del motor i la mesura de la tacodinamo. Una restricció de control limita la interacció entre dos o més components. Es suposen conegudes totes les interaccions entre components limitades pel sistema de control, podent-se expressar mitjançant un o varis generadors. Així doncs, la restricció de control k està descrita pel generador $\mathcal{G}_{RC,k}$. Es requereix que l'alfabet de la restricció de control k cobreixi tots els esdeveniments associats als components $i \in [l, p] \subset \mathbb{N}^*$ que relaciona. És a dir, $\Sigma_{RC,k} = \bigcup_{i=l}^p \Sigma_{COM,i}$.

Planta. És el procés real controlat complert. Contempla la dinàmica dels components, de les restriccions físiques i de les restriccions de control. En alguns casos s'identifica també una planta en *llaç obert*, en la que no es contemplen les restriccions de control. El comportament de la planta s'expressa mitjançant un generador \mathcal{G}_{PLA} . L'alfabet de la planta es correspon amb l'alfabet del procés: $\Sigma_{PLA} = \Sigma_{PRO}$.

L'objectiu de la metodologia és obtenir el model \mathcal{G}_{PLA} , que descriu tots

els ordenaments acceptables d'esdeveniments significatius del procés, tenint en compte tant la física del procés com la intervenció del sistema de control. D'aquest model, no es té interès en avaluar propietats com el no blocatge o la seguretat. Per tant, en la representació mitjançant un generador no té cap interès el considerar estats finals. Per això, per qualsevol generador es considerarà $X_f = \emptyset$.

Aquesta metodologia es basa primer en la identificació dels components i de les restriccions, que descriuen parcialment el comportament del procés real. El model de la planta s'obté per composició de tots ells. La metodologia s'estructura en les 7 etapes següents:

1. Enumeració d'esdeveniments significatius.
2. Identificació i modelatge de tots els components.
3. Obtenció del model augmentat de la planta en llaç obert.
4. Identificació i modelatge de totes les restriccions físiques.
5. Obtenció del model de la planta en llaç obert.
6. Identificació i modelatge de totes les restriccions de control.
7. Obtenció del model de la planta en llaç tancat.

En cadascuna d'aquestes etapes serà necessari disposar d'un coneixement expert sobre el procés $\Xi[\mathcal{P}]$ (veure la def. 0.0.1). Principalment es concretarà en el coneixement de la dinàmica del procés \mathbb{D} : les interaccions i relacions entre els diversos senyals del procés \mathbb{S} . Aquest coneixement farà possible la identificació dels ordenaments admissibles d'esdeveniments significatius.

A continuació, s'expliquen en detall totes les etapes de la metodologia. El primer que cal fer és determinar quins són els esdeveniments significatius que interessa detectar. S'han d'escollir uns esdeveniments amb els que es pugui conèixer en tot moment l'estat del procés. Per tant, serà necessari un gran coneixement expert sobre el procés. Es suposa que aquests esdeveniments es poden detectar a partir dels senyals mesurats \mathbb{S} , mitjançant un pla de detecció $\mathfrak{B}\mathcal{D}$ adequat. Tots els esdeveniments significatius constitueixen el conjunt Σ_{PRO} .

Interessa identificar els esdeveniments que són controlables i els que són no controlables. Aquells que corresponen a senyals mesurats originats en els sensors són esdeveniments no controlables. Aquells que corresponen a senyals mesurats originats en el sistema de control i destinats als actuadors són esdeveniments controlables. Amb aquestes indicacions es poden constituir els conjunts Σ_C i Σ_{NC} .

A continuació, s'han d'identificar els diversos components del procés, així com els esdeveniments associats $\Sigma_{COM,i}$. Existeixen bàsicament dues tipologies de components: els *simples* i els *complexos*.

Component simple. Es correspon amb un sensor o un actuator. En general, cada sensor o actuator s'associa a un senyal. La metodologia de les finestres lliscants permet que es pugui dissenyar més d'un camí de detecció \mathcal{CD} d'un mateix senyal. Cada camí de detecció proporciona un conjunt d'esdeveniments significatius G_{S_S} ³. Un component simple s'associa només al conjunt d'esdeveniments significatius proporcionat per un sol camí de detecció, *i.e.*, $\mathcal{CD} \Rightarrow \Sigma_{COM,i} = G_{S_S}$.

Component complex. Es correspon amb elements del propi procés físic, que no siguin actuadors ni sensors. Per exemple, un motor, un dipòsit o bé una reacció química. La informació relativa a cadascun d'aquests elements s'ha d'obtenir a partir de l'anàlisi d'un o més senyals mitjançant un o més camins de detecció. Per exemple, el comportament d'un motor es pot expressar a partir de l'anàlisi de la tensió d'alimentació i del senyal d'una tacodinamo. Un component complex s'associa a un conjunt d'esdeveniments significatius obtinguts, en general, per un conjunt de camins de detecció, *i.e.*, $\{\mathcal{CD}_l\} \Rightarrow \Sigma_{COM,i} = \bigcup_l G_{S_{S,l}}$.

Un cop identificats els components i els alfabet associats, s'han de determinar quins ordenaments d'esdeveniments són admissibles i definir un llenguatge regular $L_{COM,i}$ per a cada component. A partir d'aquí, es pot determinar un generador finit $G_{COM,i}$ tal que $L(G_{COM,i}) = L_{COM,i}$ (veure la prop. 3.2.7). Tanmateix, moltes vegades resulta més pràctic descriure directament el comportament mitjançant un generador finit.

Convindrà tenir present la possibilitat que es produeixin discontinuïtats en la detecció d'esdeveniments, ja que poden conduir a l'alteració del comportament esperat. Les discontinuïtats s'han tractat en l'apartat 2.5. Allí es proposava l'opció d'interpol·lar esdeveniments, per tal de minimitzar aquesta problemàtica. Se'ns plantegen dues alternatives, davant la possibilitat que es puguin produir discontinuïtats:

- Si s'interpol·len esdeveniments, no cal que el model reconegui les possibles discontinuïtats.
- Si no s'interpol·len esdeveniments, cal que el model les reconegui.

