UNIVERSIDAD NACIONAL DEL SUR

TESIS DE DOCTOR EN CIENCIAS DE LA COMPUTACIÓN

"ANÁLISIS ESTRUCTURAL ORIENTADO A SU APLICACIÓN EN CIENCIAS DE LA INFORMACIÓN Y EN INGENIERÍA"

Eduardo Xamena

BAHIA BLANCA ARGENTINA

2015

PREFACIO

Esta Tesis se presenta como parte de los requisitos para optar al grado Académico de Doctor en Ciencias de la Computación, de la Universidad Nacional del Sur y no ha sido presentada previamente para la obtención de otro título en esta Universidad u otra. La misma contiene los resultados obtenidos en investigaciones llevadas a cabo en el ámbito del Departamento de Ciencias e Ingeniería de la Computación durante el período comprendido entre el 7 de septiembre de 2010 y el 10 de febrero de 2015, bajo la dirección de la Dra. Nélida Beatriz Brignole, Profesora Adjunta del Dpto. de Ciencias e Ingeniería de la Computación e Investigadora Independiente del CONICET y la Dra. Ana Gabriela Maguitman, Profesora Adjunta del Dpto. de Ciencias e Ingeniería de la Computación e Investigadora Adjunta del CONICET.

Eduardo Xamena

UNIVERSIDAD NACIONAL DEL SUR

Secretaría General de Posgrado y Educación Continua

La presente tesis ha sido aprobada el 10/02/2015, mereciendo la calificación de 10 (Sobresaliente)

AGRADECIMIENTOS

En este pequeño apartado quiero expresar mi más sincero agradecimiento a todas las personas e instituciones que de una u otra manera formaron parte de este proyecto. En primer lugar, a mis Directoras, la Doctora Nélida Beatriz Brignole y la Doctora Ana Gabriela Maguitman, dos personas sumamente creativas y apasionadas por la investigación, con quienes tuve el privilegio de llevar adelante todos los trabajos de esta tesis. Muchas gracias por todas sus enseñanzas, y su colaboración incondicional.

Ningún sueño puede cumplirse si no contamos con aquellos que nos acompañan día a día, en nuestras alegrías y pesares. Este trabajo está dedicado a los dos soles que iluminan mi camino, mis amados Facundo y Claudia, mi hijo y mi esposa. Son infinitos el amor y la felicidad con que llenan mi corazón, y durante todos estos años han sido mi sustento, mi fuerza y el objetivo de todo mi trabajo.

También quiero hacer parte de este agradecimiento a quienes con tanto cariño nos ayudan siempre. A mis padres y hermanos, a mis amigos y compañeros de la Planta Piloto de Ingeniería Química, y a todos los familiares y amigos que de una u otra manera han sido parte de nuestras vidas a lo largo de estos años.

Por último, agradezco el apoyo de todas las instituciones académicas que me permitieron llevar a cabo mis investigaciones: La Facultad de Ciencias Exactas y el Consejo de Investigación de la Universidad Nacional de Salta; el Departamento de Ciencias e Ingeniería de la Computación de la Universidad Nacional del Sur, Bahía Blanca; El Consejo Nacional de Investigaciones Científicas y Técnicas; y la Planta Piloto de Ingeniería Química, Bahía Blanca.

A mi hijo

A mi esposa

A mis padres

RESUMEN

El objetivo de esta tesis es el desarrollo de métodos de particionamiento y análisis estructural de modelos matemáticos y estructuras de datos provenientes de distintas áreas del conocimiento. Los requerimientos que se desean cumplir son los siguientes: I. Manejo más eficiente de los modelos y las estructuras y II. Uso más extensivo de la información contenida en ellos. Como punto de partida para el análisis estructural de los diversos casos estudiados, se utilizó el Método Directo Extendido (MDE). Además, uno de los algoritmos sobre grafos que comprenden las distintas etapas de su funcionamiento fue utilizado sobre la estructura de un directorio web.

Con respecto a los temas abarcados en la investigación, esta tesis está concentrada en dos grandes campos de aplicación: I. Particionamiento estructural de modelos matemáticos provenientes de la Ingeniería Química, y II. Modelos estructurales que se utilizan como base para las Ciencias de la Información. En la primera sección se describe el trabajo realizado sobre distintos modelos matemáticos de Ingeniería de Sistemas de Proceso. Luego, la investigación fue enfocada en el tratamiento estructural de grandes volúmenes de datos que son utilizados para el cálculo de medidas de similitud semántica.

De acuerdo con las dos áreas de aplicación mencionadas, se enumeran las contribuciones del trabajo realizado:

- En Ingeniería de Sistemas de Proceso se perfeccionó el MDE, que genera un particionamiento de la estructura del sistema de ecuaciones. Con ello se hizo posible no sólo aumentar la cantidad de variables determinables en los modelos matemáticos implementados, sino también reducir la complejidad de cálculo. Esto último se logró con un mejor ordenamiento de las ecuaciones, poniendo énfasis en la selección guiada de ecuaciones con características deseables, como por ejemplo un bajo grado de no linealidad.

Tanto para problemas de simulación como de optimización, este algoritmo puede hacer más sencilla la tarea de su resolución y disminuir la cantidad de información inicial requerida en los modelos correspondientes.

- En Ciencias de la Información, fueron elaborados distintos modelos de propagación de relevancia sobre un directorio conocido de sitios de internet. Algunos se obtuvieron mediante la utilización de operaciones sobre matrices ralas de gran porte. Los otros fueron generados con un algoritmo sobre grafos que lleva a cabo la detección de componentes fuertemente conexas en un grafo dirigido. Este algoritmo está implementado en una de las etapas del MDE, descripto anteriormente. Además, estos modelos fueron validados experimentalmente en base a criterios estadísticos. Los mismos pueden aumentar la precisión en la determinación de valores de similitud semántica entre documentos, lo cual puede apreciarse en pruebas estadísticas llevadas a cabo a lo largo de la investigación. De acuerdo a los resultados obtenidos, su información sobre propagación de relevancia puede ser útil para diferentes propósitos en Ciencias de la Información.

ABSTRACT

The objective of this thesis is the development of structural partitioning and analysis methods for mathematical models and data structures that come up in different knowledge areas. The desired requirements to be fulfilled were: I. more efficient handling of the structures and models and II. wider usage of the information. As a starting point for the structural analysis of the various case studies, the Extended Direct Method (EDM) was employed. Besides, one of its graph-based algorithms that take part in some of EDM stages was applied to web-directory structures.

Regarding the research topics involved, this work concentrated in two big application fields: I. Structural partitioning of mathematical models from Chemical Engineering, and II. Structural models used as a basis in Information Sciences. In the first place, different mathematical models of Process Systems Engineering were analysed. Then, the research was focused on structural treatment of big data stores that are useful for semantic similarity measurements calculation.

According to these application areas, the contributions of this thesis are the following:

- In Process Systems Engineering the EDM algorithm, which generates a partitioning of the equations system structure, was improved. This improvement made it possible not only to augment the amount of determinable variables in the implemented mathematical models, but also to reduce the calculation complexity. The latter was achieved by means of a better equation ordering with emphasis on the guided selection of equations with desirable properties, e.g. a low degree of non-linearity. For both simulation and optimization problems, the improved algorithm can make their resolution task easier and diminish the amount of required initial information about the corresponding models.

- In Information Sciences various relevance propagation models over a known Web sites Directory were developed. Some of these models were obtained by using matrix operations on broad-range sparse matrices. The other ones were generated with a graph algorithm that performs the strongly connected components detection over a directed graph. This algorithm is implemented in one of the stages of the EDM, described previously. Besides, they were empirically proved, based on statistical criteria. The developed models can raise accuracy on determining semantic similarity between documents. They had an average accuracy of 65% in the implemented tests. According to this result, their information about relevance propagation could be useful for different purposes in Information Sciences.

CONTRIBUCIONES CIENTÍFICAS

Eduardo Xamena, Nélida Beatriz Brignole, y Ana Gabriela Maguitman, "Computational Models of Relevance Propagation in Web Directories," en el *Simposio Argentino de Inteligencia Artificial* de las *40º Jornadas Argentinas de Informática e Investigación Operativa*, Córdoba, Córdoba, Argentina, 29/8/2011 al 2/9/2011.

Eduardo Xamena, Ana Gabriela Maguitman, y Nélida Beatriz Brignole, "Optimized resolution of systems of equations," en el *XVIII Congreso Argentino de Ciencias de la Computación*, Bahía Blanca, Buenos Aires, 2012.

Eduardo Xamena, Nélida B Brignole, y Ana Gabriela Maguitman, "Strongly Connected Components Detection in Open Directory Project Graphs," en el *XXXI Congreso de Mecánica Computacional* de la *Asociación de Mecánica Computacional Argentina (AMCA)*, Salta, Salta, Argentina, Noviembre 2012, pp. 3411-3422.

Eduardo Xamena, Benjamín Cañete, Ana Gabriela Maguitman, y Nélida Beatriz Brignole, "Particionamiento estructural de modelos de plantas de procesos," en el *11º Congreso Interamericano de Computación Aplicada a la Industria de Procesos* de la *Pontificia Universidad Católica del Perú*, Lima, Perú, Octubre 2013.

Eduardo Xamena, Nélida B Brignole, y Ana G Maguitman, "A study of relevance propagation in large topic ontologies," publicado en la revista internacional *Journal of the American Society for Information Science and Technology*, Jhon Wiley & sons, vol. 64, n 11, pp 2238-2255, Noviembre de 2013 (Publicado online: Septiembre de 2013).

ÍNDICE

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

ÍNDICE DE ALGORITMOS

Capítulo 1:

Introducción

1.1 Contexto y Motivaciones

En los problemas que surgen a lo largo de las distintas áreas de la ingeniería y las ciencias de la información, es muy frecuente encontrarnos con modelos matemáticos ralos de gran porte [1]. Desde el diseño óptimo del instrumental de una planta de procesamiento físico/químico [2] hasta la elaboración de un conjunto de medidas de interrelación entre las páginas de un set de datos de un directorio web [3], nos podemos encontrar con modelos con características estructurales similares. La explotación de estas características mediante diferentes herramientas puede significar un impacto dramático en la eficiencia de su tratamiento, y aumentar considerablemente la información útil que puede obtenerse por su empleo.

Muchos aspectos de la estructura subyacente en grandes modelos matemáticos pueden ser plasmados en esquemas gráficos o grafos [4]. Estos esquemas nos pueden dar un panorama general de ciertas relaciones entre los componentes de los modelos. A su vez, estos grafos se pueden representar mediante matrices ralas o de baja densidad y de gran tamaño. Los métodos de particionamiento que sirven de base para este trabajo se sustentan en la utilización de matrices de adyacencia o incidencia, asociadas a grandes conjuntos de datos. Existe una gran diversidad de trabajos relacionados con este tipo de matrices y con la representación estructural de modelos mediante grafos ([5], [6], [7], [8], [9], [10], [11]). En general, las matrices de adyacencia o incidencia estructural que expresan las relaciones entre las componentes de un modelo suelen ser ralas. Por ello, se hace muy útil su tratamiento para obtener beneficios interesantes en los objetivos perseguidos.

1.1.1 Modelos expresados mediante sistemas de ecuaciones

Diversos autores han explorado modelos matemáticos que derivan en grandes sistemas de ecuaciones lineales, por ejemplo [12] y [13]. En muchos de sus trabajos se puede denotar claramente el aumento en la eficiencia de cálculo, que se logra cuando las matrices de coeficientes que representan estos sistemas son abordadas con técnicas de particionamiento de distintas características. También, los sistemas de ecuaciones con una determinada estructura no lineal pueden ser tratados con otras técnicas de particionamiento o reorganización [14]. Cabe destacar al respecto de los sistemas no lineales, que la gran diferencia entre las matrices que representan a estos últimos respecto de los sistemas lineales, es que una matriz que representa a un sistema lineal contiene los coeficientes que acompañan a cada variable en las respectivas ecuaciones, mientras que una matriz de incidencia asociada a un sistema no lineal contiene solamente unos y ceros, de acuerdo a la presencia o no de cada variable en las distintas ecuaciones. Esta situación puede apreciarse en la Figura 1-1, en la cual se muestra un sistema lineal a la izquierda, con su correspondiente matriz de coeficientes, y uno no lineal a la derecha, con su matriz de incidencia asociada debajo. Esta diferencia de representación se sustenta en la imposibilidad de utilizar coeficientes para representar una ecuación no lineal genérica, ya que los términos que conforman una ecuación no lineal pueden estar compuestos por expresiones funcionales no lineales sobre las variables. Por ejemplo, una ecuación no lineal puede consistir en un polinomio sobre alguna variable, y además contener términos lineales o no lineales en otras variables (ecuación f_2 de la Figura 1-1).

$$S_1:\begin{cases} e_1: 9x_1 + 4x_2 - 7x_3 = 1 \\ e_2: \quad 2x_2 + x_3 = -2 \\ e_3: x_1 - x_2 = 0 \end{cases}$$

9	**4**	**-7**
0	**2**	**1**
1	**-1**	0

Matriz de coeficientes

$$S_2:\begin{cases} f_1: e^{x_1} + \cos(6 * x_2 - x_3) + 3.5 = 0 \\ f_2: x_3^4 + \sqrt[3]{x_2} - 2 = 0 \\ f_3: x_2 - 2x_2^3 + 5x_2^5 - 28 = 0 \end{cases}$$

1	**1**	**1**
0	**1**	**1**
0	**1**	0

Matriz de incidencia

Figura 1-1: Sistemas de ecuaciones con sus correspondientes matrices de representación.

La estructura de un sistema de ecuaciones no lineales no puede plasmarse en una matriz de coeficientes convencional, porque pueden coexistir diferentes expresiones en una misma ecuación. Una matriz común de coeficientes está limitada a ecuaciones con términos lineales, salvo que se construya una representación ad-hoc, como por ejemplo matrices de polinomios sobre un grupo definido de variables [15]. Aun así, la situación se complica más cuando la estructura no lineal de las ecuaciones no tiene una forma estandarizada. Un ejemplo de esta situación puede ser la incorporación de términos con funciones más o menos complejas sobre algunas variables, como podría ser un logaritmo o una función exponencial o potencial. En general, la matriz de incidencia asociada a un sistema no lineal se utiliza para determinar subsistemas más pequeños de ecuaciones para resolver en diferentes etapas o en paralelo, y también para descubrir singularidades del sistema desde el punto de vista estructural [16]. Dichas singularidades pueden entenderse como bloques de la matriz con una cantidad insuficiente de ecuaciones para la determinación de ciertas variables, lo cual determinaría un conjunto infinito o nulo de soluciones para el subsistema dado, como puede verse en la Figura 1-2. Allí se muestra, a la izquierda un sistema compuesto por una ecuación y dos incógnitas, y a la derecha un sistema que consiste en dos ecuaciones y cuatro incógnitas. Ambos son casos de singularidades estructurales.

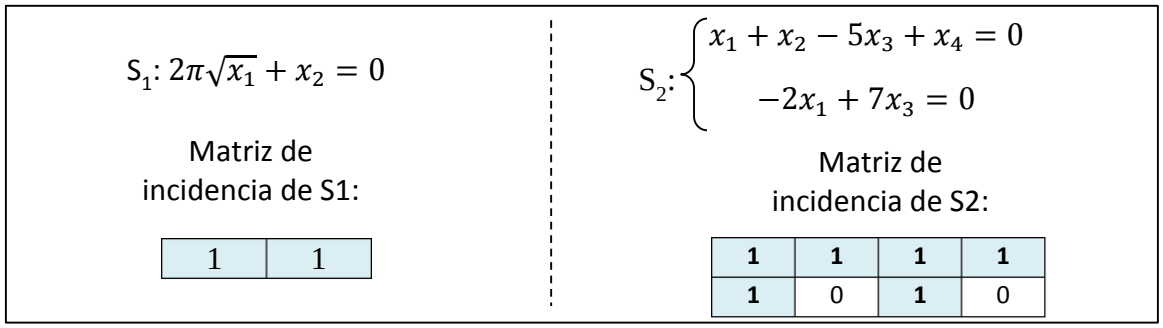


Figura 1-2: Ejemplos de singularidades estructurales en sistemas de ecuaciones.

A medida que aumenta el tamaño de los sistemas modelados, su resolución puede tornarse mucho más compleja. El tiempo requerido para ello crece de forma no lineal en función del tamaño de los modelos. Una de las circunstancias que hacen muy valorable el particionamiento de los modelos es justamente este aumento de tamaño. Si logramos obtener particionamientos equivalentes a raíz de las matrices que representan a dichos modelos, su resolución puede verse muy beneficiada siguiendo el principio de "dividir para vencer". De esta forma, los valores que deben obtenerse para el correcto ajuste del modelo pueden provenir de la resolución de partes más pequeñas del mismo. Estos particionamientos, que tal vez resulten mucho más sencillos de resolver que el modelo completo, podrán brindan mayor eficiencia en las tareas de cálculo que sean requeridas. Otra ventaja que surge de tratar los modelos de forma particionada, es la posibilidad de paralelizar los cálculos que deben ser llevados a cabo, siempre y cuando no exista acoplamiento o dependencias entre las ecuaciones de los distintos bloques.

La complejidad de diversos modelos puede aumentar considerablemente cuando se incluye el tiempo como factor dinámico en los mismos. Como consecuencia de la necesidad de estudiar este comportamiento dinámico, deben ser analizados los perfiles de ciertas variables en el tiempo. Este tipo de situaciones produce ecuaciones diferenciales que deben ser exploradas con diversos métodos de aproximación, como por ejemplo Colocación

Ortogonal [17]. Estos métodos son también generadores de sistemas de ecuaciones no lineales con estructuras particulares, que también podrían ser resueltos de manera más eficiente utilizando técnicas de particionamiento estructural.

1.1.2 Modelos para grandes volúmenes de datos

Cuando se trabaja sobre grandes volúmenes de datos, como por ejemplo los millones de páginas web que pueden estar catalogadas en un directorio, es muy común representar las relaciones existentes entre las entidades que contienen los datos mediante grafos y sus correspondientes matrices de adyacencia [10]. Desde la determinación de fórmulas y modelos de relevancia entre documentos [18] hasta el cálculo de medidas de similitud semántica dentro de los mismos [3], las matrices que reflejan relaciones entre las categorías de un directorio son un soporte de extremada importancia para llevar a cabo las operaciones necesarias y aumentar la eficiencia en el cálculo.

A través de la exploración de la estructura de la matriz de adyacencia de este tipo de modelos, se pueden descubrir patrones subyacentes que permitan derivar propiedades impensadas de los mismos, como en [10] o en [19]. El método para descubrir patrones estructurales puede contener entre sus pasos el procesamiento de matrices de adyacencia mediante algoritmos sobre sus grafos asociados. Un algoritmo que ha demostrado gran aplicabilidad en el ámbito de los grandes almacenes de datos es el de *Detección de Componentes Fuertemente Conexas* [20]. Éste último permite detectar grupos de nodos relacionados en un grafo, con la característica distintiva de poder hacer recorridos de ida y vuelta de un nodo a otro dentro de cada componente. En [10], justamente se hace uso del algoritmo mencionado para determinar grandes grupos de sitios web interconectados. Se

lleva a cabo un particionamiento dentro de la estructura de un almacén de datos, con el propósito de identificar grupos representativos de la estructura del corpus.

Otra forma de particionamiento puede ser la obtención de clústeres. En particular, esta tarea puede resultar muy útil en disciplinas como Data Mining [21]. Al representar documentos y contenidos mediante matrices de relación de conceptos o categorías, y a su vez utilizar un enfoque estructural para el reconocimiento de patrones en grandes volúmenes de datos [22], es factible un aumento en la eficiencia de determinación de agrupamientos. La obtención de clústeres puede llevarse a cabo mediante la utilización de diversos algoritmos ampliamente estudiados. Por ejemplo, en [23] se detallan dos algoritmos para la tarea de formación de clústeres, generando una nueva técnica para esta tarea, que incluye también los conceptos fundamentales de la metaheurística conocida como "*Simulated Annealing*" [24]. En [23] también es destacable el uso que se hace de distintos conjuntos de datos, lo cual refleja la robustez de las técnicas utilizadas.

1.1.3 Otros modelos con características estructurales interesantes

Las particularidades estructurales de las matrices que representan a un sinnúmero de clases de modelos, pueden ser objeto de trabajo para aumentar la eficiencia y mejorar el uso de los mismos. Cualquier operación sobre un gran conjunto de información puede beneficiarse por su tratamiento estructural. En [25] por ejemplo, se destaca la importancia de particionar grafos y mallas correspondientes a los datos de una aplicación que se ejecuta en paralelo. Al dividir el conjunto principal de datos de cualquier problema grande, podemos balancear correctamente la cantidad de información con la que debe trabajar cada procesador, y de esa manera minimizar el tiempo requerido para los cálculos y procesamientos que debieran realizarse sobre dichos datos. Para esa tarea, el autor hace uso

de una librería de software que brinda diversas facilidades para la implementación de algoritmos en forma paralela y concurrente, descripta a su vez en [26]. También en [27] se puede apreciar la ventaja del trabajo en paralelo con datos distribuidos. En esa publicación, se distingue el tratamiento de grupos de nodos de grandes grafos como subgrafos, que pasan a ser unidades independientes o particionamientos de la entidad mayor. Siguiendo ese razonamiento, se pueden llevar a cabo estudios estructurales que permitan el hallazgo de nuevas informaciones sobre los modelos analizados.

En suma a la labor sobre los datos principales de cada modelo, un tratamiento estructural adecuado sobre las matrices que contienen la información correspondiente a la distribución del trabajo sobre un entorno multiproceso en problemas de gran envergadura, también podría significar un gran avance en la eficiencia de resolución de los mismos. Al escalar problemas cuya resolución determinística requiere tiempos computacionales no polinomiales en función de su tamaño (NP-completos), como podrían ser los problemas de ruteo de vehículos -VRP- [28] o problemas de Planeamiento de recorridos óptimos -Motion Planning- [29], las matrices estructurales que vinculan los datos de distintas localizaciones podrían ser abordadas con las diversas técnicas cubiertas en esta tesis. Esto significa no solamente trabajar sobre la estructura de datos primaria que describe a cada modelo, sino también sobre la estructura asociada a la red de procesamiento que se utiliza para planificar y monitorear los algoritmos de resolución paralela. En los algoritmos utilizados en [29], por ejemplo, podrían visualizarse mejoras interesantes en términos de tiempo requerido para obtener una solución factible y consumo de recursos computacionales. Esto se debe a que se utilizan grandes cantidades de operaciones aritméticas, lo cual podría beneficiarse del uso de estructuras de datos optimizadas.

Un tema del que se puede sacar mucho provecho mediante el particionamiento y análisis estructural es la *Expresión Génica*. En esta disciplina se hace uso de estructuras de datos de tamaños colosales, para descubrir la influencia que puede tener un gen o grupo de genes sobre alguna característica particular en un individuo. Los datos que constituyen las estructuras generadas provienen de distintos análisis realizados sobre la composición genética de diversos organismos. Uno de los objetivos de llevar a cabo estos estudios es la detección de grupos de genes que manifiestan patrones de expresión similares [30]. Para poder explotar correctamente la información contenida en estos grandes almacenes, también se requiere un estudio previo de las similitudes entre los datos contenidos en dichas estructuras. Para ello, se efectúa la construcción de matrices de similitud entre los distintos genes, basadas en determinadas fórmulas para el cálculo de las medidas de similitud correspondientes ([30], [31]). Después de haber elaborado estas matrices de similitud, se puede construir los clústeres de genes con comportamientos similares mediante la utilización de diversas técnicas y algoritmos. Un ejemplo de esta clase de algoritmos puede consultarse en [32], y en [33] se busca utilizar para los agrupamientos, además de los datos de Expresión Génica, la información de almacenes de datos de relaciones entre proteínas vinculadas a la interacción de los genes. En [34] se hace uso de la Expresión Génica para un propósito particular: El reconocimiento de rasgos faciales mediante datos genéticos. Sin duda alguna, este y otros temas de importancia crucial para el ámbito científico y el desarrollo humano pueden verse favorecidos por el correcto manejo de las grandes estructuras de información que subyacen entre sus características.

