

Diagnosis de Sistemas Dinámicos: Un Modelo de Conocimiento que Permite Integrar la Evolución del Sistema en el Proceso de Diagnosis*

Diagnosis of Dynamic Systems: A Knowledge Model that Allows to Integrate System Evolution During the Diagnosis Process

Carlos J. Alonso González, César Llamas Bello, José A. Maestro Prieto y Belarmino Pulido

Grupo de Sistemas Inteligentes, Departamento de Informática
ETSI Informática, Universidad de Valladolid
Plaza de Santa Cruz No. 8, E-47002, Valladolid, España
E-mail: {calonso, cllamas, jose, belar}@infor.uva.es

Artículo recibido en Marzo 10, 2002; aceptado en September 28, 2002

Resumen

Este trabajo presenta un modelo basado en conocimiento para la diagnosis en línea de sistemas dinámicos complejos. El conocimiento del dominio se modela mediante un grafo causal que incluye relaciones temporales entre síntomas y causas. El conocimiento de inferencia y tarea, necesario para la obtención del diagnóstico se modela utilizando la metodología COMMONKADS. La principal novedad de la propuesta es que, a partir de modelos causales y temporales sencillos, permite diseñar una tarea de diagnosis que es capaz de examinar la evolución del sistema integrando la nueva información disponible en el proceso de diagnosis. En este trabajo, la diagnosis se concibe como una tarea más a realizar en el contexto de un sistema de supervisión global, de forma que la red causal que soporta el diagnóstico puede seleccionarse en función de la configuración y del punto de operación. La propuesta está diseñada para la diagnosis de procesos industriales continuos, pero el modelo que se proporciona es independiente del dominio y puede ser utilizado para diagnosticar otros tipos de sistemas dinámicos.

Palabras clave: Modelo de Conocimiento, Diagnosis basada en conocimiento, Diagnosis de sistemas dinámicos, Supervisión.

Abstract

A knowledge based model for on line diagnosis of complex dynamic systems is proposed. Domain knowledge is modelled via causal networks which consider temporal relationships among symptoms and causes. Inference and task knowledge is included using the COMMONKADS methodology. The main feature of the proposal is that the diagnosis task is able to track the evolution of the system incorporating new symptoms to the diagnosis process. Diagnosis is conceived as a task to be carry out by a supervisory systems, which could select the current causal network to perform diagnosis, depending on system configuration and operation point. Although the proposal has been design for diagnosis of industrial continuous processes, the knowledge model is domain independent and may support the diagnosis of different kinds of dynamic systems.

Keywords: Knowledge model, Knowledge based diagnosis, Diagnosis of dynamic systems, Supervision.

*Este trabajo ha sido financiado por el Proyecto CICYT TAP99-0344.

1 Introducción

Durante las dos últimas décadas, la diagnosis de sistemas dinámicos ha sido abordada desde distintas aproximaciones y con distintos grados de éxito: diagnosis basada en modelos desde la perspectiva de la teoría de sistemas dinámicos [Chen y Patton, 1999], diagnosis basada en modelos desde la perspectiva de la inteligencia artificial [Dressler, 1996] o diagnosis mediante sistemas basados en conocimiento [Cauvin y otros, 1998]. Sin embargo, ninguna de estas aproximaciones resuelve satisfactoriamente el problema de la diagnosis de sistemas dinámicos complejo, que aún es un problema abierto. Es por ello que los sistemas basados en conocimiento siguen siendo una aproximación razonable [Balakrishnan y Honavar, 1998], especialmente para abordar problemas complejos para los que existe un cuerpo de conocimiento experto.

Price en [Price, 1999], proporciona una síntesis actual de la metodología de la aproximación de los sistemas basados en conocimiento a la diagnosis de sistemas dinámicos, al mismo tiempo que pone de manifiesto una de las motivaciones de este trabajo: la mayoría de los sistemas basados en conocimiento eluden el aspecto dinámico del problema, realizando el diagnóstico bajo petición del usuario y analizando los datos disponibles hasta el momento en que se realiza la petición.

Sin embargo en numerosos sistemas dinámicos, como son las plantas industriales continuas, se precisan técnicas que monitoricen continuamente el funcionamiento del sistema, iniciando automáticamente un proceso de diagnosis tan pronto como se detecte alguna anomalía y, posiblemente, examinando la evolución del sistema a partir de la detección, como se señala en [Price, 1999] o [Acosta y otros, 2002]. Aunque existen algunas propuestas de sistemas basados en conocimiento que aportan algunas de estas capacidades, como SIDIA y MUDIA [Guckenbiehl y Schäfer-Richter, 1992] o MIDAS [Oyeleye y otros, 1990], estos sistemas no han alcanzado la etapa de producción. Excepciones notables son los sistemas basados en "crónicas" [Cordier y Dousson, 2000] o las distintas aproximaciones de la "generación OLID" [Alonso y otros, 1998, 2001]. Ambas familias de sistemas son capaces

de integrar nueva información en una tarea de diagnóstico ya iniciada, utilizando modelos temporales sencillos que permiten examinar, en línea, la evolución del sistema.

En este artículo se propone un modelo de conocimiento para la diagnóstico de sistemas dinámicos inspirado en la generación OLID, que puede considerarse como una generalización y formalización del trabajo presentado en [Alonso y otros, 2001]. El modelo propuesto define una estrategia de diagnóstico genérica especificando el conocimiento de tarea e inferencia de acuerdo a la metodología COMMON-KADS [Schreiber y otros, 1999]. El conocimiento específico del dominio se modela mediante una red causal extendida con información temporal.

Además, la diagnóstico se concibe como una tarea más de un sistema de supervisión global, como se indica en [Acosta y otros, 2002], de modo que la tarea de diagnóstico puede adaptarse a las condiciones de trabajo de la planta, reflejadas en el Protocolo de Operación con el que se gobierna la instalación industrial. Esto puede conseguirse manteniendo la misma estrategia de diagnóstico sin más que seleccionar la red causal apropiada para el Protocolo de Operación activo.

Antes de proceder a la presentación del modelo, profundizaremos en la interpretación de la diagnóstico como una tarea de supervisión. Seguidamente presentaremos el modelo de conocimiento, comenzando por el conocimiento de dominio que soporta la diagnóstico, para introducir a continuación las sub tareas en que se divide aquella, diseñadas para que la tarea de diagnóstico pueda examinar de forma activa la evolución del sistema durante el proceso de diagnóstico. El trabajo finaliza discutiendo las principales aportaciones de la propuesta.

2 La diagnóstico como una tarea de supervisión

La diagnóstico de un sistema de mediana o gran complejidad no puede concebirse como una tarea aislada que podamos invocar en cualquier circunstancia. En general, cualquier sistema de diagnóstico, con independencia de la técnica de diagnóstico que emplee, recurrirá a una serie de hipótesis simplificadores que faciliten la tarea de diagnóstico. Consecuentemente, antes de invocar un proceso de diagnóstico es preciso conocer la situación en que se encuentra el sistema a analizar, para saber si las hipótesis de trabajo, generalmente implícitas en el sistema de diagnóstico, se satisfacen.