En el cas d'un component simple, el model es construeix tenint en compte si la magnitud del senyal de sortida és ordenada o no, i si es considera que es poden produir discontinuïtats o no. Els estats del generador resultant s'associen a les magnituds qualitatives que pot assolir aquest senyal, mentre que les transicions s'associen als esdeveniments significatius.

En el cas d'esdeveniments associats a un sensor o a un actuator, el generador ha de reproduir el comportament físicament lògic del mateix. Tanma-

³Recordi's que S_S és el senyal de sortida obtingut mitjançant el camí d'anàlisi \mathcal{CA} que forma part del camí de detecció considerat.

teix, en el cas dels actuadors, també s'ha de tenir en compte el comportament imposat pel sistema de control que el governa.

Exemple 3.3.1: Amb la finalitat de detectar les tendències del senyal mesurat d'un sensor de nivell, es dissenya un camí de detecció de tendències (veure l'apartat 2.6.1). El senyal de sortida corresponent pot assolir tres magnituds qualitatives: *creixent c*, *estable e* o *decreixent d*. Es suposarà que no hi ha possibilitat de produir-se discontinuïtats. Els esdeveniments significatius s'indiquen a través de la següent matriu de canvis:

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

El generador que modela el comportament del sensor de nivell mitjançant aquests esdeveniments es mostra a la fig. 3.7.

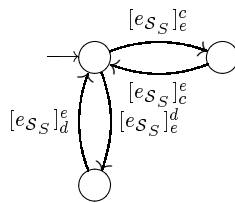


Figura 3.7: Diagrama de transicions corresponent a l'ex. 3.3.1.

El senyal de sortida corresponent és de magnitud ordenada. Cada estat del model indica que el senyal ha assolit una magnitud qualitativa diferent: l'inicial, que està *estable*, el de la dreta, que està *creixent*, i el inferior, que està *decreixent*. A més, reflexa precisament els canvis de magnitud que són acceptables d'acord amb el comportament físic del sensor de nivell. \diamond

Exemple 3.3.2: Pel mateix sensor de nivell de l'ex. 3.3.1, es dissenya un camí de detecció de tendències quasi idèntic. Suposi's ara que no és significatiu detectar quan el senyal s'estabilitza. En aquest cas, la matriu de canvis és la següent:

$$\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

El generador que descriu el comportament del sensor d'acord amb aquests esdeveniments significatius és ara el de la fig. 3.8.

En aquest cas, el model indica que el senyal és inicialment *estable*. L'estat de la dreta s'associa a que el senyal és *estable* o *creixent*, i el de l'esquerra

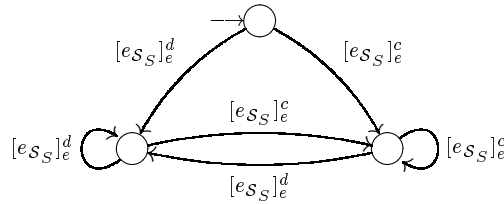


Figura 3.8: Diagrama de transicions corresponent a l'ex. 3.3.2.

a que el senyal és *estable* o *decreixent*. Aquesta imprecisió és conseqüència del conjunt d'esdeveniments significatius escollits. \diamond

Exemple 3.3.3: Novament, consideri's el sensor de nivell de l'ex. 3.3.1. Però ara, el camí de detecció de tendències s'ha dissenyat amb un senyal de sortida que assoleix només dues magnituds qualitatives: *creixent c* i *decreixent d*. D'altra banda, la matriu de canvis és:

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

El generador que descriu el comportament del sensor d'acord amb aquests esdeveniments significatius és ara el de la fig. 3.9.

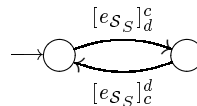


Figura 3.9: Diagrama de transicions corresponent a l'ex. 3.3.3.

En aquest cas, l'estat de la dreta s'associa a que el senyal està *creixent*, i el de l'esquerra a que el senyal està *decreixent*. El model reflecteix els canvis de magnitud que són acceptables d'acord amb el comportament físic del sensor de nivell. \diamond

En els exemples precedents es pot observar com el model depèn de com s'hagi dissenyat el camí de detecció, dels esdeveniments que es considerin significatius i del comportament físic del component associat.

En el cas de tractar-se d'un component complex, el model es construeix amb l'ajuda del coneixement expert sobre el mateix. El generador resultant ha de reflectir el comportament físicament lògic, tenint en compte la intervenció del sistema de control, si s'escau.

Un cop obtinguts els generadors corresponents a tots els components $G_{COM,i}$, es procedirà a la seva composició paral·lela, per obtenir el *model augmentat de la planta en llaç obert*: $\tilde{G}_{PLA} = G_{COM,1} || G_{COM,2} || \cdots || G_{COM,n}$. Es denomina model augmentat ja que resulta de la composició dels comportaments lliures dels components individuals: ara per ara no té en compte la interacció entre components.

La següent etapa de la metodologia consisteix en modelar les restriccions físiques mitjançant generadors $G_{RF,j}$. Aquestes han de caracteritzar les interaccions que sorgeixen entre components degut a fenòmens propis del procés físic. La física del procés regula l'admissibilitat dels esdeveniments no controlables. Per això, una restricció física s'expressa en base a impedir o permetre l'aconteixement d'esdeveniments no controlables.

Cada restricció física s'associa a dos o més components definits anteriorment. Això vol dir que cada restricció física ha d'establir els ordenaments que són admissibles físicament dels esdeveniments d'aquests components.

Existeixen dues formes d'obtenir un generador d'una restricció física:

- Partir del generador resultant de la composició dels components dels que es vol acotar el comportament. Llavors, coneixent el significat de cada estat es tracta d'eliminar transicions no controlables allí on convingui, i fer accessible el generador resultant.
- Partir del generador d'un dels components. Llavors, coneixent el significat de cada estat es tracta d'afegir transicions reflexives no controlables en determinats estats, allí on convingui.

Exemple 3.3.4: La velocitat de rotació ω d'un motor de c.c. es pot governar a partir de la tensió d'alimentació V . Es disposa d'una tacodinamo per mesurar aquesta velocitat. La fig. 3.10 mostra els generadors que modelen el comportament de la tensió d'alimentació $G_{COM,1}$ i de la tacodinamo $G_{COM,2}$, en funció de les tendències del senyal (*i.e.*, *creixent c* i *decreixent d*).

