



**UNIVERSIDAD NACIONAL DE EDUCACIÓN A
DISTANCIA**

Dpto. Inteligencia Artificial. UNED

**Diagnóstico parcial de hemorragias: Modelado
e implementación.**

Introducción.

Se pretende en este documento mostrar la utilización de diferentes métodos de *diagnóstico* para el modelado del conocimiento del problema del diagnóstico de hemorragias que plantearemos en el siguiente apartado. Gracias a que la aplicación, tal cual es aquí planteada, es un ejemplo de juguete, podremos llegar hasta la implementación, e incluso dar diferentes posibilidades para ésta.

Diagnóstico de hemorragias.

Para el desarrollo de un sistema de diagnóstico de hemorragias, vamos a partir del modelo de producción de coágulos de la figura 1 debido a [Nathan & Oski, 87]. De él se extrae el esquema de la figura 2, que describe el procedimiento de diagnóstico de la causa del fallo en la producción del coágulo, con las pruebas necesarias para llegar a conclusiones, en algunos casos parciales.

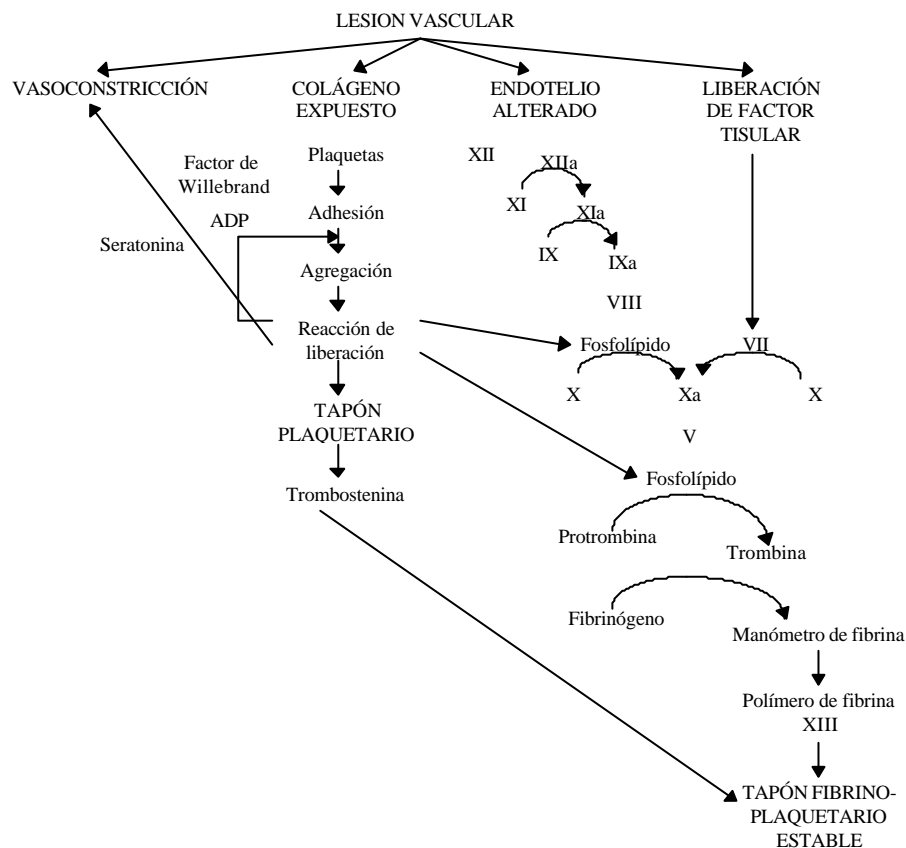


Fig. 1 Esquema del mecanismo de formación del coágulo.

El procedimiento comienza con la prueba de Rumpell-Leede para medir la fragilidad vascular y continúa con diferentes pruebas, según se muestra en la figura, hasta llegar a un diagnóstico, final en nuestro ejemplo, parcial en la realidad. La señal de alarma es un nivel bajo de hemoglobina, cuyas posibles causas son en primera instancia: vasoconstricción, colágeno



expuesto, endotelio alterado y liberación de factor tisular . Se aplica inicialmente la prueba de Rumpell-Leede para evaluar la fragilidad vascular. Si ésta es positiva llegamos ya a un diagnóstico. Si es negativa pasamos al siguiente elemento de diferenciación causal, las plaquetas, y para ello realizamos un recuento de ellas. En el caso de que se observe un número bajo, puede ser debido a: Aplasia medular, causas inmunes o secuestro. Se realizan las pruebas pertinentes para la confirmación de cualquiera de ellas.

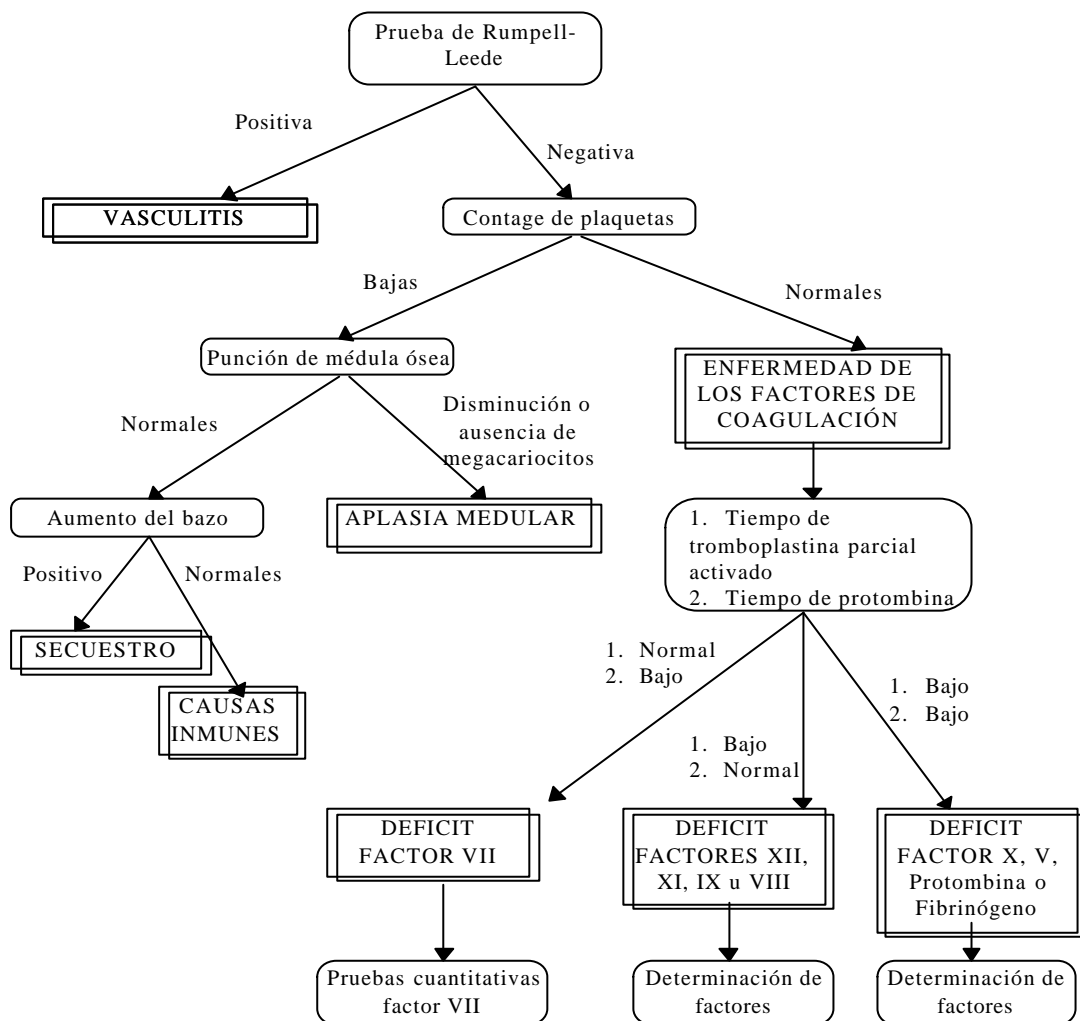
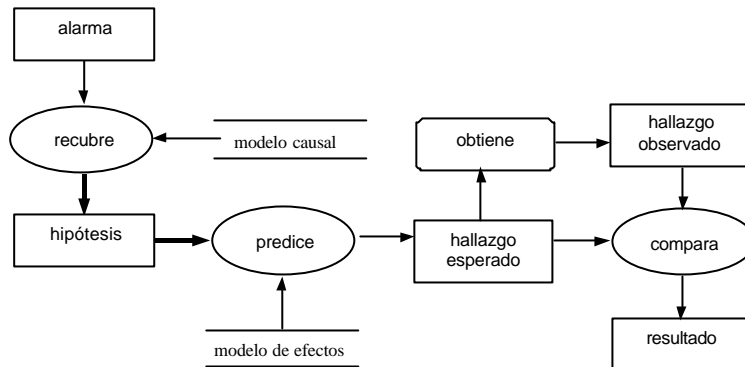


Fig. 2 Esquema del diagnóstico de hemorragia.

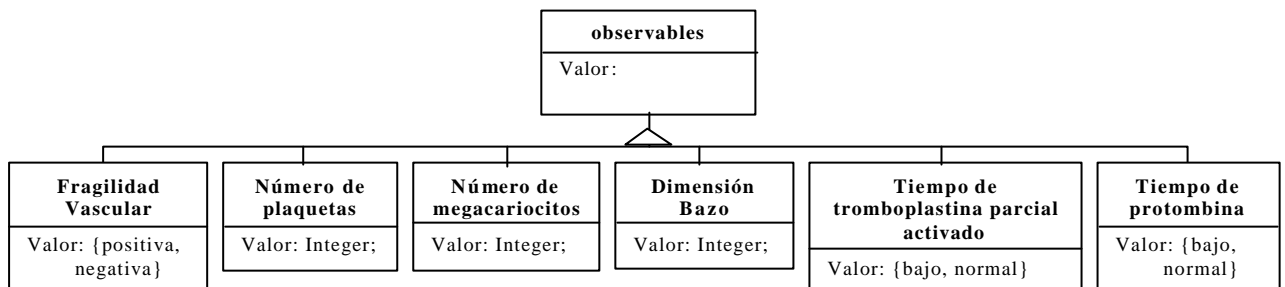
Modelo de Pericia.

El procedimiento es prácticamente idéntico al descrito para el diagnóstico de fallos en un equipo musical. En aquel ejemplo utilizamos una variante jerárquica-sistemática del método recubre-predice-compara, cuyo esquema inferencial se muestra en la siguiente figura.

