V. 8, n.º 2 (juny 1984) pp. 93-110

MINIQUERY ORAL PARA UNA BASE DE DATOS BIBLIOGRAFICA

H. RULOT, E. SANCHIS, E. VIDAL, F. CASACUBERTA

UNIVERSIDAD DE VALENCIA

En este trabajo se unen los resultados prácticos de las investigaciones del CIUV en Reconocimiento Automático del habla con aplicaciones ya consolidadas en el área de las Bases de Datos, abordándose el problema de la obtención de información de una Base de Datos Bibliográfica mediante un reducido lenguaje de consulta oral ("Mini-query oral"). Se estudian los problemas planteados por este tipo de sistemas en los campos de Bases de Datos, Reconocimiento Sintáctico-difuso de frases, y Reconocimiento de Palabras Aisladas, presentándose las bases y detalles para su implementación.

Keywords: BUROTICA, BASES DE DATOS, RECONOCIMIENTO DEL HABLA.

1. INTRODUCCION.

El espectacular desarrollo de la informática en los últimos decenios ha hecho que las máquinas de tratamiento de la información, tanto los potentes ordenadores como los distintos microprocesadores, se vayan introduciendo cada vez más en la vida cotidiana de los individuos, impulsando el nacimiento de nuevas necesidades, cada vez más imperiosas de comunicación hombre-máquina. En todos los centros de investigación, paralelamente al desarrollo de la telemática, se llevan a cabo profundos estudios con el fin de elaborar sistemas informaticos capaces de entender lenguajes más naturales que los actuales y que requieran menos aprendizaje por parte de la persona que que deba comunicarse con ellos. Esta adaptación cada vez mayor, de la máquina al hombre, sólo se ha hecho factible recientemente gracias al exponencial crecimiento de potencia que han experimentado los computadores de uso general.

Sin embargo, las frases de estos perfeccionados lenguajes aún deben ser transmitidas a la máquina mediante un teclado, a pesar de que el medio de comunicación humana más natural es la palabra hablada. Un paso decisivo hacia la meta de la adaptación ideal se logrará el día en que los ordenadores sean capaces de utilizar y entender el habla. No obstante, no debe suponerse que es sólo en la meta de la adaptación, en la simplicidad de uso, donde reside el único interés de utilizar el habla como método de comunicación con los ordenadores. A pesar de ser un procedimiento sensiblemente más lento y menos fiable que, pongamos por caso, un teclado mecanográfico, las ventajas que se le pueden encontrar son múltiples:

Instantaneidad. Al evitar la necesidad de cualquier movimiento del cuerpo para dar una orden.

Posibilidad de utilizar las redes telefónicas ya existentes para control a distancia sin necesidad de accesorios especiales. Esto engloba todo tipo de acción controlable por ordenador y si se utiliza además un sintetizador, la consulta por teléfono a cualquier clase de base de datos.

No ocupa ni las manos ni los ojos. Lo cual aparte de ser muy útil para gran número de minusválidos, permitiría a cualquier persona el consultar u ordenar sin distraer su atención de una tarea en curso.

Autoriza los desplazamientos. Acabando con la obligación de permanecer atado al termi-

- Hector Rulot Segovia; Emilio Sanchis Arnal; Enrique Vidal Ruiz; Francisco Casacuberta Nolla
Centro de Informática de la Universidad de Valencia - Valencia

- Article rebut el Febrer de 1984.

nal todo el tiempo que dure la comunicación. Las órdenes se pueden dar desde cualquier lugar al que se pueda llevar un micrófono.

Permite la identificación del locutor. Se podrá decidir si la persona que da la orden es la que está autorizada para ello o interpretar el mensaje según quien lo pronuncie. Aunque en general los sistemas actuales no están diseñados con este fin, y se le aborde como un problema aparte, su utilidad está fuera de duda.

De las aplicaciones que se derivan de estas cualidades sólo se pueden dar aquí algunos ejemplos:

- -La consulta por teléfono del estado de cuentas en un Banco, de los espectáculos del día, de las existencias en el comercio.
- -Reserva automática por teléfono de plazas en aviones, trenes o barcos, y hasta de espectáculos públicos.
- -Entrada de datos y/o programas en ordenador.
- -Cobro revertido por teléfono, composición del número por la voz.
- -La máquina de mecanografiar dictados.
- -Sistemas de ayuda al aprendizaje y a la comunicación para sordomudos.
- -Control de silla de ruedas, coches y vehículos en general para minusválidos.
- -Control de accesorios en los automóviles, control de máquinas herramienta.
- -Ayuda a la clasificación postal y de piezas de maquinaria.
- -Simplificación de la tarea de comprobación de calidad en la industria.
- -Control de electrodomésticos (lavadoras, estufas,...) directo o por teléfono.
- -Etc...

Algunas de estas aplicaciones ya están en fase de realización (Control de maquinaria, comprobación de calidad...), para otras (La máquina de dictados, por ejemplo) quedan aún muchos progresos por hacer antes de que se pueda empezar a contar con ellas.

Muchas de las aplicaciones citadas caen dentro del área de la "BUROTICA", de la cual nos vamos a ocupar en el presente trabajo. De hecho, en este área existen aplicaciones potenciales, pero ya abordables en el actual estado de conocimientos y tecnología cuyas repercusiones van a revolucionar sin duda las concepciones bajo las que se abordan los problemas en la actualidad. La "oficina del futuro (próximo)" no cabe ya concebirlas únicamente como una intrincada red de pantallas alfanuméricas y dispositivos productores de documentos escritos, sino como una combinación racional de dispositivos alfanumericos, gráficos y orales.

En esta dirección, y con el objeto fundamental de mantener una posición de vanguardia en este área, se desarrolla el presente trabajo en el que se pretenden unir los resultados prácticos de las investigaciones del CIUV en Reconocimiento Automático del Habla, con aplicaciones ya consolidadas en el área de Bases de Datos. Como ejemplo de aplicación, se va a abordar el problema de obtención de información de una base de datos bibliográfica, para lo cual se ha definido un lenguage reducido de consulta oral ("Miniquery Oral") mediante el que se pueden ordenar operaciones clásicas de este tipo de bases de datos. Esta aplicación se ha elegido fundamentalmente por su simplicidad, la cual permite llegar a la implementación de una maqueta del sistema en un tiempo relativamente reducido. Otras aplicaciones más reales podrán ser abordadas en un futuro próximo con una filosofía idéntica a la que se describirá en esta exposición.

2. BASES DE DATOS. LENGUAJE DE "QUERY".

Un Sistema de Base de Datos es, básicamente, un sistema cuya misión es el registro y el mantenimiento de información. Los datos, almacenados en dispositivos de memoria masiva (ej.discos magnéticos), forman, bajo una determinada organización, una Base de Datos (BD). Entre la base de datos y los usuarios del sistema existe un logicial llamado Sistema de Gestión de Base de Datos (SGBD) que mediante una serie de procedimientos permite una utilización flexible por parte del usuario para actualizaciones, modificación de esquemas y subesquemas, consultas, etc.

Las estructuras de la BD y los procedimientos

del SGBD depende del tipo de modelo que se utilice (Relacional, Jerárquico y en Red) /9/.