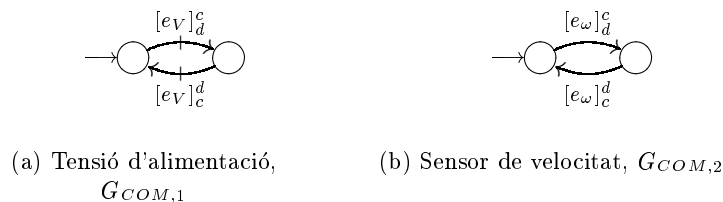


Figura 3.10: Generadors corresponents als components de l'ex. 3.3.4.

Es desitja modelar la restricció física que imposa el motor sobre ambdós components. Tal com s'ha indicat abans, això es pot aconseguir de dues

maneres.

D'una banda, la composició dels dos autòmats resulta en el generador de la fig. 3.11(a). Segons el comportament lògic del motor, la velocitat de rotació no pot augmentar si no ha augmentat abans la tensió d'alimentació. D'altra banda, la velocitat no pot disminuir a menys que la tensió hagi disminuït prèviament. Per això, interessa eliminar dues transicions. La fig. 3.11(b) mostra la restricció física resultant.

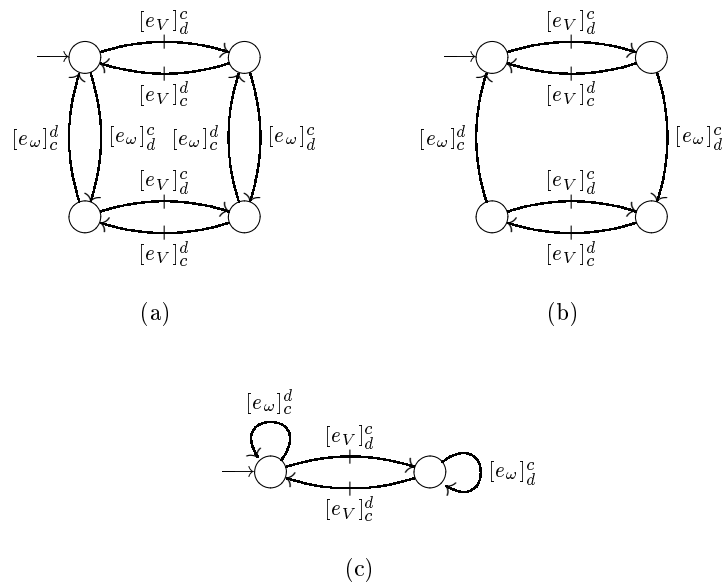


Figura 3.11: Generadors corresponents a les restriccions físiques de l'ex. 3.3.4.

D'altra banda, l'altre mètode per obtenir una restricció física consisteix en agafar el model de la tensió d'alimentació i afegir transicions reflexives associades als esdeveniments del sensor de velocitat. La fig. 3.11(c) mostra la restricció física resultant.

Ambdues restriccions són igualment vàlides, però només convindrà considerar-ne una, atès que són equivalents. \diamond

En determinades circumstàncies, una restricció física imposa una dinàmica que es podria assimilar al comportament d'un component més. Per exemple, la restricció física de l'ex. 3.3.4 podria interpretar-se com el model d'un motor. Per aquest motiu, a vegades una restricció física es pot confondre amb un component, i a l'inversa. Aquest fet no és important. El que és realment important és que la dinàmica que representa es tingui en compte, ja sigui com a component o bé com a restricció física.

La composició paral·lela del model augmentat de la planta en llaç obert \hat{G}_{PLA} amb les restriccions físiques $G_{RF,j}$ proporciona el *model de la planta en llaç obert*: $\bar{G}_{PLA} = \hat{G}_{PLA} || G_{RF,1} || G_{RF,2} || \dots || G_{RF,m}$. Normalment aquesta operació suposa una reducció en número d'estats i transicions del model augmentat de la planta en llaç obert.

A continuació, cal definir les restriccions de control $G_{RC,k}$ necessàries per incorporar les dinàmiques imposades pel sistema de control. El sistema de control modifica la dinàmica en llaç obert del procés amb l'objectiu d'acomplir certes especificacions de control. Aquesta alteració del comportament s'aconsegueix regulant l'admissibilitat dels esdeveniments controlables. Per això, una restricció de control s'expressa en base a impedir o permetre l'aconteixement d'esdeveniments controlables.

Cada restricció de control limita la interacció entre dos o més components definits anteriorment. Això vol dir que cada restricció de control ha d'establir els ordenaments que són admissibles pel sistema de control, dels esdeveniments d'aquests components.

Igual que passava amb les restriccions físiques, existeixen també dues formes d'obtenir un generador d'una restricció de control:

- Partir del generador resultant de la composició dels components dels que es vol acotar el comportament. Llavors, coneixent el significat de cada estat, es tracta d'eliminar transicions controlables allí on convingui, i fer accessible el generador resultant.
- Partir del generador d'un dels components. Llavors, coneixent el significat de cada estat, es tracta d'afegir transicions reflexives controlables en determinats estats, allí on convingui.

Exemple 3.3.5: Pel mateix motor de l'ex. 3.3.4 suposem que s'ha modelat igualment el comportament de la tensió d'alimentació en funció de les tendències del senyal (veure la fig. 3.10(a)). No obstant, s'ha decidit modelar la velocitat del motor en funció de si aquesta ha superat o no un determinat llindar de velocitat màxima: *normal* n i *alarma* a . El generador resultant es mostra en la fig. 3.12.

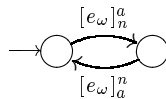


Figura 3.12: Generador corresponent al sensor de velocitat de l'ex. 3.3.5.

Es desitja modelar la següent restricció de control, que imposa el sistema de control al motor: en cas de superar la velocitat màxima, el sistema de

control ha de reduir la tensió d'alimentació. Tal com s'ha indicat abans, això es pot aconseguir de dues maneres.