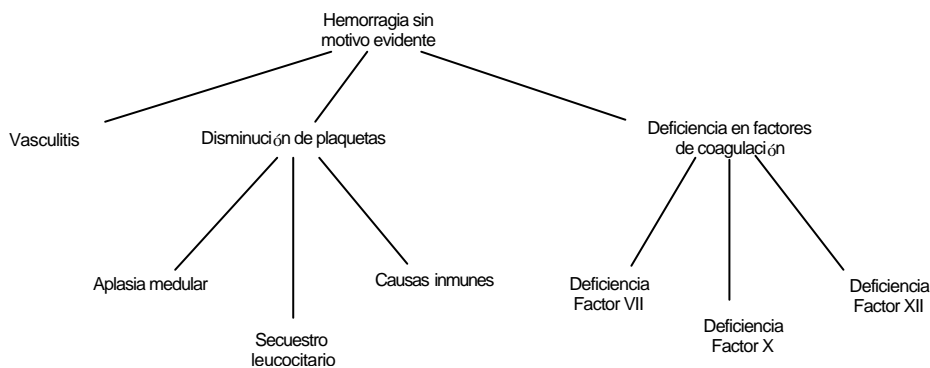




En cuanto al dominio, la descripción del experto médico en este caso, muestra la existencia de una jerarquía de refinamiento de estados patológicos del paciente y cada uno de ellos lleva asociado un hallazgo que lo confirma. Así, el modelo del dominio puede ser también casi completamente adaptado del ejemplo del sistema hi-fi. Ahora no hay una estructura de componentes sino de observables:

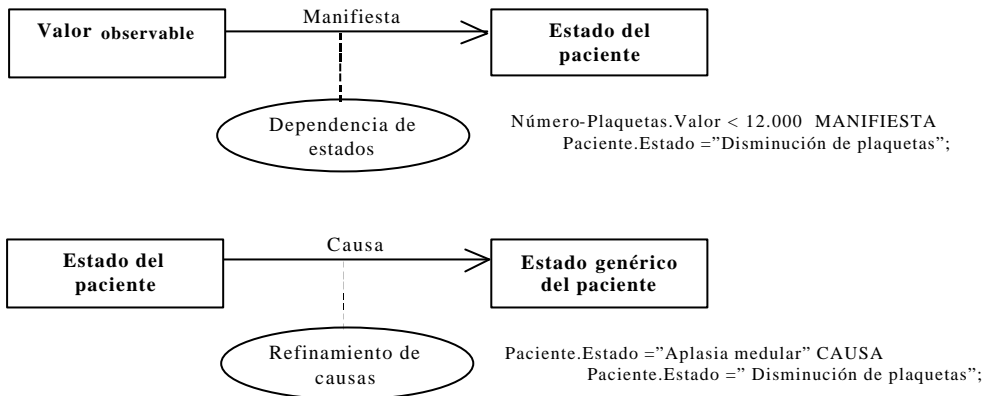


Con una jerarquía de hipótesis o estados patológicos que se muestra en la siguiente figura:

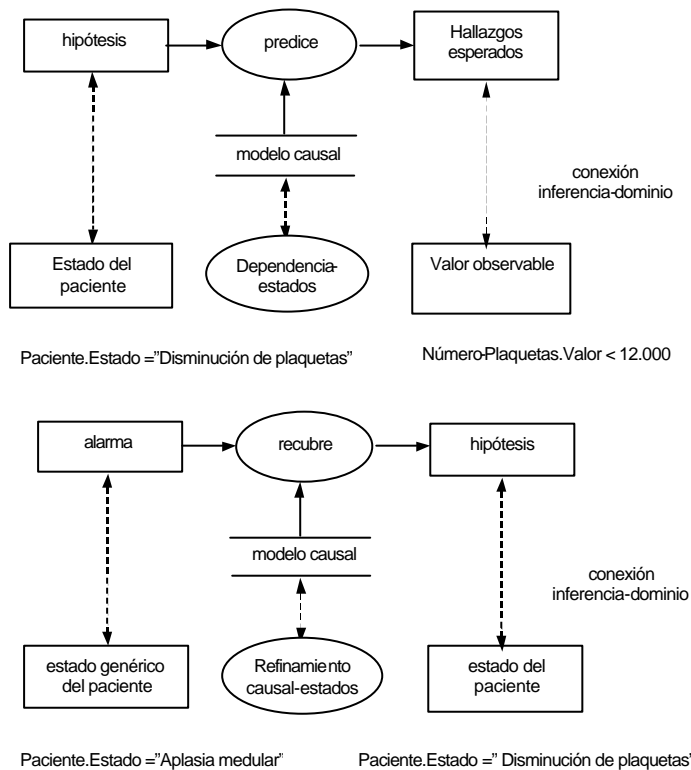


Las relaciones entre expresiones pueden ser prácticamente las mismas. La relación de refinamiento de causas o estados es idéntica. Por el contrario, en la relación de dependencia de estados se han cambiado las etiquetas de papeles y relación para adecuarlos a este dominio.





A través de estas relaciones, las inferencias quedarían conectadas con el dominio como sigue:



Para terminar, trasladando las pequeñas modificaciones que acabamos de ver a la descripción de la tarea y el método, tenemos acabado el modelo.



END TASK diagnóstico-hemorragia;

```

TASK-METHOD diagnóstico-a- través-sugiere-y-comprueba-sistemático;
  REALIZES: diagnóstico-hemorragiai;
  DECOMPOSITION:
    INFERENCES: recubre, predice, compara;
    TRANSFER-FUNCTIONS: obtiene;
  ROLES:
    INTERMEDIATE:
      hipótesis: "una posible solución";
      hallazgo-esperado: "un hallazgo que se debería observar si la hipótesis fuera cierta";
      hallazgo-observado : "el hallazgo realmente observado";
      resultado : "el resultado de la comparación";
    CONTROL-STRUCTURE:
      WHILE NEW-SOLUTION recubre(alarma-> hipótesis)
        WHILE  $\exists$  hipótesis no evaluada DO
          predice(hipótesis->hallazgo-esperado);
          obtiene(hallazgo-esperado-> hallazgo-observado);
          evidencia:=evidencia ADD hallazgo-observado);
          compara(hallazgo-observado+ hallazgo-esperado -> resultado);
          IF resultado == igual THEN BREAK-LOOP;
        ENDWHILE;
        IF resultado == igual THEN tipo-fallo:= hipótesis
                                alarma:=hipótesis
                                ELSE "no se ha encontrado solución"
        END IF;
      END TASK-METHOD diagnóstico-a- través-sugiere-y-comprueba-sistemático;

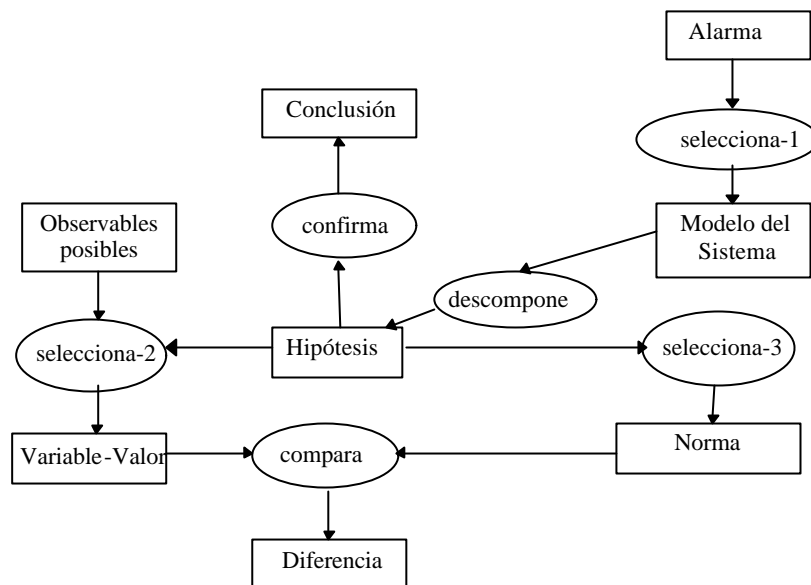
```

Modelos alternativos

Aunque el modelo que hemos presentado se adapta perfectamente a la descripción que hace el experto de su tarea, es posible tener diferentes “visiones” del mismo problema que supongan modelos alternativos. En esta sección vamos a resolver la misma tarea pero a través de dos PSMs diferentes, primeramente el método sistemático y posteriormente el heurístico.

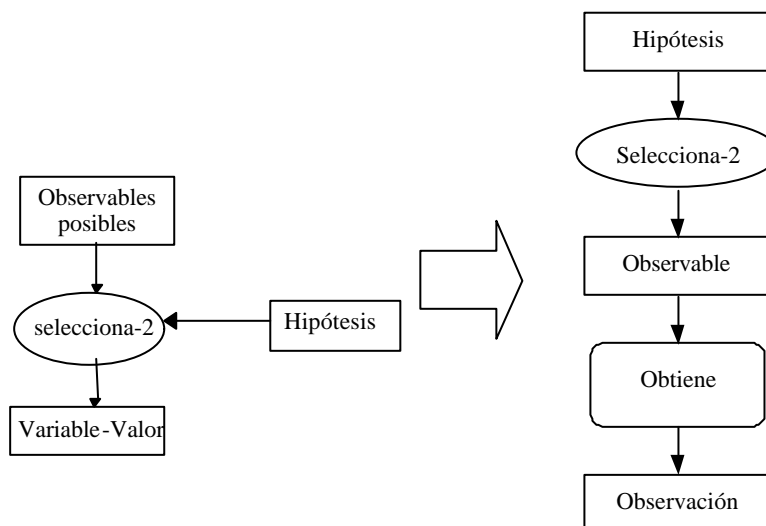
Diagnóstico sistemático. Recordamos en la siguiente figura el esquema inferencial de este método y su secuencia de control:





Diagnosis Sistemático (+alarma, + observables posibles, - modelo del sistema) by
 selecciona 1 (+alarma, - modelo del sistema)
 REPEAT
 decompose (+modelo del sistema, -hipótesis)
 WHILE número de hipótesis > 1
 select 2 (+observables posibles, +hipótesis, - valor de variable)
 select 3 (+hipótesis, -norma)
 compare (+valor de variable, +norma, -diferencia)
 modelo del sistema \leftarrow descomposición actual
 UNTIL confirma (+ hipótesis).