Este proceso de situación en contexto de la tarea de diagnóstico puede abordarse considerando la diagnóstico como una parte integrante de una tarea de supervisión global de un sistema dinámico. El concepto de supervisión global en el que vamos a encuadrar la tarea de diagnóstico obedece a la propuesta desarrollada en [Acosta y otros, 2002], donde se introducen ocho tareas para la supervisión en línea de procesos continuos: Determinación del estado (de la planta), Monitorización, Diagnóstico de fallos, Modos de operación, Pronosis,

Planificación, Interfaz hombre-máquina y Validación de datos. Para situar en contexto la diagnóstico, basta con considerar las cuatro primeras tareas: Determinación del estado, Monitorización, Diagnóstico de fallos y Modos de operación. Estas cuatro tareas podrían considerarse el núcleo de un sistema supervisor global. Estas tareas están diseñadas específicamente para el control de procesos continuos, aunque creemos que son susceptibles de extrapolación a la supervisión de sistemas dinámicos continuos.

Esta taxonomía recurre a los conceptos de estado de la planta y protocolo de operación, que introducimos brevemente parafraseando a [Alonso y otros, 2001]. El estado de la planta se define como un vector que describe las propiedades de la materia prima disponible, las propiedades deseadas del producto final y las restricciones físicas o tecnológicas a medio o largo plazo. Vamos a considerar que un estado es válido si las restricciones actuales permiten al sistema obtener el producto final deseado a partir de la materia prima disponible. Cada estado válido suele tener asociado un conjunto de protocolos de operación bien establecidos, que permiten obtener el producto deseado. Estos protocolos de operación fijan aspectos como la interconexión de componentes del sistema físico, la funcionalidad de las unidades básicas, las leyes de control y los puntos de consigna de las variables de interés. La selección del protocolo de operación para un estado válido puede depender de las restricciones activas.

Una vez introducidos los conceptos de estado de la planta y protocolo de operación, podemos describir sucintamente las cuatro tareas básicas del sistema supervisor:

Determinación del estado Se define como la tarea que establece el estado actual de la planta y su protocolo de operación.

Monitorización Dado un conjunto de variables, la monitorización identifica todas aquellas variables cuya trayectoria se aparta de la trayectoria deseada.

Modo de Operación Esta tarea comprueba que la planta se gestiona de acuerdo con el protocolo de operación seleccionado.

Diagnóstico de fallos Esta tarea localiza los componentes que fallan. Adicionalmente, puede identificar las causas del fallo.

La tarea de Determinación del Estado sería la tarea responsable de situar en contexto el sistema supervisor, determinando el estado y el protocolo deseados. Esta tarea es, probablemente, la más difícil de automatizar; pero es posible diseñar sistemas de ayuda a la toma de decisión que, dejando la decisión final en manos del personal responsable, faciliten el desarrollo de la tarea e informen al resto de tareas de las decisiones adoptadas. La tarea de Monitorización se ha desligado de la diagnóstico por varias razones, como que algunas variables van a ser monitorizadas porque la tarea Modo de Operación lo requiere o para permitir inyectar conocimiento adicional en la etapa de detección. La tarea Modo de Ope-

ración se ocupa de aquellos problemas relacionados con una operación errónea del sistema. La tarea de Modo de Operación ha de identificar las posibles violaciones del Protocolo de Operación. Este concepto es similar al que algunos autores, por ejemplo [Cauvin y otros, 1998], denominan fallos externos, a pesar de que no se corresponden con el concepto habitual de fallo.

El modelo de conocimiento que proponemos para la tarea de diagnóstico asume que conocemos el Protocolo de Operación del sistema que estamos diagnosticando, que disponemos de una tarea de Monitorización que vigila el sistema de forma continua y de una tarea de Modos de Operación. La tarea de diagnóstico será invocada cuando la Monitorización detecte un problema y la tarea de Modos de Operación no encuentre ninguna anomalía. De esta forma podemos comprobar si el contexto en el que se va a invocar la tarea de diagnóstico se corresponde con el contexto para el que esta ha sido diseñada, ya que este contexto viene definido por el Protocolo de Operación activo. Más aún, la tarea de diagnóstico solamente se ha de ocupar de los fallos físicos o fallos internos.

3 Modelo de conocimiento de la tarea de diagnóstico

Como se indica en la sección anterior, la tarea de Diagnóstico se ha diseñado asumiendo que existe una tarea de Monitorización, responsable de detectar anomalías. Esto permite focalizar el proceso de diagnóstico en torno a las variables que vigila la monitorización, denominadas variables monitorizadas. Así, cada vez que se detecta un problema con una variable monitorizada, y una vez descartada una desviación respecto al Protocolo de Operación, se pone en marcha la tarea de Diagnóstico para intentar buscar una explicación causal al comportamiento anómalo. El proceso de diagnóstico es esencialmente local a cada variable monitorizada, si bien el método de diagnóstico intentará explicar el mayor número posible de anomalías con el menor número de causas. La principal característica de la tarea de Diagnóstico es que es capaz de examinar la evolución del sistema antes de proporcionar un diagnóstico definitivo. Como hipótesis simplificadora se utiliza la hipótesis de fallo único por variable monitorizada.

A continuación se describe el modelo de red causal temporal que soporta el proceso de diagnóstico, para presentar seguidamente el modelo de conocimiento utilizando la notación de COMMONKADS.

3.1 Organización del conocimiento de dominio: Red causal con extensiones temporales

La tarea de Diagnóstico se apoya en una red causal, que denotaremos por RC , que tiene estructura de grafo dirigido

acíclico. La red causal contiene tres tipos de nodos: nodos variable monitorizada, NVM , nodos indicio, NI , y nodos causa, NC . Si denotamos por N al conjunto de nodos de la red y por A al conjunto de arcos, podemos definir de forma precisa la red causal que soporta el diagnóstico.

Definición 1 Red Causal, RC — *El grafo dirigido, acíclico y finito $RC = \{N, A\}$, con $N = NVM \cup NI \cup NC$ y sujeto a*

- a) *NVM , NI y NC disjuntos dos a dos,*
 - b) *el grafo no dirigido subyacente es conexo*
- es una Red Causal si cumple las siguientes condiciones:*
1. *Nodos variable monitorizada, NVM :*
 - a. *No tienen padre.*
 - b. *No pueden ser nodos terminales.*
 - c. *Si n es un nodo variable monitorizada, n es el nodo raíz de un árbol dirigido $T(n) = \{N(n), A(n)\}$, con $N(n) \subset N$, $A(n) \subset A$ y $\forall k \in N$ tal que $\exists [k, j, \dots, n]$, camino de k a n , los nodos $k, j \in N(n)$ y el arco $(k, j) \in A(n)$. Denominaremos a $T(n)$ como árbol asociado a n .*
 2. *Nodos indicio, NI :*
 - a. *Tienen al menos un padre n , con $n \in NVM \cup NC$.*
 - b. *No pueden ser nodos terminales.*
 - c. *Si n es un nodo hijo de un nodo indicio, $n \in NC$.*
 3. *Nodos Causa, NC :*
 - a. *Tienen al menos un nodo padre.*

La condición 1) es especialmente restrictiva y es la que permite focalizar la diagnóstico en el entorno de las variables monitorizadas. Esta condición, junto al hecho de que los nodos variable monitorizada son los únicos que no tienen padre, garantiza que la red causal esta formada por la unión de los árboles asociados a los nodos variables monitorizada: $RC = \cup_{n \in NVM} T(n)$. Los nodos causa se utilizarán para representar las causas que pueden generar desviaciones en las variables monitorizadas. Los arcos tendrán asociados los efectos necesarios para determinar la presencia de las causas. Los nodos indicios se utilizarán para establecer un conjunto de causas sospechosas en base a los efectos ya observados, pero cuya confirmación será función de los efectos que se observen en el futuro.

