

## **CAPÍTULO 3      Trabajos relacionados**

### **3.1 Proyecto Pech**

Como se comentó anteriormente, el estudiante de doctorado Manuel Pech Palacio se encuentra participando en el proyecto “Habitar y vivir. Análisis del espacio habitacional de la ciudad de Puebla 1690 -1890”. Este proyecto está dirigido por la Dra. Rosalva Loreto López, de la Benemérita Universidad Autónoma de Puebla (BUAP), quien es una investigadora en el tema de Historia Urbana. El objetivo del proyecto es encontrar patrones y relaciones interesantes entre la población y el espacio habitacional en la zona centro de la ciudad de Puebla en ese período de tiempo.

De aquí en adelante, nos referiremos a los trabajos de Manuel Pech Palacio, Víctor González Carrillo y Daniel Pérez Armería como TMPP, TVG y TDP, respectivamente.

### **3.2 Minería en el proyecto Pech**

La aplicación usada por Manuel Pech Palacio dentro de su participación en el proyecto trabaja con datos geográficos obtenidos de la base de datos geográfica del proyecto [10]. Esta base de datos contiene la información de la ciudad de Puebla, como la localización de iglesias, arroyos, familias y vialidades.

El módulo de *Map*, que se encuentra dentro de TMPP, es en el que se introducen los parámetros y los datos correspondientes para la minería de datos. Aquí se pueden seleccionar cualquier cantidad de capas para ser representadas en pantalla. Por ser capas geográficas, estos datos se pueden visualizar como si fueran un mapa (ejemplo en la siguiente sección). Es a partir de este mapa donde se pueden seleccionar los datos que se van a minar. Estos datos y sus relaciones se representan en forma de grafo, que a su vez se convierte en la entrada del algoritmo de minería de datos SUBDUE. El primer método de minería asistida (el sistema de guía en tiempo real) se implementará en esta parte de la aplicación.

Los mejores patrones arrojados por Subdue se pueden ver también en forma de grafo (TVG), y es ahí donde se implementará el segundo método de minería asistida (anotaciones sobre los patrones encontrados).

### **3.2.1 INTERFAZ DE USUARIO**

El módulo *Map* consta de una interfaz de usuario [10]:

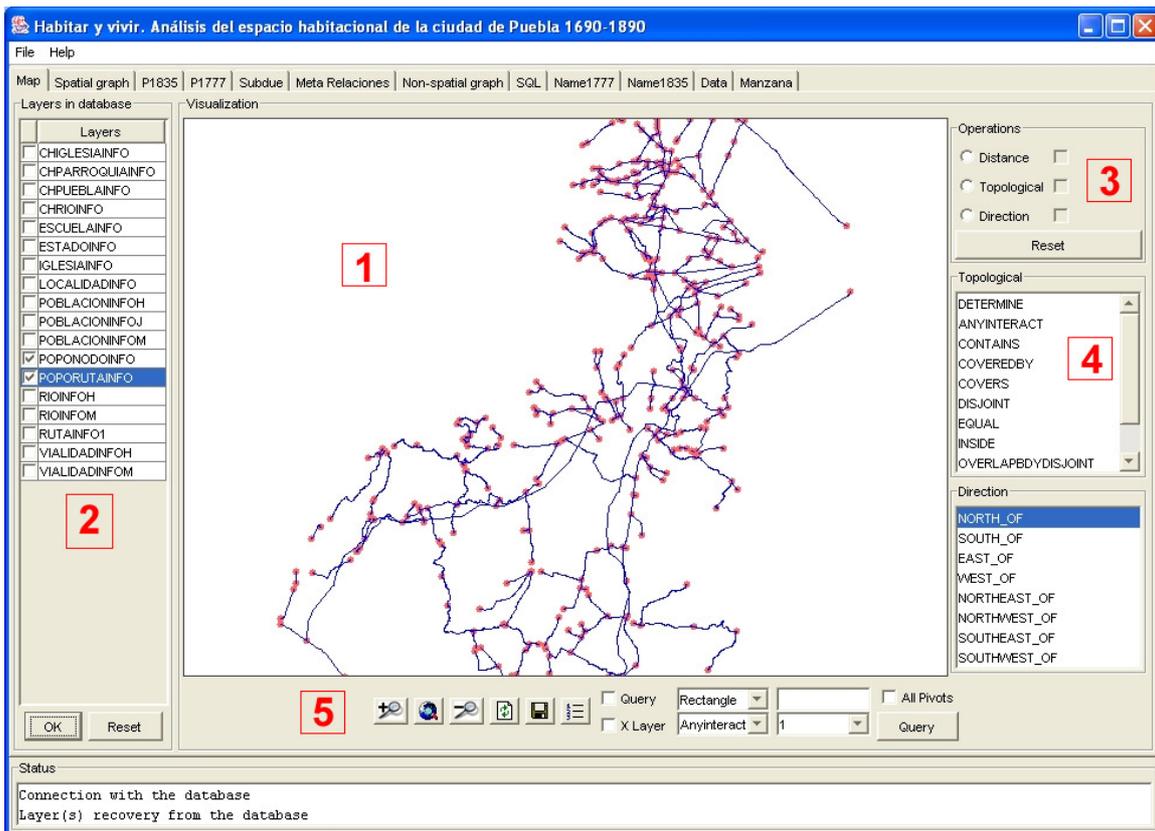


Figura 3.1 - Interfaz del módulo  
Map de TMPP

- 1) Mapa de visualización
- 2) Capas geográficas para representar en el mapa. Actualmente, está representada la capa **POPORUTAINFO**
- 3) Tipo de operación
- 4) Operaciones disponibles
- 5) Panel de control

Dentro de TMPP, esta es la única interfaz relevante para este proyecto.

### **3.3 Visualización de resultados en el proyecto Pech**

La aplicación de Manuel Pech Palacio tiene un módulo de visualización que él deseaba que fuera mejorado, por lo que lo propuso como un tema de tesis de licenciatura. Este es el propósito de la tesis de Víctor Manuel González Carrillo [11], estudiante de la carrera de Ingeniería en Sistemas Computacionales de la Universidad de las Américas (UDLA), Puebla.

Originalmente, los grafos generados eran manualmente convertidos en imágenes. Sin embargo, Víctor Manuel propuso otro método de visualización que pretende hacerla más amigable y estéticamente más agradable al usuario. Propuesto por el mismo Víctor Manuel, su tesis “tiene como propósito aplicativo proporcionar una interfaz de visualización para la interpretación de resultados obtenidos de la aplicación de minería de datos a una base de datos geográfica integrada por el Mtro. Manuel Pech Palacio” [11].

Uno de los cambios que propuso Víctor Manuel, y que es un requerimiento del presente proyecto, es la implementación de un *hiper-grafo*.

#### **3.3.1 HIPER-GRAFOS**

Definido dentro de la tesis de Víctor Manuel González Carrillo, un hiper-grafo toma su nombre del famoso *hiper-texto* de Internet, el cual contiene una liga a otro documento [11].

Un hiper-grafo contiene al menos un vértice que representa a otro grafo. En términos de representación gráfica, al presionar el botón del ratón sobre un vértice que contenga a un subgrafo, el grafo original se expande para representar al grafo completo.

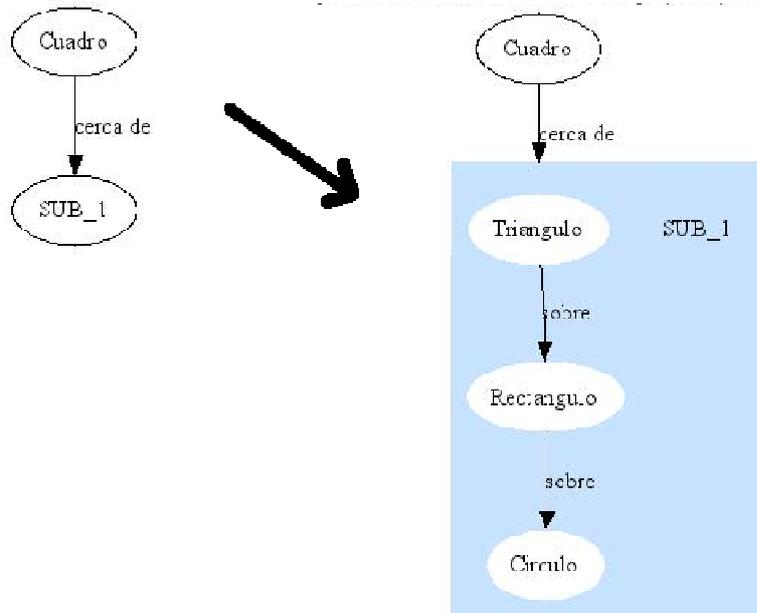


Figura 3.2

En el caso de la figura anterior, el vértice SUB\_1 representa a otro grafo, que al ser presionado tiene que insertarse en el grafo original. De manera inversa, al ser presionado el subgrafo, tiene que ser sustituido por un vértice que lo represente.