D'una banda, la composició dels dos components resulta en el generador de la fig. 3.13(a). Segons l'especificació de control desitjada, amb la velocitat per sobre del màxim, el sistema de control mai ordenarà que la tensió d'alimentació augmenti. Per això, interessa eliminar una transició. La fig. 3.13(b) mostra la restricció de control resultant.

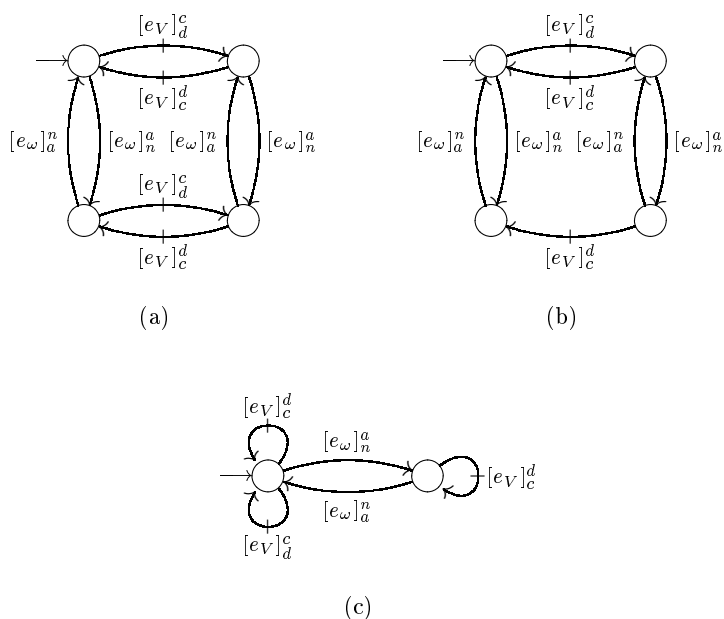


Figura 3.13: Generadors corresponents a les restriccions de control de l'ex. 3.3.5.

D'altra banda, l'altre mètode per obtenir una restricció de control consisteix en agafar el model del sensor de velocitat i afegir transicions reflexives associades als esdeveniments de la tensió d'alimentació. La fig. 3.13(c) mostra la restricció de control resultant.

Ambdues restriccions són igualment vàlides, però només convindrà considerar-ne una, atès que són equivalents. \diamond

De fet, les restriccions de control només expressen algunes de les especificacions de control: aquelles que es poden descriure fàcilment en base a l'ordenament d'esdeveniments. Altres especificacions de control basades en criteris d'optimització o temporals, com un algorisme PID, per exemple, no es podran descriure mitjançant una especificació de control. Per tant, aquests criteris de control no podran acotar el comportament de la planta.

La composició paral·lela del model de la planta en llaç obert \bar{G}_{PLA} amb les restriccions de control $G_{RC,k}$ proporciona el *model de la planta en llaç tancat*: $G_{PLA} = \bar{G}_{PLA} || G_{RC,1} || G_{RC,2} || \dots || G_{RC,p}$. S'arriba així al final de la metodologia que ha de permetre obtenir un model a esdeveniments discrets del procés.

En definitiva, és molt important que tot el coneixement expert que es tingui sobre el procés quedi incorporat en el model final. Aquest coneixement fa referència a dos aspectes:

- definir un alfabet Σ_{PRO} realment significatiu, que permeti conèixer l'estat del procés, i
- expressar tots els ordenaments que siguin significatius i lògics.

Amb aquestes premisses, s'obtindrà un bon model, en que l'assoliment d'un estat permetrà conèixer l'estat de funcionament del procés real. Per exemple, es podrà saber si una determinada reacció química està en una fase o una altra, si hi ha hagut alguna fallada en el procés, si s'ha arribat a una situació de perill, si el funcionament és normal, etc. Com més coneixement expert es tingui *a priori* sobre aquests estats i s'hagi incorporat al model, més útil serà aquest model de cara a identificar-los posteriorment en el procés real, sense l'ajuda de l'expert.

Finalment, cal dir que l'entorn informàtic de desenvolupament que es suggereix per treballar amb models a esdeveniments discrets es descriu amb més detall al Capítol 4: com a programari de desenvolupament i disseny es proposa treballar amb CTCT (veure l'apartat 4.1.3), i com a programari d'implementació i execució, STATEFLOWTM (veure 4.1.2).

La necessitat de treballar amb dues aplicacions informàtiques es deu a la impossibilitat d'expressar la composició paral·lela amb els Statecharts (veure el comentari de l'Apèndix D). Això fa que sigui necessari utilitzar el CTCT per obtenir el model a esdeveniments discrets de la planta a partir de la composició paral·lela dels models dels components i de les restriccions. Posteriorment, el model resultant s'implementa en STATEFLOWTM.

Una proposta de millora seria estendre la descomposició *AND* dels Statecharts per tal que permetés sincronitzar autòmats finits segons les regles de la composició paral·lela. La incorporació d'aquesta modificació en STATEFLOWTM permetria simplificar el procés de modelatge. No seria necessari obtenir el model DES de la planta per combinació dels components, sinó que simplement caldria introduir els models de tots els components i les restriccions directament com a descomposició *AND* de la planta. Llavors, STATEFLOWTM tindria la capacitat de simular el comportament de la planta a partir del funcionament síncron dels components. Això simplificaria notablement la metodologia de disseny d'un model DES del procés, ja que estalviaria l'ús del CTCT.

3.4 Exemple de modelatge d'un sistema de dipòsits pulmó

En aquest apartat s'il·lustra la metodologia d'identificació d'un model a esdeveniments discrets aplicada al procés de la fig. 3.14.