Desde el punto de vista de su uso en la tarea de Diagnóstico, los nodos variable monitorizada desempeñan el papel de *queja*, o síntoma que provoca el inicio del proceso de diagnóstico. Su establecimiento es función de la tarea de Monitorización. Los nodos causa jugarán el papel de *hipótesis* que pueden explicar la queja. Los nodos indicio interpretan el papel de *sospecha*, que permitirán vislumbrar las *hipótesis diferidas*, o causas cuya confirmación sólo podrá realizarse en el futuro.

Nos va a interesar restringir aún más el tipo de red que utilizaremos para modelar el conocimiento causal. Por ello, se impone una condición adicional sobre las ramas de los

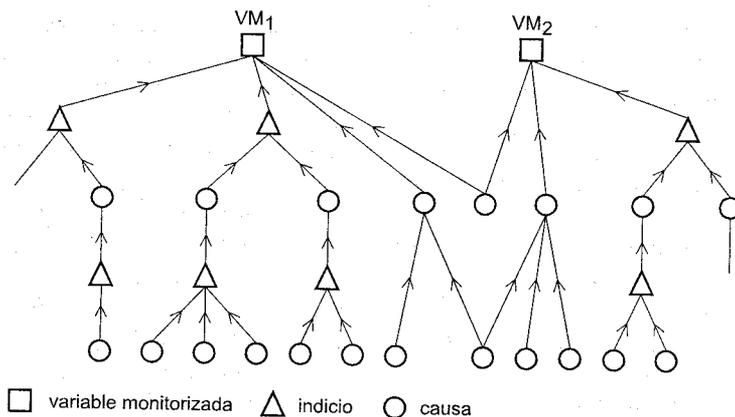


Figura 1: Red causal genérica.

árboles asociados a las variables monitorizadas que contienen nodos indicio.

Condición de separación. Sea $C = [j, \dots, s, t]$ un camino de RC que contiene al menos un nodo indicio, con $j \in NC$ un nodo terminal y $t \in NVM$. El camino C satisface la condición de separación si y sólo si todos los arcos del camino (n_i, n_{i+1}) , verifican

$$n_i \in NC \text{ y } n_{i+1} \in NI, \text{ o}$$

$$n_i \in NI \text{ y } n_{i+1} \in NC$$

salvo el último arco (s, t) , siendo $s \in NI$.

Definición 2 Red Causal Separable, RCS — Una red causal RCS es separable si y sólo si todos los caminos de la red, con origen en un nodo terminal y destino en un nodo variable monitorizada, que contienen nodos indicios cumplen la condición de separación.

La condición de separación garantiza que todos los caminos de una red causal separable, con origen en un nodo terminal y final en un nodo variable monitorizada y en los que haya alguna ocurrencia de un nodo indicio, tienen la estructura $[nc_1, ni_1, nc_2, ni_2, \dots, nc_{n_{NI}}, ni_{n_{NI}}, n]$ con $nc_k \in NC, ni_k \in NI$ y $n \in NVM$, para $k = 1, 2, \dots, n_{NI}$.

La Figura 1 ilustra, mediante un fragmento de red, algunas de las posibles configuraciones permitidas en una red causal separable. Por definición, cada nodo variable monitorizada, n , es el nodo raíz de un árbol $T(n) \subset RCS$. Este árbol admite una descomposición única, como expresa la siguiente proposición.

Proposición 1 Sea $RCS = \{N, A\}$ una red causal separable y $n \in NVM \subset N$. Sea $T(n) = \{N(n), A(n)\}$ el árbol asociado a n . $T(n)$ admite la descomposición única en dos subárboles, $T_r(n) = \{N_r(n), A_r(n)\}$ y $T_l(n) = \{N_l(n), A_l(n)\}$, de forma que $T_r(n)$ incluya únicamente todos los caminos que no contienen nodos indicio

y $T_l(n)$ los restantes. $T_r(n)$ y $T_l(n)$ verifican las siguientes propiedades:

1. $T(n) = T_r(n) \cup T_l(n)$. Esto es, $N(n) = N_r(n) \cup N_l(n)$ y $A(n) = A_r(n) \cup A_l(n)$
2. $T_r(n) \cap T_l(n) = \{\{n\}, \emptyset\}$. Esto es, $N_r(n) \cap N_l(n) = \{n\}$ y $A_r(n) \cap A_l(n) = \emptyset$
3. $N_r(n) \cap NI = \emptyset$
4. $N_l(n) \cap NI \neq \emptyset$, salvo si $T_l(n) = \{\{n\}, \emptyset\}$

Como expondremos a continuación, la condición de separación se introduce para simplificar el procesamiento en línea de la red causal cuando se incluyen consideraciones temporales.

La red causal permite modelar relaciones causa-efecto del dominio en el entorno de cada variable monitorizada. Pero las redes causales también se han utilizado, [Pople, 1982], [Schreiber y otros, 1999], para incluir información relativa a los síntomas y las causas que los provocan: relaciones de necesidad, contingencia, etc. En esta aproximación al diagnóstico, vamos a extender la red causal asociando a cada arco el conjunto de efectos necesarios para establecer el nodo origen del arco, suponiendo que el nodo destino ya ha sido establecido o confirmado. Además, vamos a ampliar el conocimiento causal añadiendo algunos elementos de su dimensión temporal, pues la localización temporal de las causas y sus efectos, [Console y Torasso, 1991], aporta información de interés para el diagnóstico.

Para incluir la información temporal, se va a recurrir a un modelo de tiempo lineal, discreto, con pasado y futuro acotados y basado en intervalos, similar al que propone Torsun en [Torsun, 1995] para la lógica temporal USF, que es suficiente para nuestras necesidades de representación. Esto es así porque en el contexto de los sistemas continuos gobernados por reguladores discretos es habitual muestrear todas las señales con frecuencias múltiplo de una frecuencia base, suponiendo constantes todos los valores entre periodos de muestro. Esta frecuencia base fijaría la unidad de discretización.

Utilizaremos la propuesta clásica de McDermott, [Charniak y McDermott, 1985], y distinguiremos entre proposiciones tipo, sin localización temporal, y proposiciones token, que sitúan las proposiciones tipo en el tiempo aportando una referencia temporal absoluta. Estas referencias temporales absolutas se determinarán en tiempo de ejecución.

Definición 3 Nodo token — Se denomina nodo token al par (n, t_a) donde $n \in NVM \cup NI \cup NC$ es un nodo (tipo) y t_a una referencia temporal absoluta. Si $n \in NVM$, t_a ha de ser fijado por la tarea de monitorización, registrando el instante en que se genera una queja. Si $n \in NI \cup NC$, t_a será determinado por la tarea de diagnosis. En el modelo de diagnosis que se describe en este trabajo, t_a tendrá el valor que se asigna al nodo variable monitorizada cuyo árbol se esté examinando; dicho valor será asignado cuando la causa o indicio representada por el nodo sea establecida como cierta.