En este trabajo se ha utilizado un modelo en red muy simple de una BD bibliográfica, en la cual se han definido tres tipos de registros:

LIBRO, LECTOR, PRESTAMO que se describen en la fig. 2.1, donde CODIGO y NUMERO son las palabras claves de los registros maestros: LIBRO y LECTOR, respectivamente y PRESTAMO es un enlace ("link").

Los registros descritos dan lugar a los conjuntos singulares ("sets" en BD en red): LIBROS, LECTORES y PRESTAMOS, estando relacionados mediante listas ligadas, tal como se indica en el ejemplo de la fig. 2.2, en el cual sólo se indican, explicitamente, las claves de los registros.

Las relaciones de la fig. 2.2 dan lugar a los conjuntos ("sets") LIBROP (libro prestado relación 1 a 1) y el de PLECTOR (préstamo a lector, relación 1 a varios), que puede ser representado de forma abstracta, mediante el diagrama de la figura 2.3.

Una vez que una BD ha sido diseñada e implementada, el usuario final necesita comunicarse con el SGBD mediante un lenguaje de interrogación de BD (lenguaje "query") que le permita obtener informaciones parciales o totales de dicha BD.

A continuación describiremos mediante la gramática de la figura 2.4 el lenguaje "query" que utilizaremos, donde las palabras en minúsculas entre paréntesis angulares se representan los símbolos no terminales; las palabras mayúsculas subrayadas representan los terminales y el símbolo inicial es "comando".

A continuación se describe la semántica de los símbolos terminales de la gramática utilizada:

*EMPEZAR sirve como delimitador entre conjuntos de comandos, para restaurar la situación inicial de la sesión, sin tener que finalizar ésta.

*FIN finalización de la sesión de "query".

*SELECCIONA permite referenciar a los "sets" singulares aisladamente.

*ENLAZA permite referenciar a los "sets" LIBROP y PLECTOR.

*LIBROS, LECTORES y PRESTAMOS hacen referencia a los correspondientes "sets" singulares.

*CLASIFICAPOR permite realizar una ordenación de los registros por el concepto indicado por item, correspondiente a algún campo del registro.

*INTERRUMPEPOR permite realizar subdivisiones de los registros ordenados por algún concepto indicado en item (agrupamientos).

*LISTA comando que indica la salida de los resultados de la ejecución de alguno (o ambos) de los dos comandos anteriores, por el dispositivo de salida, previamente definido por el sistema.

*Los terminales agrupados en item se corresponden con los distintos campos de los registros descritos en la figura 2.1.

3. RECONOCIMIENTO DE PALABRAS AISLADAS.

3.1. INTRODUCCION.

A pesar del considerable desarrollo experimentado en proyectos de reconocimiento de la palabra, el discurso hablado en general, parece "esquivar" los más sofisticados métodos puestos en juego para abordarlo. Realmente, el habla es uno de los procesos que caracteriza la inteligencia humana, y en la actualidad se es consciente de la enorme complejidad que entraña la concepción de sistemas que intenten aproximarse a las prestaciones del cerebro humano. Para obtener sistemas reales es necesario imponer restricciones más o menos severas al problema general. Una primera aproximación conduce a los Sistemas de Reconocimiento de Palabras Aisladas (R.P.A.), en los cuales el mensaje a reconocer se reduce a una única palabra perteneciente a un cierto diccionario y pronunciada aisladamente.

Aunque el problema de R.P.A. se considera en la actualidad suficientemente resuelto, me-

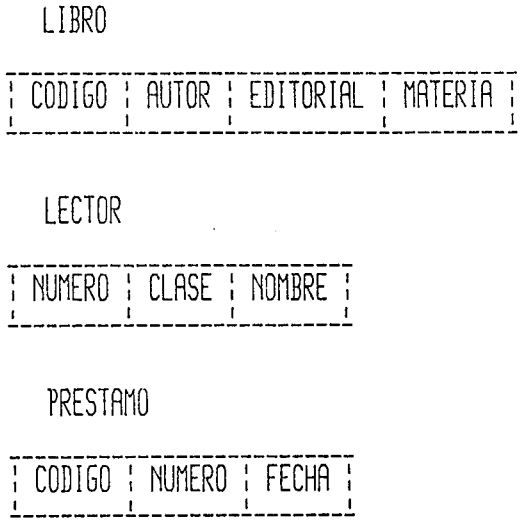

FIGURA 2.1 Descripcion de los registros.

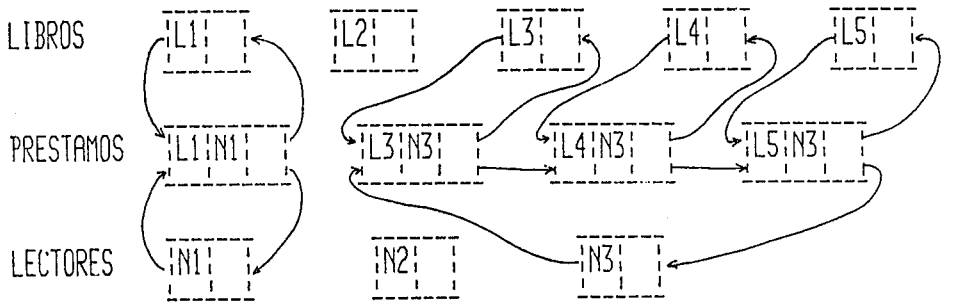

FIGURA 2.2 Ejemplo de relaciones entre registros.

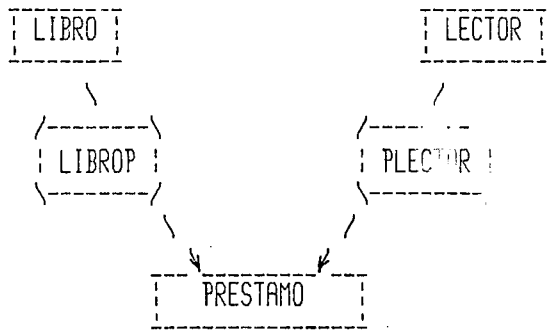

FIGURA 2.3 Conjuntos no singulares LIBROP y PLECTOR.

```
⟨comando⟩ ::= EMPEZAR ¦ LISTA ¦ FIN ¦ ENLAZA ⟨enlace⟩ ¦
              SELECCIONAPOR ⟨set⟩ ¦ CLASIFICAPOR ⟨item⟩ ¦
              INTERRUMPEPOR ⟨item⟩

⟨enlace⟩  ::= PRESTAMOS ⟨enlace_2⟩ ¦ LECTORES ⟨enlace_1⟩ ¦
              LIBROS ⟨enlace_1⟩

⟨enlace_1⟩ ::= PRESTAMOS

⟨enlace_2⟩ ::= LECTORES

⟨item⟩  ::= AUTOR ¦ EDITORIAL ¦ MATERIA ¦ CODIGO ¦ NUMERO ¦
            CLASE ¦ NOMBRE ¦ FECHA

⟨set⟩  ::= LECTORES ¦ PRESTAMOS
```

FIGURA 2.4 Gramatica de 'query'

diante la llamada aproximación global (Programación Dinámica) y existen ya en la industria sistemas comerciales con prestaciones suficientes para un gran número de aplicaciones, la complejidad y coste de estos sistemas sigue siendo relativamente elevada, tanto en lo que se refiere a la fase de parametrización como a la de reconocimiento propiamente dicho.