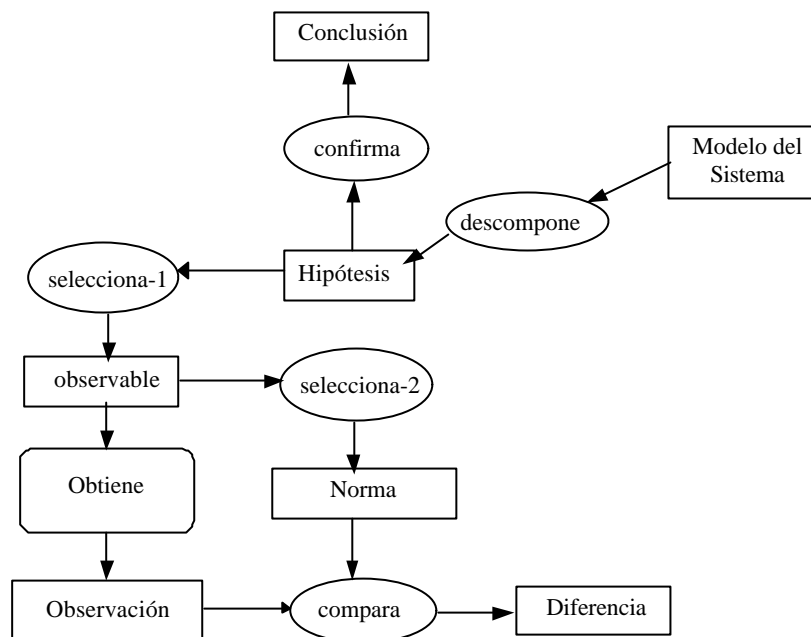
Vamos a introducir una pequeña adaptación, en tanto que requerimos una interacción con el usuario del sistema, necesitamos de la función de transferencia *obtiene*, cuyo resultado *variable-valor*, denominaremos *observación*:



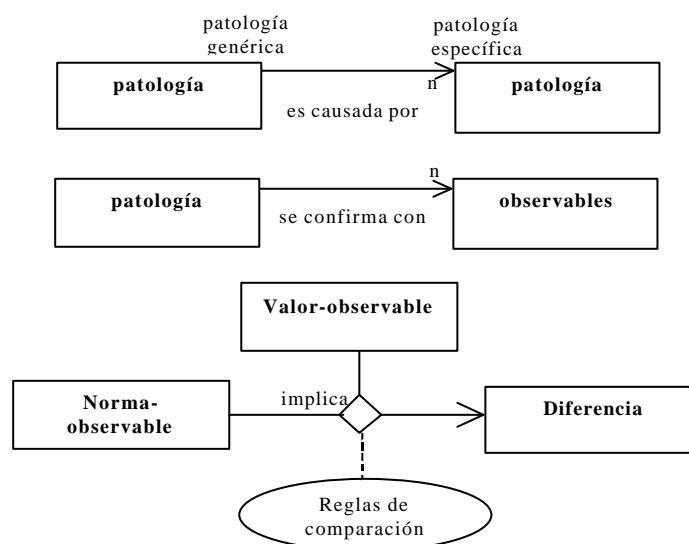
Por otro lado, en tanto que no hay una selección del modelo del sistema, vamos a obviar este paso, con lo cual nos quedan 4 inferencias una función de transferencia. Así, *Selecciona-2*



pasa a *Selecciona-1* y *Selecciona-3* a *Selecciona-2*. Hecha esta descomposición entre observable y observación, es también posible obtener la norma desde observable en lugar de obtenerla desde la hipótesis. Esto deja el esquema inferencial como sigue:



Ante este método es posible modelar el dominio de dos formas alternativas. En la primer, modelamos las patologías como objetos enlazados por una asociación “causado por” siguiente figura mostramos una versión del árbol de Fig. 2, en el que en cada nivel de descomposición tenemos las hipótesis o posibles causas. El principal atributo que incluimos en cada hipótesis es la prueba asociada. Por supuesto habría que incluir el resultado de normalidad o anormalidad de esta prueba que confirmase tal hipótesis.



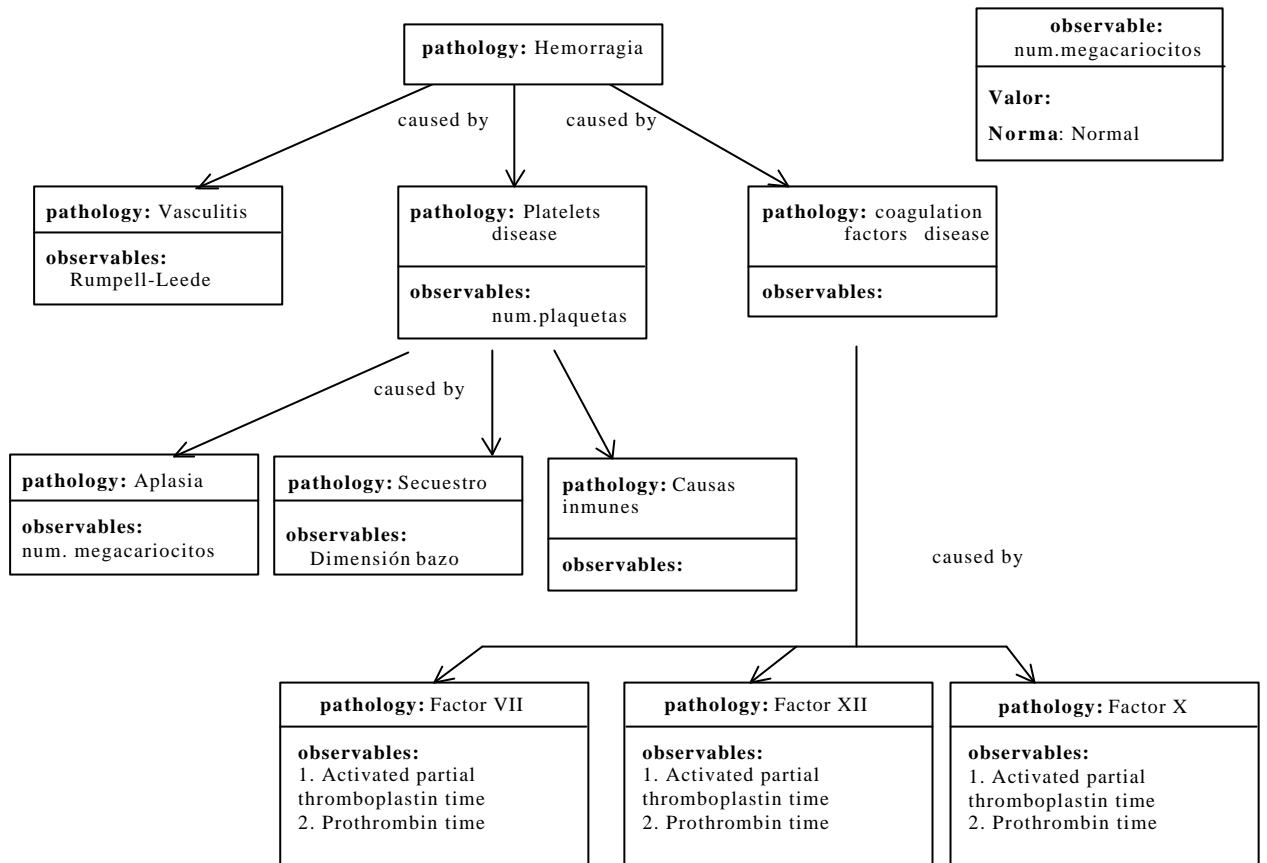
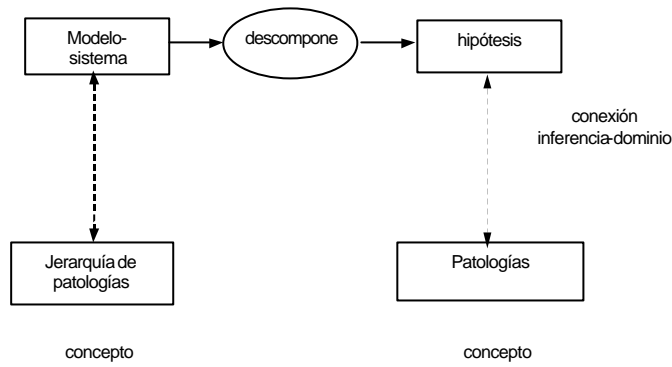


Fig. 3 Árbol causal donde cada posible causa (hipótesis) se descompone en otras causas de menor nivel.

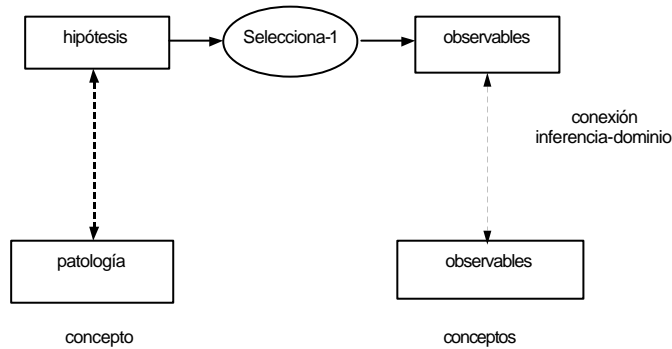




INFERENCE descompone;
 ROLES
 INPUT: modelo-sistema;
 OUTPUT: hipótesis
 STATIC
 SPECIFICATION:
 "No requiere un papel estático puesto que pueden obtenerse las hipótesis correspondientes por la asociación a través de la *es-causada-por* entre patologías abstractas y específicas"
 END INFERENCE descompone;

KNOWLEDGE-ROLE modelo-sistema;
 TYPE: DYNAMIC;
 DOMAIN-MAPPING: jerarquía-patologías;
 END KNOWLEDGE-ROLE modelo-sistema;
 KNOWLEDGE-ROLE modelo-causal;
 TYPE: STATIC;
 DOMAIN-MAPPING: dependencia-estados FROM modelo-vehículo;
 END KNOWLEDGE-ROLE modelo-causal;

KNOWLEDGE-ROLE hipótesis;
 TYPE: DYNAMIC;
 DOMAIN-MAPPING: patologías;
 END KNOWLEDGE-ROLE hipótesis;

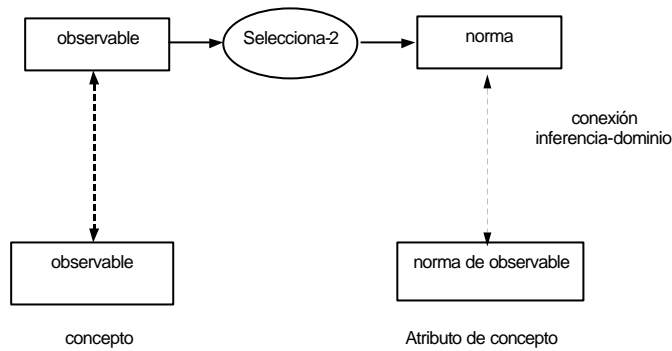


INFERENCE selecciona-1;
 ROLES
 INPUT: hipótesis;
 OUTPUT: observables;
 STATIC
 SPECIFICATION:
 "No requiere un papel estático puesto que pueden obtenerse las observables correspondientes por asociación a través de la relación *se-confirma-con* entre patologías y el valor de los
 END INFERENCE selecciona-1;

KNOWLEDGE-ROLE hipótesis;
 TYPE: DYNAMIC;
 DOMAIN-MAPPING: patología;
 END KNOWLEDGE-ROLE hipótesis;

KNOWLEDGE-ROLE observables;
 TYPE: DYNAMIC;
 DOMAIN-MAPPING: observable;
 END KNOWLEDGE-ROLE observables;





```

INFERENCE selecciona-2;
ROLES
  INPUT: observable;
  OUTPUT: norma;
  STATIC
SPECIFICATION:
  "No requiere un papel estático puesto que pueden obtenerse las
  normas desde los observables"
END INFERENCE selecciona-2;