Necesitamos ahora introducir el concepto de efecto. En la práctica, los efectos serán expresiones que comprueben alguna restricción sobre el valor actual o pasado de una variable, por ejemplo $ph-27(t-3) < 4$, para indicar que el valor de la variable $ph-27$ hace 3 segundos es inferior a 4. Este tipo de notación para establecer referencias temporales relativas es muy común en el dominio para el que se ha desarrollado este modelo, e intentaremos recogerla en él.

Para ello, haremos corresponder el concepto de efecto tipo con la dimensión atemporal de la restricción que se está comprobando, $ph-27 < 4$ en el ejemplo anterior. Desde el punto de vista del modelo de diagnosis, nos basta considerarlos como entidades atómicas que serán elementos de un Conjunto de Efectos. Para denotar referencias temporales relativas recurriremos al concepto de *efecto token relativo* y finalmente introduciremos el concepto de *efecto token*, para situar los efectos en el tiempo.

Definición 4 Efecto token relativo — Sea E un Conjunto de Efectos (tipo). Se denomina efecto token relativo al par $\langle e, j \rangle$ donde $e \in E$ es un efecto (tipo) y $j \in R^+ \cup \{0\}$ un número real no negativo. j va a indicarnos el nº de unidades de tiempo, hacia el pasado, en el que habrá que situar el efecto respecto a alguna referencia temporal absoluta. En el ejemplo previo, $j = 3$ (suponiendo que nuestra unidad de medida temporal es el segundo). Dado que j no puede ser negativo, las referencias relativas no pueden ser hacia el futuro. El valor de j será parte del conocimiento del dominio a elicitar.

Definición 5 Efecto token — Sea E un Conjunto de Efectos. Se denomina efecto token al par (e, t_a) donde $e \in E$ es un efecto (tipo) y t_a una referencia temporal absoluta. El valor de t_a se determinará durante el proceso de diagnosis.

Definición 6 Síntomas — Sea $RC = \{N, A\}$ una red causal. Denominamos Síntomas al triplete (a, Er, k) , siendo

$a \in A$ un arco de la red causal RC , Er un conjunto de efectos token relativos y $k \in R^+ \cup \{0\}$ un número real no negativo.

El concepto de Síntomas pretende capturar la noción de conjunto de efectos necesarios para establecer una causa, si bien su significado preciso dependerá del valor de k , que denota una referencia temporal relativa.

Debido a la naturaleza dinámica del sistema, los efectos necesarios para establecer una causa pueden manifestarse en instantes muy diferentes. En principio, esto no plantea ningún problema si los síntomas a observar se presentan antes de que la Monitorización realice la detección: basta con guardar los valores históricos de las variables pertinentes y examinarlos tras la detección. Por el contrario, si los síntomas a explorar sólo aparecen con posterioridad a la detección realizada por la Monitorización, será preciso observar la evolución del sistema antes de poder confirmar o rechazar su existencia. Dado que estas consideraciones son aplicables a todas las relaciones causales modeladas en el grafo, hemos considerado convenientes distinguir dos tipos de Síntomas: rápidos y lentos —respecto a la detección realizada por la monitorización. Los Síntomas rápidos sólo incluyen efectos que pueden observarse simultáneamente o con anterioridad a la aparición de la queja. Los Síntomas lentos incluyen efectos que pueden presentarse con posterioridad a la aparición de la queja.

El significado de los Síntomas es ligeramente distinto según se trate de Síntomas rápidos o lentos. Si $a = (c_1, c_2)$ es un arco con origen en c_1 y destino en c_2 y c_2 ha sido establecida, de modo que existe la causa token (c_2, t_a) tenemos:

Definición 6.1 Síntomas rápidos — tupla (a, Er, k) con $k = 0$

Interpretación: para establecer la causa (o indicio) token (c_1, t_a) , es necesaria la observación de todos los efectos token $(e, t_a - j)$, con $(e, j) \in Er$.

Definición 6.2 Síntomas lentos — tupla (a, Er, k) con $k > 0$

Interpretación: para establecer la causa (o indicio) token (c_1, t_a) , es necesaria la observación, para algún $l \in (0, k]$, $l \in R^+$, de todos los efectos token $(e, t_a - j + l)$, con $(e, j) \in Er$.

La definición de Síntomas permite diferenciar claramente aquellos síntomas que se han producido en el presente o pasado ($k = 0$) frente a aquellos que sólo se podrán observar en el futuro ($k > 0$). Además, la interpretación de los síntomas es diferente en cada caso. En los Síntomas rápidos, $k = 0$ y los efectos se sitúan en el presente o pasado, atendiendo a la referencia temporal de los efectos token relativos. En el caso de los Síntomas lentos, la propia k es una referencia temporal relativa que permitirá establecer una cota superior del instante (futuro respecto al tiempo de detección) en el que han de poder observarse los efectos. En otras palabras, k indica el tiempo máximo que el sistema puede esperar por la

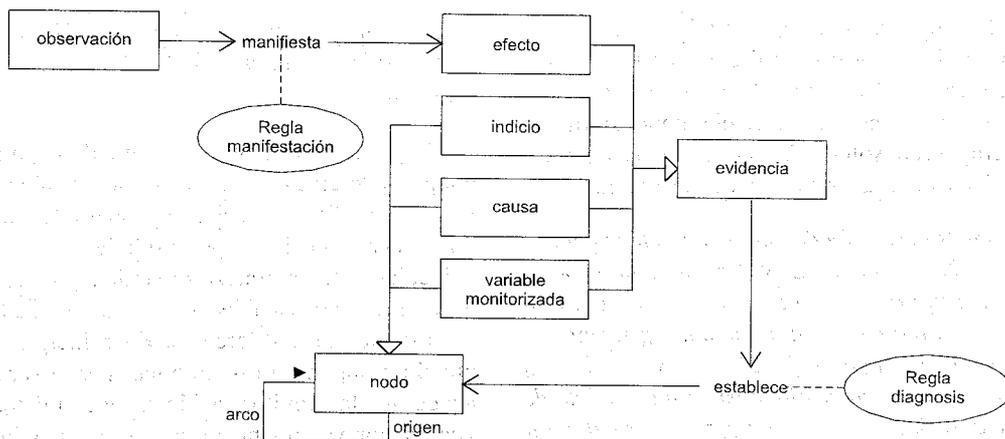


Figura 2: Un posible esquema de dominio para la diagnosis.

aparición de los síntomas, una vez que se ha producido una detección. Nótese que los síntomas lentos no sitúan con precisión los efectos, si bien mantienen su secuencia temporal dada por Er .

Definición 7 Familia de síntomas separables — Sea RCS una Red Causal Separable y $S = \{(a, Er(a), k(a)), a \in A\}$ una Familia de Síntomas, con $k : A \rightarrow R^+ \cup \{0\}$ y $Er : A \rightarrow \mathcal{P}(ER)$ funciones, donde ER denota el conjunto finito de todos los Efectos Relativos considerados por el sistema y $\mathcal{P}(ER)$ el conjunto de las partes del mismo. S es una Familia de Síntomas Separables para RCS si y sólo si se cumplen las siguientes condiciones:

1. $\forall a \in A, \exists (a, Er(a), k(a)) \in S$
2. $k(a) > 0$ si y sólo si $a = (c, i)$ con $c \in NC$ e $i \in NI$

La propiedad esencial de una Familia de Síntomas Separables radica en que todos los síntomas son rápidos, excepto, y necesariamente, los asociados a arcos con destino en nodos indicio.