También en la rama de las ciencias biológicas, podemos mencionar el trabajo de [35], en el cual se evidencia la utilidad del tratamiento estructural. En este trabajo se destaca la importancia de contar con herramientas que sean capaces de gestionar

correctamente grandes volúmenes de información, proveniente de diversas fuentes. Mediante la sistematización de anotaciones semánticas acerca de distintos fenómenos biológicos de la naturaleza, se logra elaborar medidas de similitud sobre diversos modelos, los cuales brindan la posibilidad de determinar información relevante para responder preguntas científicas trascendentales. El trabajo sobre estos grandes conjuntos de modelos biológicos permite también llevar a cabo diversos análisis con gran valor estadístico, que pueden ser de utilidad para validar o refutar toda clase de hipótesis. El manejo de estos modelos representados mediante redes complejas es explicado en detalle también en [36]. En pocas palabras, el tratamiento estructural de los modelos biológicos descriptos es crucial para la obtención de resultados científicos de suma importancia.

1.2 Antecedentes

1.2.1 Análisis y Particionamiento Estructural en Ingeniería Química

Muchos modelos matemáticos asociados a la Ingeniería Química son susceptibles de ser modificados a lo largo del tiempo, y estas modificaciones en general pueden aumentar sus tamaños y hacerlos mucho más complejos. Si tomamos una perspectiva más abstracta del funcionamiento de una planta de procesos químicos, trabajando sobre el diagrama de flujo de la misma, podremos detectar ciertas particularidades que nos darán la posibilidad de optimizar distintos aspectos de su funcionamiento. Tal enfoque es tenido en cuenta en [37], en donde se utilizan algoritmos evolutivos basados en la teoría de grafos para abordar casos de estudio de optimización estructural de plantas químicas.

Algunos aspectos dinámicos del funcionamiento de plantas químicas también han sido objeto de estudio mediante el enfoque estructural. El concepto de Observabilidad, entendido como la cantidad de variables que pueden ser determinadas mediante el modelo

matemático de una planta, es explorado en [38]. En dicho trabajo, la propuesta consiste en analizar la observabilidad en sistemas de reacción. A diferencia de esta clase de investigaciones, los métodos de tratamiento estructural utilizados en esta tesis fueron elaborados para explotar aspectos estáticos de una planta química en estado estacionario.

En disciplinas más específicas dentro de la Ingeniería Química como por ejemplo el análisis de fallas, se puede mencionar el trabajo llevado a cabo en [39] y refinado en [40], que involucra aspectos estructurales que pueden ser la base de buenos diagnósticos sobre el esquema de datos del funcionamiento de una planta química. En estos trabajos se utilizan grafos dirigidos con signo, lo cual tiene la ventaja de indicar la dirección de los flujos modelados en el sistema. La teoría de grafos se hace presente en estos trabajos, ya que se aprovechan algunos algoritmos para el hallazgo de ciertas agrupaciones en la información, que permiten lograr el objetivo propuesto.

El particionamiento estructural en modelos de Ingeniería Química ha sido objeto de estudio de dos tesis doctorales dentro del grupo de investigación en que se enmarca este trabajo. Primeramente fue aplicado para identificar conjuntos de asignación de variables y ecuaciones dentro de sistemas de ecuaciones asociados a modelos de Diseño de Instrumentación en Ingeniería Química. Luego fue refinado en esta misma área, con el agregado de la determinación de la complejidad de las ecuaciones, para lograr particionamientos que permitieran un manejo más sencillo del modelo. También fue implementado en otras clases de modelos, para extender su uso en diversas áreas de aplicación.

En primer lugar, en el trabajo de [41], se muestra el desarrollo del Método Directo (MD) para Análisis de Observabilidad en Diseño de Instrumentación, el cual fue plasmado en la tesis del mismo autor principal, [42]. Entre los resultados de dicho trabajo se puede

destacar la posibilidad que brinda el MD para determinar los valores de una gran cantidad de variables, dado un conjunto de valores iniciales y dependiendo de las características del modelo. La herramienta generada ha sido utilizada en conjunto con distintos programas de computación que generan y validan modelos de Ingeniería Química. En [43], por ejemplo, se muestra un sistema de soporte para decisiones (DSS) que hace uso del MD para la etapa de análisis de observabilidad del diseño de instrumentación de una planta química. El MD no lleva a cabo ningún procesamiento computacional sobre el grado de complejidad de las ecuaciones y variables en los sistemas, lo cual es abordado en la tesis que se describe en el siguiente párrafo.

En segundo lugar, fue elaborada la tesis de [44], en la cual se trabaja sobre el MD elaborado en la tesis de I. Ponzoni. Parte del aporte de esta tesis es la generación de un nuevo método de particionamiento estructural, el Método Directo Extendido (MDE), publicado en [45]. Este nuevo método complementa al MD en cuanto a la generación de valores de Grado de No-Linealidad (GNL) para variables, términos y ecuaciones. Con estos valores, el proceso de particionamiento estructural del sistema de ecuaciones es conducido de una manera diferente que en el MD. La diferencia radica en el hecho de la incorporación de la información contenida en el GNL, para privilegiar la formación de bloques lo más lineales posible. Así, las ecuaciones que se utilizan para determinar los valores de las variables determinables tendrán una complejidad menor. Esto facilita la determinación de los valores de las variables correspondientes de una manera más sencilla, aumentando la eficiencia de resolución del modelo. Se evita de este modo la necesidad de resolver sistemas de ecuaciones muy complejos, o que simplemente tengan entre sus ecuaciones un término no lineal. Las ecuaciones no lineales del modelo pueden ser utilizadas para otros fines, como por ejemplo, reconciliación de datos o control, lo cual implica solamente

reemplazar los valores de variables y efectuar cálculos para determinar el valor resultante de cada ecuación.

1.2.2 Análisis estructural en Ciencias de la Información

En el gran ámbito de las Ciencias de la Información, el manejo de volúmenes de datos o documentos de gran envergadura con diferentes propósitos suele ser abordado mediante diferentes técnicas ampliamente estudiadas. La utilidad del tratamiento de semejantes estructuras suele desprenderse de diversas necesidades de información, como puede ser la consulta de un usuario sobre información específica contenida en un corpus, o la determinación de los valores correspondientes de interrelación entre los documentos o datos contenidos en el mismo corpus. Con frecuencia, el enfoque utilizado para lograr el hallazgo de información valiosa consiste en un tratamiento estructural de la información contenida en alguna base de conocimiento. Este tratamiento puede involucrar la detección de porciones más pequeñas dentro del corpus trabajado, las cuales identifican grupos de elementos relacionados entre sí por algún criterio. La búsqueda de tales relaciones suele realizarse mediante la representación matricial o por grafos de la estructura del corpus, identificando a cada objeto de información con un nodo de un grafo, y a las relaciones que pudieran existir entre ellos como aristas con distintos niveles de ponderación, por ejemplo. En el trabajo de [46] se puede ver una forma de representar objetos de datos mediante grafos, donde cada nodo representa un proceso correspondiente a alguna aplicación en una red de datos, y las relaciones establecidas se sustentan en distintos criterios, como pueden ser los tipos de procesos que representan los nodos o los recursos que utilizan. Las agrupaciones de objetos descriptas suelen denominarse clústeres, y existen diversos algoritmos para obtenerlos, como fue mencionado anteriormente en este capítulo.

Una cuestión determinante en la búsqueda de información relevante en grandes volúmenes de información lo constituye el aspecto semántico de los objetos de datos que se manipulan. Por ejemplo, los usuarios de diferentes bases de datos pueden utilizar una gran diversidad de palabras para referirse a un mismo documento u objeto. Esto conlleva una gran dificultad, ya que torna muy difícil encontrar todos los datos que estén asociados con una consulta determinada. Para lidiar con este tipo de problemas, frecuentemente se utilizan matrices que representan la relación existente entre los términos utilizados en los documentos u objetos contenidos en una base de datos, con dichos documentos u objetos respectivamente. De esa manera se puede determinar con mayor precisión el grado de relación de algún término con un objeto de datos. En [47] y [48] se hace uso de este tipo de representaciones y de la técnica de Descomposición en Valores Singulares (SVD por sus siglas en inglés) sobre matrices para llevar a cabo tareas de recuperación de información y de textos.

Consecuentemente con el tema de la semántica de los términos asociados a un documento, la relevancia entre dos objetos de información juega un papel primordial en las tareas relacionadas con la búsqueda de información. Dicho concepto está vinculado fuertemente al contexto y al tipo de relaciones que pudieran existir entre objetos de una base de datos, y será descripto en la sección correspondiente de esta tesis. Por ejemplo, en [49] se hace uso del concepto de relevancia en la vinculación de especialistas con tareas requeridas por una empresa. En dicho trabajo se emplean diferentes representaciones mediante grafos para los objetos modelados. En este contexto, los objetos modelados pueden representar igualmente a trabajadores, contratos o trabajos llevados a cabo con anterioridad, así como a otras clases de documentos de la empresa. En este caso y mayormente en los trabajos de investigación de diversas áreas, la relevancia entre objetos

se define en función de aspectos estructurales, modelados, por ejemplo, mediante matrices o grafos.

El particionamiento estructural puede ser utilizado como una herramienta muy poderosa para, entre otros propósitos, la determinación de valores de similitud semántica dentro de grandes volúmenes de documentos. Esto puede ser logrado, por ejemplo, mediante la utilización de modelos de propagación de relevancia que aumenten la precisión de las relaciones existentes entre tópicos de un directorio de internet. En [3], se muestra el proceso de cálculo de valores de similitud semántica sobre el directorio web de Open Directory Project (ODP). Dicho directorio y las distintas representaciones de la información que contiene son descriptos en detalle en las secciones correspondientes de esta tesis. El cálculo de los valores de similitud semántica mencionados se llevó a cabo utilizando modelos de propagación de relevancia generados a partir de la topología del grafo asociado a ODP, y operaciones matriciales sobre el mismo. Una posibilidad que quedaba abierta en este trabajo era el tratamiento estructural de estos modelos, para aumentar la precisión de la medida de similitud semántica obtenida para el directorio.

1.3 Objetivos

El objetivo general de este trabajo es lograr un aumento en la eficiencia y utilidad de los distintos modelos desarrollados y analizados, mediante la aplicación de diversas formas de análisis sobre su estructura subyacente. Teniendo en cuenta el desarrollo de los métodos de particionamiento explicados en la sección anterior, el objetivo esencial de esta tesis es el abordaje integral de tales métodos y otras técnicas de análisis estructural para su aplicación en dos áreas principales: I. Ingeniería de Sistemas de Proceso y II. Ciencias de la Información. Los objetivos específicos para cada área de trabajo son los siguientes:

a) Estudio del particionamiento estructural en Ingeniería:

En Ingeniería de Sistemas de Procesos, el propósito del particionamiento estructural es el de disminuir la complejidad de resolución de los modelos surgidos de distintos problemas. Su aporte está dado por la reorganización en bloques de menor complejidad del sistema de ecuaciones que representa a cada uno de estos modelos. En este trabajo se pretende realizar un estudio más profundo del GNL de ecuaciones y variables de dichos sistemas, para aumentar la eficiencia en su resolución. Esto constituye un objetivo específico, para el cual fueron desarrollados diversos trabajos de investigación, entre ellos [50] y [51].

b) Modelos de propagación de relevancia para grandes volúmenes de información:

En Ciencias de la Información, la meta es el desarrollo de modelos estructurales que permitan aumentar la precisión en diversas métricas asociadas a la recuperación de información, las cuales pueden ser útiles para numerosos fines. En particular, se desarrollaron modelos de propagación de relevancia entre tópicos de un directorio de sitios web, mediante el uso de distintas herramientas ([52], [18]).

1.3.1 Metas a alcanzar en las distintas áreas de aplicación

Los objetivos que se desea alcanzar por medio del particionamiento y el análisis estructural varían de acuerdo al campo de aplicación. Podemos mencionar, por ejemplo, las siguientes proposiciones de acuerdo a los distintos problemas explorados en esta tesis:

1. En el campo del Diseño de Instrumentación de plantas químicas, resulta muy conveniente realizar un preprocesamiento de la matriz de incidencia correspondiente a un proceso, reordenando las ecuaciones del mismo. Con esto podemos lograr acercarnos un poco más al valor teórico de observabilidad de variables para el modelo

correspondiente. Dicho valor determina la cantidad máxima de información del proceso que puede ser obtenida mediante la resolución del sistema de ecuaciones asociado. Por lo tanto, el objetivo del particionamiento en este contexto es *el incremento en la información que pueda obtenerse del proceso*.

2. Los problemas de Simulación/Optimización, debido a sus requerimientos inherentes de eficiencia, también pueden ser resueltos de una manera más adecuada mediante el uso del particionamiento. Si llevamos a cabo alguna de las técnicas estructurales en sus sistemas de ecuaciones asociados, la resolución del modelo correspondiente puede realizarse con un menor esfuerzo computacional. Respecto de la información inicial requerida para resolver un modelo, se pretende, mediante el particionamiento, encontrar configuraciones estructurales óptimas para cada sistema de ecuaciones que permitan la obtención del valor de la máxima cantidad posible de variables, con el mínimo conjunto necesario de valores de entrada. En este caso, la meta es lograr una resolución más eficiente y una menor cantidad de información inicial necesaria del modelo.
3. Para la determinación de modelos precisos de propagación de relevancia en directorios de internet, es necesario el tratamiento estructural de diversos y grandes conjuntos de datos. El objetivo aquí es la obtención de modelos más certeros que permitan determinar nuevas relaciones significativas de relevancia entre tópicos. De esta forma, el análisis y particionamiento estructural de dichos modelos se lleva a cabo con el propósito de obtener nuevos modelos más precisos.

1.4 Organización

1.4.1 Parte I: Particionamiento estructural en modelos de Ingeniería

El **capítulo 2** de esta tesis contiene una explicación detallada de los algoritmos sobre grafos utilizados a lo largo de este trabajo de tesis, y los métodos de particionamiento empleados en Ingeniería de Sistemas de Proceso y Diseño de Instrumentación. Se detalla en profundidad el marco teórico que constituye la base del funcionamiento de los mismos. También se mencionan sus aplicaciones en distintos ámbitos de la ciencia.

Siguiendo con la aplicación del particionamiento estructural en Ingeniería, el **capítulo 3** expone la aplicación de los métodos descriptos en el capítulo 2 en casos estudiados en diversos trabajos publicados. Los resultados de estos casos son discutidos, y se proponen algunas mejoras adicionales a los métodos estructurales aplicados. También se destacan las ventajas y desventajas del uso de estos métodos en la optimización con restricciones.

El **capítulo 4** explica la necesidad de una validación estadística experimental sobre los métodos de particionamiento estructural, y el marco de trabajo desarrollado en esta tesis para lograr tal objetivo. Se explica allí la elaboración de las herramientas correspondientes para lograr la generación automatizada de casos de estudio, en base a parámetros estadísticos que se corresponden con casos reales del diseño de instrumentación de plantas de proceso.

1.4.2 Parte II: Modelos de propagación de relevancia sobre un directorio de internet

Para iniciar esta sección, el **capítulo 5** detalla los aspectos fundamentales del análisis estructural en sus diversas formas, utilizado en el área de las Ciencias de la

Información. Se describen conceptos como por ejemplo el de Ontologías Informáticas, y la teoría de grafos aplicada en catálogos estructurales. También se introduce el tema de la Relevancia en grandes corpus de información, y se presenta el catálogo de sitios web del Proyecto de Directorio Abierto (Open Directory Project, ODP), sobre el cual se trabajará posteriormente.

El concepto de Propagación de Relevancia es incluido en el **capítulo 6**, junto con el conjunto de modelos elaborados en este trabajo de tesis para tal efecto sobre el directorio de ODP. Son mostrados los fundamentos matemáticos de las operaciones sobre matrices llevadas a cabo para obtener dichos modelos, y también se ilustra una herramienta de visualización desarrollada para analizar los modelos.

Una vez descriptos los modelos de propagación de relevancia, en el **capítulo 7** la redacción gira sobre el eje de la validación empírica estadística de los mismos. Se muestran distintos análisis cualitativos y cuantitativos en base a la información provista por los modelos, y experimentos con usuarios humanos llevados a cabo sobre las relaciones que estos modelos infieren. En base a todo esto, fueron plasmadas las conclusiones correspondientes sobre la utilidad de la información obtenida por la generación de los modelos de propagación de relevancia.

Finalmente, el **capítulo 8** expone las conclusiones resultantes de este trabajo, y las líneas de trabajo abiertas como consecuencia del mismo.

Parte I:

Particionamiento Estructural en Modelos de Ingeniería

Capítulo 2:

Métodos de Particionamiento sobre Modelos de Ingeniería

2.1 Introducción

En las distintas áreas de la Ingeniería y la ciencia en general, es frecuente trabajar con modelos matemáticos que crecen paulatinamente en tamaño y complejidad. A menudo encontramos modelos representados por grandes sistemas ralos de ecuaciones, con diferentes grados de complejidad ([53], [54], [55], [56]). Estos sistemas pueden ser lineales, los cuales son evaluados por métodos conocidos de resolución, tales como los llevados a cabo en [57], [58], o sistemas de ecuaciones con diferentes grados de no linealidad. En este último grupo se pueden encontrar sistemas computacionalmente muy complicados para resolver, debido a la presencia de ecuaciones no lineales más o menos complejas [59], [60]. Una manera de optimizar este proceso puede ser la reorganización estructural del sistema. Dicha reorganización puede reducir drásticamente el número de pasos computacionales necesarios para la resolución de los mismos.

El método directo (MD) [42] y el método directo extendido (MDE) [44] son herramientas surgidas en el campo del diseño de instrumentación de plantas químicas. Permiten llevar a cabo el análisis de observabilidad de las variables que intervienen en el sistema de ecuaciones del modelo que describe la planta correspondiente. El MD, por su parte, realiza una reorganización estructural de la matriz de incidencia asociada al sistema de ecuaciones antes mencionado, generando así un sistema equivalente que se puede resolver en etapas y de manera más eficiente. Siguiendo la misma idea, el MDE efectúa este particionamiento teniendo en cuenta el grado de no linealidad (GNL) de las ecuaciones, variables y términos involucrados en las ecuaciones, para lograr una reorganización aún mejor. Ambas técnicas se basan en la aplicación de estos algoritmos sobre grafos, que corresponden a diferentes representaciones del grafo asociado al sistema:

I. Algoritmo de pareamiento maximal en bigrafos [61], II. Algoritmo de detección de componentes fuertemente conexas en digrafos [20].

Dentro de este capítulo se desarrolla el marco teórico concerniente a los dos algoritmos sobre grafos que se aplican en los métodos de particionamiento, y una descripción del funcionamiento de dichos métodos, destacando las diferencias esenciales entre ambos.

2.2 Teoría de grafos

Muchos problemas que se presentan en diferentes ámbitos de la vida cotidiana pueden representarse mediante grafos, con las infinitas posibilidades que estas representaciones nos dan para su resolución. El origen de su teoría se remonta al año 1736, cuando Leonhard Euler utilizó este tipo de abstracciones para resolver el célebre "*Problema de los Puentes de Königsberg*".

Si bien existe una base teórica fuertemente desarrollada a lo largo de los años ([62], [63], [64]), en las últimas décadas el trabajo de investigación se ha volcado al desarrollo de algoritmos eficientes para resolver problemas de gran tamaño. Esto se debe al advenimiento de equipos informáticos con gran capacidad de almacenamiento y velocidad de procesamiento de la información, y al crecimiento del tamaño de los problemas que se necesita resolver. También, un condimento importante para este tipo de investigaciones es el surgimiento y masificación de los equipos multiprocesadores o *multicore*, que posibilitan la aplicación del principio fundamental *"divide y vencerás"*. Con este tipo de procesadores, la implementación de los algoritmos tiene el trabajo agregado de la distribución de los datos y algoritmos entre los distintos procesadores, pero la eficiencia en la resolución de los problemas tiene un incremento dramático en muchos casos. En el trabajo de [65] se aprecia

el aprovechamiento de las ventajas del paralelismo, utilizando procesadores gráficos (*GPU*) para implementar una estructura paralela de instrucciones simples y datos múltiples (*Single Instruction, Multiple Data, SIMD*) para el tratamiento de grandes grafos.

Los métodos de particionamiento trabajados en esta sección están basados en procedimientos estructurales sobre modelos ingenieriles. Estas rutinas se apoyan en la teoría de grafos para hacer uso de sus ventajas en cuanto a la representación de relaciones existentes entre las entidades modeladas. Para el caso general de los modelos matemáticos derivados de la ingeniería, los grafos empleados reflejan las relaciones existentes entre las ecuaciones y variables de cada modelo. Las dos clases de grafos que se explotan en la implementación del MD y el MDE son los *grafos bipartitos* y los *grafos dirigidos*. Cada una de estas estructuras es utilizada con un propósito diferente en cada etapa de particionamiento, esencialmente de la misma manera en los dos métodos. La diferencia entre los métodos radica en la incorporación de información del grado de complejidad de ecuaciones y variables, llevada a cabo por el MDE.

2.2.1 Conceptos básicos

La información contenida en esta sección corresponde a los conceptos descritos en [62]. Allí se desarrollan en detalle todos los conceptos concernientes a grafos no dirigidos, grafos dirigidos y bigrafos, y también se explica el funcionamiento de algunos de los algoritmos utilizados aquí, entre ellos la detección de componentes fuertemente conexas en digrafos o una variante de la búsqueda de pareamientos maximales en bigrafos.

2.2.1.1 Grafos no dirigidos

Un grafo no dirigido se define como un conjunto de puntos o nodos llamado *V*, interconectados por otro conjunto de aristas llamado *E*. Una arista puede ser identificada

con el par no ordenado de los puntos que conecta: $e=(v_i,v_j)$. Las aristas se representan gráficamente mediante líneas que conectan a los nodos respectivos, y no tienen un sentido determinado. Analíticamente, un grafo no dirigido *G* con *n* nodos y *m* aristas puede ser representado de la siguiente manera:

$G=(V,E)$; $V=\{v_1,v_2 \ldots v_n\}$; $E=\{(v_{i1},v_{j1}),(v_{i2},v_{j2}) \ldots (v_{im},v_{jm})\}$

En la Figura 2-1 se muestra la representación gráfica y analítica de un grafo no dirigido.

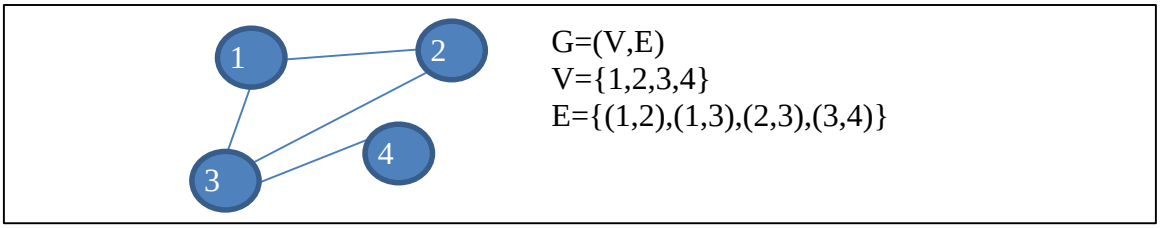


Figura 2-1: Ejemplo de representación gráfica y analítica de un grafo no dirigido.