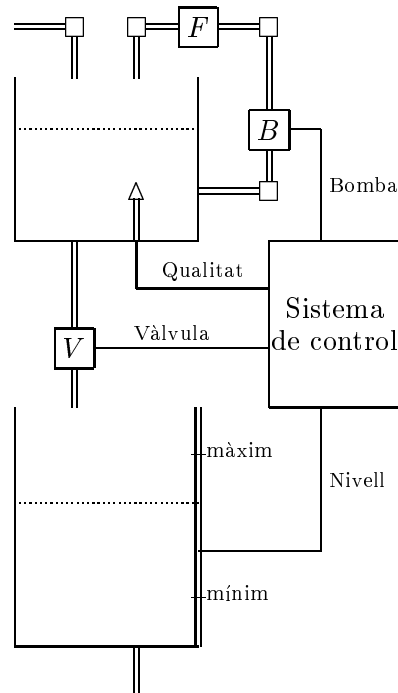


Figura 3.14: Sistema de dipòsits pulmó

Aquest procés consisteix en un sistema de dipòsits pulmó per l'alimentació d'una determinada indústria química. Es suposa que l'alimentació del dipòsit superior sempre està garantida, de forma que mai es pot quedar buit. D'altra banda, la canalització d'evacuació del dipòsit inferior sempre està oberta.

En funcionament normal el sistema de control regula el nivell del dipòsit inferior d'acord amb una consigna determinada. Existeix un sensor N que proporciona una mesura contínua del nivell del dipòsit. El nivell es controla actuant sobre la vàlvula tot-o-res V : quan està oberta el dipòsit s'omple, i quan està tancada el dipòsit es buida. És a dir, amb la vàlvula oberta el cabal d'entrada sempre és superior al cabal d'evacuació.

Interessa que el nivell sempre es mantingui entre un mínim i un màxim. Si en algun moment es supera el nivell màxim de seguretat, el sistema de control ordena el tancament de la vàlvula, a fi d'impedir el vessament del

líquid a l'exterior. En canvi, en cas de baixar per sota el nivell mínim de seguretat, el sistema de control ordena l'obertura de la vàlvula, a fi d'impedir que el dipòsit es buidi i deixi d'alimentar la indústria.

D'altra banda, el sistema de control també vetlla per la qualitat del líquid d'alimentació. Un sensor C proporciona una mesura contínua de la qualitat del líquid en funció de la concentració de poluents en el medi. Si supera un determinat llindar superior, vol dir que la qualitat és dolenta i s'ha d'iniciar un procediment d'emergència. Aquest procediment consisteix primer en assegurar que la vàlvula V estigui tancada. Així s'evita l'alimentació de la indústria amb líquid de baixa qualitat. Després, s'inicia la depuració del líquid, fent-lo recircular a través d'un filtre F mitjançant una bomba B . Tant bon punt la concentració de poluents disminueix per sota d'un llindar inferior, es continua amb el procediment de funcionament normal.

Es considera que inicialment el dipòsit està buit, el líquid en bon estat, la vàlvula oberta i la bomba aturada.

Tenint en compte que es disposa de les mesures del sensor de nivell N , del sensor de qualitat C , de l'accionament de la bomba B i de l'accionament de la vàlvula V , es tracta d'obtenir un model a esdeveniments discrets que permeti determinar en tot moment l'estat de funcionament del procés. Aquest model es sintetitzarà seguint pas per pas la metodologia abans exposada.

Primer de tot, cal decidir quins esdeveniments són significatius per a cada senyal disponible. Tots els senyals són similars, en el sentit que el fet significatiu és franquejar un determinat valor en un sentit o l'altre.

- Pel senyal provinent del sensor de nivell és significatiu el franqueig en ambdós sentits dels nivells màxim i mínim de seguretat. S'identifiquen, per tant, quatre esdeveniments significatius: NSa quan supera el nivell màxim en sentit ascendent, NSd quan supera el nivell màxim en sentit descendent, NIa quan supera el nivell mínim en sentit ascendent i $NI d$ quan supera el nivell mínim en sentit descendent.
- Amb el senyal provinent del sensor de qualitat s'identifiquen dos esdeveniments significatius: Ca quan la concentració augmenta més enllà del llindar superior i Cb quan disminueix per sota del llindar inferior.
- Amb la vàlvula s'identifiquen dos esdeveniments: Vo obrir la vàlvula (quan el senyal passa a tenir un valor alt) i Vt tancar la vàlvula (quan el senyal passa a tenir un valor baix).
- Igualment amb el senyal provinent de la bomba s'identifiquen dos esdeveniments significatius: Ba bomba activada (quan el senyal passa a tenir un valor alt) i Bd bomba desactivada (quan el senyal passa a tenir un valor baix).

Suposarem que s'ha aplicat la metodologia de les finestres lliscants per dissenyar i configurar els quatre camins de detecció que proporcionen aquests

esdeveniments (un per a cada senyal mesurat). Només s'apuntarà que es basen en un camí d'anàlisi constituït per un sol analitzador estadístic que calcula l'atribut *moda*. S'ha utilitzat el bloc Detector de canvis de quantificació de la llibreria ABSALON (veure l'apartat 4.1.1). També convé aclarir que s'ha utilitzat una notació simplificada pels esdeveniments. En definitiva, es disposa de 9 esdeveniments significatius del procés $\Sigma_{PRO} = \{NSa, NSd, NIa, NId, Ca, Cb, Vo, Vt, Va, Vd\}$.

D'altra banda, coneixent l'origen de cada senyal és fàcil distingir els esdeveniments en controlables i no controlables. Els esdeveniments obtinguts a partir dels senyals governats pel sistema de control són controlables $\Sigma_C = \{Vo, Vt, Ba, Bd\}$. En canvi, els esdeveniments obtinguts a partir dels senyals associats a sensors del procés són no controlables $\Sigma_{NC} = \{NSa, NSd, NIa, NId, Ca, Cb\}$.

S'identifiquen 4 components, un associat a cada senyal mesurat. La fig. 3.15 mostra els 4 generadors corresponents. Tots reflecteixen una dinàmica similar basada en la lògica del procés controlat real. Observi's que cada component considera un subconjunt disjunt d'esdeveniments de Σ_{PRO} .

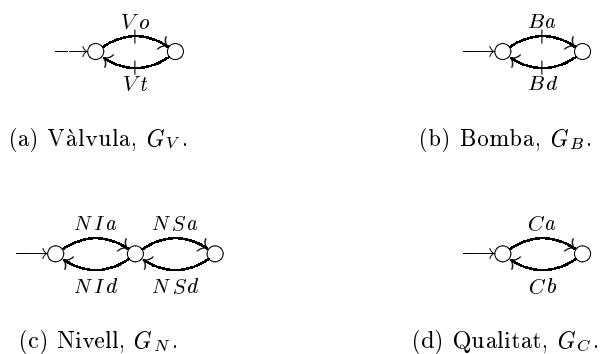


Figura 3.15: Generadors dels components.

