

1.1 Antecedentes

En la presente propuesta de tesis, se plantea el uso de *Algoritmos de Retroceso* como método de solución para el problema del diseño de la distribución de datos modelado por FURD. Se presentan algunos conceptos básicos sobre las técnicas que se pretenden emplear, así como la descripción de varios algoritmos de retroceso, en particular de aquellos basados en la técnica de Consistencia de Arco.

La razón más importante por la cuál se decidió enfocar el proyecto de residencia en resolver este tipo de problemas fue el hecho de que el modelo FURD se ha resuelto por varios algoritmos heurísticos: Recocido Simulado [22], Búsqueda Tabú [18], Algoritmos Genéticos [11] y Redes Neuronales; pero sólo se ha empleado un algoritmo exacto: Ramificación y Acotamiento [7]. Es por eso que se ha decidido llevar a cabo la investigación y tratar de darle solución al modelo FURD mediante otro método exacto: *Algoritmos de Retroceso*, tratando de verificar el comportamiento de otras técnicas, no utilizadas, y ver que nuevos resultados se obtienen.

Ahora bien, relativo a la importancia de este tema de estudio se hace referencia al hecho de que actualmente las bases de datos distribuidas representan una importante área de estudio debido a la globalización actual. en particular las fases de diseño y distribución de la base de datos son de gran importancia ya que influyen mucho en el desempeño de la misma. Debido a que el Modelo FURD permite optimizar el diseño de la distribución de datos en una red entonces si se profundiza en este tema de estudio, se podrá tener mejores herramientas para el futuro de las bases de datos distribuidas.

1.2 Objetivos

La presente tesis tiene por objetivo el diseñar e implementar un conjunto de algoritmos que permitan resolver el problema de la distribución de datos Modelado por FURD (Fragmentación, Ubicación y Reubicación Dinámica de Datos) presentado en los sitios de una red al momento de compartir información proveniente de una base de datos distribuida. En particular se implementarán algoritmos de retroceso para la solución del problema.

El objetivo específico es dar pauta al rediseño y adaptación de programas capaces de resolver el Modelo FURD empleando técnicas algorítmicas de solución exacta y reglas heurísticas que permitan hallar una solución de manera rápida y eficiente, sin consumir mucha memoria.

1.3 Restricciones

Las restricciones que se pueden encontrar durante la implementación de las técnicas algorítmicas pueden afectar tanto a: el tiempo de solución empleado, como a la cantidad de memoria consumida, entre otros.

El tiempo de solución empleado se ve afectado por la velocidad del procesador con el cual se esté trabajando, mientras más rápido sea éste, mejor será el tiempo de solución de la instancia tratada. Sin embargo, existe otro factor que también influye en este aspecto, y éste es el hecho de que se encuentren programas en ejecución al mismo tiempo que se corre la instancia. Por eso es recomendable trabajar sin ningún programa residente en memoria.

La memoria consumida depende del tamaño de la instancia y del número de variables que esta genere. Estos dos factores determinan una limitante para la instancia, si tienden a consumir mucha memoria

principal entonces se necesitará contar con suficiente memoria para resolverlos, de lo contrario será imposible.

En conclusión, es posible que las limitantes básicas sean referentes al equipo de trabajo y al tiempo requerido para la solución. Para solucionar estas restricciones se recomienda emplear equipos más potentes.

1.4 Justificación

Las bases de datos de hoy en día y las redes computacionales luchan por la necesidad de contar con una mejor distribución de sus recursos. La información es el recurso más importante de ambos, por eso es necesario involucrarle cierto grado de interés. Mientras mejor esté distribuida la información, más fácil y eficiente será su acceso. El Modelo FURD trata de encontrar la mejor ubicación de los datos de acuerdo a las características de los sitios que posee una red, basándose para ello en la fragmentación inicial de datos y en la posterior ubicación de fragmentos.

Además de lo mencionado anteriormente, también se incluye el deseo de ampliar los conocimientos existentes sobre técnicas algorítmicas que dan solución al Modelo FURD, tratando de proporcionar con este proyecto nuevos resultados que pudieran ser relevantes al área.

1.5 Producto Final

Como producto final de la tesis se entrega un programa que permite resolver el problema de la distribución de datos en bases de datos distribuidas modelado como CSP, empleando para ello el modelo matemático conocido como FURD (Fragmentación, Ubicación y Reubicación dinámica de Datos) y una serie de algoritmos de retroceso. Se espera que sirva de base para futuras investigaciones relacionadas con el tema.

1.6 Contribución Esperada

La contribución que se da con esta investigación es un marco de referencia sobre el comportamiento de los algoritmos de retroceso en la solución del Modelo FURD. Es decir, en base a los datos recolectados mediante la experimentación se indican las ventajas y desventajas que posee el procedimiento desarrollado para resolver el problema de la distribución.