Definición 8 Red Causal Extendida — Sea RCS una Red Causal Separable y S una familia de síntomas separables para RCS. Denominamos Red Causal Extendida al par $\langle RCS, S \rangle$.

Una Red Causal Extendida, por definición, contiene relaciones entre las causas que pueden generar un problema detectado por una tarea de monitorización, así como los síntomas adicionales para poder establecer las causas. Pero incluye, además, información temporal sobre el tiempo de observación de los síntomas respecto al tiempo de detección. El tipo de información temporal considerado y las condiciones de separación que se exigen a la red causal y a los síntomas han sido establecidos buscando un compromiso entre la capacidad de representación de información causal-temporal y la eficacia en el cómputo. Como expresa la Proposición 1, cada

árbol asociado a un nodo variable monitorizada, $T(n)$, admite la descomposición única $T(n) = T_r(n) \cup T_l(n)$; $T_r(n)$ no contiene nodos indicio y todos sus síntomas son rápidos, de modo que podemos recorrer el árbol y comprobar la presencia de efectos una vez realizada la detección. Por el contrario $T_l(n)$ contiene síntomas lentos y su procesamiento puede requerir examinar la evolución posterior del sistema. La exigencia de que en las ramas de $T_l(n)$ se alternen los nodos indicio y los nodos causa — y consecuentemente los síntomas rápidos y lentos — puede parecer restrictiva, pero permite unificar el procesamiento de los síntomas lentos sin perder capacidad de representación, puesto que cualquier arco puede tener asociado un conjunto de efectos relativo vacío. Además, la representación captura la heurística de que las referencias temporales hacia el futuro sólo se examinarán cuando sean necesarias para discriminar entre causas con algunos síntomas comunes ya observados. Estos síntomas comunes se modelan mediante los Síntomas rápidos asociados a arcos con origen en nodos indicios. El recorrido de los árboles con nodos indicios es sensiblemente más complejo (computacional y conceptualmente).

3.2 Modelo de Conocimiento: Conocimiento de dominio

La Figura 2 es un fragmento del esquema de dominio que permite conceptualizar la Red Causal Extendida utilizando las especificaciones de esquemas de dominio proporcionadas por COMMONKADS. Los distintos tipos de Nodos y el Arco permiten describir la topología de la red, es decir, la Red Causal Separable. Los síntomas se representan mediante el tipo Regla-diagnosis, cuya descripción textual incluye la referencia temporal relativa del síntoma.

A modo de ejemplo, consideremos el síntoma (*fuga-en-T1-o-T3, fuga-en-T3*), $(\Delta TT3.value \approx 0, 1)$, $(\Delta TT3.value > 0, 0)$, 2) donde:

```

TASK diagnosis ;
GOAL : "Encontrar una explicación causal a una queja" ;
ROLES :
  INPUT :
    queja : "Discrepancia observada" ;
    protocolo : "Protocolo de operación activo" ;
  OUTPUT :
    explicación-causal : "Camino causal desde la causa hasta la queja" ;
SPECIFICATION : "Seleccionar una causa examinando las que han provocado problemas previamente,
  las que se pueden establecer con los síntomas actuales y si es necesario, esperando síntomas futuros" ;
END TASK

TASK-METHOD diagnosis-tres-fases ;
REALIZES : diagnosis ;
DECOMPOSITION :
  TASKS : diagnosis-previa, diagnosis-instantánea, diagnosis-diferida ;
CONTROL-STRUCTURE :
  diagnosis-previa (queja, protocolo → explicación-causal);
  IF NOT explicación-causal THEN diagnosis-instantánea (queja, protocolo → explicación-causal) ; END IF
  IF NOT explicación-causal THEN diagnosis-diferida (queja, protocolo → explicación-causal) ; END IF
  IF NOT explicación-causal THEN explicación-causal = DESCONOCIDA ; END IF
END TASK-METHOD
    
```

Figura 3: Tarea y método de tarea para la diagnosis.

- ((*fuga-en-T1-o-T3*, *fuga-en-T3*) es un arco de la red, con *fuga-en-T1-o-T3* un nodo indicio y *fuga-en-T3* un nodo causa,
 - ($\Delta TT3.value \approx 0, 1$), ($\Delta TT3.value > 0, 0$) es un conjunto de efectos token relativos, con $\Delta TT3.value \approx 0$ y $\Delta TT3.value > 0$ efectos tipo,
 - 2 es la referencia temporal relativa del síntoma.
- Dicho síntoma se representa mediante la regla:

```

fuga-en-T1-o-T3 == TRUE AND
SOMETIME t IN [NOW, NOW + 2]
  ( $\Delta TT3.value(t - 1) \approx 0$ ) AND ( $\Delta TT3.value(t) > 0$ )
ESTABLECE
fuga-en-T3 = TRUE
    
```

3.3 Modelo de Conocimiento: Conocimiento de tarea e inferencia

Como indica la Figura 3, la tarea de diagnosis tiene por entradas una *queja* y el *protocolo* de operación activo. El papel de *queja* solamente puede ser desempeñado por los nodos variable monitorizada. Para poder realizar la tarea bajo distintos protocolos, será preciso disponer de una red causal extendida para cada protocolo. El papel estático que soporta la realización de la tarea, *modelo causal*, estará instanciado por el conjunto de Redes Causales Extendidas disponibles. En consecuencia, todas las inferencias que utilizan el papel estático *modelo causal*, tendrán como entrada adicional el papel *protocolo*, para poder seleccionar la red causal perti-

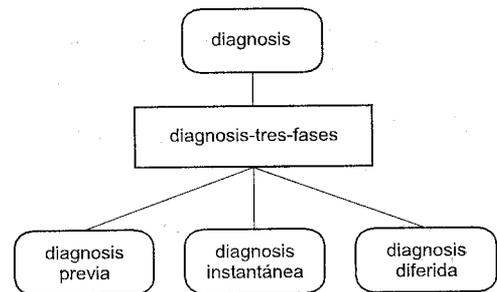


Figura 4: Descomposición de la diagnosis.

nente. Para no sobrecargar los diagramas, esta entrada adicional no se mostrará en las estructuras de inferencia, aunque se incluirá en la descripción del método de tarea. La salida de la tarea es una *explicación causal*, o secuencia de causas de un camino de la red causal extendida con origen en un nodo causa terminal y destino en el nodo variable monitorizada con el que se instancia la queja.

La Figura 4 muestra la descomposición a alto nivel de la tarea de Diagnosis, que es soportada por el método de tarea DIAGNOSIS-TRES-FASES, cuya especificación detallada se encuentra en la Figura 3. De acuerdo con dicha especificación, las tres sub tareas de diagnosis, Previa, Instantánea y Diferida, se invocan secuencialmente, hasta que alguna de ellas es capaz de encontrar una explicación causal.

```

TASK diagnosis-previa ;
GOAL : "Encontrar una explicación causal a una queja" ;
ROLES :
INPUT :
    queja : "Discrepancia observada" ;
    protocolo : "Protocolo de operación activo" ;
OUTPUT :
    explicación-causal : "Camino causal desde la causa hasta la queja" ;
SPECIFICATION : "Verificar si la causa de algún problema identificado previamente puede considerarse la causa
del problema actual" ;
END TASK

TASK-METHOD probar-causa-activa ;
REALIZES : diagnosis-previa ;
DECOMPOSITION :
    INFERENCES : cover, select, generate ;
ROLES :
    INTERMEDIATE :
        hipótesis-original : "causa terminal en el grafo causal" ;
        hipótesis-activa : "causa que explica otra queja" ;
CONTROL-STRUCTURE :
    cover (queja, protocolo → hipótesis-originales) ;
    select (hipótesis-originales → hipótesis-activa) ;
    IF hipótesis-activa THEN generate (hipótesis-activa, protocolo → explicación-causal) ; END IF
END TASK-METHOD
    
```

Figura 5: Descripción de la tarea y método de tarea para la diagnosis previa.