El mayor esfuerzo tecnológico para reducir esta complejidad apunta en la dirección de desarrollo de procesadores especializados integrados (VLSI). No obstante, se pueden utilizar aproximaciones alternativas para conseguir sistemas más eficientes y/o de menor coste como el que se va a describir a continuación.

En la aproximación global, las palabras a reconocer (palabra muestra) así como las palabras patrón del diccionario, se consideran como puntos en un espacio multidimensional. Definida una "distancia" en dicho espacio, la tarea de reconocer una palabra-muestra consiste en compararla con cada una de las palabras-patrón y elegir como reconocida aquella que arroje la mínima distancia.

Tanto patrones como muestras se suponen representados por cadenas de vectores obtenidos en la fase de parametrización mediante el análisis dependiente del tiempo /1/ de la señal vocal; es decir, la extracción de parámetros se realiza a través de una ventana temporal que se "desliza" sobre la señal a intervalos discretos de tiempo (típicamente de 10 a 20 ms.).

La definición de una "distancia" que mida la disimilitud entre dos de estas cadenas de parámetros, es un punto crucial en esta aproximación. Como una misma palabra puede pronunciarse a distintas velocidades, y como las duraciones de los distintos segmentos de dicha palabra son susceptibles de variación independiente, es necesario un método de normalización temporal no lineal de los patrones y muestras para posibilitar la necesaria definición de distancia.

Dicho método debe tolerar la variabilidad temporal con que en el habla corriente se pronuncian los distintos fonemas, y debe imponer tan sólo restricciones físicas de continuidad, monotonicidad, etc., en la forma de variación de los parámetros.

Aunque se han propuesto diversas alternativas para la normalización arriba citada, los métodos que finalmente se consideran como óptimos están basados en algoritmos de Programación Dinámica. Dichos algoritmos proporcionan simultáneamente la normalización temporal buscada, así como la definición de distancias, cuyo cómputo es asimismo realizado por el algoritmo.

En este apartado se expondrá el método de parametrización y el algoritmo de reconocimiento utilizados en un sistema de R.P.A. desarrollado en nuestros laboratorios (fig.3.1) así como los resultados con él obtenidos.

3.2. PARAMETRIZACION.

El sistema de RPA aquí presentado utiliza como parámetros los M primeros valores de la función de Autocorrelación de la señal vocal muestreada, cuantizada a dos niveles y observada a través de una ventana rectangular w(l) que se desplaza sin solapamiento sobre la señal (FA2N). Esta función la podemos definir como:

$$(3.1)\ \hat{R}(m) = \sum_{\substack{n=0 \\ m=1.M}}^{N-m-1} s'(n)s'(n+m);$$

$$s'(\ell) = w(l)\,sig(s(l));\ w(l) = \begin{cases} 1 \text{ si } 0 \le l \le L \\ 0 \text{ si } l<0 \text{ o } l>L \end{cases}$$

El recorte a dos niveles de la señal produce una fuerte reducción de la cantidad de información poseida por la señal inicial, por lo que las propiedades originales de la Función de Autocorrelación quedan alteradas o desaparecen. Teóricamente se puede predecir el error introducido por la cuantización para señales periódicas o aleatorias /2/. Para señales vocales reales se han realizado experimentos que indican que dicho error es tolerable si se utiliza una ventana de análisis suficientemente grande (del orden de 20 ms o mayor); asimismo, se ha estudiado experimentalmente el comportamiento de la FA2N con respecto al deslizamiento de la ventana en segmentos estacionarios; y con respecto a la discriminación entre diferentes fonemas castellanos /3/. Los resultados

de estos estudios mostraron la posibilidad de uso de los valores de esta función como vectores-parámetros en sistemas de R.P.A. Posteriormente se comprobó su comportamiento en un sistema experimental de R.P.A. /4/ cuyos resultados positivos fueron la base del sistema aquí presentado.

El cálculo de los M≤32 (o M≤8) primeros valores de la FA2N se realiza en tiempo real sobre un mini (o micro) ordenador de propósito general. El algoritmo (fig. 3.2) combina este cálculo con la detección de las fronteras de la palabra a reconocer así como un control de calidad de la señal adquirida, basado en valores de la amplitud media, saturación y duración de la señal.

Asimismo, el algoritmo realiza un preénfasis opcional sobre la señal de entrada para acentuar la importancia de los formantes superiores al primero. En el sistema material empleado, la señal es suministrada por un conversor A/D convencional; no obstante, para un sistema específico este material se puede reducir notablemente.

El algoritmo utilizado funciona "a micrófono abierto" almacenando los parámetros obtenidos en un buffer circular. El proceso funciona ininterrumpidamente hasta que se detecta una señal de calidad suficiente (palabra correctamente pronunciada). A partir de este momento el control es cedido al algoritmo de reconocimiento.

3.3. RECONOCIMIENTO.

Dada una métrica d en el espacio de vectores de parámetros, se trata aquí de comparar dos palabras A y B definidas por sus cadenas de estos vectores

$$A \triangleq \{a_1, a_2, \ldots, a_i, \ldots, a_I\} ;$$

$$B \triangleq \{b_1, b_2, \ldots, b_j, \ldots b_J\} .$$

en general, las duraciones I y J de ambas palabras son distintas ($I \neq J$). La normalización temporal buscada la especifica en el plano I-J una de las "funciones de alineamiento temporal" definidas mediante los caminos:

$$(3.2)\quad G \triangleq \{c_1, c_2, \ldots, c_k, \ldots c_K\} ;$$

$$c_k = (i(k), j(k))$$

para cada camino G, definimos la distancia normalizada entre A, B como:

$$(3.3)\quad D_G(A,B) \triangleq \left[\sum_{k=1}^{K} d(c_k).p(k)\right] / \left[\sum_{k=1}^{K} p(k)\right]$$

donde $d(c_k) = d(a_{i(k)}, b_{j(k)})$ es la métrica arriba indicada y p(k) es una función de ponderación.

La distancia global entre las palabras A y B la podemos ahora definir como:

$$(3.4)\quad D(A,B) = D_{G_o}(A,B) = \min_{G} D_G(A,B)$$

y la función de alineamiento temporal buscada quedará especificada por el camino G_o que minimiza la distancia normalizada entre A y D.

La minimización (3.4) de la función racional (3.3) es abordable mediante técnicas de PROGRAMACION DINAMICA /5/, con la condición de que el denominador sea independiente del camino G: $N \triangleq \sum_{k=1}^{k} p(k)$; N independiente de G. Esta condición solamente se cumple para dos definiciones simples de función de ponderación.

*SIMETRICA: $p(k) \triangleq (i(k) - i(k-1)) + (j(k) - j(k-1))$ con $N = I+J$.

En este caso, $D(A,B) = D(B,A)$.