De esta forma, mediante un *click* del ratón, se puede pasar de la representación simplificada a la expandida y viceversa. Naturalmente, se pueden tener grafos muy grandes, y con este concepto de hiper-grafo es más sencilla su visualización.

e6m1\_i10 Iter: 10 Substructure: 2 Graph(5v,4e) Instancias: 1

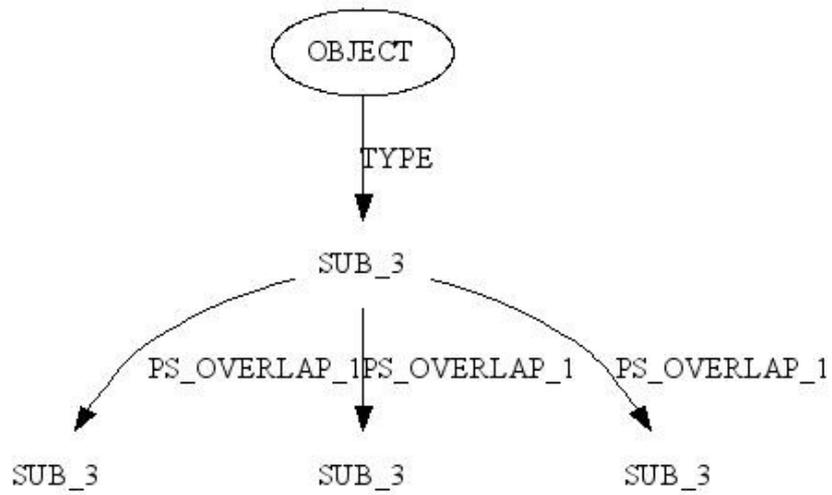


Figura 3.3 – Grafo simplificado

e6m1\_i10 Iter: 10 Substructure: 2 Graph(5v,4e) Instancias: 1

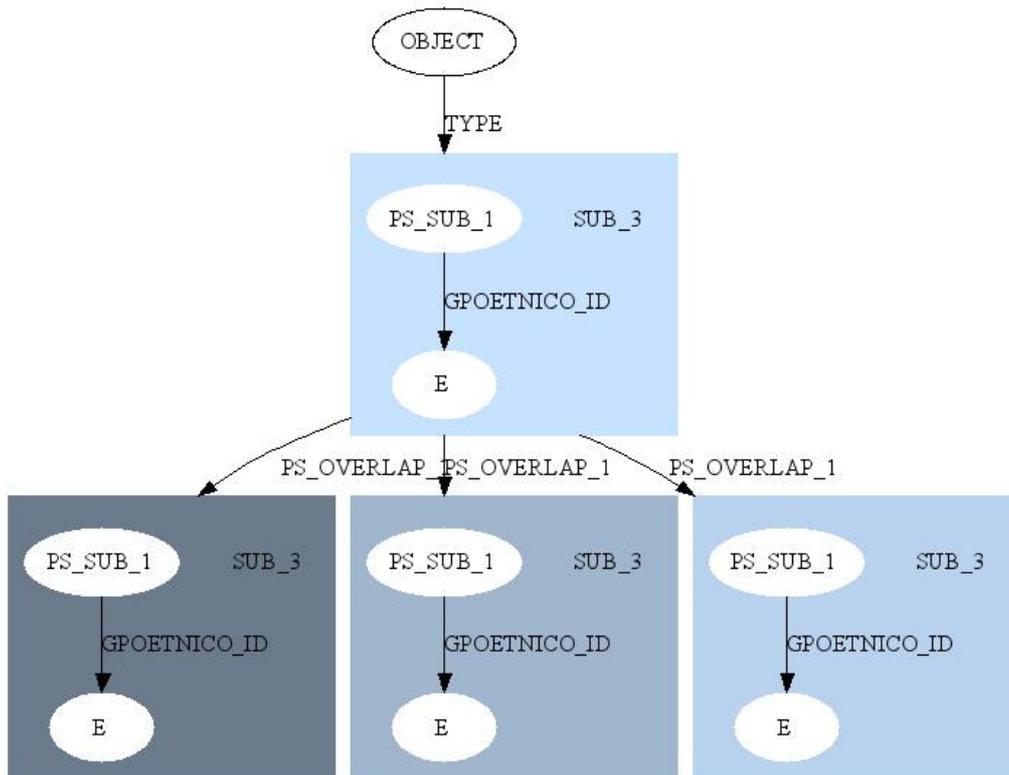


Figura 3.4 – Mismo grafo, pero expandido

### 3.3.2 PASOS PARA LA VISUALIZACIÓN

Al implementarse el programa de Víctor Manuel, su intervención queda resumida en el siguiente orden:

1. Se obtiene el archivo *.sub*, que es la salida generada por Subdue
2. Ese archivo se hace pasar por el programa *dot*, para generar los grafos necesarios, que se guardan primero en archivos con extensión *.dot*.
3. Usando Grappa, esos archivos *.dot* se leen para generar los objetos Java de los grafos.
4. Esos grafos se muestran en un **JList**, para poder seleccionar el que se desee.
5. Una vez seleccionado un grafo, se muestra en el **GrappaPanel** de la interfaz.
6. Obviamente, los grafos mostrados tienen la implementación de hipergrafo, por lo que es posible ver varios niveles del grafo de manera interactiva, usando el ratón.
7. También es posible abrir otro archivo *.sub* para ver otro grupo de grafos generados a partir de una minería distinta.

Esta es una imagen de la interfaz principal del proyecto de Víctor Manuel, contenida en su clase DemoTesis [11]:

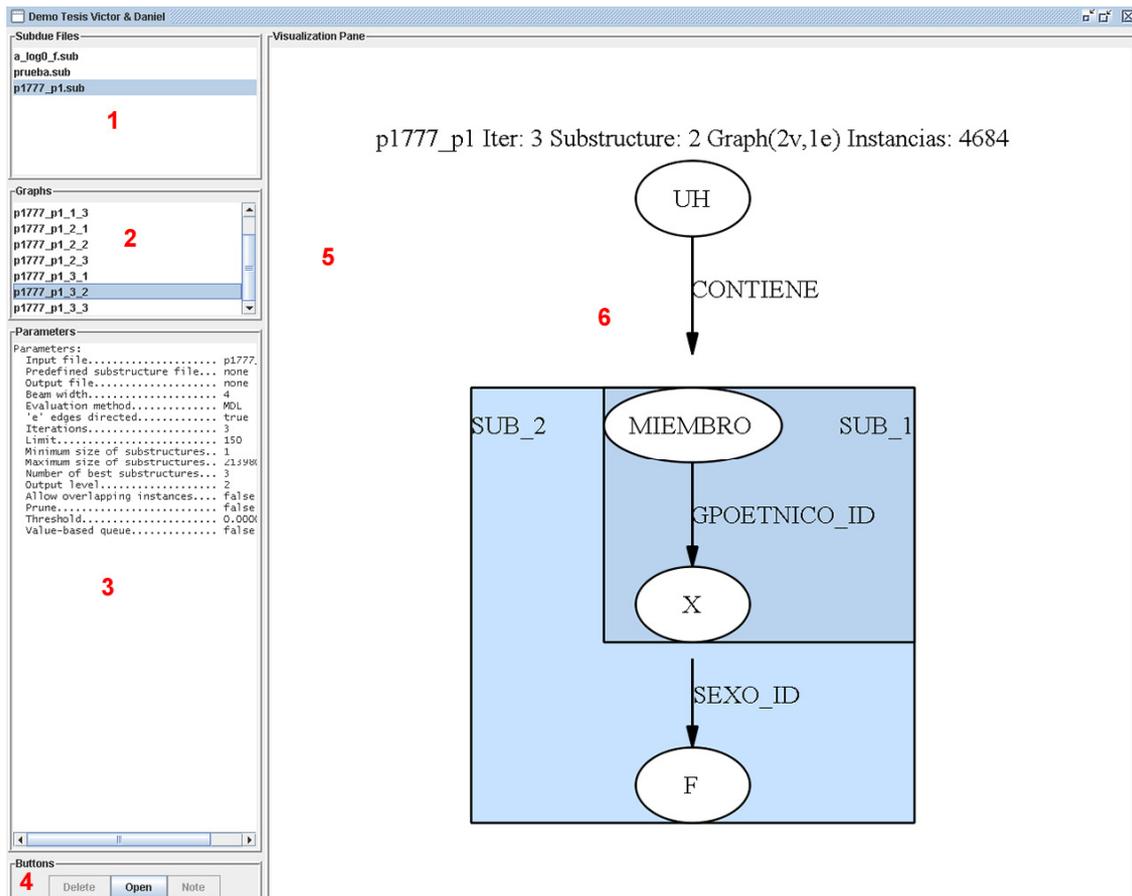


Figura 3.5

- 1) Lista de los archivos *.sub* abiertos
- 2) Lista de los grafos contenidos en el archivo *.sub* seleccionado
- 3) Parámetros de la minería de datos hecha en SUBDUE
- 4) Botones
- 5) El GrappaPanel
- 6) El grafo seleccionado