```

```

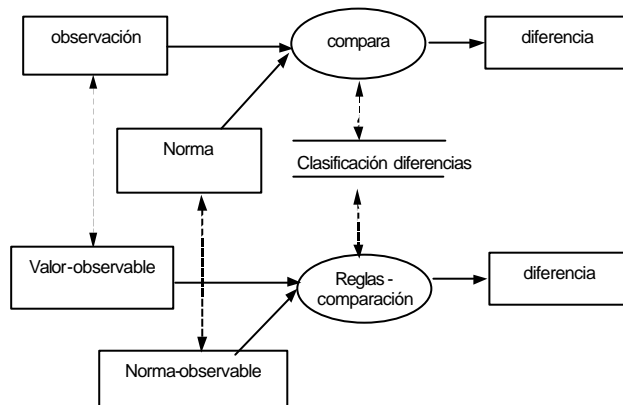
KNOWLEDGE-ROLE observable;
  TYPE: DYNAMIC;
  DOMAIN-MAPPING: observable;
END KNOWLEDGE-ROLE observable;

```

```

KNOWLEDGE-ROLE normas;
  TYPE: DYNAMIC;
  DOMAIN-MAPPING: observable.norma;
END KNOWLEDGE-ROLE normas;

```



```

INFERENCE compara;
ROLES
  INPUT: observación;
  norma;
  OUTPUT: diferencia;
  STATIC: clasificación de diferencias
SPECIFICATION:
  "Compara el valor de las observaciones con la norma de los
  observables, si es significativa la diferencia para considerar la
  hipótesis"
END INFERENCE compara;

```

```

KNOWLEDGE-ROLE observación;
  TYPE: DYNAMIC;
  DOMAIN-MAPPING: observable.valor;
END KNOWLEDGE-ROLE observación;

```

```

KNOWLEDGE-ROLE clasificación-diferencias;
  TYPE: STATIC;
  DOMAIN-MAPPING: reglas-comparación;
END KNOWLEDGE-ROLE clasificación-diferencias;

```

```

KNOWLEDGE-ROLE norma;
  TYPE: DYNAMIC;
  DOMAIN-MAPPING: observable.norma;
END KNOWLEDGE-ROLE norma;

```



Implementación.

En este apartado veremos diferentes implementaciones del mismo problema base, la tarea de diagnóstico según el método sistemático, cuyo modelo de pericia hemos estudiado en apartados previos. Obviamos la fase de diseño por no alargarnos demasiado y, gracias a la simplicidad del ejemplo, podemos ir directamente a la implementación, donde podremos observar características de aquél.

Implementación lineal.

En la primera solución que vamos a ofrecer, la más simple, adaptaremos el árbol de Las pruebas se integran como slots en un solo objeto Pruebas. Cada una de ellas es de tipo booleano, de tal forma que se interroga al usuario en terminos de sí o no (o verdadero o falso):

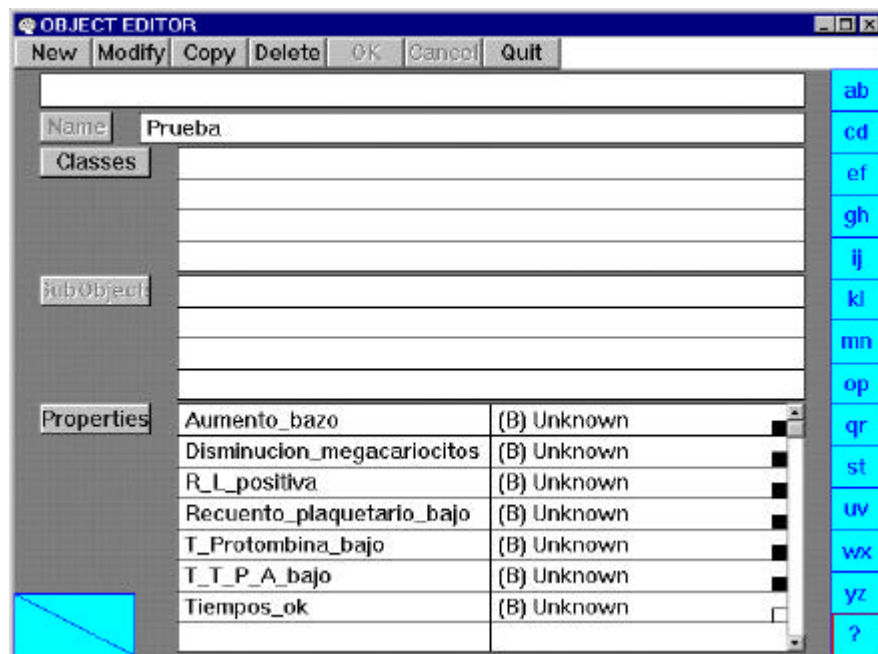


Fig. 4 Pantalla de edición del objeto Pruebas.



Teniendo en cuenta que vamos a utilizar un entorno como Nexpert orientado a la programación por reglas y que únicamente disponemos de 10 reglas por la limitación del programa demo disponibles por los alumnos, la opción más sencilla para una clasificación de este tipo es escribir tantas reglas como hipótesis de diagnóstico, en donde cada regla contiene en sus premisas el encadenamiento de condiciones para la confirmación de esas hipótesis. Es la denominada **solución lineal**. Así, únicamente se requiere una hipótesis (hipótesis de reglas no hipótesis de diagnóstico) para todas las reglas, variable que denominamos *realizado*. Cada regla deja como resultado el nombre de la hipótesis confirmada y no descomponible en la variable resultado. Ambas, *resultado* y *realizado* se integran como slots del objeto *diagnóstico*:



Fig. 5 Volcado del editor de objetos del entorno Nexpert Object

Por ejemplo en la imagen de la Fig. 6, volcado de una pantalla Nexpert, observamos la regla que conduce a la confirmación de un diagnóstico de Secuestro leucocitario, a partir de la concatenación de una serie de condiciones.



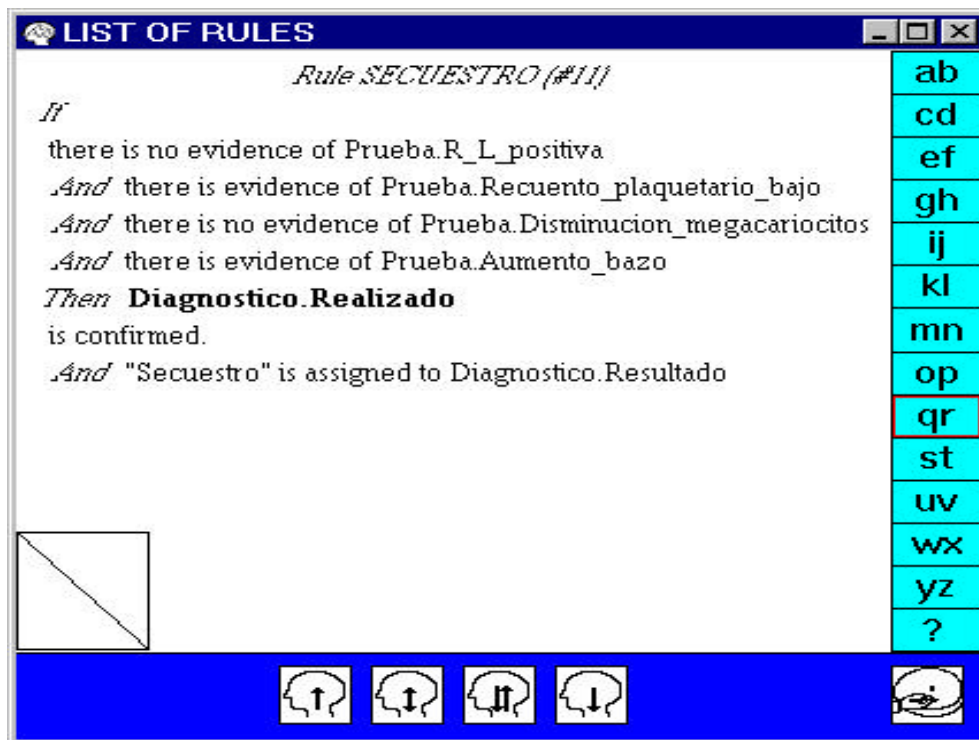


Fig. 6 Volcado de la pantalla List of Rules del entorno Nexpert Object que nos muestra una de las reglas.

En lenguaje Nexpert, esta regla sería:

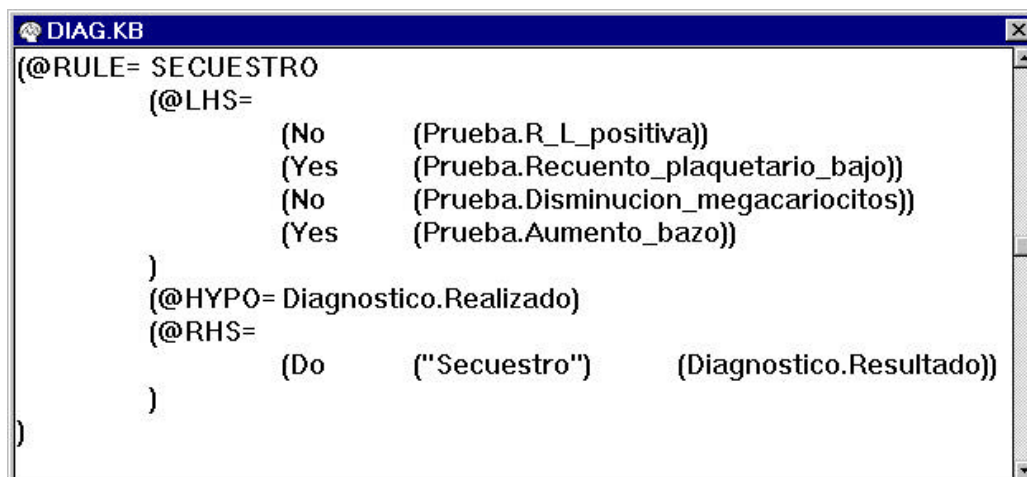


Fig. 7 Regla *Secuestro* en lenguaje Nexpert Object

Puesto que las premisas de la reglas se han expresado en el mismo orden que seguirían al recorrer el árbol, el conjunto de estas reglas se comportarían como un recorrido sistemático

l significa que sí, que estamos trabajando según el método sistemático, aunque el conjunto de reglas tenga poco parecido estructural con el modelo de pericia expuesto.