2.2.1.2 Grafos dirigidos

Un grafo dirigido o digrafo, en cambio, es cualquier grafo en el cual cada arista tiene un sentido determinado. Por ello, en el par ordenado que representa una arista en un grafo dirigido, el orden de los puntos adquiere relevancia. De esta manera, si existe en el grafo una arista que va desde el nodo v_i al nodo v_j, el único par ordenado que la representa es (v_i,v_j), ya que el par (v_j,v_i) representaría una arista con sentido opuesto. Gáficamente, las aristas se representan como flechas que van del nodo origen al nodo destino. Un ejemplo de un grafo dirigido, con su representación gráfica y analítica, puede verse en la Figura 2-2.

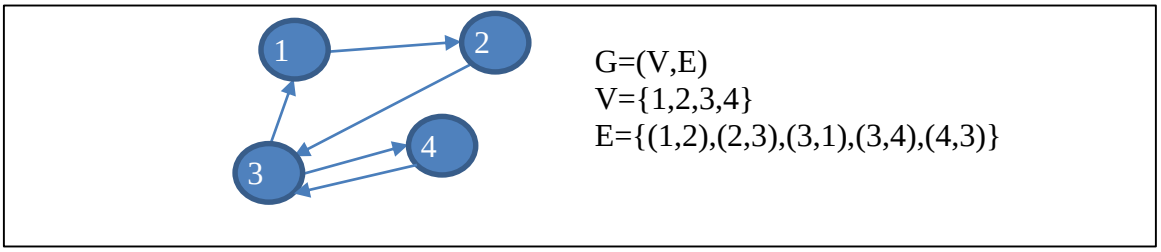


Figura 2-2: Ejemplo de representación gráfica y analítica de un grafo dirigido.

2.2.1.3 Bigrafos

Existe una clase de grafos que permite representar las relaciones entre objetos de dos clases diferentes. Estos son los denominados bigrafos o grafos bipartitos. Consisten en dos conjuntos de nodos *R* y *C*, uno para cada clase de objetos, y un conjunto *E* de aristas, las cuales vinculan cada una a un objeto de la primera clase con uno de la segunda clase. En la Figura 2-3 se puede ver la representación gráfica y analítica de un bigrafo.

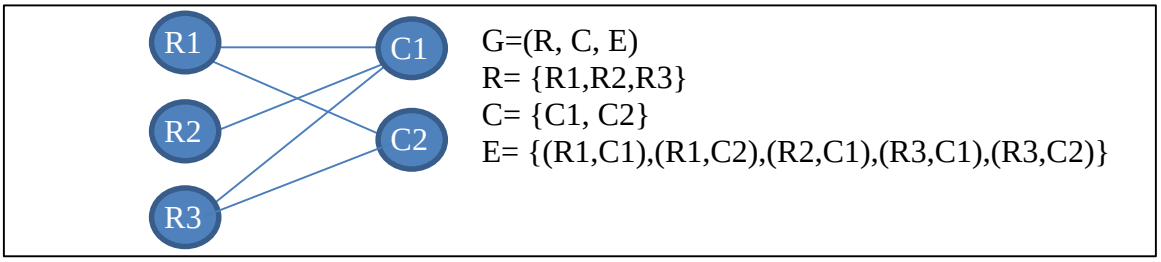


Figura 2-3: Ejemplo de representación gráfica y analítica de un bigrafo.

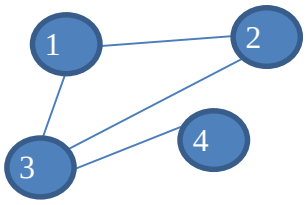
2.2.2 Matrices para la representación de grafos

Los distintos tipos de grafos enumerados anteriormente pueden ser representados mediante matrices que reflejan las relaciones entre sus nodos, denotadas por las aristas. Las matrices que se utilizan en los métodos de particionamiento utilizados en este trabajo simplemente reflejan la existencia o no de una arista que vincula a dos nodos. Por consiguiente, todas las matrices que se describen a continuación son binarias, es decir, solamente admiten un 1 o un 0 como valor en cada celda.

2.2.2.1 Matrices de adyacencia

Una matriz de adyacencia es una matriz cuadrada binaria que representa a los nodos y aristas de un grafo. Cada nodo se asocia con una fila y una columna de la matriz, y las aristas se indican con un 1 en la celda de la matriz correspondiente a los nodos que une. Si tuviéramos un grafo como el de la Figura 2-4, en su representación matricial tendríamos,

por ejemplo, un 1 en la componente (1,2) de la matriz (fila 1, columna 2) indicando la existencia de la arista que conecta los nodos 1 y 2. Y tendríamos un 0 en la componente (2,4), ya que no existe ninguna arista que una los nodos correspondientes en el grafo.

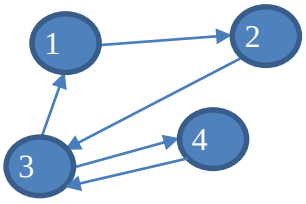


M=

	1	2	3	4
1	0	1	1	0
2	1	0	1	0
3	1	1	0	1
4	0	0	1	0

Figura 2-4: Grafo no dirigido de la Figura 2-1 y matriz de adyacencia correspondiente.

Cabe destacar que una matriz de adyacencia para un grafo no dirigido resulta simétrica, debido al doble sentido de las aristas. No es necesariamente así en el caso de los digrafos, ya que cada arista (v_i,v_j) tiene un sentido determinado, y se indica en la matriz en la fila *i-ésima* y la columna *j-ésima*, pero no en la columna *i-ésima* y en la fila *j-ésima*. En la Figura 2-5 se muestra un grafo dirigido con su correspondiente matriz de adyacencia.



M=

	1	2	3	4
1	0	1	0	0
2	0	0	1	0
3	1	0	0	1
4	0	0	1	0

Figura 2-5: Digrafo de la Figura 2-2 y matriz de adyacencia correspondiente.

2.2.2.2 Matrices de incidencia

Las matrices de incidencia son representaciones de las relaciones que existen entre objetos de dos clases distintas. A diferencia de las matrices de adyacencia, las matrices de incidencia pueden ser rectangulares. Esta situación se da cuando existen cantidades diferentes de elementos entre una clase y otra, lo cual puede verse en la matriz de la Figura 2-6. Un bigrafo, por ejemplo, puede ser descrito mediante una matriz de incidencia estructural. En ese caso, uno de los grupos de nodos se corresponderá con las filas de la

matriz y el otro con las columnas. Cada una de las celdas de la matriz de incidencia tiene un 1 o un 0 dependiendo de la existencia o no respectivamente de una relación entre el nodo fila y el nodo columna correspondientes. La Figura 2-6 también denota la representación gráfica y la matriz de incidencia correspondiente para un bigrafo.

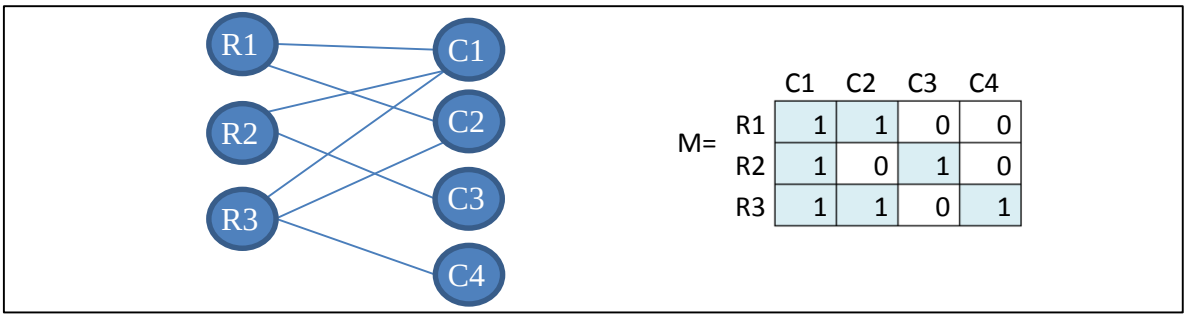


Figura 2-6: Bigrafo y matriz de incidencia correspondiente.

2.2.3 Detección de componentes fuertemente conexas en grafos dirigidos

Una de las tareas más comunes en la teoría de grafos es el reconocimiento de grupos fuertemente conectados de nodos. Dentro de un set de datos representado mediante un digrafo, pueden identificarse diferentes particiones con un alto grado de relación entre sus elementos. Ejemplos del uso de esta técnica pueden ser la búsqueda de comunidades temáticas o de usuarios dentro de redes sociales [66], o el estudio estructural de un gran set de datos [10]. En esta tesis se hace uso del algoritmo de detección de componentes fuertemente conexas en digrafos, definido en [20]. Dicho algoritmo constituye una parte esencial de los métodos de particionamiento desarrollados aquí.

En un primer apartado de esta sección, se detallan algunos conceptos necesarios para comprender el funcionamiento del algoritmo de detección de componentes fuertes. Luego, se explica la implementación del algoritmo mediante el uso de un algoritmo básico, el de *búsqueda en profundidad en digrafos*. Finalmente, se explica el algoritmo de

detección de componentes fuertes, con una estructura recursiva basada en la búsqueda en profundidad.

2.2.3.1 Conceptos adicionales sobre grafos

Para comprender el funcionamiento del algoritmo de detección de componentes fuertes, es necesario definir algunos conceptos más avanzados sobre grafos:

- **Subgrafo**: Grafo $G'=(V',E')$ compuesto por el subconjunto de nodos V' y aristas E' de un grafo $G=(V,E)$. Si todas las aristas de G' tienen sus extremos dentro de V', entonces se dice que G' es un *subgrafo inducido* de G. En la Figura 2-7.b) se muestra un subgrafo inducido del grafo de la Figura 2-7.a).
- **Caminata dirigida**: Dados dos nodos p y q de un digrafo G, una *caminata dirigida* (*directed walk*) de p a q se define como una secuencia finita y alternada de nodos y aristas de G: $p=v_0, e_1, v_1, e_2, v_2, ..., e_n, v_n=q$. Esta secuencia indica un recorrido de nodos a través de aristas adyacentes, que comienza en el nodo origen p y termina en el nodo destino q. Una caminata tiene una longitud dada por su número de aristas.
- **Camino dirigido**: Un *camino dirigido* (*directed path*) consiste en una caminata sin nodos repetidos en la secuencia. Un camino dirigido puede ser el existente entre los nodos 1 y 2 de la Figura 2-7.c).
- **Nodos conectados:** Se dice que dos nodos u y v de un grafo $G=(V,E)$ están conectados entre sí cuando existe un camino dentro de G que los une. Si G es un grafo dirigido y existe un camino de u a v pero no de v a u o viceversa, se dice que u y v están débilmente conectados, mientras que si existe un camino de ida y vuelta entre los dos, se dice que están fuertemente conectados. En la Figura 2-7.d) se muestra que los nodos 1 y 2 del grafo correspondiente están fuertemente conectados, y que los nodos 6

y 7 están débilmente conectados, ya que existe un camino desde 6 hasta 7 pero no desde 7 hacia 6.

- **Ciclo**: Un ciclo es un caso particular de un camino dirigido entre dos nodos *p* y *q*, en el cual el nodo origen es igual al nodo destino, es decir *p=q*. Además, un ciclo no debe tener aristas o nodos repetidos, salvo el nodo inicial y el final. En la Figura 2-7.e) se puede apreciar la existencia de dos ciclos, uno entre los nodos 1, 2 y 6, y otro entre los nodos 3 y 7.
- **Componente fuertemente conexa**: Una *componente fuertemente conexa* de un grafo *G*, o simplemente una *componente fuerte*, es un subgrafo *S* de *G*, tal que si tomamos cualquier par de nodos *u* y *v* de *S*, tenemos un *ciclo* que los contiene. Esto es equivalente a decir que para los nodos mencionados *u* y *v*, existe un camino que va desde *u* hacia *v* y otro camino que va desde *v* hacia *u*. En el grafo de la Figura 2-7.e) se distinguen dos componentes fuertemente conexas, las cuales coinciden con los ciclos encontrados anteriormente.
- **Árbol**: Un árbol (*tree*) es un grafo que no contiene ciclos. En la Figura 2-7.f) se puede ver un árbol formado por los nodos conectados por las aristas de trazo continuo.
- **Árbol de expansión**: Un árbol de expansión (*spanning tree*) sobre un grafo G es un árbol que contiene a todos los nodos conectados de G que pueden visitarse desde un nodo de origen *v*.
- **Bosque de expansión**: Un bosque de expansión (*spanning forest*) sobre un grafo G se obtiene incluyendo todos los nodos del grafo en un conjunto de árboles que los contenga. En la Figura 2-7.f) se puede observar un bosque de expansión sobre el grafo

de la Figura 2-7.a). Allí se tienen dos árboles, uno formado por el nodo aislado 4 y el otro compuesto por el resto de los nodos y las aristas de trazo continuo.

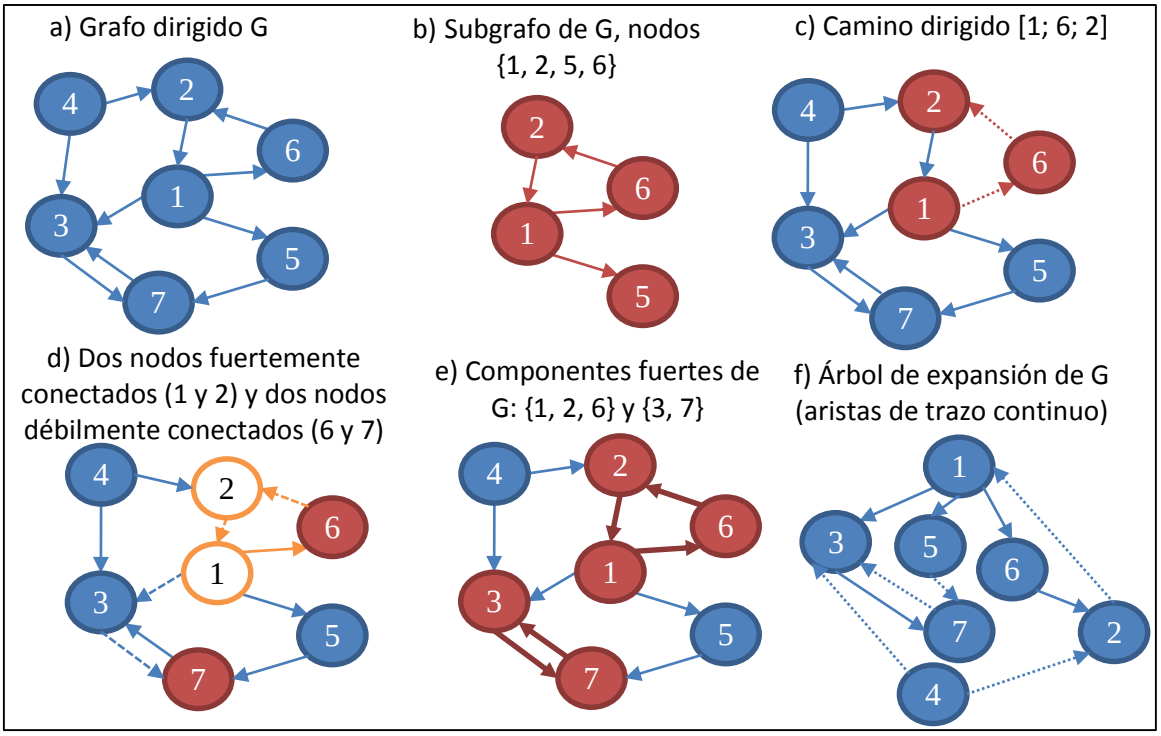


Figura 2-7: Representación visual de conceptos de grafos dirigidos.

Tomando como base estos conceptos, se puede describir los dos algoritmos que se presentan a continuación. El primero, Búsqueda en Profundidad (DFS), permite realizar una recorrida vertical completa de un grafo, y el segundo, Detección de Componentes Fuertes, usa la idea del DFS para detectar las componentes fuertemente conexas de cualquier grafo.

2.2.3.2 Búsqueda en Profundidad (DFS)

El algoritmo de Búsqueda en Profundidad ó *Depth-First Search* (DFS) [62] es una poderosa herramienta para implementar una recorrida sistemática por todos los nodos de un grafo, partiendo de un nodo inicial y visitando progresivamente sus hijos y todos sus descendientes. Como su nombre lo indica, lleva a cabo una recorrida vertical del grafo, ya que lo cubre primero en profundidad. Esto significa que antes de recorrer todos los nodos

adyacentes de un nodo en particular *v* -lo cual se denomina búsqueda en anchura o *Breadth-First Search* (BFS)-, agota sucesivamente las ramas de los hijos de *v*, sin empezar con otro nodo adyacente a *v* hasta tanto no se haya agotado la rama del anterior. En la Figura 2-8 se ilustran estos conceptos, distinguiendo la búsqueda en profundidad (a la izquierda) de la búsqueda en anchura (a la derecha).

Cuando se lleva a cabo una búsqueda en profundidad, se privilegia la verticalidad en el recorrido. De este modo, si visitamos primero el nodo 1 del grafo G de la Figura 2-8, el primer nodo adyacente que será encontrado por el algoritmo será el nodo 3 (Paso 1 Figura 2-8), siguiendo la arista correspondiente. Luego, el siguiente paso es buscar los descendientes del nodo 3 para encontrar nodos que no hayan sido visitados hasta ese momento. El nodo 7, alcanzado directamente desde el nodo 3, es el siguiente en ser incluido (Paso 2 Figura 2-8). En ese momento, se agota la rama actual en el algoritmo, ya que desde 7 se puede ir hacia 3 solamente. Después de terminar con el nodo 7, y dado que desde 3 tampoco puede visitarse ningún nodo nuevo, se inicia la rama siguiente, yendo desde el nodo 1 hacia el 5 (Paso 3 Figura 2-8). Desde 5, el único nodo alcanzable es el 7, el cual ya forma parte de una rama anterior, por lo que se termina otra rama. De esa misma forma se genera la rama correspondiente a los nodos 6 y 2 (Pasos 4 y 5 Figura 2-8).

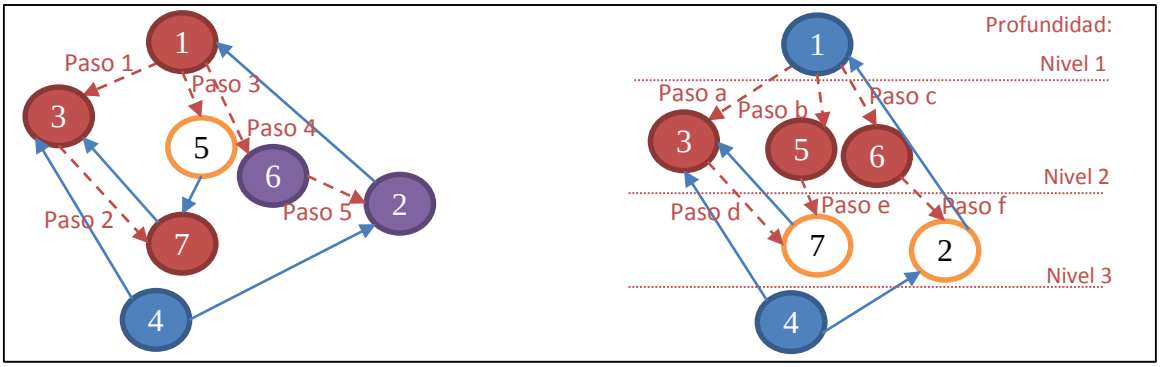


Figura 2-8: Búsqueda en profundidad (izquierda) y en anchura (derecha) sobre el grafo G de la Figura 2-7.

A diferencia del procedimiento anterior, la búsqueda en anchura recorre los nodos adyacentes a un nodo determinado antes de agotar alguna rama. Por ello decimos que BFS efectúa un recorrido horizontal en lugar de vertical sobre la estructura del grafo. Si observamos la Figura 2-8 a la derecha, podemos ver los pasos que sigue este algoritmo sobre el grafo correspondiente. En primera instancia, se recorren todos los vecinos del nodo 1. Estos son el nodo 3 (Paso *a* Figura 2-8), el 5 (Paso *b* Figura 2-8) y el 6 (Paso *c* Figura 2-8). Luego, se sigue con el nodo 7, adyacente a 3 (Paso *d* Figura 2-8). El nodo 5 tiene como descendiente al 7 (Paso *e* Figura 2-8), pero como ya fue visitado no se agrega. El 6 tiene a 2 como descendiente (Paso *f* Figura 2-8), y allí finaliza el procedimiento. En la Figura 2-8 se puede ver como el nodo 1 corresponde al primer nivel de profundidad; los nodos 3, 5 y 6 al segundo; y los nodos 7 y 2 al tercer nivel. El nodo 4 no es alcanzable desde ninguno de los anteriores, por lo tanto resulta ser un nodo aislado.

El Algoritmo 2-1 muestra el pseudocódigo que lleva a cabo la búsqueda en profundidad [62]. Este algoritmo se compone de dos procedimientos: Uno para verificar que todos los nodos hayan sido visitados, y otro para recorrer los adyacentes de cada nodo. El primero, DFST (DFS Transversal), inicializa todos los nodos, busca iterativamente aquellos que no hayan sido visitados hasta el momento y ejecuta sobre ellos el segundo procedimiento. Éste último, DFS, itera sobre los adyacentes de un nodo dado para ir etiquetando aquellos que no hayan sido visitados antes. Para llevar cuenta del orden de los nodos recorridos a cada momento, se utiliza un índice de visita o *Índice de Búsqueda en Profundidad* (DFI, *Depth First Index*).

```
1. Procedimiento DFST:
2.      Entrada: N,E
3.      Salida: Ef (Aristas finales de la DFS)
4.      i ← 1
5.      Ef ← ∅
6.      Para todo v ∈ N hacer: DFI(v) ← 0
7.          Mientras exista algún u ∈ N tal que DFI(u)=0 hacer
8.          DFS(i,u,N,E,Ef)
9.      fin-para
10. Procedimiento DFS:
11.     Entrada-Salida: (i,u,N,E,Ef)
12.     DFI(u) ← i
13.     i ← i+1
14.     Para todo nodo v adyacente a u hacer:
15.         Si DFI(v)=0
16.             Ef ← Ef ∪{(u,v)}
17.             DFS(i, v, N, E, Ef)
18.         fin-si
19.     fin-para
```

Algoritmo 2-1: Procedimientos DFST y DFS para búsqueda en profundidad.

Los datos requeridos como entrada para el Algoritmo 2-1 son los nodos y aristas del grafo *G* al cual se le aplicará la búsqueda: *N, E*. La salida proporcionada será el conjunto final ordenado de aristas, que permiten una recorrida completa por el grafo: E_f. Dicho conjunto de aristas nos da un bosque de expansión sobre el grafo *G*. En la Figura 2-9 se puede visualizar el valor de *DFI* de cada uno de los nodos del grafo de la Figura 2-7.a), y el conjunto E_f de aristas obtenido al ejecutar el Algoritmo 2-1 sobre el grafo mencionado.

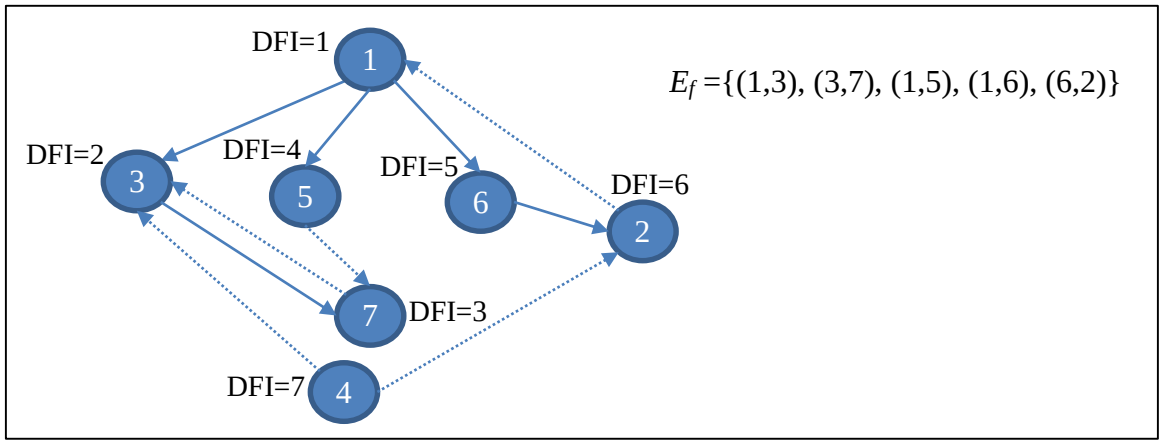


Figura 2-9: Valores de *DFI* y conjunto ordenado de aristas para una recorrida en profundidad del grafo de la Figura 2-7.a).