*ASIMETRICA: $p(k) = (i(k) - i(k-1))$ si $N=I$.

o $p(k) = (j(k) - j(k-1))$ si $N = J$.

En estos casos, $D(A,B) \neq D(B,A)$.

No todos los caminos G definidos en (3.2) tienen significado físico, por lo que puede ocurrir que el camino G_o definido por (3.4) no sea un camino admisible. Es, por tanto, necesario imponer ciertas restricciones adicionales:

*LIMITES: $i(1) = j(1) = 1$;

$i(K) = I$: $j(K) = J$.

*MONOTONICIDAD: $i(k-1) \leq i(k)$;

$j(k-1) \leq j(k)$.

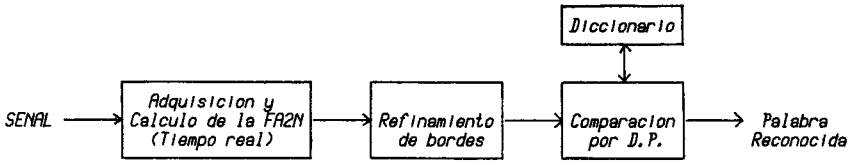

Fig 3.1 Diagrama de bloques del sistema 1.

TABLA 3.3

Diccionario	Numero de Palabras	Silabas/ Palabra	Tasa de Reconoc.	Tiempo de Respuesta	Observaciones
NOMBRES	12	2.8	96%	0.7 s.	Nombr. de Personas
ARITMETICA	16	1.6	95%	0.6 s.	Digitos y Operad.
MINI-QUERY	17	2.6	97%	1.0 s.	Consulta Bibliogr.
CIUDADES	30	3.0	97%	1.9 s.	Capitales Prov.

Tasa de reconocimiento del Subsistema de R.P.A.

```
Datos:  ESPERA_MAXIMA, UMBRAL_AMPLITUD, TIEMPO_DE_SILENCIO_MAX, NUM_PARAMETROS, TALLA_VENTANA,
        NIVEL_RUIDO.
-------------------------------------------------------------------------------------------------
Principio
  Inicializar  puntero_buf_ciclico  A  1.
  Inicializar  tiempo_de_espera  A  0.

  Repetir
    Parametrizar (BUF_CICL, puntero_buf_ciclico).
    Incrementar ciclicamente puntero_buf_ciclico.
    Incrementar tiempo_de_espera.
  Hasta tiempo_de_espera  >  ESPERA_MAXIMA  O  amplitud_en_ventana > UMBRAL_AMPLITUD.

  Si tiempo_de_espera > ESPERA_MAXIMA  Entonces error: 'espera demasiado large'.

  Inicializar puntero_buffer  A  talla_buf_ciclico+1.
  Inicializar tiempo_silencio  A  0.

  Repetir
    parametrizar (BUFFER, puntero_buffer).
    Si amplitud_en_ventana > UMBRAL_AMPLITUD  Entonces Inicializar tiempo_de_silencio  A  0.
                                              Sino     Incrementar tiempo_de_silencio.
    Incrementar puntero_buffer.
  Hasta puntero_buffer  >  talla_buffer  O  tiempo_de_silencio > TIEMPO_DE_SILENCIO_MAX.

  Si puntero_buffer > talla_buffer  Entonces error: 'palabra demasiado large'.
  Copiar ordenadamente  BUF_CICLICO  A  BUFFER[1..talla_buf_ciclico].
Fin.
-------------------------------------------------------------------------------------------------
Procedimiento parametrizar (BUFF, PUNTERO).

{ SHIFT y SHIFT_AUX son registros de desplazamiento de NUM_PARAMETROS bits }

Principio
  Inicializar BUFF [puntero..puntero+NUM_PARAMETROS]  A  0.
  Inicializar amplitud_en_ventana  A  0.
  Inicializar muestras_leidas  A  0.

  Repetir
    Inicializar punt  A  puntero.
    Inicializar SHIFT_AUX  A  SHIFT.
    Incrementar muestras_leidas.
    Leer_muestra.
    Incrementar amplitud_en_ventana  En  Imuestra!.
    Si muestra  >  NIVEL_RUIDO  Entonces  Repetir
                                            Incrementar punt.
                                            Desplazar a la izquierda SHIFT_AUX.
                                            Si carry=0 Entonces Incrementar BUFF[punt].
                                          Hasta punt = puntero + NUM_PARAMETROS.
                                          Desplazar a la izq. SHIFT inicializando carry A 0.
                                Sino      Repetir
                                            Incrementar punt.
                                            Desplazar a la izquierda SHIFT_AUX.
                                            Si carry = 1 Entonces Incrementar BUFF[punt].
                                          Hasta punt = puntero + NUM_PARAMETROS.
                                          Desplazar a la izq. SHIFT inicializando carry A 1.
  Hasta muestras_leidas = TALLA_VENTANA.

  BUFF[PUNTERO]:= amplitud_en_ventana.
Fin.
-------------------------------------------------------------------------------------------------
Procedimiento leer.
Principio
  Esperar impulso de reloj de muestreo.
  Adquirir una muestra del conversor A/D.
  Si PREENFASIS entonces decrementar muestra en muestra_anterior.
                         inicializar muestra_anterior a muestra.
Fin.
```

Fig.3.2. Algoritmo de adquisicion y obtencion de la F. autocorrelacion a 1 bit (FA2N).

*CONTINUIDAD: $i(k) - i(k-1) \leq i$;

$j(k) - j(k-1) \leq 1$.

*PENDIENTE: Limitación de exceso o defecto de la pendiente media local de G en el plano i-j.

*VENTANA DE EXPLORACION: Limita las máximas diferencias temporales toleradas.

Una definición clásica de ventana es:

$|i(k) - j(k)| \leq r$; $r \in z \geq 0$ (ventana de paralelas).

La forma general de los algoritmos de Programación Dinámica basados en los principios expuestos, consiste en un cálculo recursivo de las funciones D_{c_k}

*CONDICION INICIAL: $D_{c_1} = d(c_1).P(1)$,

*RELACION DE RECURRENCIA:

$$D_k = \min_{c_{k-1}} \left\{ D_{c_{k-1}} + d(c_k).p(k) \right\}$$

*RESULTADO FINAL: $D(A,B) = - D_K/N$

Si se utiliza una ventana de exploración, de anchura r y no se imponen restricciones de pendiente, la complejidad de estos algoritmos es $0(r\ j)$ operaciones de cálculo de la función $d(a_i,b_j)$. Gracias a las condiciones de Monotonicidad y Continuidad, estos algoritmos se pueden implementar de forma simple mediante un barrido ascendente del área de comparación, lo que exige tan sólo una memoria de talla I.

En el S.R.P.A. aquí descrito se ha utilizado la forma simétrica sin restricciones de pendiente con ventana de exploración de rectas paralelas de pendiente unidad. Como métrica se ha utilizado, por su simplicidad de cálculo, la distancia de Hamming:

$$d(a_i,b_j) = \sum_{m=0}^{M} |\hat{R}_i(m) - \hat{R}_j(m)|$$

donde $\hat{R}(m)$ son los valores de la FA2N definidos anteriormente en (3.1).