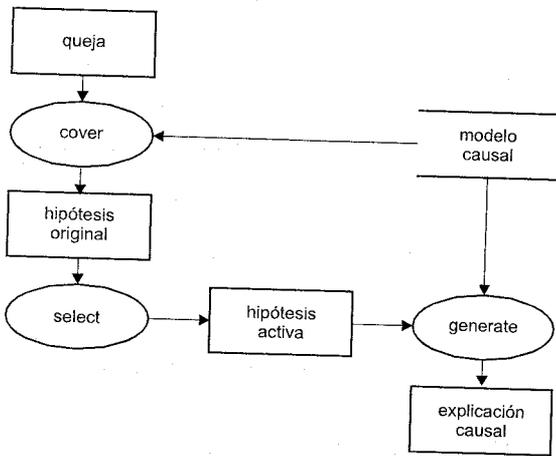


Figura 6: Estructura de inferencia para la diagnosis previa.

3.3.1 Diagnósis Previa

Esta tarea se introduce para reducir al mínimo el número de causas consideradas por el sistema, comprobando, antes de buscar una nueva explicación causal, si alguna de las ya presentes es capaz de explicar la nueva queja detectada.

Las inferencias a realizar, esquematizadas en la Figura 6, son:

cover queja: genera el conjunto de hipótesis originales (causas terminales) de acuerdo a la red causal.

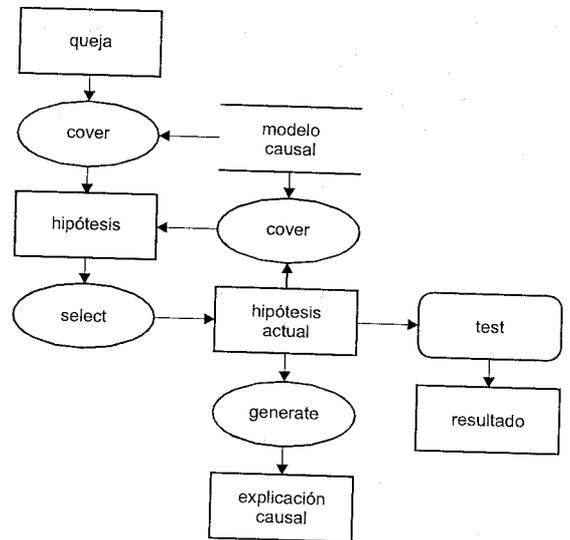


Figura 7: Estructura de inferencia para la diagnosis instantánea.

select hipótesis original: comprueba si algunas de estas hipótesis están establecidas como causas de otras quejas; en ese caso, selecciona una aleatoriamente.

generate explicación causal: crea una explicación recorriendo la red causal a partir de la hipótesis activa seleccionada.

```

TASK diagnosis-instantánea ;
GOAL : "Encontrar una explicación causal a una queja" ;
ROLES :
  INPUT :
    queja : "Discrepancia observada" ;
    protocolo : "Protocolo de operación activo" ;
  OUTPUT :
    explicación-causal : "Camino causal desde la causa hasta la queja" ;
  SPECIFICATION: "Encontrar un camino causal que explique la queja, consistente con observaciones pasadas y presentes";
END TASK

TASK-METHOD recubrimiento-simple ;
REALIZES : diagnosis-instantánea ;
DECOMPOSITION :
  INFERENCES : cover, select, generate ;
  TRANSFER-FUNCTIONS : test ;
ROLES :
  INTERMEDIATE :
    hipótesis : "Conjunto de causas potenciales" ;
    hipótesis-actual : "Causa que se comprueba" ;
    resultado : "Valor de verdad de la comprobación de la causa" ;
    hipótesis-adicionales : "Posible causa de una hipótesis actual" ;
CONTROL-STRUCTURE :
  cover (queja, protocolo → hipótesis) ;
DO
  select (hipótesis → hipótesis-actual) ;
  test (hipótesis-actual → resultado) ;
  hipótesis = hipótesis SUBTRACT hipótesis-actual ;
  IF resultado == TRUE THEN
    cover (hipótesis-actual, protocolo → hipótesis-adicionales) ;
    hipótesis = hipótesis ADD hipótesis-adicionales ;
  END IF
UNTIL SIZE hipótesis < 1 OR "se ha cumplido alguna causa original" ;
IF "se ha cumplido alguna causa original" THEN generate (hipótesis-actual, protocolo → explicación-causal) ; END IF
END TASK-METHOD

```

Figura 8: Descripción de la tarea y método de tarea para la diagnosis instantánea.

La Figura 5 muestra la especificación de la tarea y el método que la soporta, *probar-causa-activa*.

3.3.2 Diagnósis Instantánea

Esta tarea intenta encontrar una explicación causal para la queja actual utilizando únicamente las observaciones disponibles en el momento en que se realiza la detección. Para ello, se limita a recorrer la parte del árbol asociado al nodo variable monitorizada que sólo contiene síntomas rápidos.

Su estructura de inferencias, Figura 7, exige las siguientes inferencias o subtareas:

cover queja: esta inferencia genera el conjunto de hipótesis hijas de la queja, sin considerar los nodos indicio.

select hipótesis: selecciona una hipótesis, quizás según prioridad.

test hipótesis actual: esta subtask examina los síntomas asociados al arco con origen en la hipótesis actual, ob-

teniéndose un resultado cierto si todos los efectos token requeridos por los síntomas, ver Definición 6.1, están presentes.

generate explicación causal: crea una explicación causal recorriendo la red a partir de la hipótesis actual, que además de haber sido establecida ha de ser terminal.

cover hipótesis actual: genera el conjunto de hipótesis hijas de la hipótesis actual.

La Figura 8 muestra la especificación de la tarea y el método que la soporta, *recubrimiento-simple*, que es una variante simplificada del método de recubrimiento causal propuesto por [Schreiber y otros, 1999]. Como se desprende de la especificación de las inferencias y del método de tarea, esta tarea se limita a explorar la parte del grafo causal que tiene asociados síntomas con dinámica rápida respecto a la variable monitorizada que da lugar a la queja; esto es, el árbol de la red causal que denominamos $T_r(n)$ en la Proposición 1.