1.7 Organización del Documento

Este documento se encuentra dividido en 6 capítulos. En el capítulo uno se encuentra la introducción a la tesis; se mencionan sus objetivos y restricciones, así como las razones por las cuales se decidió elaborarse; también se incluyen una pequeña descripción del producto final que se obtiene y la contribución que se hace con la tesis. Es en esta parte donde se encuentra incluida esta sección.

El capítulo dos desarrolla el marco teórico en el cual se fundamenta la tesis. Contiene conceptos básicos sobre técnicas algorítmicas y heurísticas, y describe las herramientas empleadas para desarrollar el programa implementado. En este capítulo se concentra todo el material recopilado de las diferentes literaturas consultadas.

El capítulo tres es el capítulo de la experimentación. Describe la metodología empleada para poder darle seguimiento a la tesis, como se seleccionaron los casos de prueba, que equipo se empleó, cuál fue el orden de desarrollo del trabajo.

Dentro del contenido del capítulo cuatro se mencionan los resultados obtenidos por la experimentación que se menciona el capítulo tres. Se describe en forma exhaustiva el producto final de la tesis.

El capítulo 5 describe la contribución de la tesis en toda su extensión. Por último, en el capítulo 6 se dan las conclusiones y se presentan los trabajos futuros para este documento. Aparte de esta organización inicial, también se incluye una sección de bibliografía empleada, otra de glosario y una de anexos.

Capítulo 2

Capítulo 2. Revisión Bibliográfica

2.1 Antecedentes

Dentro de este documento se encuentra un marco de referencia bibliográfico que orientará, a quien lo lea, sobre el contenido de la propuesta de tesis que se plantea elaborar.

En los párrafos que se mostrarán en la sección de contenido se incluirán la razones por las cuales se seleccionaron las referencia bibliográficas que se tienen al final del documento, así como algunas de las ideas principales que fueron tomadas de ellas.

.

A manera de resumen, el contexto que se muestra refleja lo que son los *Algoritmos de Retroceso*, su aplicación en el campo de la investigación, cómo se define al problema de la distribución de datos y algunos otros conceptos básicos.

2.2 Definición de Problemas de Satisfacción de Restricciones (CSP)

Según [20], [10] y [6] un Problema de Satisfacción de Restricciones o *CSP* (por sus siglas en inglés, *Constraint, Satisfaction, Problem*) es una tripleta (X, D, C) donde X es un conjunto de n variables, D es un conjunto de dominios finitos sobre las variables y C es un conjunto de m restricciones aplicables al conjunto de las variables en X . La meta de esta tripleta es encontrar una o todas las soluciones para el problema *CSP*, en el cuál una solución representara una asignación de valores $a \in D(X)$ a sus variables X de tal forma que satisfagan cada una de las restricciones C que se presentaran para el ejemplar del problema tratado.

En esta tesis se identifican dos tipos de restricciones para los problemas *CSP*, las restricciones explícitas y las restricciones implícitas. Las primeras son aquellas que se pueden incluir como parte del conjunto de restricciones del problema, poseen una serie de tuplas válidas que las identifican, y forman parte de cualquier ejemplar que represente al tipo de problema básico (*CSP*). Por otra parte se encuentran las restricciones implícitas, que se basan en condicionales y, debido a esta característica, no poseen un conjunto de tuplas válidas, se construyen mediante código (sentencias condicionales) dentro del programa que los resuelve y no se pueden manipular directamente en los ejemplos de problema. Otra forma de definir a estos tipos de restricciones es como se menciona en [24], que dice que cuando las restricciones son especificadas en forma explícita, es decir, cuando se enumera todo el conjunto de tuplas válidas, se dice que son almacenadas *en extensión* o *extensionalmente* de otra forma estarán almacenadas *intencionalmente*.

Con el objeto de ejemplificar el problema *CSP*, a continuación se muestra el *Problema del Crucigrama* (crossword).

El *Problema del Crucigrama* se encuentra formado por un tablero y un diccionario. El tablero es un matriz bidimensional (como la mostrado en la Figura 2.1) formada por espacios y blancos. El diccionario es el conjunto de palabras válidas que se pueden admitir para dar solución al *Problema del Crucigrama*.



Figura 2.1. Tablero. Una *B* representa un blanco, mientras que una *E* representa un espacio.