El algoritmo se ha realizado en lenguaje ensamblador y permite la comparación entre dos palabras en, aproximadamente, 1/10 de tiempo real sobre un ordenador ECLIPSE C-350.

El sistema se completa con un programa de creación del diccionario de palabras-patrón (aprendizaje). La creación de este diccionario se realiza mediante la elección de una palabra-patrón entre varias pronunciaciones de la misma palabra. El criterio de optimización utilizado hace uso de la distancia D(A,B) entre palabras, definida por el algoritmo de Programación Dinámica, para seleccionar como patrón aquella palabra cuya suma de distancias a las demás sea mínima.

Algunos de los resultados monolocutor obtenidos por este sistema los resumimos en la tabla 3.3. Estos resultados se han obtenido utilizando los siguientes parámetros del sistema:

Frec. muestreo = 6400 hz.

Preénfasis de 6 db/8.

Vectores de 8 parámetros (7 valores de la FA2N + amplitud).

Frecuencia de submuestreo = 50 hz. (Ventana = 128 puntos (20 ms)).

Ventana de exploración = 7 ventanas de análisis (140 ms).

3.4. CONCLUSION.

El sistema expuesto apunta a la simplificación del método de parametrización necesario en el reconocimiento de palabras aisladas, desarrollando el resto del sistema a partir de la aproximación clásica. La parametrización se basa en la evaluación de los primeros valores de la F.A. de la señal muestreada a 1 bit y se realiza con un material mínimo y con un logicial que funciona en tiempo real sobre un mini o un microordenador de propósito general. Este sistema monolocutor, de prestaciones aceptables para la aplicación propuesta, manejará para este caso el diccionario de 18 palabras reseñado a continuación:

EMPEZAR
ENLAZA
PRESTAMOS
LECTORES
LIBROS
SELECCIONA
CLASIFICA

INTERRUMPE POR
AUTOR
EDITORIAL
MATERIA
CODIGO
NUMERO
CLASE
NOMBRE
FECHA
LISTA
FIN

Para cada palabra pronunciada este subsistema reconocedor producirá una "tabla de distancias" (de certezas) entre dicha palabra y cada una de las palabras del diccionario. Tras la pronunciación de una frase, las tablas asociadas a las palabras pronunciadas serán utilizadas por el reconocedor sintácti co-difuso para la interpretación de dicha frase.

4. SINTAXIS DEL MINI-QUERY: RECONOCIMIENTO SINTACTICO-DIFUSO.

Los diversos tratamientos sobre la base de datos se gestionarán mediante el lenguaje de comandos descrito en el capítulo II. El objetivo de esta etapa consiste en obtener, a -- partir de la descripción dada por el reconocedor de palabras aisladas, la frase pronunciada por el locutor para realizar en una fase posterior las acciones asociadas a ese comando.

El lenguaje de comandos utilizado en este mini-query bibliográfico está generado por una gramática (fig. 2.4) y está representado en el autómata determinista de estados finitos presentado en sus dos variantes en las figuras 4.1 y 4.2.

En los sistemas de reconocimiento automático del habla hay que tener en cuenta las ambigüedades de los datos de entrada debidas a la variabilidad de la señal vocal /6/.

Para ello partiremos de una representación difusa de las cadenas de entrada al autómata y realizaremos un reconocimiento sintáctico difuso de la frase que nos proporcione el emparejamiento de dicha cadena con todas las posibles cadenas del lenguaje de un modo semejante al utilizado en /7/.

Las restricciones sintácticas del lenguaje ayudarán a superar las posibles ambigüedades en los resultados del reconocedor de palabras aisladas.

La representación sintáctica es la siguiente:

Sea Σ un alfabeto formado por todas las palabras del vocabulario, que son los símbolos terminales de la gramática descritos en el capítulo II.

Del reconocedor de palabras aisladas se obtiene un subconjunto difuso de Σ formado por todas las palabras del vocabulario con sus funciones de pertenencia que determinan la evidencia de que hayan sido pronunciadas.

Estos subconjuntos difusos son los símbolos de entrada del autómata de la figura 4.2.

En el proceso de reconocimiento sintáctico se evaluan todas las transiciones posibles para cada símbolo de entrada acumulando las evidencias de cada transición. De este modo se obtienen todas las secuencias de estados permitidas, o caminos a través del autómata, con sus evidencias correspondientes, evidencias que representan la compatibilidad de la cadena de entrada con las posibles frases del lenguaje.

Estas evidencias pueden considerarse como las funciones de pertenencia de un subconjunto difuso de todas las frases del lenguaje, de manera que la que presenta más evidencia será la frase reconocida.

Para la obtención del mejor camino a través del autómata, se ha desarrollado un algoritmo de búsqueda basado en el algoritmo de Viterbi /8/, con las siguientes particularidades:

- La evidencia acumulada en las transiciones viene determinada sólo por las evidencias de los símbolos de entrada.

- Sólo pueden crecer caminos que presenten continuidad en el incremento de su evidencia acumulada.

- El algoritmo actúa sobre el autómata de la Fig. 4.2 en el que existen varios estados finales que corresponden a cada una de las fa-

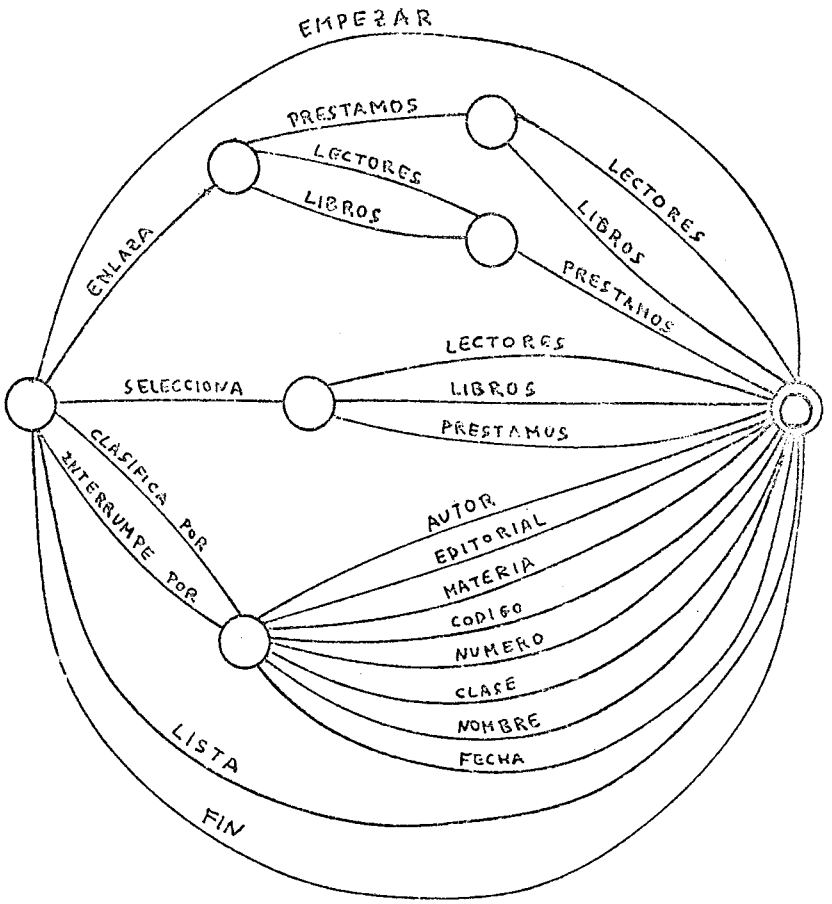

Figura 4.1 Autómata del lenguage de comandos.