A continuación incluimos todo el listado del programa Nexpert según la implementación lineal:

```
(@VERSION= 020)
(@PROPERTY= Aumento_bazo @TYPE=Boolean;)
(@PROPERTY= Disminucion_megacariocitos @TYPE=Boolean;)
(@PROPERTY= Puncion @TYPE=String;)
(@PROPERTY= R_L_positiva @TYPE=Boolean;)
(@PROPERTY= Realizado @TYPE=Boolean;)
(@PROPERTY= Recuento_plaquetario_bajo @TYPE=Boolean;)
(@PROPERTY= Resultado @TYPE=String;)
(@PROPERTY= T_Protombina_bajo @TYPE=Boolean;)
(@PROPERTY= T_T_P_A_bajo @TYPE=Boolean;)
(@PROPERTY= Tiempos_ok @TYPE=Boolean;)

(@OBJECT= Diagnostico
  (@PROPERTIES=
    Realizado
    Resultado
  )
)
(@OBJECT= Prueba
  (@PROPERTIES=
    Aumento_bazo
    Disminucion_megacariocitos
    R_L_positiva
    Recuento_plaquetario_bajo
    T_Protombina_bajo
    T_T_P_A_bajo
    Tiempos_ok
  )
)
(@SLOT= Diagnostico.Resultado
  (@CACTIONS=
    (Execute("WriteTo") (@ATOMID=Diagnostico.Resultado;@STRING="@TRANSCRIPT,\
@TEXT=El resultado del proceso es:");)
  )
)
(@SLOT= Prueba.Aumento_bazo
  @PROMPT="Se ha observado aumento del bazo?";
  @FORMAT="Si;No";
)
(@SLOT= Prueba.Disminucion_megacariocitos
  @PROMPT="En la puncion, se ha observado disminucion o ausencia de megacariocitos?";
  @FORMAT="Si;No";
)
(@SLOT= Prueba.R_L_positiva
  @PROMPT="La prueba de Rumpell_Leede ha resultado positiva?";
  @FORMAT="Si;No";
)
(@SLOT= Prueba.Recuento_plaquetario_bajo
  @PROMPT="Ha resultado bajo el recuento de plaquetas?";
  @FORMAT="Si;No";
)
(@SLOT= Prueba.T_Protombina_bajo
  @PROMPT="Es bajo el tiempo de Protombina?";
  @FORMAT="Si;No";
```



```

)
(@SLOT=      Prueba.T_T_P_A_bajo
  @PROMPT="Es bajo el tiempo de tromboplastina parcial activado?";
  @FORMAT="Si;No";
)
(@RULE=      VASCULITIS
  (@LHS=
    (Yes      (Prueba.R_L_positiva))
  )
  (@HYPO=    Diagnostico.Realizado)
  (@RHS=
    (Do       ("Vasculitis")    (Diagnostico.Resultado))
  )
)
)
(@RULE=      SECUESTRO
  (@LHS=
    (No       (Prueba.R_L_positiva))
    (Yes      (Prueba.Recuento_plaquetario_bajo))
    (No       (Prueba.Disminucion_megacariocitos))
    (Yes      (Prueba.Aumento_bazo))
  )
  (@HYPO=    Diagnostico.Realizado)
  (@RHS=
    (Do       ("Secuestro")    (Diagnostico.Resultado))
  )
)
)
(@RULE=      ENFERMEDAD_FACTORES_COAGULACION
  (@LHS=
    (No       (Prueba.R_L_positiva))
    (No       (Prueba.Recuento_plaquetario_bajo))
    (Yes      (Prueba.Tiempos_ok))
  )
  (@HYPO=    Diagnostico.Realizado)
)
)
(@RULE=      CAUSAS_IMUNES
  (@LHS=
    (No       (Prueba.R_L_positiva))
    (Yes      (Prueba.Recuento_plaquetario_bajo))
    (No       (Prueba.Disminucion_megacariocitos))
    (No       (Prueba.Aumento_bazo))
  )
  (@HYPO=    Diagnostico.Realizado)
  (@RHS=
    (Do       ("Causas inmunes")    (Diagnostico.Resultado))
  )
)
)
(@RULE=      APLASIA_MEDULAR
  (@LHS=
    (No       (Prueba.R_L_positiva))
    (Yes      (Prueba.Recuento_plaquetario_bajo))
    (Yes      (Prueba.Disminucion_megacariocitos))
  )
  (@HYPO=    Diagnostico.Realizado)
  (@RHS=
    (Do       ("Aplasia medular")    (Diagnostico.Resultado))
  )
)
)

```




```

)
(@RULE=      DEFICIENCIA_FACTOR_XII
  (@LHS=
    (Yes      (Prueba.T_T_P_A_bajo))
    (No       (Prueba.T_Protombina_bajo))
  )
  (@HYPO=    Prueba.Tiempos_ok)
  (@RHS=
    (Do      ("Deficiencia factor VIII, IX, XI, o XII")    (Diagnostico.Resultado))
  )
)
)
(@RULE=      DEFICIENCIA_FACTOR_X
  (@LHS=
    (Yes      (Prueba.T_T_P_A_bajo))
    (Yes      (Prueba.T_Protombina_bajo))
  )
  (@HYPO=    Prueba.Tiempos_ok)
  (@RHS=
    (Do      ("Deficiencia factor V, X, Protombina o Fibrinogeno")    (Diagnostico.Resultado))
  )
)
)
(@RULE=      DEFICIENCIA_FACTOR_VII
  (@LHS=
    (No       (Prueba.T_T_P_A_bajo))
    (Yes      (Prueba.T_Protombina_bajo))
  )
  (@HYPO=    Prueba.Tiempos_ok)
  (@RHS=
    (Do      ("Deficiencia factor VII") (Diagnostico.Resultado))
  )
)
)
(@GLOBALS=
  @INHVALUP=FALSE;
  @INHVALDOWN=TRUE;
  @INHOBJUP=FALSE;
  @INHOBJDOWN=FALSE;
  @INHCLASSUP=FALSE;
  @INHCLASSDOWN=TRUE;
  @INHBREADTH=TRUE;
  @INHPARENT=FALSE;
  @PWTRUE=TRUE;
  @PWFALSE=FALSE;
  @PWNOTKNOWN=FALSE;
  @EXHBWRD=TRUE;
  @PTGATES=TRUE;
  @PFACTIONS=TRUE;
  @SOURCESON=TRUE;
  @CACTIONSON=TRUE;
  @SUGLIST=Diagnostico.Realizado;
)

```

Implementación diferencial.

Una segunda posible codificación mediante reglas (ya no admisible en Nexpert Demo, pues consta de 12 reglas) utiliza la estrategia “backward chaining” de Nexpert para encadenar



reglas y descender por el árbol del diagnóstico. Este tipo de implementación se denomina “por diferenciación”. Las reglas ya no forman una estructura plana (como en la versión anterior) sino que su estructura es un reflejo bastante directo de la estructura del dominio, del árbol del diagnóstico. Por ejemplo en la siguiente figura se observa una fracción del árbol de reglas que coincide con el subárbol que desciende desde la hipótesis “Problema plaquetario”:

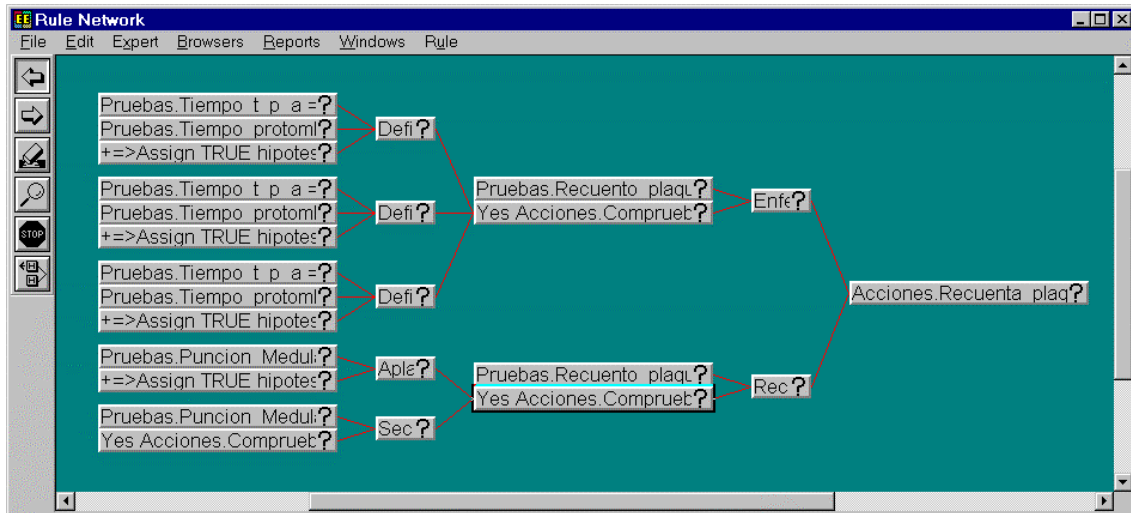


Fig. 8 Sección del árbol de reglas para la implementación por “diferenciación”. Se muestran las reglas encadenadas para el recorrido del árbol desde la hipótesis

El código completo de esta versión es el siguiente:

```
(@VERSION= 040)
(@PROPERTY= Aplasia_medular @TYPE=Boolean;)
(@PROPERTY= Causas_inmunes @TYPE=Boolean;)
(@PROPERTY= Comprueba_bazo @TYPE=Boolean;)
(@PROPERTY= Comprueba_factores @TYPE=Boolean;)
(@PROPERTY= Comprueba_medula_osea @TYPE=Boolean;)
(@PROPERTY= comprueba_vasculitis @TYPE=Boolean;)
(@PROPERTY= Deficiencia_factor_VII @TYPE=Boolean;)
(@PROPERTY= Deficiencia_factor_X @TYPE=Boolean;)
(@PROPERTY= Deficiencia_factor_XII @TYPE=Boolean;)
(@PROPERTY= Hemorragia @TYPE=Boolean;)
(@PROPERTY= Puncion_Medula_osea @TYPE=String;)
(@PROPERTY= Recuenta_plaquetas @TYPE=Boolean;)
(@PROPERTY= Recuento_plaquetas @TYPE=String;)
(@PROPERTY= Rumpell_leede @TYPE=String;)
(@PROPERTY= Secuestro_leucocitario @TYPE=Boolean;)
(@PROPERTY= Tamanho_bazo @TYPE=String;)
(@PROPERTY= Tiempo_protombina @TYPE=String;)
(@PROPERTY= Tiempo_t_p_a @TYPE=String;)
(@PROPERTY= Vasculitis @TYPE=Boolean;)
```

```
(@OBJECT= Acciones
(@PUBLICPROPS=
```



```

    Comprueba_bazo
    Comprueba_factores
    Comprueba_medula_osea
    comprueba_vasculitis
    Recuenta_plaquetas
)
)
(@OBJECT=    hipotesis
  (@PUBLICPROPS=
    Aplasia_medular
    Causas_inmunes
    Deficiencia_factor_VII
    Deficiencia_factor_X
    Deficiencia_factor_XII
    Hemorragia
    Secuestro_leucocitario
    Vasculitis
  )
)
)
(@OBJECT=    Pruebas
  (@PUBLICPROPS=
    Puncion_Medula_osea
    Recuento_plaquetas
    Rumpell_leede
    Tamanho_bazo
    Tiempo_protombina
    Tiempo_t_p_a
  )
)
)
(@META=      Pruebas.Puncion_Medula_osea
  @PROMPT="El resultado de la puncion en medula osea fue?";
)
(@META=      Pruebas.Recuento_plaquetas
  @PROMPT="El resultado del recuento de plaquetas fue?";
)
(@META=      Pruebas.Rumpell_leede
  @PROMPT="El resultado de la prueba de Rumpell-Leede fue?";
)
(@META=      Pruebas.Tamanho_bazo
  @PROMPT="El tamanho del bazo es?";
)
(@META=      Pruebas.Tiempo_protombina
  @PROMPT="El Tiempo de protombina es?";
)
(@META=      Pruebas.Tiempo_t_p_a
  @PROMPT="El tiempo de tromboplastina parcial activado es?";
)
)

(@RULE=      Aplasia_medular
  (@LHS=
    (=      (Pruebas.Puncion_Medula_osea) ("Disminucion megacariocitos"))
  )
  (@HYPO=    Acciones.Comprueba_medula_osea)
  (@RHS=