### 3.4 Anotaciones para la minería asistida

Es sobre un panel del tipo mostrado en la figura anterior donde se implementará el método de anotaciones para patrones encontrados. El

usuario, al estar visualizando un patrón, también deberá de ser capaz de almacenar sus observaciones o notas al respecto, para poder consultarlas en otro momento. Esas notas aparecerán al pie del patrón, y el usuario podrá realizar cualquier cantidad de notas que quiera.

### **3.5 Arquitectura básica**

La figura 3.5 muestra la arquitectura superficial del sistema completo.

La TMPP consta de varios módulos para realizar sus tareas, de los cuales sólo se han resaltado los que son importantes para este proyecto. El módulo *Map* proporciona la visualización de los datos espaciales almacenados en la base de datos geográfica, y además prepara estos datos en forma estructural y de grafo para pasárselos al módulo llamado apropiadamente *SUBDUE*. Este módulo realiza la minería correspondiente y le lanza sus resultados al módulo de TVG (*Visualización de resultados de minería de datos hecha en SUBDUE*), encargado de su adecuada visualización.

Naturalmente, la TMPP contiene más módulos, como el de SQL o el de Meta-relaciones, pero no son relevantes para el desarrollo de este proyecto, por lo que no serán vistos aquí.

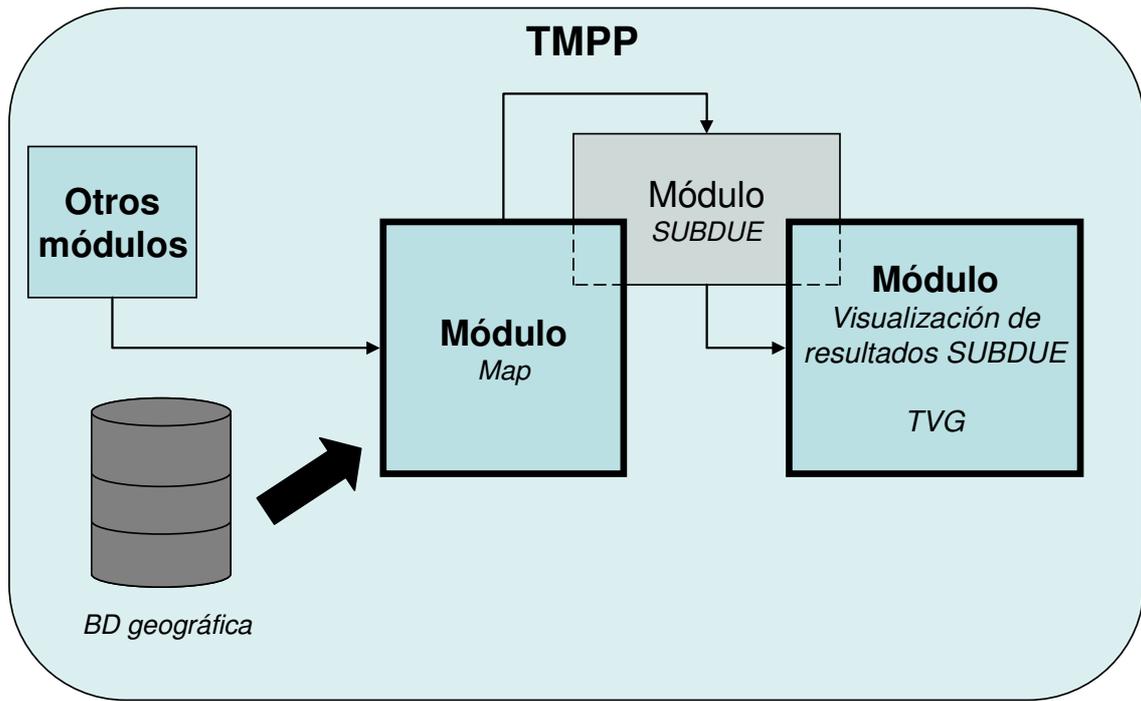


Figura 3.6 Arquitectura básica de TMP y TVG