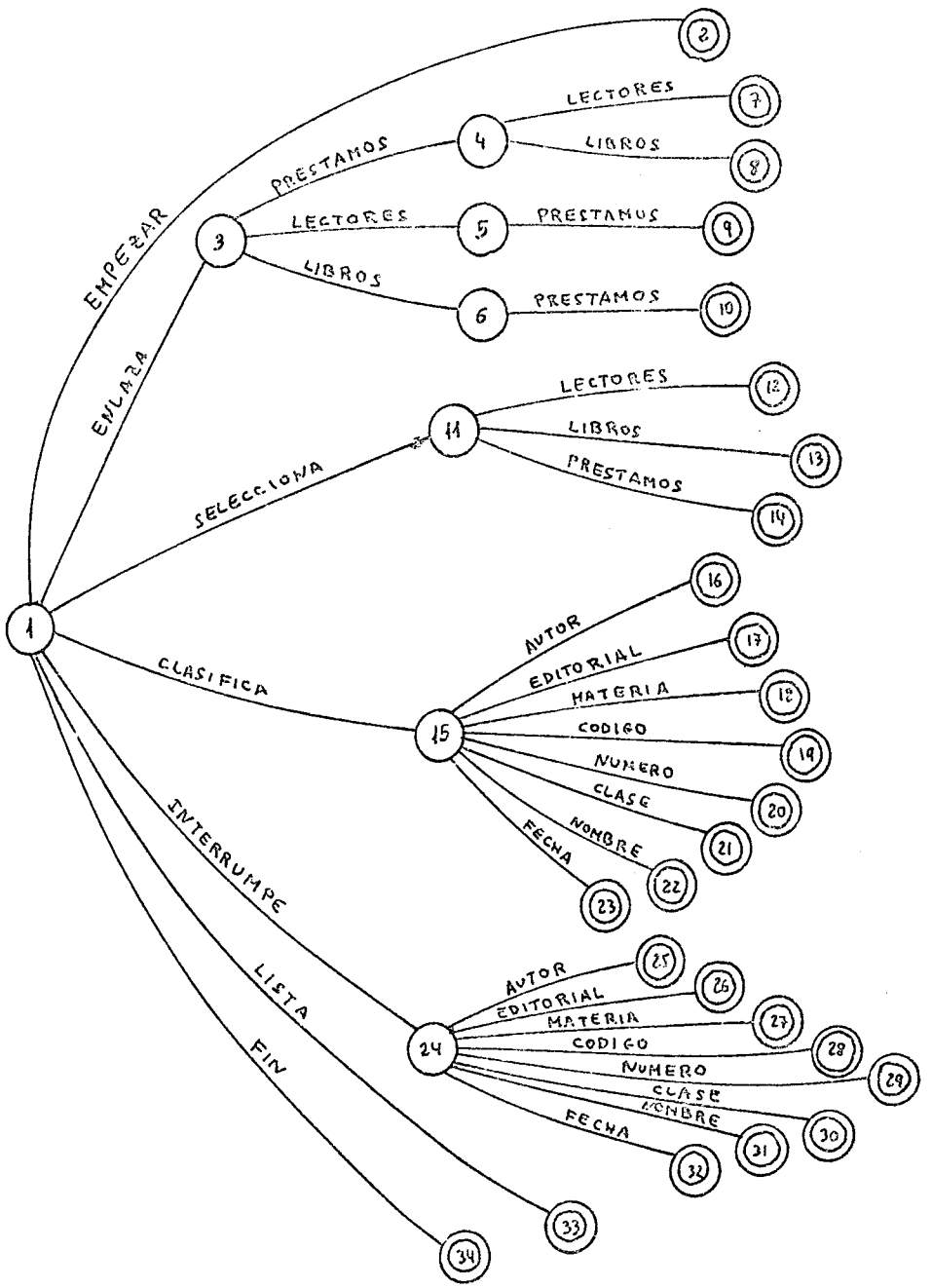

Figura 4.2 Autómata en árbol del lenguage de comandos.

ses del lenguaje, de modo que una frase es reconocida cuando su correspondiente estado final es el de mayor evidencia.

Finalmente, una vez reconocida la frase pronunciada, pasarán a ejecutarse las acciones asociadas a ese comando.

Si no se obtiene ninguna frase cuya evidencia destaque claramente de las demás o ninguna frase presente una evidencia mayor de un cierto umbral se dará un mensaje de error.

5. INTEGRACION DEL SISTEMA E IMPLEMENTACION.

5.1. INTRODUCCION.

En los capítulos anteriores se ha descrito la estructura y funciones de las partes básicas e que consta la base de datos bibliográfica controlada por la voz (BDBCV) que se pretende construir:

- El sistema de reconocimiento de palabras aisladas.

- El analizador sintáctico difuso.

- El gestor de la base de datos con su "lenguaje de query".

Para que la BDBCV resulte utilizable deben añadirse otros elementos y/o subsistemas que aseguren la correcta interacción entre los subsistemas básicos y la comunicación del sistema en conjunto con el exterior.

5.2. LOS SUBSISTEMAS COMPLEMENTARIOS.

- EL SUBSISTEMA DE CONTROL, para sincronizar los diversos elementos que forman la BDBCV, así como para tomar las decisiones y decidir la estrategia a seguir en caso de malfuncionamiento o error y también para gestionar la transmisión de datos o información entre subsistemas, se requiere algún tipo de control. El subsistema de control deberá decidir qué hace y en qué momento lo hace cada uno de otros subsistemas. Existen infinidad de algoritmos de control, que se separan en dos clases fundamentales: los que activan en paralelo los subsistemas controlados y los secuenciales que no permiten que dos subsistemas estén activos simultáneamente. Un sencillo ejemplo de algoritmo secuencial de control para la BDBCV presentada sería: el presentado en la figura 5.1.

Siendo el subsistema de control el que supervisa y dirige la acción de los otros subsistemas deberá disponerse de sentencias o frases de control dirigidas a modificar el estado general del sistema. En el caso más simple la única frase de control será la palabra 'FIN'.