Es importante resaltar que la utilidad del primer procedimiento del Algoritmo 2-1, DFST, es identificar los diferentes árboles del grafo. Esto se hace evidente cuando encontramos árboles separados o nodos aislados. El algoritmo DFS solamente recorre los caminos que pueden seguirse desde el nodo origen *u* que se le envía como parámetro. Por lo tanto, si existe un nodo *v* tal que no exista ningún camino desde *u* hasta *v*, *v* no formará parte de su árbol. Por ello, para garantizar que se recorran todos los nodos del grafo, el procedimiento DFST recorre iterativamente los nodos que hasta el momento no fueron visitados.

Una característica clave del Algoritmo 2-1 es el índice de visita (DFI) de cada nodo. Cuando se inicia el procedimiento DFST, el DFI de todos los nodos se coloca en cero, indicando que todavía ninguno ha sido visitado. Luego, en las sucesivas iteraciones del segundo procedimiento (DFS), el DFI se actualiza para indicar en qué paso del algoritmo cada nodo es agregado. Esto genera un orden de precedencia para los nodos y aristas del grafo, que resulta muy útil tanto en la recorrida en profundidad del grafo como en otros algoritmos, por ejemplo el de detección de componentes fuertes.

La esencia del funcionamiento del Algoritmo 2-1 se encuentra en las líneas del procedimiento DFS que buscan nodos adyacentes al nodo origen (líneas 14 a 19). Para cada uno de los nodos *v* adyacentes al nodo origen *u*, si *v* no ha sido visitado se agrega al conjunto E_f la arista correspondiente *(u,v)*. Para verificar que el nodo *v* no fue visitado todavía, se chequea que su valor *DFI* esté en 0.

Otra cuestión importante del Algoritmo 2-1, que constituye la clave de la búsqueda en profundidad, es la forma recursiva del procedimiento DFS. Una vez que se encontró entre los adyacentes al nodo *u* un nodo *v* no visitado (línea 15) y se agregó la arista correspondiente en E_f (línea 16), el siguiente paso es ejecutar el mismo procedimiento DFS sobre *v* (línea 17), en forma recursiva, para agotar la rama correspondiente a este nodo. Con esta forma de funcionamiento se privilegia la recorrida en profundidad sobre la recorrida en anchura.

2.2.3.3 Detección de componentes fuertes mediante DFS

El algoritmo de recorrida en profundidad sirve como plataforma para llevar a cabo la detección de componentes fuertemente conexas en un grafo. Siguiendo los nodos en el orden que va elaborando la búsqueda DFS e incorporando la lógica de detección de componentes fuertes, se obtiene un algoritmo muy eficiente para esta tarea [20].

Para comprender mejor el funcionamiento del Algoritmo de Detección de Componentes Fuertemente Conexas ó *Strongly Connected Components Detection* (SCCD), es necesario identificar los conjuntos de aristas que pueden encontrarse en un árbol de expansión. Esto se debe a que los algoritmos basados en DFS se apoyan en el concepto de bosque de expansión, identificando árboles de expansión en el grafo correspondiente, para

cubrir todos los nodos del mismo de una manera eficiente. Las distintas clases de aristas que podemos encontrarnos son:

- **Aristas salientes (E_f):** Son las correspondientes al bosque de expansión.
- **Aristas de retroceso (B_1):** Unen un nodo cualquiera con alguno de sus ancestros.
- **Aristas de avance (B_2):** No están en E_f, pero unen un nodo cualquiera con alguno de sus descendientes.
- **Aristas de cruce (E_c):** Unen un par de nodos de dos árboles o ramas diferentes.

Teniendo en cuenta esta clasificación, el algoritmo SCCD funciona llevando a cabo una búsqueda de aristas de cruce o retroceso, mientras se va construyendo el bosque de expansión provisto por la base del algoritmo DFS. Dentro de la estructura recursiva de DFS, para lograr identificar la raíz de una componente fuerte y extraer todos los nodos de la misma, se utiliza un valor asociado a cada nodo recorrido *u*, identificado como *Q(u)*. Este valor es calculado en función del valor DFI de *u* y sus descendientes. Si se encontraran aristas de cruce o retroceso (del conjunto B_1 o E_c) desde algún descendiente *v* de *u* hacia algún ancestro de *u*, entonces el valor *Q(u)* se actualiza para identificar un ciclo dentro del grafo. La particularidad de cada raíz r_i de cada componente fuerte es que el valor $Q(r_i)$ coincide con el valor $DFI(r_i)$: $Q(r_i)=DFI(r_i)$. Para mayor detalle acerca de los teoremas y conceptos que sirven de soporte teórico a estas técnicas, consultar [62]. El valor *Q(u)* se obtiene con la siguiente fórmula recursiva:

$$Q(u) = mínimo(\{DFI(u)\} \cup \{Q(v) | v \text{ es un hijo de } u\} \cup \{DFI(v) \mid (u, v) \text{ está en } B_1 \text{ o } E_c, \text{ tal que la raíz de la componente fuerte que contiene a } v \text{ es un ancestro de } u\})$$

```
1.  Procedimiento SCCDT:
2.      Entrada: N,E
3.      Entrada-Salida: CF
4.      i←1
5.      j←1
6.      Pila←∅
7.      CF←∅
8.      Para todo v ∈ N hacer:
9.          DFI(v) ←0
10.         Apilado(v) ←falso
11.     fin-para
12.     Mientras exista algún nodo u tal que DFI(u)=0 hacer:
13.         DFSSCCD(i, u, DFI, N, E, Pila, j, CF)
14.     fin-mientras
15. Procedimiento DFSSCCD:
16.     Entrada: i, v, N, E, Pila
17.     Entrada-Salida: j, CF
18.     DFI(v)←i
19.     Q(v)←DFI(v)
20.     i←i+1
21.     Poner v en Pila
22.     Apilado(v)←verdadero
23.     Para cada nodo v' adyacente a v hacer:
24.         Si DFI(v')=0
25.             DFSSCCD(i, v', DFI, N, E, Pila, j, CF)
26.             Q(v) ←min(Q(v),Q(v'))
27.         Sino
28.             Si DFI(v')<DFI(v) y Apilado(v')
29.                 Q(v) ←min(Q(v),DFI(v')).
30.             fin-si
31.         fin-si
32.     fin-para
33.     Si Q(v)=DFI(v)
34.         Desapilar todos los nodos de Pila hasta  alcanzar v
35.         Almacenar todos los nodos desapilados en CF(j), incluyendo v
36.         Desapilar v
37.         Para cada nodo u ∈ CF(j) hacer
38.             Apilado(u) ←falso
39.         fin-para
40.         j←j+1.
41.     fin-si
```

Algoritmo 2-2: Procedimientos SCCDT y DFSSCCD para búsqueda transversal y detección de una componente fuerte a partir de un nodo raíz.

En el Algoritmo 2-2 se muestra el pseudocódigo de la rutina que utiliza el valor *Q(u)* para determinar las componentes fuertemente conexas de un grafo. Consiste en dos procedimientos basados en la búsqueda en profundidad. Uno de ellos, SCCDT (SCCD Transversal), inicializa y recorre el grafo en busca de nodos no visitados, para asegurar la recorrida completa del mismo. El otro, DFSSCCD (SCCD basado en DFS), identifica componentes fuertes partiendo desde un nodo inicial. La finalidad de utilizar dos procedimientos es justamente la necesidad de cubrir todo el grafo, teniendo en cuenta que pueden existir componentes aisladas (árboles de expansión separados).

El procedimiento SCCDT recibe como parámetros el conjunto de nodos *N* y el de aristas *E* del grafo *G* con el cual se desea trabajar. Comienza colocando los valores iniciales a *i* y *j*, los cuales representarán el nodo visitado y el número de la componente fuerte que se está armando en cada momento. También se inicializan en vacíos a los conjuntos *Pila*, que irá almacenando los vértices de la componente fuerte actual, y *CF*, que almacenará las distintas componentes fuertes generadas. La salida proporcionada por este procedimiento será el mismo conjunto *CF*. También, para cada nodo *v* se establece inicialmente en 0 el valor *DFI(v)*, y se indica que ningún nodo está incluido todavía en *Pila* (línea 10 del Algoritmo 2-2). El ciclo de instrucciones cuyo inicio se ubica en la línea 12 del Algoritmo 2-2, es el que inicia la ejecución de la detección de componentes fuertes propiamente dicha, y termina cuando todos los nodos del grafo han sido visitados. Esta característica proviene del algoritmo DFS, y cumple la función de verificar que todos los nodos hayan sido incluidos en la detección de componentes fuertes.

En el Algoritmo 2-2, cada vez que el procedimiento TSCCD ha encontrado un nodo *v* sin visitar, delega el control al procedimiento DFSSCCD, dándole como parámetros de entrada los valores actuales de *i, v, N, E, Pila, j* y *CF* (línea 13). El funcionamiento de este

último procedimiento se basa en el recorrido de los hijos del nodo *v* que viene como parámetro, para determinar el valor recursivo *Q(v)*. Dado un nodo *v'* adyacente a *v*, si *v'* no había sido visitado anteriormente (*DFI(v')=0*, línea 24), se hace una llamada recursiva de DFSSCCD sobre *v'*, luego de la cual se actualiza el valor de *Q(v)* con el mínimo entre *Q(v)* y *DFI(v')*. De lo contrario, si *v'* ya había sido visitado, se verifica si la arista *(v,v')* es una arista de retroceso o de cruce (línea 28). Si esta última condición se cumple, se actualiza el valor *Q(v)* con el mínimo entre *Q(v)* y *DFI(v')*. Finalmente y luego de recorrer todos los nodos adyacentes a *v*, se evalúa si se ha encontrado la raíz de una componente fuertemente conexa (línea 33). La condición que *v* debería cumplir para ser raíz de una componente fuerte es que su valor *Q(v)* sea igual al valor *DFI(v)*. Si eso se cumple, entonces se agregan los nodos de la componente fuerte hallada al conjunto *CF* y se reorganizan las demás estructuras de datos para continuar con la ejecución del algoritmo, sin seguir iterando sobre los nodos de la componente fuerte obtenida (líneas 34 a 40).

La forma recursiva del procedimiento DFSSCCD del Algoritmo 2-2 asegura que todos los caminos posibles que inicien y terminen en la raíz de una componente fuerte sean recorridos antes de que termine la ejecución del mismo procedimiento sobre dicha raíz. También es muy importante el trabajo con la estructura de datos *Pila*, que como su nombre lo indica es un arreglo de nodos con la forma *Last In First Out* (LIFO, último en entrar primero en salir). Esta estructura permite identificar exactamente los nodos correspondientes a una componente fuerte de forma eficiente, agregándolos a medida que aparecen, y teniendo en cuenta que el último a extraer de la lista será la raíz encontrada por el algoritmo.

2.2.4 Pareamientos maximales en bigrafos

En muchas ocasiones podemos encontrarnos problemas que, al ser modelados, quedan descriptos por un conjunto de objetos y relaciones entre los mismos. En el caso particular de los problemas en los cuales se puede identificar dos clases de objetos, y relaciones que van de una clase a otra únicamente, suele ser muy importante hallar combinaciones óptimas sobre las relaciones entre estos objetos para alcanzar algún objetivo en particular. Como se describió anteriormente en este capítulo, en la sección correspondiente, los *bigrafos* nos permiten representar adecuadamente estas situaciones.

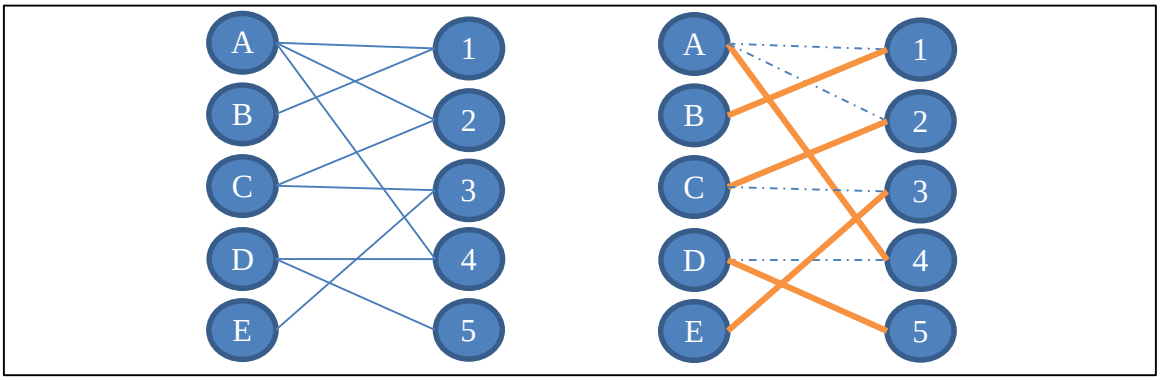


Figura 2-10: Bigrafo G y un pareamiento maximal sobre el mismo.

Un pareamiento P sobre un bigrafo G es un conjunto de aristas que no tienen nodos en común. En particular, un pareamiento maximal PM es un pareamiento que incluye la mayor cantidad posible de los nodos del bigrafo. En la Figura 2-10 vemos un ejemplo de un bigrafo a la izquierda, y un pareamiento maximal sobre el mismo a la derecha. Un bigrafo puede tener más de un pareamiento maximal, dependiendo de la manera de recorrer las aristas y los nodos. Por ejemplo, podemos obtener diferentes pareamientos si recorremos en distinto orden las aristas y los nodos del bigrafo, o si utilizamos diferentes puntos de partida.

2.2.4.1 Conceptos adicionales sobre bigrafos

Para ilustrar correctamente el algoritmo de obtención de pareamientos maximales, es necesario definir los siguientes conceptos:

- **Cardinalidad:** En un pareamiento, la cardinalidad del mismo está definida por la cantidad de aristas que este contiene.
- **Camino alternante:** Si tenemos un pareamiento *P*, un camino alternante *A* sobre *P* es aquel cuyas aristas alternan en *P*, es decir que si la primera arista de *A* pertenece a *P*, entonces todas las aristas en posiciones impares también pertenecerán a *P*.
- **Camino aumentado:** Es un camino alternante relativo a un pareamiento *P* cuyos extremos son nodos no apareados por *P*. Mediante la utilización de un camino aumentado respecto de un pareamiento *P*, se puede obtener otro pareamiento *P'* con una cardinalidad mayor que *P* en una unidad.

2.2.4.2 Algoritmo de pareamientos maximales en bigrafos

En particular, la rutina que se describe en [61] busca obtener un pareamiento con la mayor cardinalidad posible. Para lograr esto, utiliza el concepto de *camino aumentado*, incrementando en una unidad la cardinalidad del pareamiento en cada iteración. En el primer paso, simplemente genera un pareamiento inicial apareando cada nodo del primer grupo con su primer adyacente del segundo, si esto es posible. Luego, itera sobre este pareamiento buscando caminos aumentados. Las iteraciones terminan una vez que no es posible encontrar más caminos aumentados. En el Algoritmo 2-3 se visualiza en detalle este procedimiento.

```
1.  Datos de entrada: R, C, A
2.  Datos de salida: P_M
3.  P_M ←∅
4.  R_NA ←∅

5.  //Pareamiento inicial

6.  Para cada nodo r ∈ R hacer
7.      Aparear r con algún nodo adyacente no apareado c ∈ C
8.      Si existe tal nodo entonces P_M ← P_M ∪ {r,c}
9.      Sino R_NA ← R_NA ∪ {r}
10. fin-para

11. //Caminos aumentados

12. Repetir
13.     Buscar un camino aumentado CA desde algún r ∈ R_NA, visitando solamente
        nodos de C que no hayan sido visitados antes durante este paso
14.     Marcar todos los nodos alcanzados como visitados
15.     Si se ha encontrado un camino aumentado CA
16.         Aumentar PM con CA
17.         Quitar r de R_NA
18.     fin-si
19. hasta que no se encuentre ningún camino aumentado en un paso
```

Algoritmo 2-3: Búsqueda de un pareamiento maximal en un bigrafo.

En el Algoritmo 2-3, el conjunto P_M es el que corresponde a las aristas del pareamiento maximal, y R_{NA} es el conjunto de nodos no apareados pertenecientes al grupo *R*. El bloque de obtención de un pareamiento inicial recorre todos los nodos del grupo *R* para ver si pueden ser apareados con algún nodo adyacente del conjunto *C* que no haya sido apareado hasta ese momento. Si para algún nodo $r \in R$ no existe ningún nodo $c \in C$ con estas características, simplemente se agrega el nodo *r* al conjunto R_{NA} para indicar que no está apareado.

El mecanismo de búsqueda de caminos aumentados (línea 13 del Algoritmo 2-3) consiste simplemente en buscar algún nodo u_1 de cualquiera de los dos grupos que no haya sido apareado hasta el momento, y luego seguir alguna arista e_1 hacia algún otro nodo

apareado si existiera. Luego se siguen atravesando nodos u_i y aristas e_j, de forma tal que las aristas con índice *j* par pertenezcan al pareamiento y aquellas con *j* impar no pertenezcan a éste. La búsqueda concluye cuando se encuentra algún nodo no apareado u_j. Una vez encontrado el camino aumentado *CA*, se obtiene un nuevo pareamiento *P'* que tiene una cardinalidad mayor en una unidad que el pareamiento anterior *P*, es decir, $|P'|=|P|+1$. Para obtener *P'* se debe quitar de *P* todas las aristas que éste último tiene en común con el camino aumentado *CA*, y luego se agregan las aristas que solamente estaban presentes en *CA*.

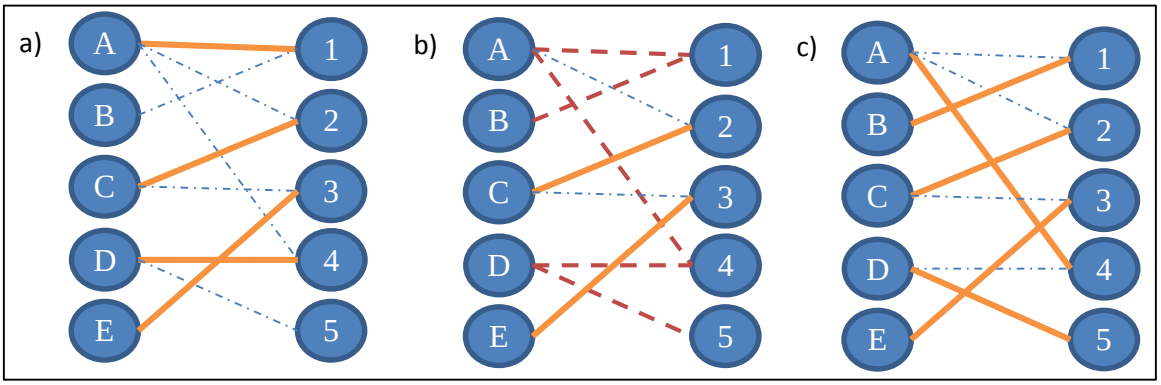


Figura 2-11: Etapas del Algoritmo 2-3 para la obtención de pareamientos maximales.

El ciclo que comprende las líneas 12 a 19 del Algoritmo 2-3 es el encargado del proceso iterativo de buscar caminos aumentados. En cada paso de esta estructura se recorren los nodos del conjunto *R* para encontrar algún camino alternante que pueda aumentar la cardinalidad del pareamiento. Una vez que en alguno de los pasos no se logra hallar ningún camino aumentado termina el ciclo, lo que indica que se ha llegado a un pareamiento maximal. En la Figura 2-11 se puede ver un ejemplo de la búsqueda de caminos aumentados y la obtención final de un pareamiento maximal. La sección a) de dicha figura muestra el pareamiento inicial, la sección b) un camino aumentado sobre el pareamiento inicial, y la c) el pareamiento maximal obtenido.

2.3 Método Directo

En el campo de la Ingeniería Química, en particular en el Diseño de Instrumentación de plantas químicas, resulta muy útil contar con la mayor cantidad posible de información acerca de los equipos y componentes que conforman las mismas [67]. Esto permite lograr, por ejemplo, mejoras importantes en el control y la seguridad de los procesos involucrados. El concepto de Observabilidad de una planta química justamente hace referencia a la máxima cantidad de información que puede obtenerse de ella en estado estacionario, para una configuración específica de sensores. Una manera de llegar a este valor teórico es la utilización de las distintas ecuaciones correspondientes al modelo de la planta, para determinar los valores de sus variables. Dichas variables son las que componen el total de información disponible de la planta. Los sistemas de ecuaciones asociados a los modelos de plantas químicas pueden hacerse muy grandes en la medida en que se agregan sensores, equipos o corrientes al modelo, o simplemente se incluyen cuestiones más específicas en el modelado de las plantas [17]. La matriz de incidencia asociada a estos sistemas de ecuaciones suele ser rala, y puede ser explotada para lograr una reorganización estructural de las ecuaciones y variables de los mismos. El propósito de dicha reorganización es la determinación de la máxima cantidad posible de variables del proceso en cuestión, para así acercarnos al valor teórico de observabilidad correspondiente a la configuración actual de la planta.

Se puede lograr una reorganización adecuada de la matriz de incidencia para este propósito mediante el tratamiento de dicha matriz manipulando los grafos asociados a ella [42]. En una primera instancia, se representa la estructura de la matriz mediante un bigrafo para llevar a cabo en éste el algoritmo de pareamientos maximales, descrito en la sección

correspondiente. Luego, sobre uno de los bloques de la matriz permutada de acuerdo a la nueva estructura, se construye una representación por grafo dirigido y se lleva a cabo el algoritmo de detección de componentes fuertemente conexas, también explicado en una sección anterior.

2.3.1 Forma triangular inferior en bloques

Si deseamos determinar qué cantidad de variables del modelo resultarán observables mediante la utilización del sistema de ecuaciones correspondiente, podemos hacer uso de la matriz de incidencia asociada. Para ello, se puede permutar dicha matriz para convertirla a una *forma triangular inferior en bloques (FTiB)*. Un esquema gráfico de esta estructura puede verse en la Figura 2-12.

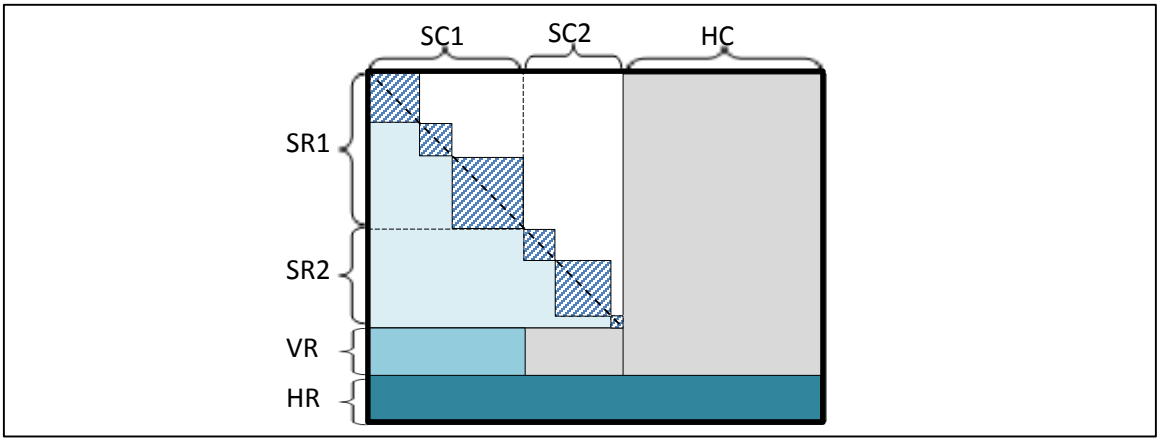


Figura 2-12: Distintos conjuntos de ecuaciones y variables para la Forma Triangular en Bloques.