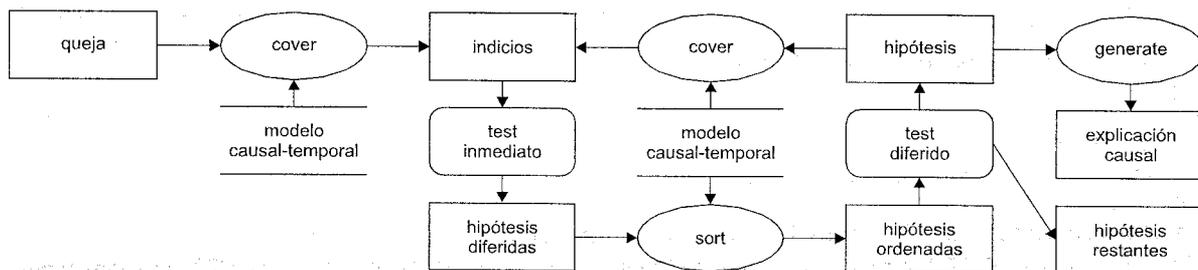


Figura 9: Estructura de inferencia para la diagnosis diferida.

3.3.3 Diagnos Diferida

Esta tarea puede permanecer a la espera de la ocurrencia de nuevos síntomas para obtener un diagnóstico. Únicamente se invoca si las restantes tareas de diagnóstico no han finalizado con éxito, lo cual es un reflejo de que la información histórica no ha sido suficiente para incriminar una causa. La tarea sólo se puede ejecutar si el nodo variable monitorizada con el que se ha instanciado la queja tiene asociado un árbol con síntomas lentos $T_i(n) \neq \{\{n\}, \emptyset\}$. Además, ha sido diseñada de forma que sólo se realiza si existen motivos para sospechar la presencia de algunas causas, porque algunos de sus síntomas ya han sido observados, manteniéndose a la espera de nuevos síntomas que permitan diferenciar entre las posibles hipótesis.

La estructura de inferencia de dicha tarea se proporciona en la Figura 9, e incluye las siguientes inferencias:

cover queja: esta inferencia genera el conjunto de indicios hijos de la queja, ignorando los nodos causas, ya procesados en la tarea diagnosis instantánea. Si la queja no tiene asociados nodos indicios, esto es $T_i(n) = \{\{n\}, \emptyset\}$, la tarea finaliza.

test inmediato indicios: esta subtarea examina, para cada indicio, los síntomas asociados al arco con origen en el indicio; cuando se ejecuta esta tarea, los efectos requeridos por los síntomas han de ser observables en el pasado, por tratarse de Síntomas rápidos. Si los efectos se satisfacen, confirmándose el indicio, se generan las hipótesis diferidas a partir de las causas hijas de los nodos indicios. Nótese que si inicialmente no se puede confirmar ningún indicio, las siguientes inferencias fracasarán y la tarea finalizará sin éxito.

sort hipótesis diferidas: se ordenan las hipótesis, por orden creciente del tiempo máximo de espera para observar los efectos que la establecen. En nuestro modelo causal, formado por Redes Causales Extendidas, el parámetro que proporciona este tiempo máximo de espera es la referencia temporal relativa que define el síntoma de cada hipótesis diferida, según la Definición 6.2.

test diferido hipótesis ordenadas: esta tarea comprueba, periódicamente, la presencia de los síntomas esperados, eliminando aquellas hipótesis diferidas cuyos síntomas

no se hayan presentado trascurrido el tiempo máximo estimado para ello. La tarea finaliza cuando puede establecer una *hipótesis*, manteniendo en *hipótesis restantes* aquellas hipótesis que aún se pueden confirmar en el futuro, y que se añadirán a las hipótesis diferidas en el siguiente ciclo de diagnosis. La tarea puede finalizar rechazando todas las hipótesis, en cuyo caso la diagnosis finalizará sin éxito.

cover hipótesis: genera el conjunto de indicios hijos de la hipótesis.

Esta tarea, descrita y especificada en la Figura 10, es capaz de discriminar entre causas con síntomas rápidos comunes, esperando la presencia de sus síntomas lentos esperados. No hemos especificado en más detalle la subtarea test diferido, porque su especificación interna no es relevante en este nivel de análisis. Tan sólo es necesario que el modelo temporal utilizado soporte la tarea de acuerdo con su especificación. En esta propuesta se utilizan las Redes Causales Extendidas ya definidas y la tarea test diferido ha de comprobar la presencia de los síntomas lentos, de acuerdo a la Definición 6.2.

Mientras que la tarea diagnosis instantánea realiza, básicamente, un recorrido primero en profundidad del grafo causal a partir del nodo variable monitorizada, la tarea diagnosis diferida explora simultáneamente todos los caminos del grafo causal que incluyen indicios, utilizando como criterio de selección del nodo a expandir aquel con menor tiempo de espera de entre los nodos que se pueden establecer.

4 Discusión

En este trabajo se ha presentado un modelo de conocimiento para la tarea de diagnosis de sistemas dinámicos. La principal característica de la propuesta es la división de la tarea de diagnosis en tres subtareas: diagnosis previa, instantánea y diferida. La articulación de esta tres subtareas permite al sistema de diagnóstico vigilar la evolución de un proceso continuo para discriminar entre hipótesis que aún pueden explicar las observaciones disponibles. Además, esto no evita que el sistema pueda obtener un diagnóstico con los datos presentes cuando se produce la detección de una anomalía, siempre que estos datos permitan incriminar una única causa.

```

TASK diagnosis-diferida ;
GOAL : "Encontrar una explicación causal a una queja" ;
ROLES :
  INPUT :
    queja : "Discrepancia observada" ;
    protocolo : "Protocolo de operación activo" ;
  OUTPUT :
    explicación-causal : "Camino causal desde la causa hasta la queja" ;
  SPECIFICATION : "Encontrar un camino causal que explique la queja consistente con observaciones pasadas,
    presentes y futuras" ;
END TASK

TASK-METHOD recubrimiento-temporal-simple ;
REALIZES : diagnosis-diferida ;
DECOMPOSITION :
  INFERENCES : cover, sort, generate ;
  TASKS : test-inmediato, test-diferido ;
ROLES :
  INTERMEDIATE :
    hipótesis : "Conjunto de causas potenciales" ;
    hipótesis-diferidas : "Conjunto de hipótesis pendientes de confirmación por observaciones futuras";
    hipótesis-ordenadas : "Conjunto de hipótesis pendientes de confirmación ordenadas por
      orden creciente del tiempo máximo de espera para su establecimiento" ;
    hipótesis-restantes : "Hipótesis pendientes de confirmación por observaciones futuras" ;
    indicios : "Entidad que permite sospechar de algunas causas" ;
CONTROL-STRUCTURE :
  cover (queja, protocolo → indicios);
  IF indicios != ∅ THEN
    DO
      test-inmediato (indicios → hipótesis-diferidas) ;
      hipótesis-diferidas = hipótesis-diferidas ADD hipótesis-restantes ;
      sort (hipótesis-diferidas → hipótesis-ordenadas) ;
      test-diferido (hipótesis-ordenadas → hipótesis + hipótesis-restantes) ;
      IF hipótesis THEN cover (hipótesis + protocolo → indicios) ; END IF
    UNTIL SIZE hipótesis-restantes < 1 OR "se ha logrado una causa terminal" ;
    IF "se ha logrado una causa terminal" THEN generate (hipótesis + protocolo → explicación-causal); END IF
  END IF
END TASK-METHOD

```

Figura 10: Descripción de la tarea y método de tarea para la diagnosis diferida.

La principal contribución de este trabajo es la formalización del modelo de conocimiento. Esta formalización incluye una descripción detallada de la Red Causal Extendida utilizada para representar el conocimiento de dominio, junto a la especificación del conocimiento de inferencia y de tarea, incluidos en el tercer apartado. Esta formalización intenta generalizar el último modelo utilizado por la generación OLID, brevemente descrita en [Alonso y otros, 2001], de sistemas de diagnóstico industrial.