- ELEMENTOS AUXILIARES: Además del ordenador, con sus memorias central y masiva, donde se integrarán los subsistemas y base de datos, se requiere, para la visualización de la información extraida de la base de datos por el gestor de ésta, de una pantalla (para los resultados cuyo almacenamiento no es necesario) y de una impresora.

```
Algoritmo secuencial de control.

Control;
  Inicializar Reconocedor Sintactico.
  Inicializar Base Datos.
  Inicializar R.P.A.
  Inicializar Sonorizedor.
  Si error de inicializacion entonces
                  Escribir Error;
                  fin de programa.
  repetir
    Sonorizar 'listo'; comprobar error.
    repetir
      Adquirir frase; comprobar error.
    hasta fin de frase y no error.
    Reconocer frase; comprobar error.

    Si no error entonces
      Ejecucion de frase por Gestor B.D.
      Comprobar error.
   hasta frase= 'fin'.
   cerrar R. Sintactico, Base Datos,
          R.P.A.,Sonorizedor.
   fin.

Procedimiento Adquirir frase.
    (puede integrarse en el R.P.A.)
  Numero palabras:= 0
  repetir
    Adquirir palabra;
    Incrementar Numero Palabras;
  hasta Numero palabras > Num.Palabras max.
        o Tiempo Silencio > Umbral o Error.
  fin de frase:= no error.
  fin.

Procedimiento Comprobar error
  si Error grave entonces
          Escribir error en pantalla;
          fin del programa.
  si Error leve entonces
          Sonorizar 'frase de error'.
          Comprobar error (Todo error del
                   sonorizador es grave)
  fin.
```

Figura 5.1.

Toda la información que pueda proporcionar la base de Datos saldrá a través de uno de estos dos periféricos, seleccionables interactivamente.

La pantalla de visualización será en general la de la consola de operación del sistema, necesaria para la inicialización del mismo y la recuperación de los errores graves. Opcionalmente el teclado de esta consola puede utilizarse como medio directo de dar órdenes al sistema sin pasar por el reconocedor de voz. La utilización de la misma sintaxis permite emplear el mismo analizador sintáctico, con solo asignar certeza "1" a las palabras introducidas mediante el teclado. Debe observarse sin embargo que esta opción implica necesariamente un algoritmo de control paralelo.

- EL SUBSISTEMA SONORIZADOR. Un sistema vocal es forzosamente lento y propenso a los errores de comprensión por parte del oyente cuando la cantidad de información a transmitir es grande. Por ello no es práctico proporcionar vocalmente los datos extraídos de la base. Es útil sin embargo disponer de un sistema de sonorización para pronunciar los mensajes cortos que informan del estado del sistema y de los errores leves en él producidos, lo que permite un manejo totalmente vocal de la BDBCV. Una vez activado el sistema sólo saldrán por impresora (pantalla) los resultados generados por el gestor de la Base de Datos.

El sistema de sonorización propuesto admite una versión muy sencilla, basada en el volcado directo a un conversor D/A de un fichero asociado a la frase a pronunciar. Dicho fichero contendrá simplemente la frase digitalizada, sin que medie ningún proceso de parametrización/desparametrización.

Un vocabulario de frases típico para la BDBCV se muestra en la tabla 5.2.

Obviamente el subsistema sonorizador no podrá informar de una alteración grave de su propio estado. Sus mensajes de error deberán dirigirse a la pantalla del sistema.

5.3. LA ESTRUCTURA DEL SISTEMA DE BDBCV.

El sistema de BDBCV tendrá finalmente una estructura lógica como la esquematizada en la figura 5.3., donde las líneas de trazo contínuo representan caminos de transmisión básicos, es decir aquellos por los que transita la información a tratar por cada uno de los subsistemas y los resultados generados a partir de ella. Las líneas de puntos indican caminos de control, por donde circulan las órdenes de sincronismo y los mensajes de error intercambiados entre el algoritmo de control y cada uno de los subsistemas.

5.4. IMPLEMENTACION EFECTIVA.

MATERIAL:

El sistema propuesto a lo largo de todo lo expuesto se instala en un ordenador Eclipse C-350 Data General, dotado con 512 Kby de memoria central y una capacidad total en disco de 70 Mby. Los periféricos auxiliares empleados son una impresora Centronics de 600 líneas/Minuto, una consola Dasher y un conversor A/D-D/A de 12 bits de precisión (muy superior a la realmente necesaria).

Del sistema AOS de Data General se aprovecha las facilidades de multi-proceso que proporciona, fundamentalmente el mecanismo de puertas IPC (inter-process-comunications) entre procesos.

La BDBCV se construirá utilizando 4 procesos, según muestra la figura 5.4.

Cada uno de los procesos (2,3,4) utiliza una puerta IPC que le comunica con el proceso 1, en el que se situa el algoritmo de control. Estas puertas sirven tanto para transmitir sus datos y/o resultados (Todos los cuales pasan por el proceso 1), como para intercambiar los mensajes de sincronismo y error con el proceso 1 de control.

Esta estructura multi-proceso no es solo útil conceptualmente para materializar el diagrama lógico de la BDBCV; es además una estructura que se ha hecho necesaria para poder disponer de las grandes cantidades de memoria central que requieren los procesos sonorizador y re-

	MENSAJE
	Listo
	Palabra demasiado larga
	Hable mas fuerte
	Hable mas bajo
	Separe mas las palabras
RPR	Palabra irreconocible
	Item no seleccionado
	Listado vacio
QUERY	Borrado de resultados
	Frase ambigua
RSF	Frase irreconocible