En el gráfico de la Figura 2-12 se puede ver un conjunto cuadrado en la parte superior izquierda, formado por dos subconjuntos de ecuaciones y variables: SR1-SC1 y SR2-SC2. Ambos conjuntos contienen las asignaciones de ecuaciones a variables, que permitirán determinar el valor de estas variables mediante la resolución de sistemas de ecuaciones más pequeños que el sistema completo original. Estos últimos sistemas son los representados como bloques cuadrados rayados dentro de los grupos SR1-SC1 y SR2-SC2.

Además de estos bloques, el conjunto VR-SC1 contiene las ecuaciones redundantes y sus variables asociadas, el conjunto HR contiene las ecuaciones con variables indeterminables, y el conjunto HC comprende el grupo de variables indeterminables.

Ya que la matriz de incidencia representa las variables que intervienen en cada ecuación, queda claro que cada bloque de asignación de los conjuntos *SR1-SC1* y *SR2-SC2* representará un subsistema de ecuaciones con igual cantidad de ecuaciones que variables. Cada subsistema puede resolverse de manera independiente, siempre y cuando se hayan resuelto los correspondientes a los bloques anteriores, por encima del mismo en la *FTiB*. Los dos bloques rectangulares que se observan en la parte inferior de la estructura se corresponden con los grupos de ecuaciones redundantes o con variables indeterminables. Las ecuaciones redundantes son aquellas que solamente contienen variables determinables por la resolución de los bloques cuadrados de *SR1-SC1* y las ecuaciones con variables indeterminables son las que contienen más de una variable indeterminable. La diferencia entre los conjuntos *SR1-SC1* y *SR2-SC2* es la no existencia de ecuaciones redundantes para el grupo *SR2-SC2*.

2.3.2 Detalle del Funcionamiento del MD

El objetivo de la utilización del MD en una matriz de incidencia es la permutación de la misma a la forma *FTiB*. Para lograr esto, se hace uso de diversas representaciones y algoritmos sobre la matriz de incidencia original. Los pasos del MD son los indicados en el Algoritmo 2-4, en el procedimiento principal.

1. ***Procedimiento DescGruesa()***
2. *Datos de entrada:* ***Bigrafo BG(M)***
3. *Datos de salida:* ***SR1, SR2, VR, HR, SC1, SC2, HC***
4. *Obtener un pareamiento maximal P_m de BG*
5. *Clasificar las filas en SR1, SR2, VR o HR según P_m*
6. *Clasificar las columnas en SC1, SC2 o HC según P_m*

Para cada fila especial que exista, clasificarla como:
- ***VR*** *si no está conectada con ninguna columna de* ***HC***
- ***SR1*** *si está conectada a una única columna de* ***HC*** *y a ninguna de* ***SC2*** *(se forma un bloque de 1x1)*
- ***SR2*** *si está conectada a una única columna de* ***HC*** *y al menos a una columna de* ***SC2*** *(se forma un bloque de 1x1)*
- ***HR*** *si está conectada a más de una columna de* ***HC***

fin DescGruesa()

Procedimiento DescFina()

Datos de entrada: ***Matriz de incidencia M y conjuntos (SR1,SC1), (SR2, SC2)***

Datos de salida: ***Componentes fuertes*** $M_{11}, M_{12}, ..., M_{1p}$ ***y*** $M_{21}, M_{22}, ..., M_{2q}$

Asociar el digrafo $G(M_1)=(V,E)$ *a la matriz* M_1 *correspondiente al conjunto* ***(SR1, SC1)***

Descomponer $G(M_1)$ *en sus componentes fuertes* $M_{11}, M_{12}, ..., M_{1p}$. *Cada* M_{1i} *corresponde a un bloque de la diagonal.*

Asociar el digrafo $G(M_2)=(V,E)$ *a la matriz* M_2 *correspondiente al conjunto* ***(SR2, SC2)***

Descomponer $G(M_2)$ *en sus componentes fuertes* $M_{21}, M_{22}, ..., M_{2q}$. *Cada* M_{2j} *corresponde a un bloque de la diagonal.*

fin DescFina()

Procedimiento Reasignar()

Datos de entrada: $\mathbf{M, M_{1i}, P_m}$***, VR, SR1, SC1***

Datos de salida: $\mathbf{M_{1i}}$***, ExitoR***

Buscar una arista ***(r,c)*** *donde* $\mathbf{r \in VR}$, $\mathbf{c}$ *a las columnas en* $\mathbf{M_{1i}}$, $\mathbf{(k,c) \in P_m}$ *y* ***k*** *no fue reasignada por* ***r*** *anteriormente*

Si *tal arista* ***(r,c)*** *existe, es posible la reasignación:*

Eliminar de $\mathbf{P_m}$ *la arista* ***(k,c)***

Agregar ***(r,c)*** *a* $\mathbf{P_m}$ *y eliminar* ***k*** *de* ***SR1*** *y* $\mathbf{M_{1i}}$

Agregar ***r*** *a* ***SR1*** *y* $\mathbf{M_{1i}}$ *y eliminar* ***r*** *de* ***VR***

Agregar ***k*** *a* ***VR***

ExitoR = verdadero

Sino ExitoR = falso

fin Reasignar()

Procedimiento ReducirBG()

Datos de entrada: $\mathbf{BG, M_{11}..M_{ij}, S}$

Datos de salida: ***BG, S***

Eliminar del bigrafo todas las filas y columnas correspondientes a las componentes fuertes anteriores a $\mathbf{M_{ij}}$ *(*$\mathbf{M_{kl}}$ *con* $\mathbf{k \leq i}$ *y* $\mathbf{l < j}$*) e incorporarlas a la solución*

Seleccionar una fila de $\mathbf{M_{ij}}$ *como fila especial (que no haya sido seleccionada anteriormente) y eliminarla de* ***BG(M)***

41. *Si todas las filas de **Mij** fueron seleccionadas anteriormente como fila especial, seleccionar dos filas especiales entre las filas de **Mij**, sin importar si habían sido elegidas antes*
42. ***(SR1, SR2, VR, HR, SC1, SC2, HC) = DescGruesa(BG)***
43. ***(M_{11}..M_{1p} , M_{21}.. M_{2q}) = DescFina(M, (SR1,SC1), (SR2, SC2))***
44. ***fin ReducirBG()***
45.
46. ***Procedimiento MetodoDirecto()***
47. *Datos de entrada:* ***M (Matriz de incidencia), Rest (Conjunto de Restricciones)***
48. *Datos de salida:* ***M reordenada a FTiB***
49. ***Etapa 0: Inicialización***
50. ***BG(M)=(R, C, E)*** *(Se construye el bigrafo BG de M)*
51. ***S = Ø*** *(Inicializar conjunto solución)*
52.
53. ***Etapa 1: Descomposición Gruesa.***
54. ***(SR1, SR2, VR, HR, SC1, SC2, HC) = DescGruesa(BG)***
55.
56. ***Etapa 2: Descomposición Fina.***
57. ***(M_{11}..M_{1p} , M_{21}.. M_{2q}) = DescFina(M, (SR1,SC1), (SR2, SC2))***
58.
59. ***Etapa 3: Test de Admisión.***
60. ***SeguirTestDeAdmision = verdadero***
61. ***Mientras SeguirTestDeAdmision == verdadero***
62. ***SeguirTestDeAdmision = falso***
63. ***Para*** *cada componente fuerte* ***M_{1i}, mientras SeguirTestDeAdmision == falso***
64. *Chequear que* ***M_{1i}*** *no esté en el conjunto de restricciones* ***Rest***
65. ***Si M_{1i}*** *está prohibido*
66. ***SeguirTestDeAdmision = verdadero***
67. ***(M_{1i}, ExitoR) = Reasignar(M, M_{1i}, P_m, VR, SR1, SC1)***
68. ***Si ExitoR == verdadero***
69. ***(M_{11}, M_{12}, ..., M_{1p} y M_{21}, M_{22}, ..., M_{2q}) = DescFina(M, (SR1,SC1), (SR2, SC2))***
70. ***Sino***
71. ***(BG, S) = ReducirBG(BG, M_{11}.. M_{1i}, S)***
72. ***fin Si*** *(ExitoR == verdadero)*
73. ***fin Si*** *(M_{1i} está prohibido)*
74. ***fin Para*** *(cada componente fuerte M_{1i})*
75. ***Para*** *cada componente fuerte* ***M_{2j}, mientras SeguirTestDeAdmision == falso***
76. *Chequear que* ***M_{2j}*** *no esté en el conjunto de restricciones* ***Rest***
77. *Si* ***M_{2j}*** *está prohibido*
78. ***SeguirTestDeAdmision = verdadero***
79. ***(BG, S) = ReducirBG(BG, M_{11}.. M_{2j}, S)***
80. ***fin Si*** *(M_{2j} está prohibido)*
81. ***fin Para*** *(cada componente fuerte M_{2j})*
82. ***fin Mientras*** *(SeguirTestDeAdmision == verdadero)*

```
83.
84.        Etapa 4: Reordenamiento
85.          Reordenar M como sigue: [M11, M12... M1p, M21, M22... M2q, (VR,SC1),
             (HR,HC)]
86. fin MetodoDirecto()
```

Algoritmo 2-4: Pseudocódigo del Método Directo.

Al principio del algoritmo se detallan cuatro procedimientos que integran distintas fases del funcionamiento del MD, y serán explicados en esta sección. Por su parte, el procedimiento principal, ***MetodoDirecto()***, acepta como entrada la matriz *M* que se desea permutar y el conjunto de restricciones *Rest* para la formación de bloques de asignación, y devuelve como salida la misma matriz *M* permutada a la forma *FTiB*. Cada bloque prohibido del conjunto *Rest* es simplemente un conjunto de filas y columnas de la matriz *M* que no pueden conformar un bloque de asignación en la matriz permutada. En la primera etapa del algoritmo (Etapa 0) se construye el bigrafo $BG(M)=(R, C, E)$ que representará a la matriz *M* a lo largo del proceso.

En la Etapa 1, se efectúa la descomposición gruesa de la matriz, utilizando el bigrafo obtenido en la etapa anterior. Sobre aquel, se lleva a cabo el algoritmo de Pareamientos Maximales descrito anteriormente, e implementado por el procedimiento ***DescGruesa()***. Mediante la aplicación de esa rutina sobre el bigrafo, se hace posible identificar los bloques correspondientes a los grandes conjuntos de variables que formarán parte de la permutación final de la matriz. Estos conjuntos son: (*SR1, SC1*); (*SR2, SC2*); (*VR, SC1*) y (*HR, HC*). El pareamiento maximal obtenido sirve de guía para reconocer las variables y ecuaciones asignadas. Cada par obtenido en el pareamiento maximal será una componente de la diagonal principal de la matriz permutada. De esta manera, se logra una característica muy importante en la matriz permutada, como lo es la diagonal principal llena en el bloque cuadrado compuesto por los conjuntos *SR1-SC1* y *SR2-SC2*.

La Etapa 2 del algoritmo consiste en la descomposición fina de los conjuntos *SR1-SC1* y *SR2-SC2* por separado. Allí se identifican los bloques de asignación candidatos para formar parte de la solución final. Esto se logra mediante la representación por digrafos de los conjuntos mencionados, y la aplicación del *algoritmo de detección de componentes fuertemente conexas* sobre aquellos. Cada componente fuerte obtenida se reflejará en un bloque de asignación en la solución, siempre y cuando el bloque formado no incluya ninguna restricción del conjunto *Rest*. En esta etapa se hace uso del procedimiento ***DescFina()***, el cual identifica los bloques de asignación.

Una vez obtenidos los bloques de asignación candidatos, en la Etapa 3 se verifica que estos sean admisibles. Para ello, se chequea que no contengan ninguna de las restricciones del conjunto *Rest*. Si se encuentra un bloque prohibido entre los obtenidos en el conjunto *SR1-SC1*, se intenta reasignar dicho bloque (procedimiento ***Reasignar()***). Si la reasignación no fue posible o se trata de un bloque prohibido del conjunto *SR2-SC2*, se lleva a cabo la reducción del bigrafo (procedimiento ***ReducirBG()***) y se trata de encontrar otro particionamiento factible de la matriz. Esta tarea de verificación se encuentra dentro de un conjunto de ciclos que concluyen cuando todos los bloques obtenidos resultan admisibles.

La reasignación sobre los bloques del conjunto *SR1-SC1* consiste en intercambiar alguna de las filas del bloque por una fila redundante del bloque *VR*. Si esto es posible, se efectúa el intercambio y se ejecuta nuevamente la descomposición fina. Al trabajar sobre los bloques de asignación del conjunto *SR2-SC2*, no es posible efectuar una reasignación ya que la característica especial de estos bloques es la no existencia de ecuaciones redundantes con las que se pueda hacer el reemplazo correspondiente para la reasignación. Si se logra

efectuar correctamente la reasignación de un bloque, el algoritmo vuelve a llevar a cabo el armado de las componentes fuertes, mediante el procedimiento ***DescFina()***.

Si la reasignación no es factible porque no se encuentra una fila redundante para reemplazar alguna fila del bloque, o el bloque actual pertenece al conjunto *SR2-SC2*, entonces se lleva a cabo la reducción del bigrafo, con el procedimiento ***ReducirBG()***. Esta reducción consiste en la incorporación a la solución final de los bloques anteriores al bloque prohibido, y la eliminación de las filas y columnas del bigrafo que corresponden a dichos bloques. Luego de eliminar estos bloques del bigrafo, se selecciona una o más filas especiales. Estas no serán tenidas en cuenta para la obtención de un nuevo pareamiento maximal, con el propósito de lograr un nuevo particionamiento diferente al anterior, ya que este último produjo el bloque prohibido imposible de reasignar. Al pasar por esta fase, se debe volver a armar la descomposición gruesa con el bigrafo reducido.

Finalmente, se obtendrá la matriz permutada a la forma *FTiB*, con el ordenamiento correspondiente de los bloques: Primero se sitúan los bloques correspondientes a los conjuntos *SR1-SC1* y *SR2-SC2* que contendrán los bloques de asignación que constituyen los subsistemas de ecuaciones generados. Luego, se coloca el conjunto de ecuaciones redundantes *(VR, SC1)* y al final el conjunto de ecuaciones indeterminables con sus correspondientes variables, *(HR, HC)*. De esta manera, el algoritmo devuelve una permutación de la matriz de entrada *M* a la forma *FTiB*.

2.4 Método Directo Extendido

La aplicación del MD en la matriz de incidencia de un sistema de ecuaciones nos brinda un reordenamiento de sus filas y columnas muy apropiado para una resolución eficiente por bloques. Aun así, este algoritmo no implementa ninguna clasificación de las

ecuaciones de acuerdo con su nivel de complejidad. El MDE, para complementar las funciones de su predecesor, efectúa un análisis en este sentido para lograr un mejor reordenamiento. Dicho análisis consiste en la determinación de la no linealidad asociada a variables, ecuaciones y términos de ecuaciones, y su utilización para la formación de los bloques de asignación. Esta no linealidad se define como un valor de ponderación que cada ecuación, variable y término tendrá adjunto, de acuerdo a la complejidad que aporta al sistema en general.

Términos	**Lineal**	**Bilineal**	**No Lineal**				
			1 variable	2 variables	3 variables	4 variables	5 o + variables
Pesos	0	1	2,2	2,4	2,6	2,8	3

Tabla 2-1: Ejemplo de valor de ponderación para cada tipo de término según su GNL.

El algoritmo del MDE lleva a cabo el cálculo del Grado de No Linealidad (GNL) para cada ecuación, variable y término del siguiente modo:

- *Para cada variable*, el GNL se calcula como el promedio del GNL de los términos en que aparece la misma, a lo largo de todas las ecuaciones del sistema.
- *Para cada ecuación*, el GNL será el promedio de los GNL de sus términos.
- *Para cada término*, el GNL se calcula en base a una tabla, que indica un número diferente de ponderación para cada tipo de término. Un ejemplo de este tipo de ponderaciones puede verse en la Tabla 2-1.

En la práctica, los valores de ponderación del grado de no linealidad pueden definirse de acuerdo a cualquier criterio apropiado para el problema [45]. Es decir, que los valores expresados en la Tabla 2-1 pueden variar para adaptarse a las necesidades de cada problema en particular, siguiendo los criterios experimentales que correspondan. Un ejemplo del cálculo del GNL de variables y ecuaciones puede verse en la Figura 2-13. En dicha figura, a la izquierda se muestra un sistema de ecuaciones de ejemplo, y al centro y a

la derecha los GNL de sus ecuaciones y variables. Para el cálculo de esos GNL se utilizan los valores de ponderación de la Tabla 2-1.

e_1: $x_1 * x_3 - x_2 = 0$ **e_2:** $x_2 - \ln(x_2/x_3) = 0$ **e_3:** $x_1^2 * x_3 + x_2 - x_1/5 = 0$	**GNL ecuaciones:** $GNL(e_1)=0,5$ $GNL(e_2)=1,2$ $GNL(e_3)=0,8$	**GNL variables:** $GNL(x_1)=1,13$ $GNL(x_2)=0,6$ $GNL(x_3)=1,93$

Figura 2-13: Sistema de ecuaciones con los GNL de sus ecuaciones y variables.

Para implementar el GNL en el algoritmo del MD, el MDE simplemente utiliza una variante del mismo en la que se modifica la búsqueda de un pareamiento maximal. La modificación sobre el algoritmo de pareamientos maximales se puede describir del siguiente modo, teniendo en cuenta el pseudocódigo del Algoritmo 2-3, y la modificación indicada en el Algoritmo 2-5:

- Se debe disponer de las estructuras de datos correspondientes a los GNL de variables y ecuaciones.
- Con las estructuras de datos mencionadas (GNL(R) y GNL(C)) es posible reordenar las listas de nodos adyacentes en orden ascendente.
- A lo largo de su iteración, el algoritmo modificado irá tomando nodos adyacentes para la formación de los distintos conjuntos de la estructura *FTiB*. Ya que estos nodos estarán ordenados ascendentemente por su GNL, entonces será más factible que los primeros conjuntos en formarse (*SR1-SC1* y *SR2-SC2*) incluyan ecuaciones y variables más lineales, es decir, con menor GNL.

1. Datos de entrada: ***R, C, A, GNL(R), GNL(C)***
1a. Reordenar las listas de filas y columnas adyacentes en orden ascendente según GNL de los nodos adyacentes
...

Algoritmo 2-5: Fragmento del Algoritmo 2-3 modificado para incluir el GNL.

El Algoritmo 2-5 muestra la modificación que se incluye en el Algoritmo 2-3, para obtención de un pareamiento maximal incluyendo el GNL en la formación de los bloques de asignación. Solamente se agregan los arreglos correspondientes a los GNL de ecuaciones y variables en la entrada de datos del algoritmo, y el paso 1.a de ordenación de los arreglos de nodos adyacentes. La recorrida de nodos adyacentes no se ve alterada, ya que es llevada a cabo de la misma manera, de forma secuencial sobre los arreglos ordenados por GNL.

Una característica fundamental del MDE es la librería que permite obtener el GNL de ecuaciones y variables. Dispone de un analizador sintáctico que reconoce los términos y variables de las ecuaciones generadas con un formato preestablecido, y de acuerdo a un criterio determinado, le asigna los valores correspondientes de GNL a cada entidad. Para mayor detalle sobre el funcionamiento de esta librería y del MDE en su totalidad, consultar [44].

2.5 Conclusiones

En este capítulo fue descripto en detalle el marco teórico concerniente a la teoría de grafos y los algoritmos que sirven de base para comprender los capítulos siguientes de esta tesis. Los algoritmos de detección de componentes fuertemente conexas y de identificación de pareamientos maximales han sido ampliamente utilizados y extendidos para diferentes ámbitos de la ciencia. El primero de ellos, la detección de componentes fuertemente conexas, puede ser, por ejemplo, el puntapié inicial para determinar grupos de objetos interrelacionados de acuerdo a diferentes criterios. En [19], por ejemplo, se utiliza este algoritmo para determinar comunidades temáticas en un directorio de internet, lo cual será explicado en la sección correspondiente de esta tesis. Por su parte, el algoritmo de

identificación de pareamientos maximales también constituye una herramienta básica y clave para los trabajos de diversas áreas. Un ejemplo de esto último es el trabajo llevado a cabo en [68], en el cual se implementa un nuevo algoritmo basado en la versión clásica de los pareamientos maximales, y la variante allí es que se buscan relaciones de un nodo de un grupo con más de un nodo del otro grupo.

Capítulo 3:

Particionamiento Estructural de Modelos Ingenieriles

3.1 Introducción

El modelado de un gran número de procesos naturales e industriales puede derivar en grandes sistemas de ecuaciones, cuya resolución nos permitirá la evaluación, control y corrección de ciertos parámetros para lograr objetivos concretos. Teniendo en cuenta los métodos de reordenamiento estructural descriptos en el capítulo anterior, se mostrará aquí su aplicación a casos académicos y algunos modelos provenientes del Diseño de Instrumentación de Plantas Químicas.

También en este capítulo se comenta en detalle la implementación de una mejora en el algoritmo del MDE, que en principio podría incrementar la cantidad de bloques lineales obtenidos a partir de un modelo matemático determinado. Ya que la modificación al algoritmo no incluye librerías externas al software del MDE y solamente se utilizan procedimientos nativos del lenguaje en que aquel fue desarrollado, no existe un impacto considerable que pudiera comprometer la eficiencia de este método. En una sección del presente capítulo se muestran resultados correspondientes a ejemplos para los cuales resulta muy provechosa la implementación de esta mejora.

Además del Diseño de Instrumentación, en este capítulo se explora la posibilidad de aplicación de los métodos de particionamiento a distintos casos de Simulación y Optimización de modelos químicos. Por otro lado, se ilustra una cuestión fundamental que impide la aplicación del particionamiento en cierta clase de problemas de optimización con restricciones de desigualdad.

3.2 Mejora implementada al MDE: Ordenamiento preliminar sobre ecuaciones

La gran ventaja de la aplicación del MDE con respecto a su antecesor el MD, es la determinación del GNL de variables y ecuaciones del sistema, que permite la obtención de

bloques más sencillos para resolver. En la implementación del MDE, el GNL calculado para cada ecuación y variable es empleado dentro de la sección del algoritmo que corresponde a la elaboración de un pareamiento maximal sobre el bigrafo que representa al sistema de ecuaciones (ver Capítulo 2). Para sacar provecho del valor de GNL, al momento de seleccionar un nodo para aparear con otro, los adyacentes al primero se ordenan en función de su GNL ascendentemente, para seleccionar de todos los adyacentes posibles aquel que tenga menor GNL.

Como parte de los trabajos elaborados en el marco de esta tesis, en [50] fue propuesta una mejora al algoritmo del MDE, para tratar de disminuir un poco más la complejidad o no linealidad de los bloques conformados como fruto del particionamiento estructural de dicho método. Esta mejora consiste en la incorporación de un ordenamiento previo sobre las ecuaciones del sistema de acuerdo a su GNL de forma ascendente. El objetivo de esta modificación al algoritmo es la orientación de la selección de ecuaciones para elaborar los pareamientos maximales, con el fin de que las ecuaciones lineales sean tenidas en cuenta prioritariamente para su inclusión en los primeros bloques de asignación. De esa forma, podría aumentarse la probabilidad de conformar bloques lineales.