Una característica importante del modelo es su relativa sencillez, que es consecuencia de la estructura de la red causal utilizada para representar el conocimiento causal-temporal del dominio. La Red Causal Extendida fuerza una separación de los síntomas en síntomas rápidos y síntomas lentos, en relación al tiempo de detección. Esta propiedad permite factorizar la tarea de diagnosis según esta dimensión

temporal. Evidentemente, la contrapartida es una limitación en la capacidad de la red causal para representar información temporal, que se limita a situar en el tiempo los efectos necesarios para determinar una causa y a describir el tiempo máximo necesario para observar una secuencia de efectos. Es cierto que existen sistemas, como DIAPO, [Cauvin y otros, 1998], que permiten incluir en el conocimiento del dominio una descripción detallada de las relaciones temporales entre los elementos del mismo. DIAPO recurre al uso de técnicas avanzadas de razonamiento abductivo y temporal [Console y Torasso, 1991] y su capacidad de representación es superior a la de la Red Causal Extendida que aquí se propone. Como contrapartida, DIAPO sólo se invoca bajo demanda, por la petición de un usuario y no automáticamente por un sistema de monitorización. Más aún, DIAPO sólo trabaja con datos actuales e históricos, lo que seguramente limita su capacidad

como herramienta de diagnóstico en línea.

Para finalizar esta breve discusión, y considerando la experiencia industrial con sistemas de diagnóstico basados en versiones preliminares del modelo que aquí se presenta, ver [Alonso y otros, 2001], podemos afirmar que el modelo propone un compromiso razonable entre capacidad de representación y coste, tanto computacional como desde la perspectiva de la ingeniería de software y conocimiento –esto es, elaboración del modelo de conocimiento junto al desarrollo y mantenimiento de la aplicación.

Referencias

- G. Acosta, C. Alonso, y B. Pulido.** Basic Tasks for Knowledge Based Supervision in Process Control. *Engineering Applications of Artificial Intelligence*, 14:441–455, 2002.
- C. Alonso, B. Pulido, y G. Acosta.** On Line Industrial Diagnosis: an attempt to apply Artificial Intelligence techniques to process control. En *11th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, IEA/AIE-98. LNAI*, volumen 1415, páginas 804–813. Springer-Verlag, 1998.
- C. Alonso, B. Pulido, G. Acosta y C. Llamas.** On-line Industrial supervision and diagnosis, knowledge level description and experimental results. *Expert Systems with Applications*, 20(2):117–132, febrero 2001.
- K. Balakrishnan y V. Honavar.** Intelligent diagnosis systems. *Journal of Intelligent Systems*, 8(3/4):239–290, 1998.
- S. Cauvin, M-O. Cordier, C. Dousson, P. Laborie, F. Lévy, J. Montmain, M. Porcheron, I. Servet y L. Travé-Massuyès.** Monitoring and Alarm Interpretation in Industrial Environments. *AI Communications*, 11(3–4):139–173, 1998.
- E. Charniak y D. McDermott.** *Introduction to Artificial Intelligence*. Addison-Wesley, 1985.
- J. Chen y R.J. Patton.** *Robust model based fault diagnosis for dynamic systems*. Kluwer Academic Publisher, 1999.
- L. Console y P. Torasso.** On the co-operation between abductive and temporal reasoning in medical diagnosis. *Artificial Intelligence in Medicine*, 3(6):291–311, 1991.
- M. Cordier y C. Dousson.** Alarm Driven Monitoring Based on Chronicles. En *Proceedings of the 4th SafeProcess*, Budapest, Hungría, junio 2000.
- O. Dressler.** On-line diagnosis and monitoring of dynamic systems based on qualitative models and dependency recording diagnosis engines. En *Proceedings of the Twelfth European Conference on Artificial Intelligence ECAI'96*, páginas 461–465, 1996.
- T. Guckenbiehl y G. Schäfer-Richter.** *Readings in Model Based Diagnosis*, capítulo SIDIA: Extending prediction based diagnosis to dynamic models, páginas 309–317. Morgan-Kaufman Pub., San Mateo, 1992.
- O.O. Oyeleye, F.E. Finch y M.A. Kramer.** Qualitative modeling and fault diagnosis of dynamic processes by MIDAS. *Chemical Engineering Communications*, 96:205–228, 1990.
- H. Pople.** Heuristic Methods for Imposing Structure on Ill-Structured Problems: The Structuring of Medical Diagnostics. *Artificial Intelligence in Medicine, AAAS Select Symposium 51*, Westview-Press, páginas 119–190, 1982.
- C. Price.** *Computer-based diagnostic systems*. Springer, 1999.
- G. Schreiber, H. Akkermans, A. Anjewierden, R. de Hoog, N. Shadbolt, W. Van de Velde y B. Wielinga.** *Knowledge Engineering and Management, The CommonKADS Methodology*. The MIT Press, 1999.
- I.S. Torsun.** *Foundations of Intelligent Knowledge-Based Systems*. Academic Press, 1995.



Carlos J. Alonso González, es Licenciado en Ciencias, 1985 y Doctor en Ciencias Físicas, 1990, en ambos casos por la Universidad de Valladolid. Ha participado en numerosos proyectos relacionados con la supervisión y diagnóstico de procesos industriales continuos. En la actualidad Profesor Titular de Universidad en el Departamento de Informática. Sus principales temas de investigación son los Sistemas Basados en Conocimiento para la Supervisión y Diagnosís sistemas dinámicos, Diagnosís Basada en Modelos, Ingeniería del Conocimiento y Aprendizaje Automático.



César Llamas Bello, es Licenciado en Ciencias Físicas y Doctor en Informática por la Universidad de Valladolid. Profesor Titular del Área de Ciencias de la Computación e Inteligencia Artificial, sus intereses actuales son: los Sistemas Multi-agente, la Supervisión y Diagnosís de Procesos Continuos, y los Sistemas de Ayuda a la Docencia.



José A. Maestro Prieto, es Ingeniero en Informática por la Universidad de Valladolid desde 2000. Actualmente es profesor del Área de Ciencias de la Computación e Inteligencia Artificial, en esta universidad. Su principal tema de investigación es la Ingeniería del Conocimiento, aplicada a la supervisión y diagnóstico de procesos continuos



Belarmino Pulido, es Licenciado en Informática por la Universidad de Valladolid desde 1992. Desde 1994 trabaja en el Departamento de Informática de la Universidad de Valladolid, donde obtuvo su MSc en Informática en 1995 y su doctorado en 2001. Actualmente es Profesor Titular de Universidad en dicha universidad. Sus áreas de investigación son el razonamiento basado en modelos (principalmente aplicaciones al diagnóstico mediante diagnóstico basada en consistencia) y la ingeniería del conocimiento (esencialmente aplicada a la supervisión y diagnóstico de procesos continuos mediante sistemas basados en conocimiento). Es miembro de la Asociación Española para la Inteligencia Artificial desde 1997 y de la IEEE desde 2001. Actualmente es el coordinador del nodo de Valladolid de la Red de Excelencia Europea MONET2 (para la difusión del razonamiento basado en modelos y el diagnóstico) y pertenece al colectivo español ARCA (para la difusión del razonamiento cualitativo).