```



```

    (Assign (TRUE) (hipotesis.Aplasia_medular))
  )
)

(@RULE=      Causas_inmunes
  (@LHS=
    (=      (Pruebas.Tamanho_bazo) ("Normal"))
  )
  (@HYPO=    Acciones.Comprueba_bazo)
  (@RHS=
    (Assign (TRUE) (hipotesis.Causas_inmunes))
  )
)

(@RULE=      Deficit_factor_VII
  (@LHS=
    (=      (Pruebas.Tiempo_t_p_a) ("Normal"))
    (=      (Pruebas.Tiempo_protombina) ("Bajo"))
  )
  (@HYPO=    Acciones.Comprueba_factores)
  (@RHS=
    (Assign (TRUE) (hipotesis.Deficiencia_factor_VII))
  )
)

(@RULE=      Deficit_factor_X_VII_Protombina_Fibrinogeno
  (@LHS=
    (=      (Pruebas.Tiempo_t_p_a) ("Bajo"))
    (=      (Pruebas.Tiempo_protombina) ("Bajo"))
  )
  (@HYPO=    Acciones.Comprueba_factores)
  (@RHS=
    (Assign (TRUE) (hipotesis.Deficiencia_factor_X))
  )
)

(@RULE=      Deficit_factor_XII_XI_IX_VII
  (@LHS=
    (=      (Pruebas.Tiempo_t_p_a) ("Bajo"))
    (=      (Pruebas.Tiempo_protombina) ("Normal"))
  )
  (@HYPO=    Acciones.Comprueba_factores)
  (@RHS=
    (Assign (TRUE) (hipotesis.Deficiencia_factor_XII))
  )
)

(@RULE=      Diagnostico_de_hemorragias
  (@LHS=
    (Yes    (Acciones.comprueba_vasculitis))
  )
  (@HYPO=    hipotesis.Hemorragia)
)

(@RULE=      Enfermedad_de_los_factores_de_coagulacion
  (@LHS=
    (=      (Pruebas.Recuento_plaquetas) ("Normal"))
    (Yes    (Acciones.Comprueba_factores))
  )
  (@HYPO=    Acciones.Recuenta_plaquetas)
)

```



```

(@RULE=      No_Vasculitis
  (@LHS=
    (=      (Pruebas.Rumpell_leede) ("Negativa"))
    (Yes    (Acciones.Recuenta_plaquetas))
  )
  (@HYPO=    Acciones.comprueba_vasculitis)
)
(@RULE=      Recuento_bajo_de_plaquetas
  (@LHS=
    (=      (Pruebas.Recuento_plaquetas) ("Bajo"))
    (Yes    (Acciones.Comprueba_medula_osea))
  )
  (@HYPO=    Acciones.Recuenta_plaquetas)
)
(@RULE=      Secuestro
  (@LHS=
    (=      (Pruebas.Tamanho_bazo) ("Excesivo"))
  )
  (@HYPO=    Acciones.Comprueba_bazo)
  (@RHS=
    (Assign (TRUE) (hipotesis.Secuestro_leucocitario))
  )
)
(@RULE=      Secuestro_o_causas_inmunes
  (@LHS=
    (=      (Pruebas.Puncion_Medula_osea) ("Normal"))
    (Yes    (Acciones.Comprueba_bazo))
  )
  (@HYPO=    Acciones.Comprueba_medula_osea)
)
(@RULE=      Vasculitis
  (@LHS=
    (=      (Pruebas.Rumpell_leede) ("Positiva"))
  )
  (@HYPO=    Acciones.comprueba_vasculitis)
  (@RHS=
    (Assign (TRUE) (hipotesis.Vasculitis))
  )
)

(@GLOBALS=
  @INHVALUP=FALSE;
  @INHVALDOWN=TRUE;
  @INHOBJUP=FALSE;
  @INHOBJDOWN=FALSE;
  @INHCLASSUP=FALSE;
  @INHCLASSDOWN=TRUE;
  @INHBREADTH=TRUE;
  @INHPARENT=FALSE;
  @PWTRUE=TRUE;
  @PWFALSE=TRUE;
  @PWNOTKNOWN=TRUE;
  @EXHBWRD=TRUE;
  @PTGATES=TRUE;
  @PFACTIONS=TRUE;
  @SOURCESON=TRUE;

```



```

@CACTIONSON=TRUE;
@VALIDUSER=FALSE;
@VALIDENGINE=FALSE;
@PFEACTIONS=FALSE;
@PFMACTIONS=GLOBAL;
@PFMEACTIONS=FALSE;
@SUGLIST=
    hipotesis.Hemorragia;
)

```

Implementación declarativa.

El siguiente ejemplo de implementación¹ que vamos a mostrar para el mismo problema, tiene la ventaja sobre las anteriores de la genericidad. Esto es, la implementación va a servir para cualquier dominio, siempre que cumpla una determinada condición. Es la solución denominada **declarativa**. En nuestro caso, esta condición es que pueda expresarse mediante un árbol binario, cosa que nuestro ejemplo cumple como se muestra en la figura:

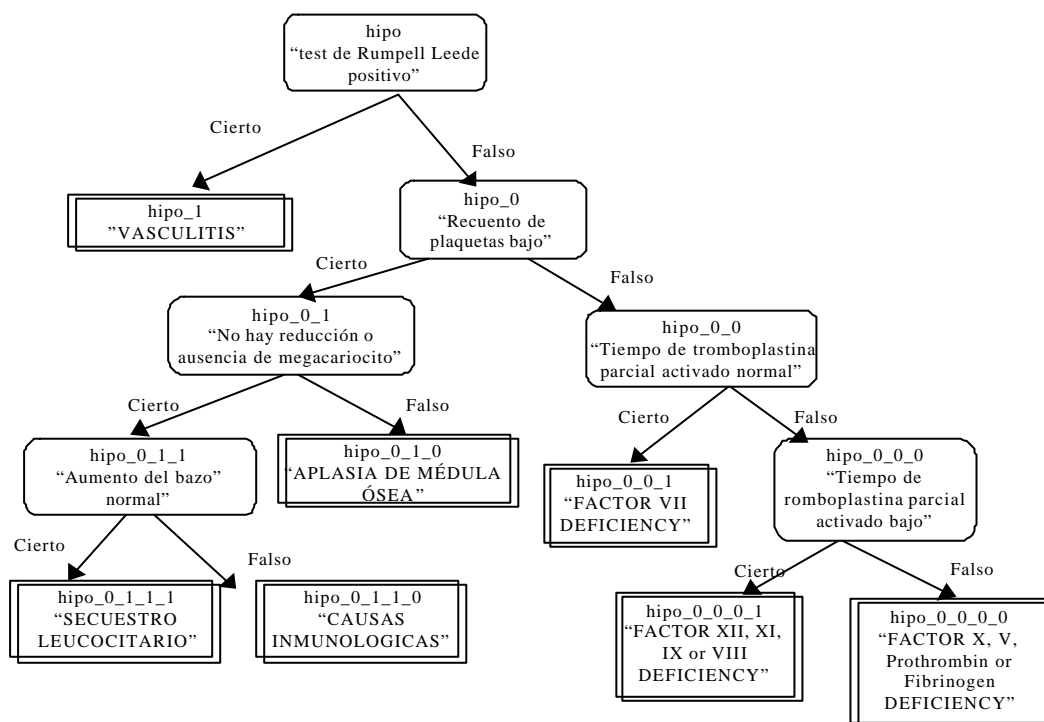


Fig. 9 Árbol de hipótesis transformado en árbol binario.

El recorrido del árbol se ha simplificado hasta el máximo (la hipótesis coincide con el resultado de la prueba, por ejemplo) y, gracias a ello, es automático, de tal forma que, con solo cuatro reglas, el encadenamiento backward recorre todo el árbol. Esto se consigue utilizando

¹ Se pueden ver más detalles de este ejemplo en el documento “Aspectos metodológicos básicos en el



variables de la mismo forma que *prolog*, el valor de un slot es variable y referencia a las sucesivas hipótesis.

Para mayor simplicidad en la codificación de las reglas, se ha utilizado una codificación especial en los nombres de las hipótesis que indica la secuencia recorrida (o para falso, 1 para cierto) hasta llegar a la hipótesis.

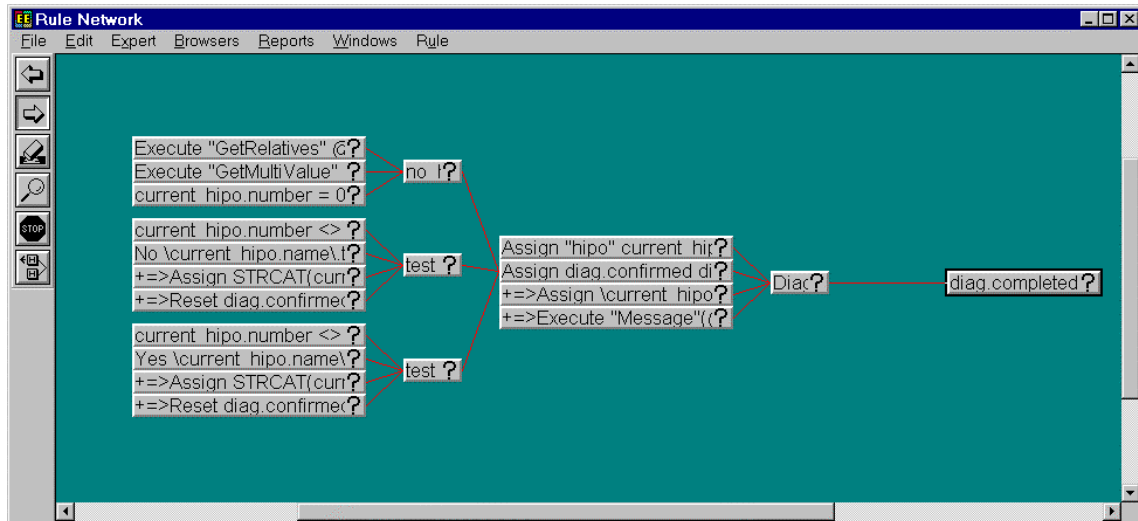


Fig. 10 Con solo cuatro reglas se puede recorrer el árbol binario de Fig. 9.