Para incluir el ordenamiento de ecuaciones descripto en el párrafo anterior dentro de la estructura del MDE, debe modificarse el algoritmo de obtención de pareamientos maximales, que forma parte de la primera sección del método. Al igual que en el Algoritmo 2-5, en el Algoritmo 3-1 se incluye un ordenamiento adicional sobre la base de la rutina de obtención de un pareamiento maximal. En este caso, el ordenamiento realizado se lleva a cabo sobre el vector que representa a las ecuaciones del sistema, para disponerlas ascendentemente de acuerdo a su GNL. El paso del Algoritmo 3-1 que refleja esta tarea es el **1.b**.

1. Datos de entrada: ***R, C, A, GNL(R), GNL(C)***
1a. Reordenar las listas de filas y columnas adyacentes en orden ascendente según GNL de los nodos adyacentes
1b. Reordenar la lista de ecuaciones en orden ascendente según GNL de las mismas
...

Algoritmo 3-1: Mejora implementada al MDE dentro del algoritmo de pareamientos maximales.

Con el paso **1.a** nos aseguramos de que cada vez que se aparean dos nodos, el nodo destino tenga el menor GNL posible. Este principio ya se encuentra incluido en el funcionamiento básico del MDE. El paso **1.b** refleja la nueva funcionalidad incorporada al MDE, y con el ordenamiento llevado a cabo se puede asegurar una mejor clasificación, ya que al momento de armar un pareamiento inicial las primeras ecuaciones tomadas serán aquellas con el menor GNL. El pareamiento inicial constituye la base del Algoritmo 2-3, ya que es sobre dicha estructura que se itera para encontrar caminos aumentados. Por ello, con la mejora del Algoritmo 3-1, el pareamiento inicial proporcionado podría brindar una mejor configuración para la obtención de los bloques de asignación en los métodos de particionamiento.

3.2.1 El lenguaje C: Alta eficiencia en ordenamientos

Para la implementación de esta mejora, el ordenamiento de elementos en un vector es un tema importante a considerar. Es sabido que esta es una de las tareas más costosas en términos de consumo de recursos en un sistema de cómputo. Tanto el MD como el MDE y el MDE mejorado fueron implementados en el lenguaje de programación C. Para evitar un impacto de gran magnitud en la eficiencia general y el desempeño del algoritmo, se utilizó una función nativa del lenguaje C para la programación del ordenamiento de las ecuaciones. Dicha función lleva a cabo el ordenamiento de los datos mediante la utilización del algoritmo *quicksort* [69], el cual ha demostrado tener muy alta eficiencia. Los

programas compilados en este lenguaje también son muy eficientes en comparación con otras plataformas.

3.3 Ejemplos de aplicación de los métodos de particionamiento

Para comprender mejor la utilidad de los métodos de particionamiento descriptos anteriormente y poder apreciar las ventajas de su utilización, se muestran en esta sección algunos modelos sobre los que se aplican dichos métodos.

3.3.1 Ejemplo académico: Sistema de ecuaciones genérico

En primera instancia, se muestra un sistema de ecuaciones genérico sobre el cual se realiza el análisis correspondiente con los métodos de particionamiento. Dicho sistema tiene 12 ecuaciones y 10 variables, y está expresado en la Figura 3-1.

$$
S_1: \begin{cases}
e_1: x_2 + 3x_7 - 2x_{10} - 10 = 0 \\
e_2: \log_{10}(x_2^3 - 17) + x_2^2 - 10 = 0 \\
e_3: -3x_2 - 3x_8 + 15 = 0 \\
e_4: 3x_2 - x_7 - x_{10} = 0 \\
e_5: x_6 + x_{10}^2 + e^{x_{10}-4} - 14 = 0 \\
e_6: 7x_1 - x_5 + \dfrac{x_2^3}{3} + x_2^2 - 23 = 0 \\
e_7: 2x_2 - 3x_7 + 9 = 0 \\
e_8: x_6 + x_7 - 2 = 0 \\
e_9: 3x_2 - x_6 + x_7 + 2x_{10} - 25 = 0 \\
e_{10}: 2x_2 + x_6 - 2x_7 - x_{10} + 11 = 0 \\
e_{11}: -8x_1 + 15x_6 - 20x_7 - x_{10} + x_2^4 * x_8 + x_2^2 - 14 = 0 \\
e_{12}: x_2^3 - e^{(x_2-3)} - x_3 - 4x_4 - x_5 + x_6 - x_7 + x_9 - x_{10} + 4 = 0
\end{cases}
$$

Figura 3-1: Sistema de ecuaciones genérico.

El sistema de la Figura 3-1 fue generado haciendo uso de una plataforma de generación de casos aleatorios de sistemas de ecuaciones, la cual es descripta en el Capítulo 4 de esta tesis. Como fue comentado en el capítulo anterior, los métodos de particionamiento analizados trabajan sobre los aspectos estructurales de los sistemas de

ecuaciones. Esos aspectos son los plasmados en la matriz de incidencia del sistema, la cual representa la existencia o no de cada una de las variables en cada una de las ecuaciones. Para el sistema que se analiza en esta sección, la matriz de incidencia correspondiente es la que se muestra en la Tabla 3-1.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
e_1	0	1	0	0	0	0	1	0	0	1
e_2	0	1	0	0	0	0	0	0	0	0
e_3	0	1	0	0	0	0	0	1	0	0
e_4	0	1	0	0	0	0	1	0	0	1
e_5	0	0	0	0	0	1	0	0	0	1
e_6	1	1	0	0	1	0	0	0	0	0
e_7	0	1	0	0	0	0	1	0	0	0
e_8	0	0	0	0	0	1	1	0	0	0
e_9	0	1	0	0	0	1	1	0	0	1
e_{10}	0	1	0	0	0	1	1	0	0	1
e_{11}	1	1	0	0	0	1	1	1	0	1
e_{12}	0	1	1	1	1	1	1	0	1	1

Tabla 3-1: Matriz de Incidencia para el sistema de ecuaciones de la Figura 3-1.

Otro aspecto importante de los sistemas de ecuaciones que surgen en el área de Ingeniería de Sistemas de Proceso es la existencia de *bloques prohibidos* o *restricciones* sobre la formación de posibles bloques de asignación (Ver Capítulo 2). El sistema presentado en la Figura 3-1 tiene asociadas 17 restricciones para la formación de bloques, las cuales están expresadas en la Tabla 3-2. Como se ve allí, cada una de las restricciones consiste en un conjunto de variables y ecuaciones que no pueden estar presentes simultáneamente en un bloque conformado. Las columnas del sector izquierdo de la tabla reflejan las combinaciones de una variable y una ecuación que no pueden componer un

bloque de asignación, y las del sector derecho indican similares restricciones de dos variables y dos ecuaciones.

Los algoritmos que llevan a cabo los particionamientos verifican, luego de haber elaborado un conjunto posible de bloques de asignación, que cada uno de estos bloques no contenga restricciones en el grupo de ecuaciones y variables que lo componen. Esta sección de los algoritmos se conoce bajo el nombre de *Test de Admisión*, y los procedimientos encargados de manejar los bloques conformados que no pasan este test son los denominados *Reasignación* y *Reducción del bigrafo*.

Tamaño 1

Variables	Ecuaciones
10	8
4	7
9	12
9	5
8	12
7	6
3	8
4	12
5	12
7	9
10	12
9	2
9	6
3	7

Tamaño 2

Variables	Ecuaciones
9	4
6	5
2	3
4	11
4	4
1	11

Tabla 3-2: Restricciones del sistema de la Figura 3-1 agrupadas por tamaño.

Al observar las características de este sistema de ecuaciones, podemos mencionar que 7 de las ecuaciones que lo componen son lineales ($e_1, e_3, e_4, e_7, e_8, e_9$ y e_{10}), mientras que 5 contienen entre sus términos no linealidades (por ejemplo, el primer término de la ecuación e_2 que contiene un logaritmo y una potencia, o el quinto término de la ecuación e_{11} que contiene un producto entre variables y una potencia), lo cual las hace ecuaciones con un grado de no linealidad mayor que cero, o simplemente ecuaciones no lineales.

Si intentáramos resolver este sistema mediante un software diseñado para ello, este tendría que llevar a cabo la resolución encarando al grupo de ecuaciones como un sistema no lineal completo, ya que la sola existencia de un término no lineal en cualquiera de las ecuaciones provoca la imposibilidad de resolverlo mediante métodos de resolución de sistemas lineales. Esto se vuelve importante si tenemos en cuenta que la resolución de sistemas lineales suele ser más precisa y menos demandante que la resolución de sistemas no lineales. Para sistemas no lineales, los métodos clásicos de resolución involucran aproximaciones que requieren un conocimiento pormenorizado de las funciones que integran al sistema y sus derivadas de primer y segundo orden, como por ejemplo el método de Newton-Raphson [60]. En la práctica, es sabido que en muchas ocasiones no se cuenta con tal información, sino simplemente, por ejemplo, con valores de entrada asociados a sus correspondientes valores de salida, teniendo las funciones respectivas expresiones muy difíciles de manejar en términos matemáticos. Por ello, el tratamiento independiente de las ecuaciones agrupadas en bloques según sus dependencias con las variables del sistema podría ser muy provechoso, ya que permitiría disponer de la precisión y eficiencia que ofrecen los métodos de resolución de sistemas lineales sobre los bloques de esa misma condición, entre otras ventajas. También, los particionamientos estudiados aquí permiten tener un panorama preliminar sobre las singularidades estructurales que pudieran presentarse en un sistema *(Ver Capítulo 1, sección 1.1)*.

Los métodos de particionamiento que se trabajan aquí, en su conjunto, utilizan dos etapas para su funcionamiento, en las cuales manipulan distintas representaciones de la matriz de incidencia. En una primera instancia, la representación de la matriz de incidencia que se emplea consiste en un bigrafo, sobre el cual se busca un pareamiento maximal entre los nodos correspondientes al grupo de ecuaciones y los del grupo de variables. Luego, la

representación del sistema es llevada a cabo mediante un grafo dirigido, que refleja las asignaciones de variables a ecuaciones obtenidas en el paso anterior (para mayor detalle, consultar capítulo 2).

En las secciones siguientes, se muestra el funcionamiento y los resultados de cada uno de los métodos de particionamiento para el sistema de ecuaciones detallado en la Figura 3-1.

3.3.1.1 Método Directo

El primero de los métodos trabajados aquí, el Método Directo (MD) [41], lleva a cabo tres tareas fundamentales de forma iterativa según las necesidades de cada caso:

- La descomposición gruesa del bigrafo asociado al sistema, en la cual se obtiene el conjunto de variables determinables y las ecuaciones asignadas para tal fin.
- La descomposición fina del grafo dirigido que refleja las asignaciones obtenidas en el paso anterior.
- El análisis de factibilidad de los bloques de asignación obtenidos.

En la Figura 3-2 se muestra el bigrafo asociado al sistema de ecuaciones de la Figura 3-1. Luego en la Figura 3-3, se muestra el conjunto de nodos y aristas del pareamiento maximal sobre dicho bigrafo, correspondiente a la tarea de descomposición gruesa del MD, luego de llevar a cabo el test de admisión sobre los bloques de asignación generados.

El bloque conformado por las variables y ecuaciones de los grupos SR1-SC1 y SR2-SC2 constituye el conjunto sobre el que se aplicará posteriormente el algoritmo de detección de componentes fuertes. Cada nodo del grafo dirigido sobre el que se aplica este último algoritmo representa la asignación de una variable a una ecuación, y las aristas de

dicho grafo se corresponden con las aristas del bigrafo para los nodos dentro del grupo de ecuaciones y variables asignadas, que no forman parte del pareamiento maximal. En la Figura 3-4 se puede apreciar a la izquierda el digrafo y las componentes fuertes correspondientes al bloque SR1-SC1, y a la derecha el digrafo y las componentes fuertes del bloque SR2-SC2. Como puede observarse de la Tabla 3-2 y de la Figura 3-3 y la Figura 3-4, los bloques finales obtenidos son admisibles, ya que no contienen ninguna de las restricciones dentro de su estructura.

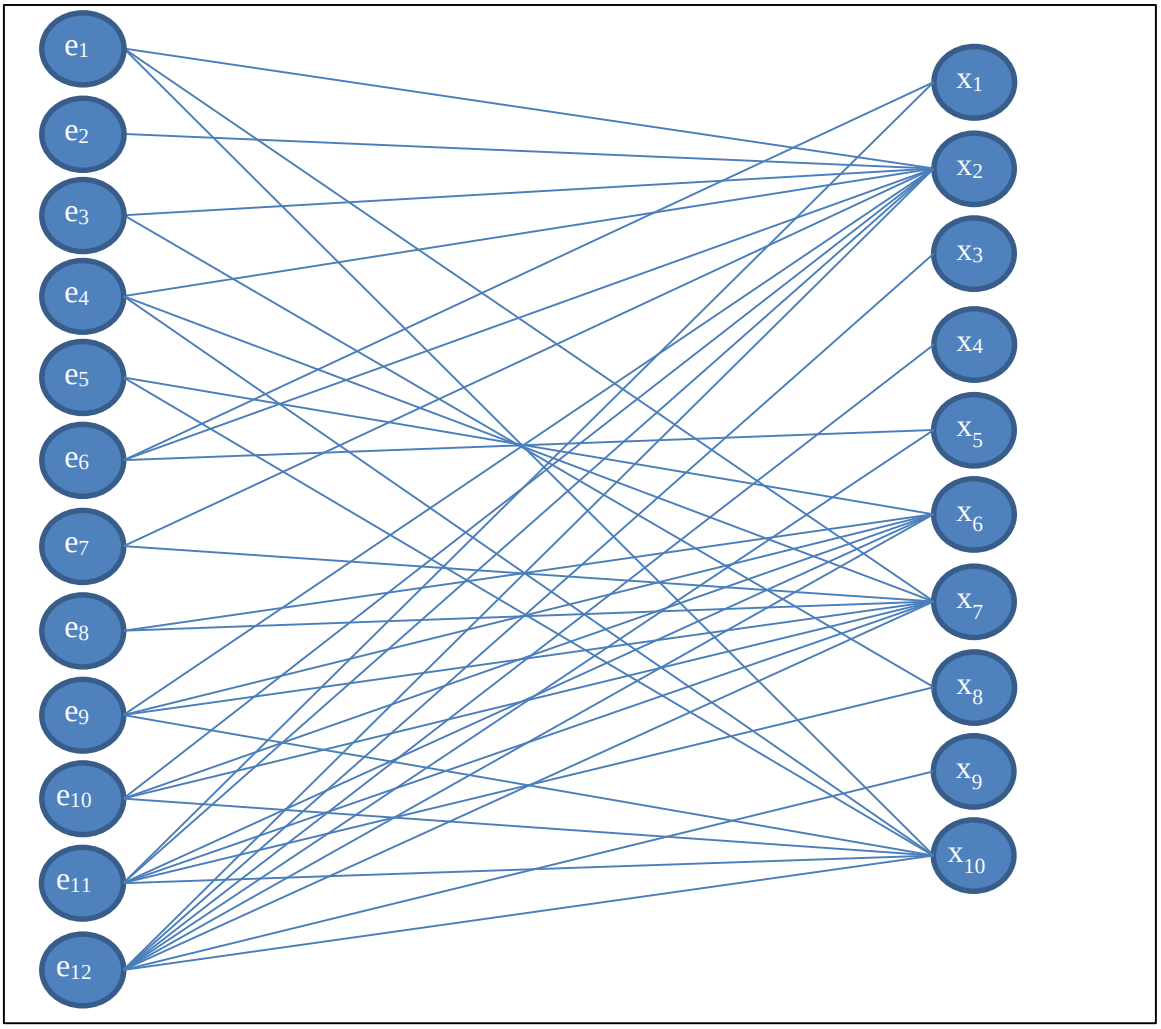


Figura 3-2: Bigrafo asociado al sistema de ecuaciones de la Figura 3-1.

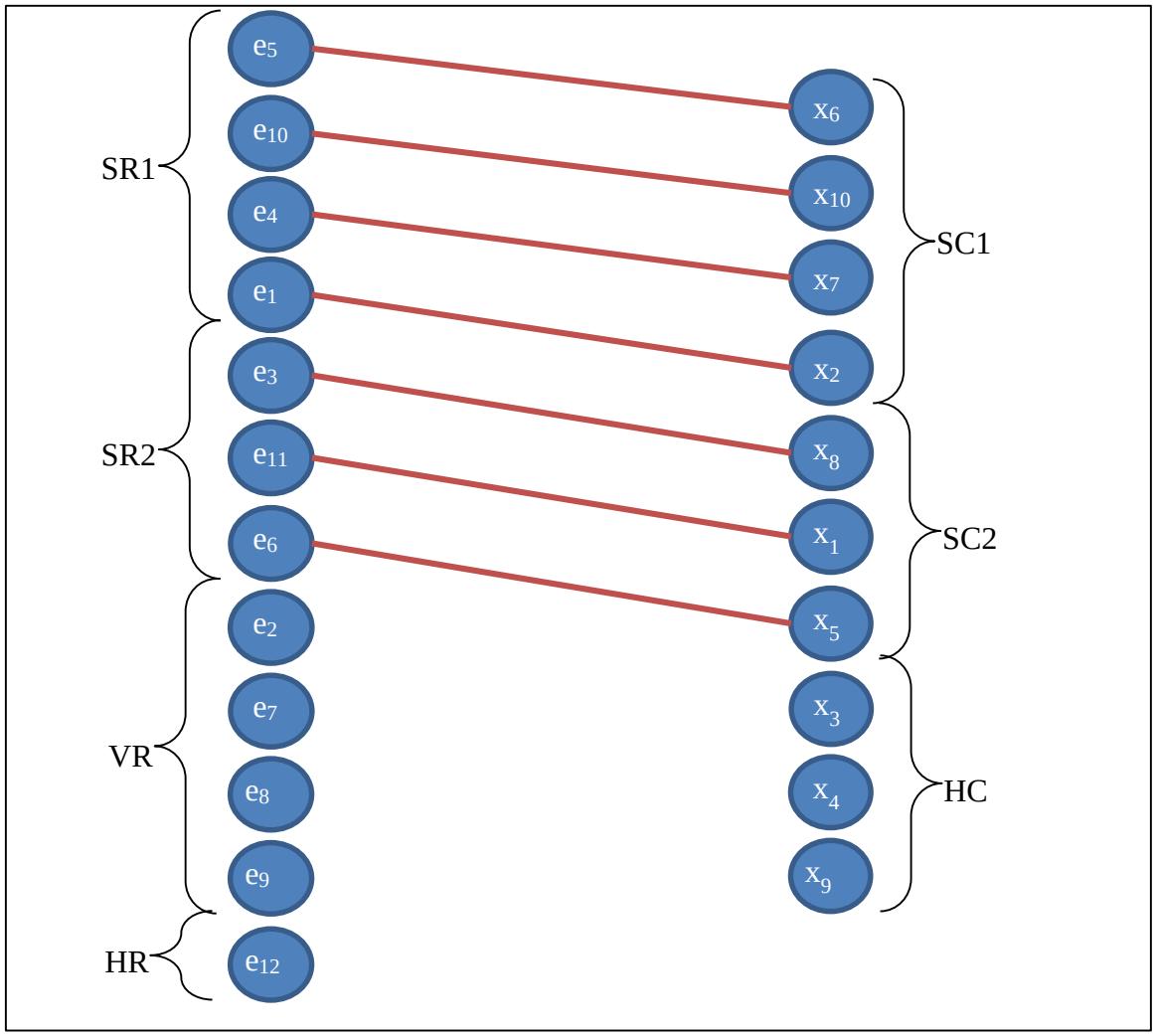


Figura 3-3: Vértices, Aristas y grupos del Pareamiento Maximal del sistema de la Figura 3-1.

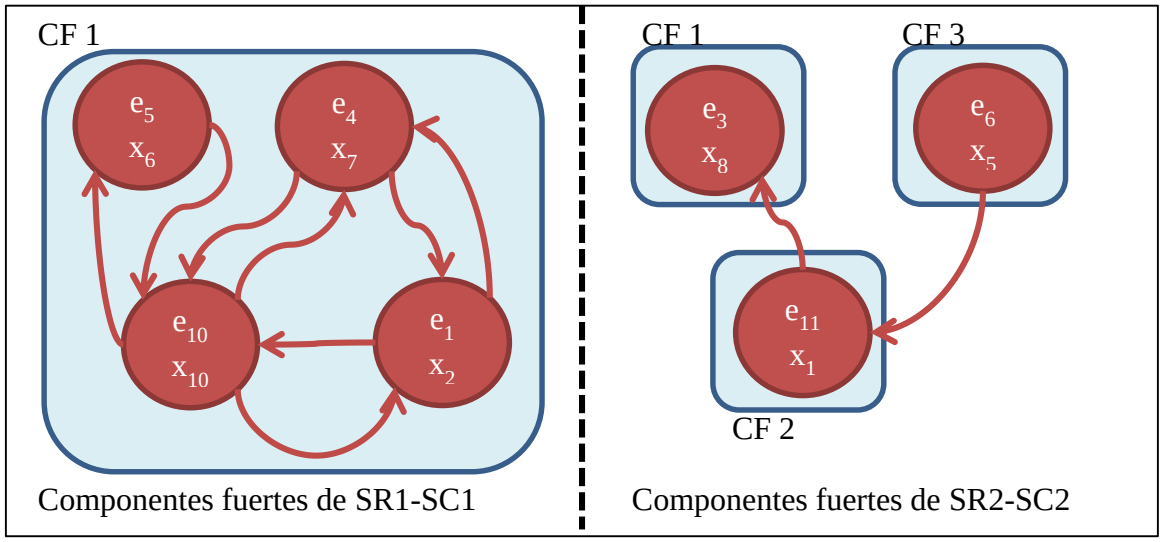


Figura 3-4: Digrafo y componentes fuertes para los grupos SR1-SC1 y SR2-SC2 del sistema de la Figura 3-1.

El resultado de la aplicación del MD sobre el ejemplo de la Figura 3-1 puede verse en detalle en la Tabla 3-3, la cual muestra la matriz de incidencia permutada a la FTiB propuesta por el MD. En ella se enumeran los bloques obtenidos, y se especifica si los mismos son lineales o no lineales, en función de las ecuaciones que contienen. Luego, en la Tabla 3-4, se resume las cantidades de bloques de acuerdo al tipo de ecuaciones que contienen según sean lineales y no lineales, especificando las cantidades de variables cuyos valores pueden determinarse por la resolución de uno u otro tipo de bloques.

	x_6	x_{10}	x_7	x_2	x_8	x_1	x_5	x_3	x_4	x_9	Tipo de bloque	Grupo
e_5	1	1	0	0	0	0	0	0	0	0	NO LINEAL	SR1
e_{10}	1	1	1	1	0	0	0	0	0	0		
e_4	0	1	1	1	0	0	0	0	0	0		
e_1	0	1	1	1	0	0	0	0	0	0		
e_3	0	0	0	1	1	0	0	0	0	0	LINEAL	SR2
e_{11}	1	1	1	1	1	1	0	0	0	0	NO LINEAL	
e_6	0	0	0	1	0	1	1	0	0	0	NO LINEAL	
e_2	0	0	0	1	0	0	0	0	0	0		VR
e_7	0	0	1	1	0	0	0	0	0	0		
e_8	1	0	1	0	0	0	0	0	0	0		
e_9	1	1	1	1	0	0	0	0	0	0		
e_{12}	1	1	1	1	0	0	1	1	1	1		HR
Grupo	SC1				SC2			HC				

Tabla 3-3: Matriz de incidencia reordenada de acuerdo al MD, para el sistema de la Figura 3-1.