Tabla 5.2
Vocabulario de frases del sonorizador

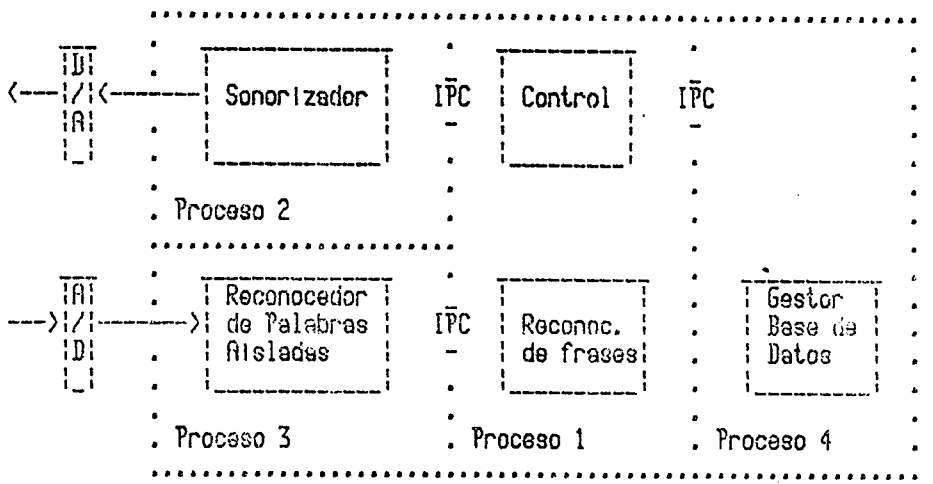

Figura 5.4
Implementacion de la BDDCV

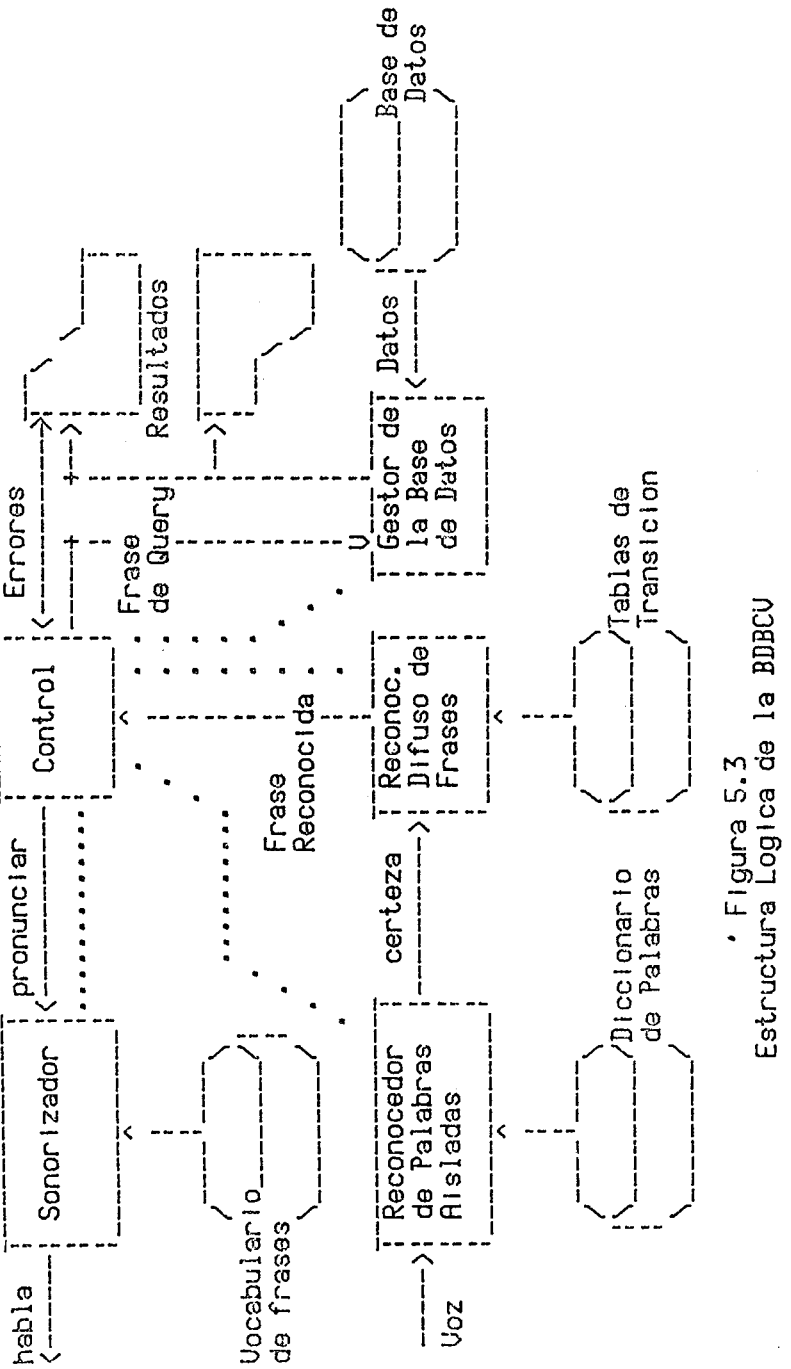

Figura 5.3
Estructura Logica de la BDBCV

conocedor de palabras aisladas (restricción propia al sistema AOS). Por otra parte, la localización de cada subsistema en un proceso diferente permite una implementación sencilla de los algoritmos de control en paralelo, aunque en la presente versión sólo se utilice un algoritmo secuencial.

Como ventaja marginal, cabe mencionar que la independencia del sonorizador del reconocedor de palabras y/o del gestor de la base de datos, autorizaría la utilización de estos subsistemas (via las mismas u otras puertas IPC) a partir de otros procesos situados en el mismo ordenador aunque éstos no tuvieran nada que ver con la BDBCV propiamente dicha.

LOS PROGRAMAS AUXILIARES.

Para la elaboración de los diversos ficheros auxiliares a los subsistemas, es decir:

- El vocabulario de frases de sonorizador.
- El diccionario de palabras del Reconocedor de Palabras aisladas.
- Las tablas de transición del autómata que forma el reconocedor sintáctico difuso.

y para la generación del contenido e la base de Datos, son necesarios una serie de programas auxiliares independientes del sistema BDBCV propiamente dicho. Gran parte de la eficiencia de la BDBCV dependerá de estos programas, que deben asegurarse tanto de la optimización de los ficheros que generan (Diccionario de palabras, Tabla de transiciones), como de que la construcción de los mismos sea cómoda para el usuario (Base de datos, vocabulario de frases).

6. CONCLUSIONES.

En este trabajo se han expuesto las bases y detalles para la implementación de un sistema de consulta oral a una base de datos bibliográfica.

Aunque la aplicación elegida se ha simplificado considerablemente, con objeto de permitir la implementación de una maqueta en un tiempo relativamente reducido, los mismos principios y métodos aquí expuestos se podrán utilizar para la puesta en marcha de aplicaciones reales en un futuro próximo.

El estado de implementación de la maqueta expuesta en el presente trabajo está avanzado, siendo ya operativos los principales módulos del sistema (Reconocedor de Palabras aisladas, Sonorizador, Reconocimiento difuso de frases...). La principal labor que queda pendiente en el momento actual es la puesta en marcha del subsistema de control y la integración total del sistema, de acuerdo con el proyecto presentado en la sección 5.

7. BIBLIOGRAFIA.

/1/ CASACUBERTA F.,VIDAL E.,VICENS M., BENEDI J.: "Sistemas Informáticos para el Análisis y Síntesis de la Palabra" Revista de Informática y Automática, n. 50, pp. 9-27, 1981.

/2/ VAN VLECK J.H., MIDDELTON D.: "The Spectrum of Clipped Noise", Proc. of IEEE, vol. 54, n.1, Jan. 66, pp. 2-19.

/3/ RULOT H., VIDAL E., CASACUBERTA F.: "La Función de Autocorrelación en el Reconocimiento de la Palabra", V Congreso de Informática y Automática, Madrid, Mayo 1982.

/4/ RULOT H., VIDAL E., CASACUBERTA F.: "Isolated Word Recognition System Based on the Autocorrelation Function", Portugal Workshop on Signal Proc. and Applications, Povoa de Varzim, Sept. 1982.

/5/ SAKOE H., CHIBA S.: "Dynamic Programming Algorithm Optimization for Spoken Recognition", IEEE Trans. on Acoustic Speech and Signal Processing, vol ASSP-26, n.1, Feb. 78.

/6/ DE MORI ., SAITTA L.: "Automatic Learning of Fuzzy Naming Relations over Finite Languages", Information Sciences, 21, pp.83-139, 1980.

/7/ VIDAL E., SANCHIS E., CASACUBERTA F.: "A Speaker-Independent Isolated Word Recognition System for Specific Dictionaries", Spain Workshop on Signal Proc. and its Applications, Sitges 83.

/8/ FORNEY G.D.: "Te Viterbi Algorithm", Proc. IEEE, vol. 61, 3, May 75.

/9/ DATE, C.J.: "An introduction to database systems", 3ª edición. Addison-Wesley -- Publ. Company, 1981.

8. NOTA.

Este trabajo forma parte del proyecto "Estudio de reorganización administrativa y proceso de informatización de la gestión universitaria, 3 fase" patrocinado por el ICE.