Esto es a cargo de que la regla se vuelve menos legible al utilizar variables que tomaran valores en las sucesivas hipótesis:



Fig. 11 La regla se hace más compleja por la utilización de variables interpretadas (o declarativas).



Lo que realmente estamos haciendo es implementar un PSM, a partir del PSM backward chaining, reutilizable como éste en diferentes dominios. Se ha elevado el nivel del PSM a costa de elevar el nivel de las restricciones.

Incluimos también aquí el código completo para mayor detalle:

```
(@VERSION= 040)
(@PROPERTY= affirmation @TYPE=String;)
(@PROPERTY= children @TYPE=String;)
(@PROPERTY= completed @TYPE=Boolean;)
(@PROPERTY= confirmed @TYPE=Boolean;)
(@PROPERTY= herency @TYPE=Boolean;)
(@PROPERTY= name @TYPE=String;)
(@PROPERTY= number @TYPE=Integer;)
(@PROPERTY= sun @TYPE=String;)
(@PROPERTY= test @TYPE=Boolean;)
```

```
(@CLASS= diagnosis
  (@PUBLICPROPS=
    affirmation
    completed
    confirmed
    name
  )
)
(@CLASS= hypothesis
  (@PUBLICPROPS=
    name
    test
  )
)
(@OBJECT= current_hipo
  (@CLASSES=
    hypothesis
  )
  (@PUBLICPROPS=
    children
    name
    number
    sun
    test
  )
)
(@OBJECT= diag
  (@CLASSES=
    diagnosis
  )
  (@PUBLICPROPS=
    affirmation
    completed
    confirmed
    herency
    name
  )
)
```




```

)
(@OBJECT=    hipo
  (@CLASSES=
    hypothesis
  )
  (@SUBOBJECTS=
    hipo_0
    hipo_1
  )
  (@PUBLICPROPS=
    name
    test
  )
)
)
(@OBJECT=    hipo_0
  (@CLASSES=
    hypothesis
  )
  (@SUBOBJECTS=
    hipo_0_0
    hipo_0_1
  )
  (@PUBLICPROPS=
    name
    test
  )
)
)
(@OBJECT=    hipo_0_0
  (@CLASSES=
    hypothesis
  )
  (@SUBOBJECTS=
    hipo_0_0_0
    hipo_0_0_1
  )
  (@PUBLICPROPS=
    name
    test
  )
)
)
(@OBJECT=    hipo_0_0_0
  (@CLASSES=
    hypothesis
  )
  (@SUBOBJECTS=
    hipo_0_0_0_0
    hipo_0_0_0_1
  )
  (@PUBLICPROPS=
    name
    test
  )
)
)
(@OBJECT=    hipo_0_0_0_0
  (@CLASSES=
    hypothesis

```



```

    )
    (@PUBLICPROPS=
        name
        test
    )
)
(@OBJECT=    hipo_0_0_0_1
    (@CLASSES=
        hypothesis
    )
    (@PUBLICPROPS=
        name
        test
    )
)
)
(@OBJECT=    hipo_0_0_1
    (@CLASSES=
        hypothesis
    )
    (@PUBLICPROPS=
        name
        test
    )
)
)
(@OBJECT=    hipo_0_1
    (@CLASSES=
        hypothesis
    )
    (@SUBOBJECTS=
        hipo_0_1_0
        hipo_0_1_1
    )
    (@PUBLICPROPS=
        name
        test
    )
)
)
(@OBJECT=    hipo_0_1_0
    (@CLASSES=
        hypothesis
    )
    (@PUBLICPROPS=
        name
        test
    )
)
)
(@OBJECT=    hipo_0_1_1
    (@CLASSES=
        hypothesis
    )
    (@SUBOBJECTS=
        hipo_0_1_1_0
        hipo_0_1_1_1
    )
    (@PUBLICPROPS=
        name

```



```

        test
    )
)
(@OBJECT=    hipo_0_1_1_0
  (@CLASSES=
    hypothesis
  )
  (@PUBLICPROPS=
    name
    test
  )
)
)
(@OBJECT=    hipo_0_1_1_1
  (@CLASSES=
    hypothesis
  )
  (@PUBLICPROPS=
    name
    test
  )
)
)
(@OBJECT=    hipo_1
  (@CLASSES=
    hypothesis
  )
  (@PUBLICPROPS=
    name
    test
  )
)
)
(@META=      |hypothesis|.test
  @PROMPT="Se cumple @V(@self.name) ?";
)

(@META=      diag.name
  @INITVAL= "Diagnostico de hemorragias")
)

(@META=      hipo.name
  @INITVAL= "La Prueba de Rumpell-Leede es positiva")
)
(@META=      hipo.test
  @INHVALUP=FALSE;
  @INHVALDOWN=FALSE;
)
(@META=      hipo_0.name
  @INHVALUP=FALSE;
  @INHVALDOWN=FALSE;
  @INITVAL= "El recuento de plaquetas es bajo")
)
(@META=      hipo_0.test
  @INHVALUP=FALSE;
  @INHVALDOWN=FALSE;
)
(@META=      hipo_0_0.name
  @INHVALUP=FALSE;

```



```

    @INHVALDOWN=FALSE;
    (@INITVAL= "El tiempo de tromboplastina es normal y el de protombina es bajo")
)
(@META=      hipo_0_0_0.name
  @INHVALUP=FALSE;
  @INHVALDOWN=FALSE;
  (@INITVAL= "El tiempo de tromboplastina es bajo y el de protombina es normal")
)
(@META=      hipo_0_0_0_0.name
  @INHVALUP=FALSE;
  @INHVALDOWN=FALSE;
  (@INITVAL= "Deficiencia en los factores X,\
V Protombina o Fibrinogeno")
)
(@META=      hipo_0_0_0_1.name
  (@INITVAL= "Deficiencia en factores XII,\
XI,IX o VIII")
)
(@META=      hipo_0_0_1.name
  (@INITVAL= "Deficiencia en el factor VII")
)
(@META=      hipo_0_1.name
  (@INITVAL= "La muestra de medula osea es normal")
)
(@META=      hipo_0_1_1.name
  (@INITVAL= "Hay aumento del bazo")
)
(@META=      hipo_0_1_1_0.name
  (@INITVAL= "Causas inmunologicas")
)
(@META=      hipo_0_1_1_1.name
  (@INITVAL= "Secuestro leucocitario")
)
(@META=      hipo_1.name
  (@INITVAL= "Vasculitis")
)
(@RULE=      Diagnostic_completed
  (@LHS=
    (Assign ("hipo")(current_hipo.name))
    (Assign (diag.confirmed) (diag.confirmed))
  )
  (@HYPO=      diag.completed)
  (@RHS=
    (Assign (\current_hipo.name\name)      (diag.name))
    (Execute ("Message")      (@WAIT=TRUE;@STRING="@TEXT=El resultado del diagnostico e\
s...@V(diag.name),@OK";))
  )
)
(@RULE=      no_hay_descendientes
  @INFCAT=2;
  (@LHS=
    (Execute ("GetRelatives") (@WAIT=TRUE;@ATOMID=\current_hipo.name\;\
@STRING="@OBJECTS,@CHILDREN,@ONELEVEL,@RETURN=current_hipo.children";))
    (Execute ("GetMultiValue")      (@WAIT=TRUE;\
@ATOMID=current_hipo.children;@STRING="@RETURN=current_hipo.number,\
@LENGTH,@STRAT=SET";))

```



```

        (=      (current_hipo.number)    (0))
    )
    (@HYPO=     diag.confirmed)
)
(@RULE=       test_negativo
  (@LHS=
    (<>      (current_hipo.number)    (0))
    (No      (\current_hipo.name\test))
  )
  (@HYPO=     diag.confirmed)
  (@RHS=
    (Assign (STRCAT(current_hipo.name,"_0"))      (current_hipo.name))
    (Reset  (diag.confirmed))
  )
)
)
(@RULE=       test_positivo
  (@LHS=
    (<>      (current_hipo.number)    (0))
    (Yes     (\current_hipo.name\test))
  )
  (@HYPO=     diag.confirmed)
  (@RHS=
    (Assign (STRCAT(current_hipo.name,"_1"))      (current_hipo.name))
    (Reset  (diag.confirmed))
  )
)
)

(@GLOBALS=
  @INHVALUP=FALSE;
  @INHVALDOWN=FALSE;
  @INHOBJUP=FALSE;
  @INHOBJDOWN=FALSE;
  @INHCLASSUP=FALSE;
  @INHCLASSDOWN=TRUE;
  @INHBREADTH=TRUE;
  @INHPARENT=FALSE;
  @PWTRUE=TRUE;
  @PWFALSE=TRUE;
  @PWNOTKNOWN=TRUE;
  @EXHBWRD=TRUE;
  @PTGATES=TRUE;
  @PFACTIONS=TRUE;
  @SOURCESON=TRUE;
  @CACTIONSON=TRUE;
  @VALIDUSER=FALSE;
  @VALIDENGINE=FALSE;
  @PFEACTIONS=FALSE;
  @PFMACTIONS=GLOBAL;
  @PFMEASCTIONS=FALSE;
  @SUGLIST=
    diag.completed;
)

```



rragias que seguimos como ejemplo. Al cambiar el método, cambia el procedimiento de diagnóstico, si el método sistemático se fundamenta en el recorrido del árbol de descomposición de hipótesis, el método heurístico se basa en el emparejamiento entre las observaciones (a un cierto nivel de abstracción) y las soluciones del diagnóstico (refinables).

Remitimos al lector de nuevo al documento “Aspectos metodológicos básicos de desarrollo de SBC’s” donde se da una explicación más detallada de este metodo aplicable a la tarea de diagnóstico y ejemplificado en otro problema diferente. Como allí se ve, la estructura inferencial es la siguiente:

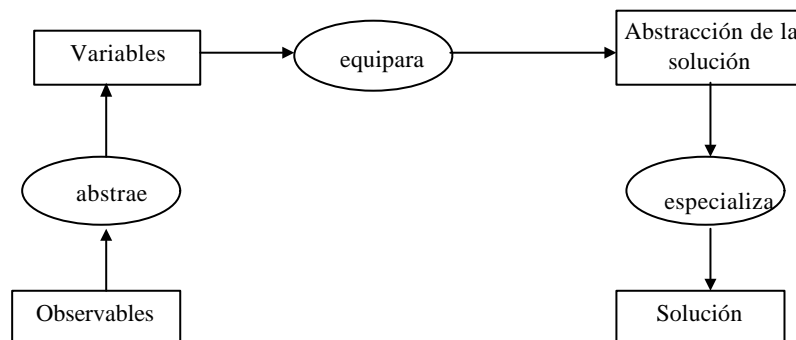


Fig. 12. Diagrama de Inferencias del método de *Clasificación Heurística*.