La composició paral·lela dels quatre components proporciona un model augmentat de la planta en llaç obert \hat{G}_{PLA} de 24 estats i 104 transicions. No es mostra gràficament degut a la seva complexitat.

Però aquest generador reflecteix certa dinàmica que no és coherent amb el procés físic. De cara a eliminar aquestes incoherències, és necessari incorporar les dues restriccions físiques de la fig. 3.16.

La restricció física de la fig. 3.16(a) afecta a dos components: la vàlvula i el sensor de nivell. L'objectiu és acotar el comportament conjunt d'ambdós components, d'acord amb els següents criteris:

- Només es poden franquejar en sentit ascendent el nivell màxim o mínim de seguretat quan la vàlvula està oberta, perquè és quan el nivell del

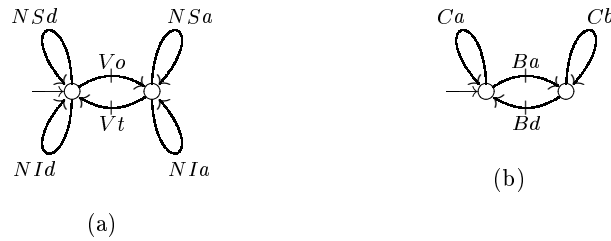


Figura 3.16: Generadors de les restriccions físiques.

líquid augmenta.

- Només es poden franquejar en sentit descendent el nivell màxim o mínim de seguretat quan la vàlvula està tancada, perquè és quan el nivell del líquid disminueix.

Aquesta restricció ha estat sintetitzada afegint al generador de la vàlvula (veure la fig. 3.15(a)) quatre transicions reflexives no controlables associades als esdeveniments del sensor de nivell.

Observi's que el generador d'aquesta restricció es podria interpretar com el model del comportament del dipòsit inferior. És a dir, es podria haver considerat aquest generador com el model d'un component més. Si així hagués estat, llavors ja no s'hauria considerat com una restricció física.

D'alta banda, la restricció física de la fig. 3.16(b) afecta a la bomba i el sensor de concentració. L'objectiu és acotar el comportament conjunt d'ambdós components d'acord amb els següents criteris:

- La qualitat del líquid només pot empitjorar quan la bomba de recirculació està aturada, és a dir, en funcionament normal.
- L'única forma de regenerar la qualitat del líquid és activant la bomba de recirculació.

Aquesta restricció ha estat sintetitzada afegint al generador de la bomba (veure la fig. 3.15(b)) dues transicions reflexives no controlables associades a esdeveniments del sensor de qualitat.

En aquest cas, es podria haver interpretat que aquest generador caracteritza la dinàmica d'alteració de la qualitat del líquid. Per tant, podia haver estat considerat un altre component.

La composició de \hat{G}_{PLA} amb les dues restriccions físiques proporciona un generador de 24 estats i 76 transicions. Aquest generador representa el model de la planta en llaç obert, \hat{G}_{PLA} . La imposició de les restriccions físiques ha acotat la dinàmica de \hat{G}_{PLA} , traduint-se en una disminució del número de

Aquesta restricció ha estat sintetitzada afegint al generador de la vàlvula (veure la fig. 3.15(a)) dues transicions reflexives controlables associades a esdeveniments de la bomba.

La restricció de control de la fig. 3.17(c) afecta a dos components: el sensor de qualitat i la bomba. L'objectiu és acotar el comportament conjunt d'ambdós components d'acord amb les següents especificacions de control:

- Només s'activarà la bomba de recirculació quan la concentració de contaminants sigui inacceptablement alta.
- No es desactivarà la bomba de recirculació fins que la concentració de contaminants hagi disminuït per sota d'un nivell acceptable.

Aquesta restricció ha estat sintetitzada afegint al generador del sensor de qualitat (veure la fig. 3.15(d)) dues transicions reflexives controlables associades a esdeveniments de la bomba.

La restricció de control de la fig. 3.17(d) afecta a tres components: el sensor de nivell, el sensor de qualitat i la bomba. L'objectiu és acotar el comportament conjunt dels tres components d'acord amb les següents especificacions de control:

- Mentre el nivell es manté entre el mínim i el màxim, la vàlvula es pot obrir i tancar d'acord amb l'algorisme de control programat (*e.g.*, un control PID).
- Si es supera el nivell màxim de seguretat, s'ha de tancar la vàlvula fins a recuperar un nivell del líquid que sigui acceptable.
- Si el nivell baixa per sota del mínim, s'ha d'obrir la vàlvula fins a recuperar un nivell de líquid que sigui acceptable.
- En qualsevol moment, si la qualitat del líquid es degrada i la concentració de poluents supera el llindar màxim, s'ha de tancar la vàlvula fins a recuperar una concentració acceptable.
- En cas de conflicte, l'especificació anterior preval per sobre de les altres.

Aquesta darrera especificació fa referència a la situació que es crea quan simultàniament el nivell està per sota del mínim i la qualitat del líquid esdevé inacceptable. Llavors, el sistema de control tancarà la vàlvula per evitar que passi líquid de mala qualitat cap a la indústria.

La restricció de la fig. 3.17(d) ha estat sintetitzada en dues etapes. Primer s'ha fet la composició d'ambdós sensors (veure fig. 3.15(c) i 3.15(d)). Tot seguit s'han afegit set transicions reflexives controlables associades a esdeveniments de la vàlvula.

Totes aquestes restriccions responen al comportament que imposen els algorismes amb que el sistema de control ha estat programat.

A l'hora de concebre les restriccions, siguin físiques o de control, interessa destacar un aspecte prou important respecte a les transicions reflexives. Afegir una transició reflexiva en un estat implica que s'està autoritzant a la detecció, en aquell estat, de l'esdeveniment corresponent. Però a la vegada, s'està impedit la detecció d'aquell esdeveniment en aquells estats en que no aparegui una transició reflexiva associada a ell.