Concepto	**Cantidad**
Bloques Lineales	**1**
Variables observables por bloques lineales	**1**
Bloques No Lineales	**3**
Variables observables por bloques no lineales	**6**
Total Bloques	**4**
Total Variables Observables	**7**

Tabla 3-4: Resumen del resultado del MD sobre el sistema de la Figura 3-1.

Si observamos en la Tabla 3-3 el detalle de los bloques de asignación obtenidos por el MD, podremos ver que solamente uno de ellos es lineal. Dicho bloque está formado por la ecuación e_3, y permite obtener el valor de la variable x_8, una vez obtenidos los valores

de x_6, x_{10}, x_7 y x_2. Tanto el primero de los bloques como el tercero y el cuarto (asignados para las variables x_1 y x_5 respectivamente) contienen términos no lineales en sus ecuaciones. En las próximas secciones, se verá que se pueden obtener otras permutaciones de la matriz de incidencia que brindan mejores descomposiciones, y permiten generar más bloques de asignación lineales.

3.3.1.2 Método Directo Extendido

En el Capítulo 2 fue explicado el MDE, cuya principal contribución científica es el tratamiento diferenciado de las ecuaciones del sistema bajo análisis según su grado de no linealidad. Luego de haber detallado el desempeño del MD en la sección anterior para el sistema de la Figura 3-1, aquí se explicarán las diferencias de la implementación del MDE sobre el mismo sistema. El grado de no linealidad de ecuaciones y variables del sistema bajo análisis está expresado en la Tabla 3-5, y constituye una de las diferencias fundamentales entre el MD por un lado y el MDE y el MDE Mejorado por el otro, ya que esa información solamente es tenida en cuenta por los dos últimos métodos. Luego, en la Tabla 3-6 se muestra la matriz de incidencia del sistema de ecuaciones reordenada conforme a los bloques obtenidos por la aplicación del MDE.

Si observamos en detalle la Tabla 3-6, podemos encontrar la conformación de bloques distintos a los de la Tabla 3-3. El MDE genera 4 bloques no lineales de una sola ecuación ($e_2 - x_2; e_5 - x_6; e_{11} - x_1; e_6 - x_5$), un bloque de dos ecuaciones lineales ($e_4 - x_{10}$ *y* $e_1 - x_7$) y un bloque lineal de una ecuación ($e_3 - x_8$). Este resultado está resumido también en la Tabla 3-7. Esta última nos muestra que el MDE logró encontrar 2 bloques lineales, con los cuales se puede determinar el valor de 3 de las variables del sistema, mientras que mediante el uso del MD solamente se podía encontrar el valor de 1 variable

por la resolución de un bloque lineal. Este resultado es una clara evidencia de la ventaja que brinda el tratamiento del GNL de las ecuaciones de un sistema, a la hora de efectuar un particionamiento estructural sobre el mismo, favoreciendo al MDE sobre el MD.

Ecuación	GNL
E1	0.00
E2	2.20
E3	0.00
E4	0.00
E5	1.47
E6	1.10
E7	0.00
E8	0.00
E9	0.00
E10	0.00
E11	0.77
E12	0.49

Variable	GNL
X1	0.00
X2	1.27
X3	0.00
X4	0.00
X5	0.00
X6	0.00
X7	0.00
X8	1.20
X9	0.00
X10	0.55

Tabla 3-5: GNL de ecuaciones y variables para el sistema de la Figura 3-1

	x_2	x_{10}	x_7	x_6	x_8	x_1	x_5	x_3	x_4	x_9	Tipo de bloque	Grupo
e_2	1	0	0	0	0	0	0	0	0	0	NO LINEAL	
e_4	1	1	1	0	0	0	0	0	0	0	LINEAL	SR1
e_1	1	1	1	0	0	0	0	0	0	0		
e_5	0	1	0	1	0	0	0	0	0	0	NO LINEAL	
e_3	1	0	0	0	1	0	0	0	0	0	LINEAL	
e_{11}	1	1	1	1	1	1	0	0	0	0	NO LINEAL	SR2
e_6	1	0	0	0	0	1	1	0	0	0	NO LINEAL	
e_7	1	0	1	0	0	0	0	0	0	0		
e_8	0	0	1	1	0	0	0	0	0	0		VR
e_9	1	1	1	1	0	0	0	0	0	0		
e_{10}	1	1	1	1	0	0	0	0	0	0		
e_{12}	1	1	1	1	0	0	1	1	1	1		HR
Grupo	SC1				SC2			HC				

Tabla 3-6: Matriz de incidencia reordenada de acuerdo al MDE, para el sistema de la Figura 3-1.

Concepto	Cantidad
Bloques Lineales	**2**
Variables observables por bloques lineales	**3**
Bloques No Lineales	**4**
Variables observables por bloques no lineales	**4**
Total Bloques	**6**
Total Variables Observables	**7**

Tabla 3-7: Resumen del resultado del MDE sobre el sistema de la Figura 3-1.

3.3.1.3 MDE Mejorado

Al tratar el sistema de ecuaciones de la Figura 3-1 con el MDE mejorado (ver sección 3.2), se obtienen los resultados descriptos en la Tabla 3-8 y en la Tabla 3-9. Analizando los bloques obtenidos mediante la utilización del MDE mejorado, detallados en la Tabla 3-8 y la Tabla 3-9, podemos decir que la cantidad de variables cuyo valor puede determinarse mediante la resolución de sistemas lineales supera a la misma cantidad, en el caso del MDE básico, cuyos resultados fueron mostrados en la sección anterior. Si bien los tres métodos de particionamiento logran determinar bloques de asignación que permiten llegar al mismo valor de observabilidad (7 variables), la gran diferencia entre sus desempeños radica en la cantidad y tamaño de bloques lineales que genera cada uno. Para visualizar más claramente la ventaja evidenciada del MDE sobre el MD y del MDE mejorado sobre el MDE, en la Tabla 3-10 se detallan los resultados en conjunto para todos los métodos sobre el sistema de la Figura 3-1. La primera columna detalla los conceptos que se analizan para el resultado de cada técnica, y las otras tres columnas indican los valores asociados a los conceptos de la primera para cada uno de los métodos.

Los valores que se observan en la Tabla 3-10 demuestran que, para este caso, existe una clara ventaja en la aplicación del MDE mejorado sobre el sistema de ecuaciones bajo estudio. Esto se aprecia al observar que la cantidad de bloques lineales es 3 para el MDE mejorado, mientras que es de 1 para el MD y de 2 para el MDE. También, y más

importante aún, existe un bloque de tres ecuaciones lineales que forma parte del particionamiento que generó el MDE mejorado, y ninguno de los otros dos métodos pudo hallar un bloque con esas características. Adicionalmente, se puede ver en la Tabla 3-10 que la mayor cantidad de variables observables por bloques lineales es 5, y este valor también corresponde al MDE mejorado.

	x_2	x_{10}	x_7	x_6	x_8	x_1	x_5	x_3	x_4	x_9	Tipo de bloque	Grupo
e_7	1	0	1	0	0	0	0	0	0	0	LINEAL	SR1
e_4	1	1	1	0	0	0	0	0	0	0		
e_1	1	1	1	0	0	0	0	0	0	0		
e_{10}	1	1	1	1	0	0	0	0	0	0	LINEAL	
e_3	1	0	0	0	1	0	0	0	0	0	LINEAL	SR2
e_{11}	1	1	1	1	1	1	0	0	0	0	NO LINEAL	
e_6	1	0	0	0	0	1	1	0	0	0	NO LINEAL	
e_2	1	0	0	0	0	0	0	0	0	0		VR
e_5	0	1	0	1	0	0	0	0	0	0		
e_8	0	0	1	1	0	0	0	0	0	0		
e_9	1	1	1	1	0	0	0	0	0	0		
e_{12}	1	1	1	1	0	0	1	1	1	1		HR
Grupo	SC1				SC2			HC				

Tabla 3-8: Matriz de incidencia reordenada de acuerdo al MDE mejorado, para el sistema de la Figura 3-1.

Concepto	**Cantidad**
Bloques Lineales	**3**
Variables observables por bloques lineales	**5**
Bloques No Lineales	**2**
Variables observables por bloques no lineales	**2**
Total Bloques	**5**
Total Variables Observables	**7**

Tabla 3-9: Resumen del resultado del MDE mejorado sobre el sistema de la Figura 3-1.

Como conclusión respecto de la implementación de estas tres técnicas sobre el sistema de ecuaciones analizado, se aprecia que la implementación del MDE mejorado podría aumentar la cantidad de bloques lineales en ciertas clases de sistemas de ecuaciones.

Esto es debido al ordenamiento previo que este método lleva a cabo sobre las ecuaciones de acuerdo a su GNL, lo cual permite incluir primeramente las ecuaciones lineales del sistema a la hora de elaborar las asociaciones que llevarán a la construcción de los bloques de asignación. Para determinar si realmente esta mejora es significativa de acuerdo a diferentes criterios, se deben realizar análisis exhaustivos sobre una gran cantidad de sistemas de ecuaciones de diversas características.

Concepto	MD	MDE	MDE mejorado
Bloques Lineales	**1**	**2**	**3**
Variables observables por bloques lineales	**1**	**3**	**5**
Bloques No Lineales	**3**	**4**	**2**
Variables observables por bloques no lineales	**6**	**4**	**2**
Total Bloques	**4**	**6**	**5**
Total Variables Observables	**7**	**7**	**7**

Tabla 3-10: Resumen conjunto para el MD, el MDE y el MDE mejorado.

3.3.2 Particionamiento para Simulación y Optimización

El contenido de esta sección está orientado al tratamiento de distintos problemas de simulación y optimización en Ingeniería Química. Los resultados se analizan desde el punto de vista de la eficiencia de resolución de los modelos matemáticos que surgen de estos problemas, y de la cantidad de variables que no sería necesario inicializar para resolverlos. Respecto de la cantidad de variables a inicializar, mediante los particionamientos se hace posible determinar el valor de las variables incluidas en los bloques de asignación (variables observables), y por ello no es necesario un valor inicial para aquellas variables. De todo esto se entiende que el foco de estudio no corresponde al grado de no linealidad de los bloques conformados luego de efectuar los particionamientos de los modelos, sino al impacto que puede tener la implementación de estos métodos en la resolución completa de

los modelos. El procedimiento de resolución de estos modelos incorporando los particionamientos estructurales puede verse reflejado en el Algoritmo 3-2.

Entrada: Formulación del problema de Simulación/Optimización
1. Identificar parámetros, variables, ecuaciones y configuraciones del problema.
2. Realizar el particionamiento estructural sobre las ecuaciones y variables.
3. Para cada bloque de ecuaciones obtenido, en forma secuencial y respetando el orden del particionamiento:
3a. Formular un subproblema con las ecuaciones del bloque únicamente.
3b. Resolver el subsistema para dichas ecuaciones.
3c. Asignar los valores obtenidos a las variables del bloque.
3d. Incluir estos valores para todos los subsistemas siguientes.
Salida: Reporte de los valores de las variables obtenidos por la resolución de los distintos bloques.

Algoritmo 3-2: Procedimiento para resolución de problemas de simulación/optimización con los métodos de particionamiento.

El procedimiento detallado en el Algoritmo 3-2 será utilizado en las próximas secciones para la resolución de distintos problemas de Simulación/Optimización. La automatización de dicho algoritmo se llevó a cabo con la implementación de algunas librerías de software utilizadas para la elaboración de la plataforma MP4SO [70]. Las formulaciones de los modelos matemáticos y su resolución se llevaron a cabo utilizando la plataforma GAMS [71]. Todos estos resultados fueron publicados en [51].

3.3.2.1 Resolución de problemas de simulación

Aquí se muestran algunos ejemplos de aplicación en simulación. Los resultados se expresan en términos de la estructura obtenida y las variables que deben inicializarse si se decide utilizar el modelo particionado.

1. Cracking catalítico de gasoil:

Este ejemplo es una variante del modelo extraído de [17]. Consiste en dos ecuaciones diferenciales que permiten determinar los perfiles de concentración de los reactivos en el tiempo. Considerando que el programa empleado para la resolución de los

modelos (GAMS, [71]) no incluye un procesador de ecuaciones diferenciales, fue empleado el método de colocación ortogonal para estimar los perfiles mencionados y los parámetros correspondientes. Para el cálculo se utilizaron cien elementos finitos y cuatro puntos de colocación. La interpolación se realizó a partir de polinomios de Lagrange. Los resultados se observan en la Tabla 3-11.

2. Columna de destilación reactiva: producción de Metil Tert-Butil Éter (MTBE):

Para continuar con la evaluación de los métodos de particionamiento, fue seleccionada como ejemplo una columna de destilación reactiva en donde se produce (MTBE) a partir de metanol e isobuteno [72]. El modelo de la columna consta de ecuaciones correspondientes a cálculos de equilibrio, balances de masa y energía, entre otros. La columna posee 15 etapas teóricas, más condensador y rebullidor. De estas 15 etapas, 8 son etapas reactivas, mientras el resto corresponde a etapas de separación.

En la Tabla 3-11 se resumen los resultados obtenidos para los dos ejemplos de simulación descriptos. Se observa el porcentaje de variables a inicializar con respecto al número total de variables, el número de ecuaciones y variables, la estructura y cantidad de bloques obtenida luego del particionamiento, y el tiempo de resolución para el modelo completo (**MC**) y el modelo particionado (**MP**).

Problema		**1.Cracking Gasoil**		**2. Columna reactiva (8 etapas)**			
Nº de ecuaciones		998		1599			
Nº de variables		998		1599			
% de variables a inicializar		0%		62%			
Estructura de	**Tamaño**	1x1	4x4	1x1	5x5	13x13	997x997
bloques	**Cantidad**	198	200	584	1	1	1
Tiempo de resoluc.	**MC**	0,289		9,019			
en segundos	**MP**	18,67		28,26			

Tabla 3-11: Resultados para los problemas de simulación.

Como se observa en la Tabla 3-11, para el problema de cracking catalítico de gasoil no es necesaria la inicialización de ninguna variable para la resolución del modelo. Al haber sido utilizado en este caso un método de resolución como el de colocación ortogonal, la estructura de los bloques obtenidos resulta intuitiva, y la tarea de particionar el sistema de ecuaciones no brinda un beneficio a la resolución del modelo. Esto se debe a que la matriz de incidencia resultante tiene una estructura de matriz banda o diagonal en bloques, como puede apreciarse en la Figura 3-5. Además de esto, por el pequeño tamaño de los bloques, no es necesaria la inicialización de ninguna variable.

Para el problema de la columna reactiva, el bloque de 997 variables y ecuaciones debe ser inicializado para la resolución del mismo, reduciendo de esta forma la cantidad de variables a inicializar a un 62%. Los tiempos de resolución para ambos modelos no presentan una mejora en el modelo particionado, lo cual puede deberse al tiempo empleado en la generación de los bloques de ecuaciones y variables, y al acople de los valores de las variables entre los bloques.

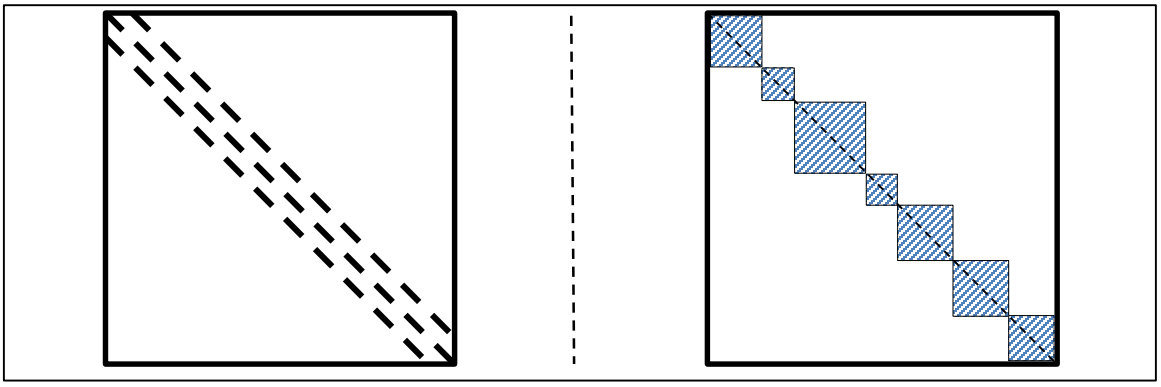


Figura 3-5: Estructura de matriz banda (izquierda); Estructura de matriz diagonal en bloques (derecha).

3.3.2.2 Resolución de problemas de optimización

Luego de explicar los resultados de algunos problemas de simulación en Ingeniería Química, a continuación se detallan los resultados obtenidos para dos problemas de optimización.

1. Proceso de alquilación de Iso-butano:

Este proceso tiene como objetivo la producción de iso-octano, uno de los alcanos más importantes empleados para el aumento del octanaje de las naftas. Su valor de número de octano se considera en 100 como valor de referencia. De manera simplificada, el proceso cuenta con un reactor, un separador, y un sistema de reciclo para la realimentación de la materia prima no reaccionada. Al reactor ingresa una corriente fresca de buteno y una corriente de reposición de iso-butano. Se agrega además ácido sulfúrico en una concentración de 98% en peso, que actúa como catalizador de la reacción. El ácido utilizado es posteriormente retirado del proceso. La corriente de producto del reactor es alimentada a un separador, donde se obtiene por tope el iso-butano no reaccionado, y por fondo el producto de la alquilación, es decir el iso-octano. La función objetivo para este ejemplo es la ganancia neta de la planta. Cabe destacar que este modelo cuenta con 4 ecuaciones de igualdad y 8 ecuaciones de desigualdad.

2. Ubicación óptima de un catalizador en un reactor tubular:

En este caso se desea determinar la distribución óptima de dos catalizadores a lo largo de un reactor tubular de flujo pistón. El modelo fue extraído del sitio web COPS[1].

Para el problema de optimización del proceso de alquilación, si bien la resolución del sistema en bloques resulta considerablemente más sencilla que la resolución del modelo completo, las soluciones de algunos bloques resultan no factibles. Esto surge como

[1] http://www.mcs.anl.gov/~more/cops/

consecuencia del tratamiento de un subconjunto de las restricciones de igualdad, y este problema es abordado en detalle en la próxima sección de este capítulo. La inicialización de aproximadamente el 35% de las variables del proceso permite arribar a una solución aproximada. Estos datos se encuentran resumidos en la Tabla 3-12.

Problema		**1.Alq. De Isobutano**		**2. Loc. Óptima de Catalizador**			
Nº de ecuaciones		12 (4 + 8)		201			
Nº de variables		11		227			
% de variables a inicializar		36%		66%			
Estructura de	**Tamaño**	1x1	2x2	2x2	3x3	4x4	6x6
bloques	**Cantidad**	5	3	14	1	2	6
Tiempo de resoluc.	**MC**	0,1		0,056			
en segundos	**MP**	0,275		0,997			

Tabla 3-12: Resultados para los problemas de optimización.

En el problema de la localización óptima del catalizador, se requiere la inicialización del 66% de las variables, lo cual puede observarse también en la Tabla 3-12, y esto corresponde al conjunto de 152 variables y 126 ecuaciones no incluido entre los bloques de asignación. Al igual que en los casos de simulación, los tiempos de ejecución son notablemente más altos para los modelos particionados, aunque la reducción en la información requerida para la resolución del modelo resulta una ventaja considerable.

3.3.2.3 Particionamiento y optimización con restricciones de desigualdad

En general, los problemas de optimización pueden contener restricciones de igualdad y desigualdad. Es bien sabido que una solución factible debe cumplir con todas y cada una de estas restricciones del problema en cuestión. En los problemas de simulación, el conjunto de ecuaciones solamente contiene igualdades, y en general se puede individualizar una única solución factible. En cambio, las restricciones de desigualdad que puedan existir en un problema de optimización hacen posible la existencia de infinitas soluciones factibles.

Minimizar $f(x, y, z) = x^2 + y^2 + z^2$

Sujeto a:
r_1: $\sin(2{,}2 * y) * \sin(0{,}4 * x) + 0{,}03 * x^2 - 0{,}2 * y^2 + 2 - z \geq 0$
r_2: $z - 0{,}5 * y + 3 \geq 0$
r_3: $-z - 2 * y - x - 1 \geq 0$

Figura 3-6: Formulación de un problema genérico de optimización con restricciones de desigualdad.

Un particionamiento en un sistema de ecuaciones asociado a un problema de simulación/optimización, consiste en la agrupación de ecuaciones en bloques, de manera tal que se pueda atenuar la complejidad de resolución del sistema completo en un solo paso. Cuando existen restricciones de desigualdad en el problema, obtener un particionamiento en el conjunto de restricciones que conlleve a una solución factible es muy complejo. Esto surge por el concepto de particionamiento en sí mismo, ya que al tomar un determinado subgrupo de restricciones no estamos teniendo en cuenta el cumplimiento de las restricciones restantes. Por ello, no podemos asegurar que se obtengan soluciones factibles. En cada subgrupo resuelto se fijan valores para algunas variables del problema, sin tener en cuenta si estos dan lugar a soluciones factibles con respecto al grupo completo de restricciones. Para explicar en detalle esta situación, se explora un problema genérico de optimización con restricciones de desigualdad. Dicho problema está formulado en la Figura 3-6.

En la Figura 3-7 podemos observar las superficies de nivel para algunos valores de la función objetivo (f: 0.1, 1, 9, 25). Cada valor de la misma produce una esfera en el espacio tridimensional, con radio igual a la raíz cuadrada del valor de la función. También se debe destacar que para todos los gráficos utilizados en este ejemplo la escala adoptada fue la siguiente: $x \in [-5; 5];\ y \in [-5; 5];\ z \in [-5; 5]$.

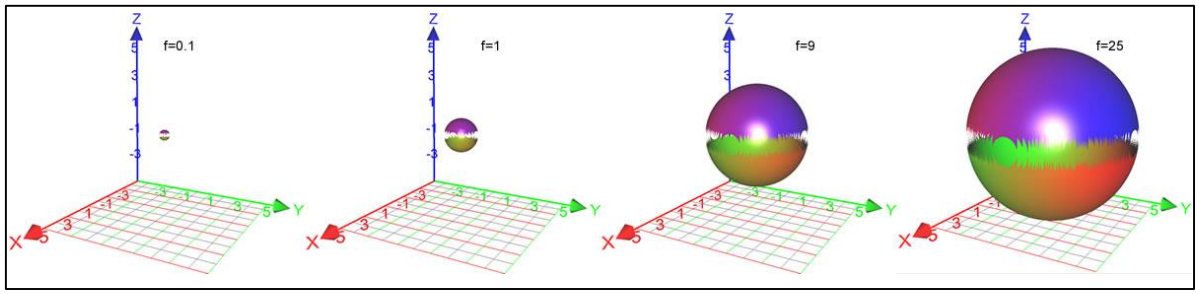


Figura 3-7: Superficies de nivel para la función objetivo.

Del análisis de la función objetivo, se desprende que el mínimo global de esta función está en el punto (0, 0, 0), con valor 0 para la misma. En la Figura 3-8 se muestran las representaciones gráficas de las restricciones r_1, r_2 y r_3, especificadas anteriormente.