Para la estructura de control existen diferentes posibilidades, aquí hemos traído una como ejemplo (expresada en pseudocódigo) en la que el control viene dado por los datos, es la *data-driven* o *forward reasoning*.

Clasificación heurística (Observables, -> Solución) abstrae (Observables -> Variables) equipara (Variables -> Abstracción de la solución) especializa (Abstracción de la solución -> Solución).
--

En el diagnóstico sistemático, la secuencia de introducción (o de consulta, si ya se han introducido previamente) de las observaciones viene dirigida por las hipótesis (conforme se descendiende por el árbol). En el método heurístico, dirigido por datos, son estos los que dirigen el control. El sistema infiere un resultado a partir de las observaciones contenidas.

Los datos se han diferenciado en observaciones abstractas (como slots del objeto *Datos_Abstractos*) y observaciones numéricas (como slot del objeto *Pruebas*). Estas últimas son las que requieren la inferencia de abstracción.

En la siguiente imagen se muestra el encadenamiento de tres reglas, las dos primeras (a la izquierda) implementan sendas abstracciones de observaciones numéricas y la tercera una

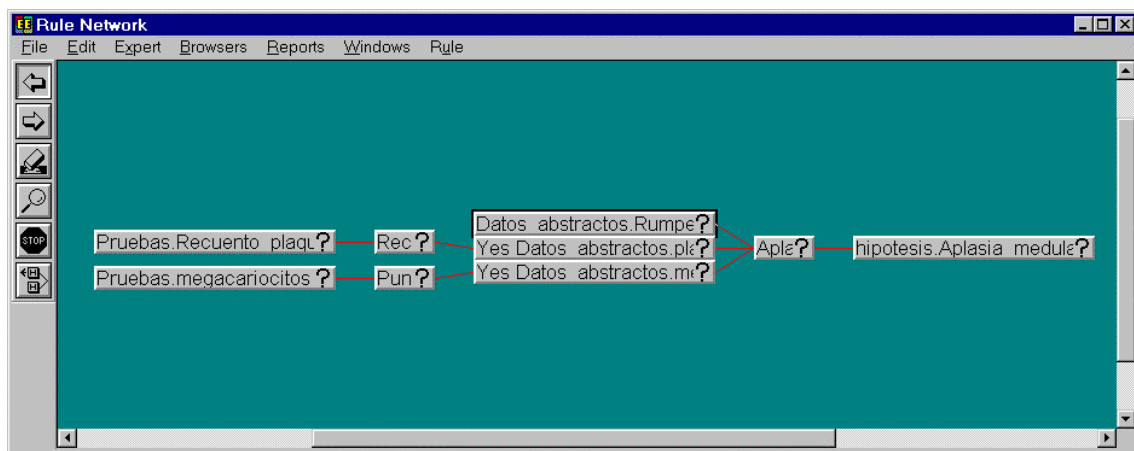


Fig. 13. Las dos reglas de la izquierda llevan a cabo una abstracción de datos, mientras que la de la derecha empareja una solución abstracta con una solución.

En código Nexpert estas tres reglas serían:

```
(@RULE=      Recuento_de_plaquetas
  (@LHS=
    (<      (Pruebas.Recuento_plaquetas)    (12000))
  )
  (@HYPO=    Datos_abstractos.plaquetas_bajas)
)
(@RULE=      Puncion_medula_osea
  (@LHS=
    (<      (Pruebas.megacariocitos) (2000))
  )
  (@HYPO=    Datos_abstractos.megacariocitos_bajos)
)
(@RULE=      Aplasia_de_medula_osea
  (@LHS=
    (=      (Datos_abstractos.Rumpell_leede) ("Negativa"))
    (Yes    (Datos_abstractos.plaquetas_bajas))
    (Yes    (Datos_abstractos.megacariocitos_bajos))
  )
)
```



(@HYPO= hipotesis.Aplasia_medular)
)

Ejercicio:

Se pide desarrollar el SBC que se requiere en el siguiente texto. Dar un modelo del conocimiento (modelo de pericia) y su implementación Nexpert. El problema es muy similar al diagnóstico visto en secciones anteriores, por lo que tanto el modelo de tarea como las posibilidades de implementación son las mismas.

El siguiente monólogo se extrajo de una entrevista con un experto de máquinas de café de la NASA, puesto que se desea enviar una de estas máquinas a unos extraterrestres acompañadas de un sistema experto para el diagnóstico de problemas”.

“Una máquina de café consiste de los siguientes componentes: una jarra, el contenedor del filtro, un tanque de agua, un tubo, el interruptor con una pequeña luz, un indicador del nivel del agua, el elemento que calienta el agua y el cable de alimentación. El calentador tiene dos funciones; mantiene caliente la jarra y evapora el agua desde tanque a través del conducto hacia el filtro. También es necesario el café, agua y el filtro.

Como experto en preparar café, a menudo me llaman otros compañeros quejándose de que no hay café en la jarra. Sé que hay bastantes explicaciones para e causado por problemas eléctricos. De estos hay cuatro ejemplos: la persona que prepara el café se olvidó del interruptor, se olvidó del enchufe, hay un fallo en la alimentación o que el cable esté cortado o roto. También hay problemas técnicos: puede se que se haya acumulado cal en el conducto y no fluya el agua, o que el calentador no funcione. Por último que la persona haya olvidado llenar el tanque de agua.

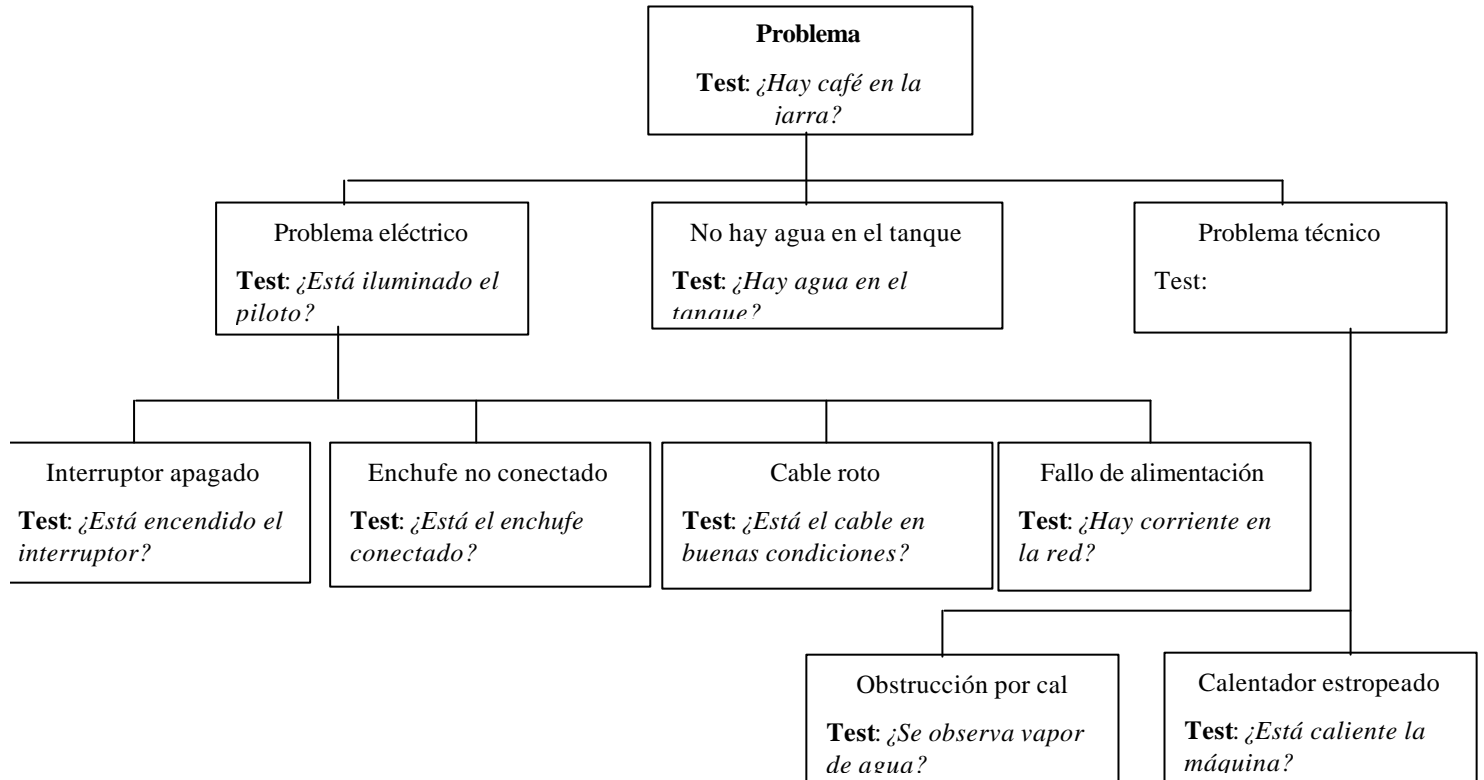
Si el que prepara el café me llama y me dice que no sale café, procedo de la siguiente forma: primero me centro en una posible explicación. Mentalmente he preparado una lista de explicaciones posibles, y tomo una de ellas, la más habitual. Para cada una de las explicaciones compruebo si es la correcta en ese caso. Para ello enuncio una pregunta al que prepara el café, así, para la explicación de que la jarra se encuentre vacía pienso en lo que esta persona debería observar y pido que se comprueben (podría haber varias observaciones relacionadas con la misma explicación).

De todas las explicaciones que me vienen a la cabeza, me centro en las que tienen un origen eléctrico. Esto se ve fácilmente por la luz de que dispone la cafetera. Si está apagada, se ha comprobado el origen eléctrico pero se desea refinar más y conocer exactamente la causa. Normalmente ha sido un olvido de encender el interruptor. Esto se comprueba por la posición del interruptor. Menos habitualmente se olvidan de conectar el enchufe. Si estas explicaciones son falsas es que hay un fallo de



alimentación. Se puede comprobar que hay corriente en la red eléctrica. En último caso puede ser una rotura del cable, siempre observable a primera vista.

Si la luz está encendida, entonces es que olvidaron poner agua en el tanque. La comprobación es fácil, mirar el nivel del agua. Si e quedan la posibilidades de que haya cal en el conducto o de que falle el calentador. En el primer caso se observaría vapor de agua y en el segundo simplemente tocando la



Referencias.

[**Nathan & Oski, 87**] Nathan D.G. Oski F.A. *Hematology of Infancy Childhood*, 3rd. ed. Philadelphia: WB Saunders, 1987.

[**Tansely & Hayball, 93**] Tansely, D.S.W. Hayball, C.C. *Knowledge-Based Systems Analylis and Design. A KADS Developer's Handbook*. Prentice Hall, 1993.