La composició de \bar{G}_{PLA} amb les quatre restriccions de control proporciona un generador de 18 estats i 34 transicions. Aquest generador representa el model de la planta en llaç tancat, G_{PLA} . En aquest cas, la imposició de les restriccions de control ha acotat la dinàmica de \bar{G}_{PLA} , traduint-se en una disminució del número de transicions. En altres casos, pot ser que no sigui així, tot dependrà de les especificacions de control desitjades. El generador que modela la planta s'il·lustra en la fig. 3.18.

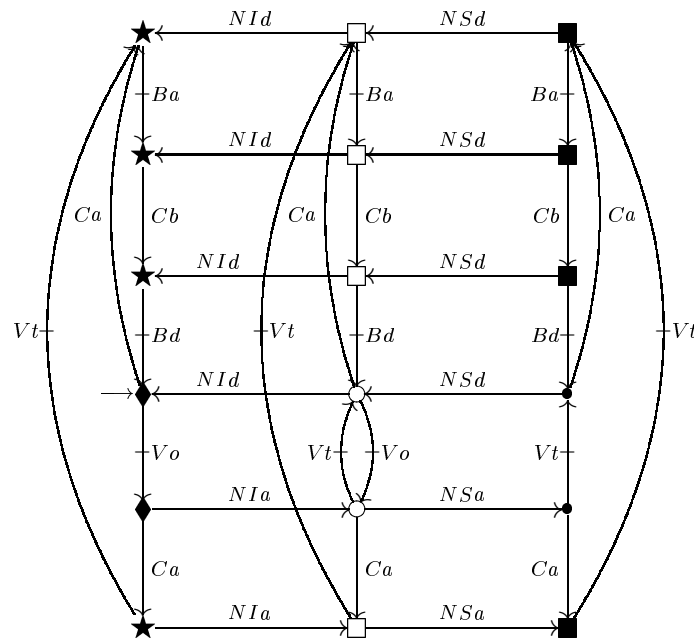


Figura 3.18: Generador de la planta en llaç tancat.

Aquest model permet identificar sis estats de funcionament del procés:

- *Funcionament normal*, en els estats simbolitzats amb \circ . El sistema de control intenta que el nivell del dipòsit segueixi un nivell de referència.
- *Funcionament de seguretat Nivell Màxim*, en els estats simbolitzats amb \bullet . El sistema de control tanca la vàlvula, per a que el nivell del dipòsit baixi per sota del llindar màxim de seguretat.

- *Funcionament de seguretat Nivell Mínim*, en els estats simbolitzats amb \blacklozenge . El sistema de control obre la vàlvula, per a que el nivell del dipòsit pugi per sobre del llindar mínim de seguretat.
- *Funcionament d'emergència*, en els estats simbolitzats amb \square . El sistema de control tanca la vàlvula si no ho està ja, i inicia el procés de filtratge, fins a recuperar una concentració del líquid baixa en poluents.
- *Funcionament d'emergència Nivell Màxim*, en els estats simbolitzats amb \blacksquare . El sistema de control tanca la vàlvula si no ho està ja, i inicia el procediment de reciclatge, a l'espera de recuperar la concentració baixa en poluents o que el nivell del dipòsit baixi per sota del llindar màxim de seguretat.
- *Funcionament d'emergència Nivell Mínim*, en els estats simbolitzats amb \star . El sistema de control tanca la vàlvula si no ho està ja, i inicia el procediment de reciclatge, a l'espera de recuperar la concentració baixa en poluents.

S'ha realitzat una implementació d'aquest model en STATEFLOWTM, tal com es mostra en la fig. 3.19.

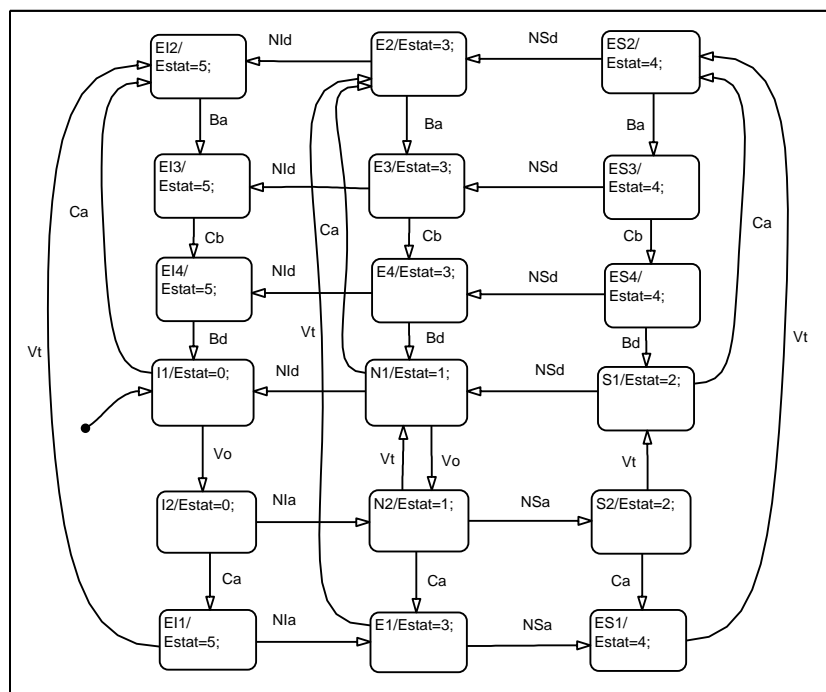
Aquest model s'ha integrat en SIMULINK[®] per constituir un SIS que permeti identificar l'estat de funcionament del sistema de dipòsits pulmó. La fig. 3.20 mostra l'esquema implementat, que recorda l'arquitectura presentada en l'apartat d'objectius de la Tesi (veure fig. 5). Es veuen clarament 4 blocs: un bloc que simula el procés físic real, un bloc que simula el sistema de control, un bloc que serveix com a interfície detectora d'esdeveniments, i un bloc supervisor que es correspon amb el model a esdeveniments discrets de la fig. 3.19.

La implementació en SIMULINK[®] ha permès realitzar la simulació mostrada a la fig. 3.21.