Para ejemplificar el Problema CSP mediante el Problema del Crucigrama se necesita transformar este último mediante alguno de los modelos existentes: el modelo letra, dual y el modelo oculto. En [24] y en [20] se muestra la definición de los tres métodos de modelado previamente mencionados, los cuales trabajan de la siguiente manera:

- El *Modelo Letra*. En este método las variables que pasarán a constituir el problema CSP serán cada uno de los espacios que contenga la matriz del Crucigrama. Por lo tanto, el dominio de cada una de las variables estará constituido por las letras del alfabeto que forman las palabras del diccionario. Por su parte, las restricciones estarán formadas por cada palabra que se encuentre en el Crucigrama.
- El *Modelo Dual*. Mediante este modelo las palabras que componen al Crucigrama pasan a formar las variables duales de su equivalente Problema CSP. Su dominio será el tamaño del diccionario con que se cuente. Las restricciones serán de carácter binario entre las variables, asegurando que las palabras que sean asignadas, sean consistentes para el problema.
- El *Modelo Oculto*. Esta técnica se podría considerar como una combinación de las anteriores, es decir, las variables del Problema CSP equivalente desarrollado por el *Modelo Oculto*, son los espacios y las palabras que componen el crucigrama. El dominio dependerá de la variable que se trate, pero solamente podrá ser cualquiera de estos dos: el alfabeto manejado o el diccionario empleado. Las restricciones fuerzan a una asignación compatible entre una variable letra (espacio) y una variable palabra.

El método completo para transformar un ejemplar del *Problema del Crucigrama* a CSP empieza seleccionando alguno de los modelos mencionados en los párrafos anteriores. Para ejemplificar esto se muestra el ejemplar de Crucigrama mostrado en la Figura 2.2.

Tablero:
Diccionario:
ASA, OSO, OSA, ASO, AS, OS, SA, SO

Figura 2.2. Un ejemplar del Problema del Crucigrama.

Una vez que se ha definido qué ejemplo va a ser el que se traduzca a *CSP*, se selecciona el método para llevar a cabo el modelado. En este caso se seleccionó el *Modelo Letra*. Al tener seleccionada la técnica, se procede a la transformación del problema.

Lo primero que se menciona en el *Modelo Letra* es que las variables que pasarán a constituir el problema *CSP*, estarán formadas por los espacios que se encuentren en el tablero del crucigrama. El resultado de este paso se ilustra en la Figura 2.3.

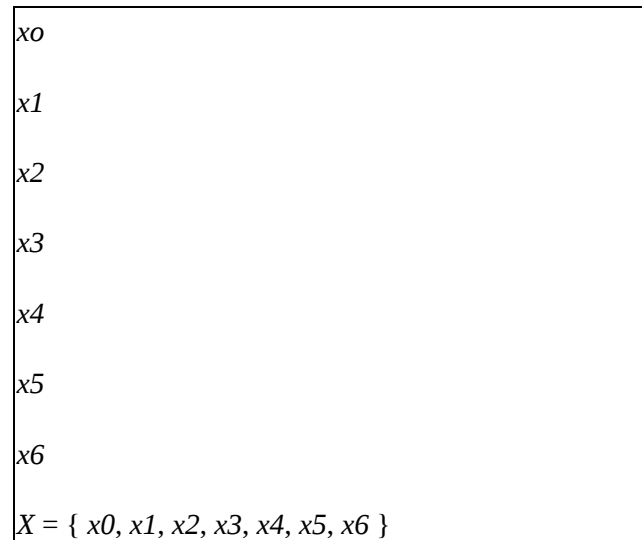


Figura 2.3. Conjunto de variables X para el Problema CSP.

Lo siguiente es determinar el dominio de cada una de estas variables. Como se muestra en la Figura 2.4, debido a que las variables representan letras del alfabeto entonces, su dominio estará comprendido por aquellas letras que se puedan usar para formar palabras del diccionario.

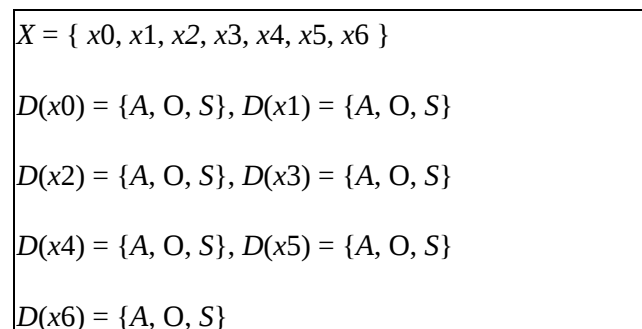


Figura 2.4. Variables x_i y sus Dominios $D(x_i)$ para el problema CSP.

Por último, se identifican las restricciones a las que estará sujeto el Problema del Crucigrama, ya como Problema CSP. En el ejemplo manejado, las restricciones encontradas son las que se muestran en la Figura 2.5. Sus respectivos dominios estarán compuestos por las palabras que posea el diccionario del crucigrama.

$x0$

$x1$

x^2

x^3

x^4

x^5

x^6