El sistema de control té l'objectiu de regular el nivell del dipòsit seguint la consigna que es mostra a la gràfica. S'ha simulat un control tot-o-res. Aquesta és la raó per la que el nivell segueix una dinàmica en dents de serra al voltant de la consigna. En dues ocasions, s'ha simulat l'empitjorament de la qualitat de líquid. En ambdós casos, el sistema de control tanca la vàlvula i posa en marxa la bomba de recirculació per a filtrar el líquid.

En el gràfic *Consigna & Nivell*, les línies discontinües marquen els nivells màxim (7) i mínim (1) de seguretat permesos. D'altra banda, en el gràfic *Qualitat*, les línies discontinües marquen els nivells màxim (2) i mínim (1) de concentració de poluents permesos.

Observi's com el supervisor permet determinar en tot moment l'estat de funcionament del procés real: de seguretat Nivell Mínim (0), normal (1), de seguretat Nivell Màxim (2), d'emergència (3), d'emergència Nivell Màxim (4) i d'emergència Nivell Mínim (5).



Printed 15-Sep-2001 18:55:13

Figura 3.19: Model de la planta en STATEFLOW™.

Inicialment, el dipòsit està buit. Per això, l'estat inicial és el de seguretat Nivell Mínim. Un cop s'ha superat el llindar inferior, s'inicia el funcionament normal amb un control tot-o-res per tal de seguir la consigna, fixada a 5.

Al voltant d'una hora i quart després d'haver-se iniciat la simulació, la concentració de poluents ha superat el nivell màxim permès, i s'ha activat el funcionament d'emergència. Observi's que durant el procés de filtratge, el nivell del dipòsit va decreixent. Això és degut a que la vàlvula es manté tancada. Un cop recuperada la qualitat desitjada, es retorna al funcionament normal, i el nivell de líquid segueix novament la consigna.

Després d'un canvi de consigna a 2, a les tres hores i mitja, la qualitat s'ha degradat altre cop, i s'inicia automàticament l'estat d'emergència. En aquesta ocasió, mentre s'està filtrant el líquid, el nivell del dipòsit inferior baixa per sota del mínim. Tot i això, la vàlvula es manté tancada, per evitar l'alimentació de la indústria amb líquid de mala qualitat. Però, aquest fet posa en perill la indústria, ja que pot patir una manca d'alimentació. Afortunadament, aquesta situació crítica pot ser identificada pel supervisor

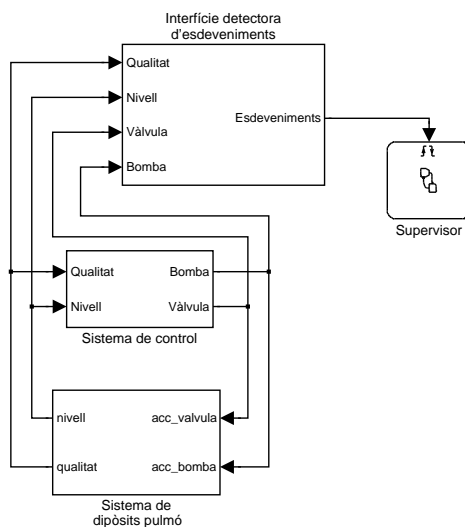


Figura 3.20: Implementació en SIMULINK[®] del SIS.

mitjançant l'assenyalament de l'estat de funcionament d'emergència Nivell Mínim. Un cop notificat a l'operador, aquest pot prendre la decisió que més convingui per tal d'evitar el mal funcionament de la maquinària de la indústria. Per exemple, pot aturar el procés productiu, o pot posar en marxa un sistema d'alimentació alternatiu.

Un cop el líquid torna a tenir una qualitat acceptable, el sistema de control obre la vàlvula per recuperar un nivell per sobre del mínim. En aquest moment es retorna novament a l'estat de funcionament normal, seguint la consigna a 2.

En resum, aquest exemple ha permès il·lustrar la metodologia proposada, i obtenir un model a esdeveniments discrets amb el que implementar el supervisor del SIS.

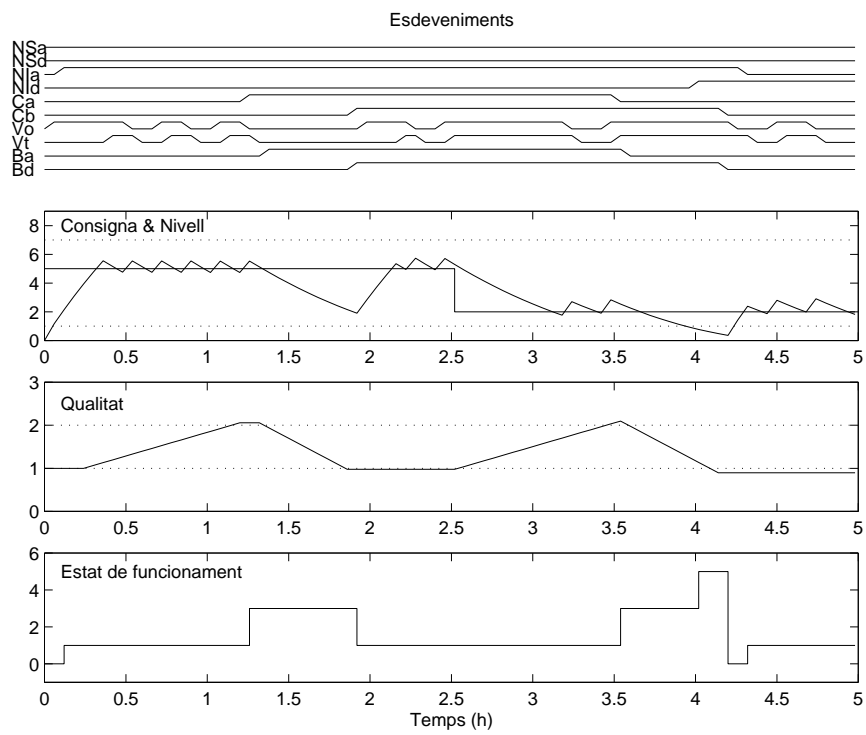


Figura 3.21: Resultats de simulació.