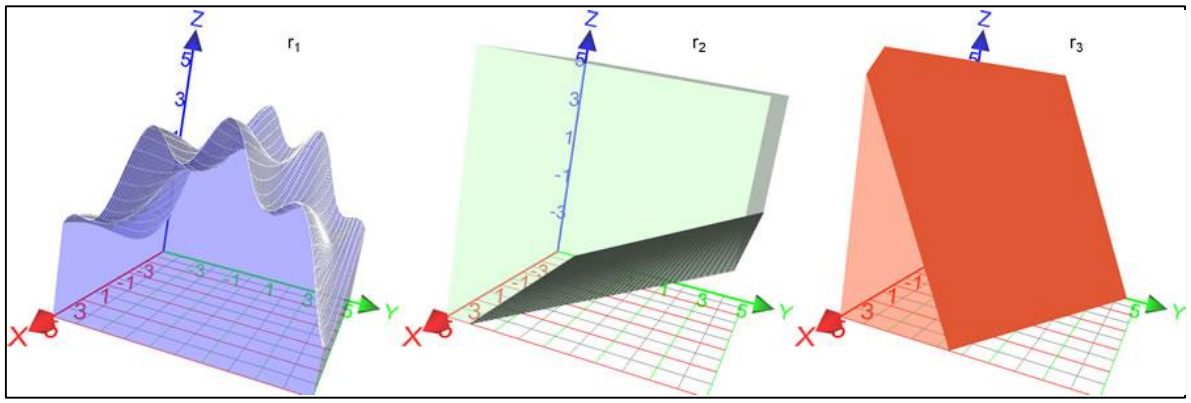


Figura 3-8: Representación gráfica de las restricciones del problema de la Figura 3-6.

Si efectuáramos un particionamiento en el sistema de ecuaciones determinado por las restricciones y la función objetivo del problema, podríamos obtener, por ejemplo, los siguientes subproblemas:

Subproblema 1:

f: $f(x, y, z) = x^2 + y^2 + z^2$

r_1: $\sin(2{,}2 * y) * \sin(0{,}4 * x) + 0{,}03 * x^2 - 0{,}2 * y^2 + 2 - z \geq 0$

r_2: $z - 0{,}5 * y + 3 \geq 0$

Subproblema 2:

r_3: $-z - 2 * y - x - 1 \geq 0$

Luego de efectuar el particionamiento, la resolución del problema principal se lograría mediante una resolución secuencial de los subproblemas obtenidos. En este caso, luego de resolver el subproblema 1, obtendríamos como resultado el óptimo global (x=0, y=0, z=0). Dicho punto cumple con las restricciones r_1 y r_2, y además minimiza la función en la región determinada por estas restricciones. La representación gráfica del punto en la región determinada por las restricciones puede verse en el lado izquierdo de la Figura 3-9. Allí se representa en color celeste el punto de la solución hallada, y las restricciones r_1 y r_2.

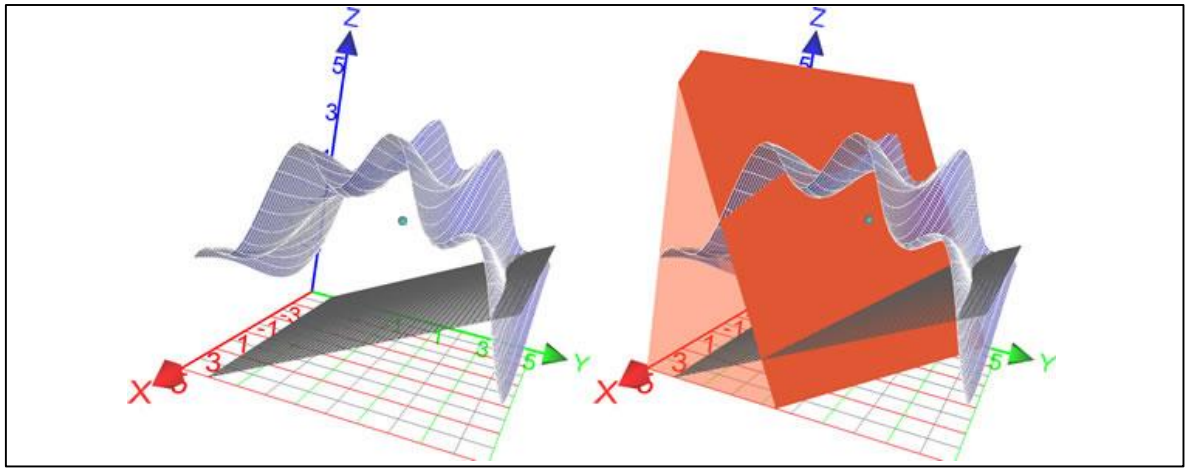


Figura 3-9: Representación gráfica de la solución no factible hallada para el problema de la Figura 3-6.

Si observamos la Figura 3-9, en el gráfico de la derecha se observa el punto (0, 0, 0) obtenido de la resolución del subproblema 1, con la restricción r_3 incluida. Se puede observar que el punto óptimo hallado no cumple con la restricción r_3 (plano y región sombreada, encima del eje X), por lo que es un punto no factible. Por lo tanto, si intentáramos resolver el subproblema 2 habiendo fijado antes los valores obtenidos para *x*, *y* y *z*, llegaríamos a que el problema no tiene solución factible. Lo anterior no es correcto, ya que este problema tiene una solución óptima factible, como se muestra en la Figura 3-10. Este error surge de la resolución parcial que se hace del problema en cada etapa, que

determina valores fijos para grupos de variables que pueden ocasionar tanto la no factibilidad de resolución como el estancamiento en puntos no óptimos.

Si resolviéramos el problema explicado utilizando un método tradicional de resolución de problemas de optimización, se obtendría que el óptimo global del mismo está determinado por los valores x= -1/6, y= -1/3, y z= -1/6, con valor de la función objetivo f=1/6. En la Figura 3-10 se ilustra el plano de la restricción r_3 y la superficie de nivel para la función objetivo en f=1/6. Se puede ver en el lado derecho de la figura que para este valor de f, el plano que identifica al límite de la restricción r_3 es tangente a la esfera que representa a f=1/6 (En esa imagen se cambia un poco el ángulo de perspectiva para poder observar la tangencia). De esta forma, podemos afirmar que el punto óptimo global de este problema es el punto que tienen en común el plano de r_3 y la esfera dada por f=1/6. Este punto tiene las coordenadas (-1/6,-1/3,-1/6).

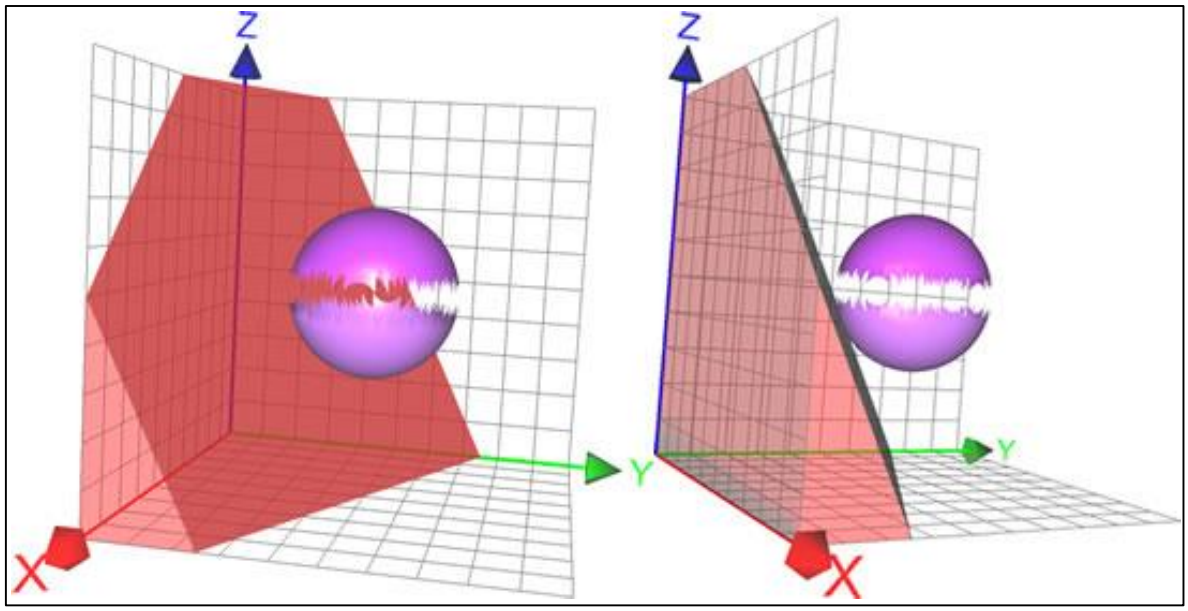


Figura 3-10: Plano y región de la restricción r_3, y esfera de nivel para f=1/6 en distintas perspectivas.

La dificultad para resolver problemas de optimización con restricciones de desigualdad mediante la utilización de particionamientos del sistema de ecuaciones, no se

presenta para problemas que poseen únicamente restricciones de igualdad. Esto se debe a que los valores que se van encontrando para las variables a cada paso, se corresponden con ecuaciones de igualdad que no deberían admitir una gran cantidad de valores para sus variables.

3.4 Conclusiones

En este capítulo se explicó con diversos ejemplos el funcionamiento y las características distintivas de los métodos de particionamiento utilizados en el marco de esta tesis. Como contribución científica principal, fue descripta la implementación de una mejora en el MDE, que de acuerdo a los resultados obtenidos en uno de los ejemplos desarrollados parece tener un mejor desempeño en el hallazgo de bloques de asignación lineales en un sistema de ecuaciones. Fueron analizados los resultados de la implementación de los métodos en el procedimiento de resolución de problemas de simulación y optimización, y se explicó una cuestión incompatible con la utilización de particionamientos en la resolución de problemas de optimización con restricciones de desigualdad.

Debido al problema mencionado con las restricciones de desigualdad, tal vez sea mejor incluir todas ellas en un solo bloque de asignación, ya que particionar este conjunto puede ocasionar el hallazgo de soluciones no factibles en un problema de optimización. Otra manera posible de abordar este problema es realizar un tratamiento de los bloques no factibles, prohibirlos y reinicializar las variables involucradas.

En el ámbito de la resolución de problemas de simulación/optimización, otra cuestión destacable es el tiempo que insume la generación y ejecución independiente de cada uno de los subproblemas derivados de los particionamientos. Este problema se debe a

la necesidad de compilación de cada uno de los bloques de asignación como un modelo independiente, en la plataforma GAMS [71]. Si se lograra realizar una generación conjunta de las particiones y una ejecución secuencial dependiente con una única compilación, tal vez podría aumentarse en una alta proporción la eficiencia de resolución de cada modelo.

Capítulo 4:

Plataforma para Validación de Métodos de Particionamiento

4.1 Introducción

El particionamiento estructural de matrices de incidencia ha sido descripto en los capítulos anteriores, y resulta una estrategia muy útil para simplificar el trabajo sobre modelos matemáticos. Aunque diversos aspectos estructurales pueden ser explotados sobre distintas clases de modelos tales como los asociados a sistemas físicos [73], o sistemas de ecuaciones diferenciales algebraicas [74], el enfoque empleado a lo largo de los capítulos 2 y 3 de esta tesis se orienta hacia las disciplinas de Diseño de Instrumentación, Simulación y Optimización de Sistemas de Procesos en Ingeniería. La investigación llevada a cabo en [41], [45] y [51] ha promovido mejoras significativas en la cantidad de variables observables para la estructura de algunas plantas químicas y la resolución numérica de problemas de simulación/optimización.

Un mecanismo para demostrar que los métodos de particionamiento desarrollados realmente pueden ser de utilidad, es la validación empírica y estadística sobre una cantidad considerable de casos teóricos y prácticos asociados a plantas químicas del mundo real. En este sentido, sería deseable contar con una herramienta que genere casos de estudio automáticamente, de acuerdo a parámetros estadísticos asociados con plantas reales y teóricas. El objetivo principal del trabajo descripto en este capítulo es la construcción de tal plataforma.

En primera instancia se describe el funcionamiento y la estructura de la plataforma de generación de casos de estudio, abarcando el diseño y la programación de la misma. Luego, se explican las cuestiones que representan los parámetros que se desea obtener, y la forma en que se planea generarlos y ajustarlos.

4.2 Plataforma de generación de casos de aplicación

Teniendo en cuenta que la representación utilizada a lo largo de esta tesis para los sistemas de ecuaciones consiste en matrices de incidencia, este es el aspecto estructural atacado mediante la plataforma desarrollada. Por ello, la primera tarea en el desarrollo fue la implementación de un generador de matrices de incidencia. Al observar la estructura matricial de la Figura 4-1, podemos vislumbrar los grupos de elementos que debería incluir una matriz de incidencia generada, luego de haber sido objeto de la implementación de algún método de particionamiento. Estos grupos son:

- Grupo de ecuaciones y variables SR1-SC1: Este es el primer grupo de la matriz en Forma Triangular Inferior en Bloques (FTiB). Sobre su diagonal principal pueden distinguirse subconjuntos cuadrados que representan los bloques de asignación que permitirán la determinación de los valores de las variables asignadas a cada uno de ellos. La característica distintiva de este conjunto respecto al conjunto SR2-SC2 es la existencia de ecuaciones redundantes dentro del sistema, que contienen solamente variables del conjunto SR1-SC1, y pueden intercambiarse por alguna ecuación de dicho conjunto si es necesario.
- Grupo de ecuaciones y variables SR2-SC2: Este conjunto también contiene un grupo de bloques de asignación, pero a diferencia de SR1-SC1, no existen ecuaciones redundantes que puedan reemplazar a las ecuaciones dentro de este grupo.
- Grupo de ecuaciones redundantes VR: Las ecuaciones de este grupo solamente contienen variables del grupo SC1, por lo que pueden reemplazar eventualmente a las ecuaciones de SR1 en una formación alternativa de bloques de asignación. En

los algoritmos de particionamiento estudiados aquí, la etapa de Reasignación justamente consiste en buscar dentro del grupo VR alguna ecuación que pueda suplantar a otra dentro de SR1, para evitar la presencia de bloques prohibidos (ver Capítulo 2).

- Grupo de ecuaciones con variables indeterminables HR: Ecuaciones que contienen variables que no pueden ser calculadas mediante la resolución del presente sistema.
- Grupo de variables indeterminables HC: Variables cuyo valor no puede ser determinado mediante la resolución del sistema.

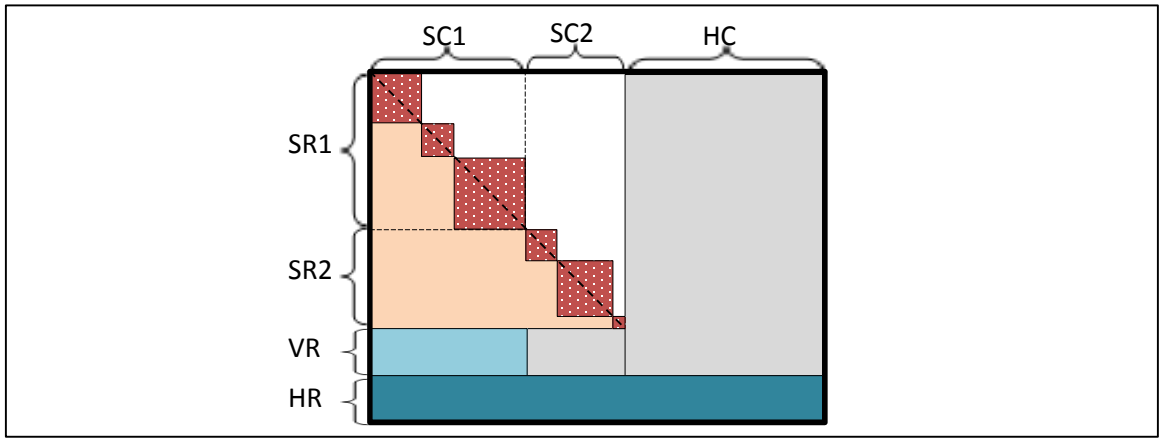


Figura 4-1: Grupos de variables y ecuaciones que constituyen la FTiB de una matriz de incidencia.

Para efectuar la generación de una matriz de incidencia que pueda ser llevada a una FTiB como la de la Figura 4-1, una estrategia posible consiste en tomar el camino inverso, conformando una estructura FTiB y luego ejecutando un ordenamiento aleatorio dentro de la misma para obtener una matriz de incidencia genérica. De esa forma nos aseguramos de que la matriz pueda ser permutada por lo menos a una FTiB.

Para lograr la formación de una estructura como la de la Figura 4-1, deben ser establecidos los valores de diversos parámetros. Estos determinarán la forma y el tamaño de los bloques finales de la FTiB preliminar. Para que la formación de las matrices de incidencia esté ligada a un proceso estocástico controlado, todos estos parámetros de su

configuración deben ser obtenidos a partir de números aleatorios o pseudoaleatorios, con funciones de distribución de probabilidad conocidas. Esto también permite la obtención de un número arbitrario de matrices de incidencia en base a parámetros preestablecidos, lo que facilita la realización de estudios estadísticos sobre los datos generados. Si los parámetros son ajustados a modelos matemáticos conocidos, los resultados estadísticos pueden ser de gran utilidad. En la Figura 4-2 puede verse un esquema que explica el funcionamiento descripto para la plataforma.

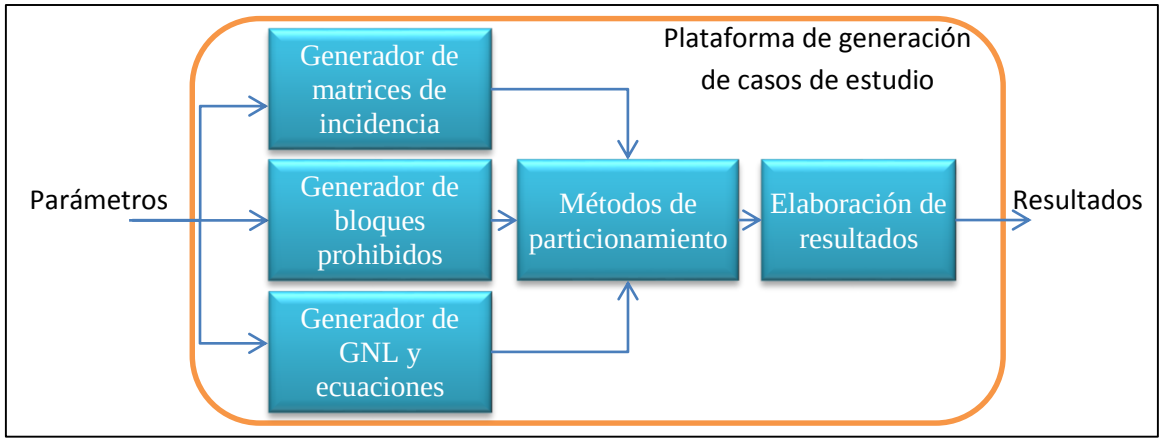


Figura 4-2: Esquema gráfico del funcionamiento de la plataforma.

Los parámetros que se deben establecer para la formación de las matrices de incidencia en la plataforma son los siguientes:

- Cantidad de ecuaciones y variables del sistema: Estos valores darán la característica fundamental para la matriz de incidencia correspondiente, determinando si la misma tendrá forma cuadrada o rectangular, y qué dimensión tendrá.
- Tamaño del Grupo completo SR1-SR2-SC1-SC2: Este valor define la cantidad de ecuaciones del bloque cuadrado correspondiente, y vendrá dado como una proporción del menor valor entre el total de ecuaciones y el de variables. Consiste en un número aleatorio, cuya distribución de probabilidad puede ser la Distribución Normal (DN) y la media y varianza serán provistas por el usuario de la plataforma.

- Tamaño del Grupo SR1-SC1: Luego de haber establecido el tamaño del grupo completo SR1-SR2-SC1-SC2, este valor será calculado como una proporción del mismo. Dicha proporción será un valor aleatorio, también con una DN, y media y varianza provistas por el usuario.
- Tamaño del Bloque SR2-SC2: Este valor viene dado simplemente por la diferencia entre la cantidad de ecuaciones del bloque total SR1-SR2-SC1-SC2 y el bloque SR1-SC1.
- Tamaños de los Bloques de asignación dentro de SR1-SC1 y SR2-SC2: El tamaño de cada bloque de asignación dentro de cada uno de los grupos cuadrados SR1-SC1 y SR2-SC2 será preestablecido mediante valores aleatorios de proporción, también distribuidos normalmente.
- Tamaño del Bloque de Ecuaciones Redundantes: La dimensión de este bloque depende de otras dos características de la matriz de incidencia que se generará: La cantidad de filas de esta matriz se obtendrá de la diferencia entre la cantidad total de filas de la matriz y la suma de las cantidades de los demás grupos (SR1, SR2 y el Bloque Indeterminable), y su cantidad de columnas es igual a SC1.
- Tamaño del Bloque Indeterminable: Este tamaño también será determinado con un valor aleatorio, el cual tendrá como restricción que la cantidad de ecuaciones del mismo deberá ser menor que la cantidad de variables indeterminables en el sistema. Si no es cumplida esta última condición, entonces no estaríamos en presencia de un bloque indeterminable.
- Cantidad de variables presentes en cada ecuación: Este valor determinará cuántos unos estarán presentes en una fila, lo que es equivalente a la cantidad de variables

presentes en la ecuación correspondiente. Se debe destacar que, para los bloques de asignación dentro del grupo SR1-SC1-SR2-SC2, este número será la suma de dos valores: la cantidad de unos de la fila dentro del bloque cuadrado y la cantidad de unos de la fila en el rectángulo que antecede a dicho bloque (ver Figura 4-3). En este caso, el valor aleatorio estará uniformemente distribuido, en lugar de presentar una distribución normal. Esto se debe a que el comportamiento de este parámetro dentro de cada ecuación es diferente al de los demás parámetros que rigen a la formación de las matrices.

- Posición de cada uno dentro de la fila: Una vez que es definida la cantidad de variables que estarán presentes en cada ecuación, el siguiente paso es determinar cuáles serán esas variables. Para ello se emplea otro valor uniformemente distribuido, que será calculado para cada posición. Este cálculo se hará la cantidad de veces que determina el parámetro de *Cantidad de variables presentes en cada ecuación*. Con ello se agrega el uno correspondiente en cada posición de la matriz.

En la Figura 4-3 se muestra un esquema gráfico que puede ayudar a entender el significado de cada uno de los parámetros descriptos.

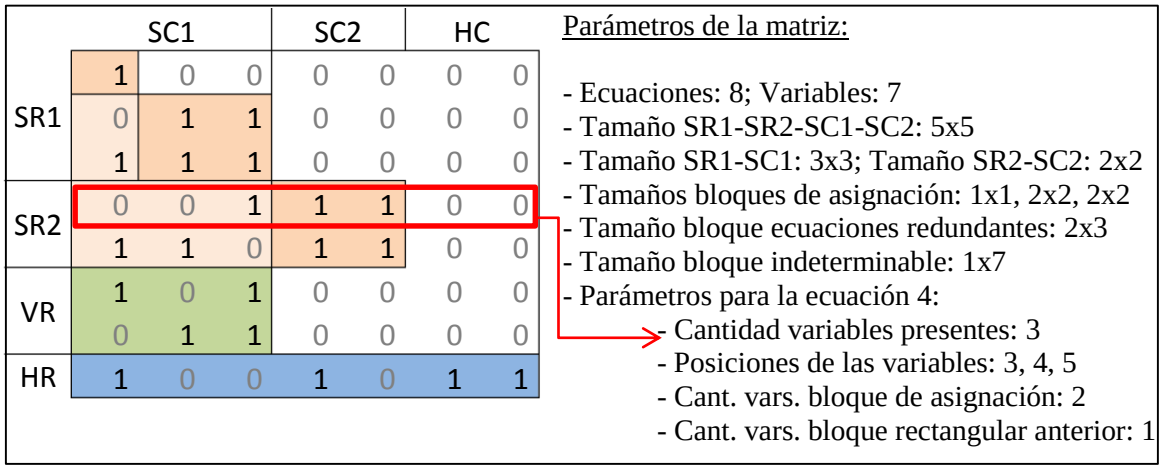


Figura 4-3: Ejemplo práctico de los parámetros de formación de matrices de incidencia.

Además de los parámetros asociados a la matriz de incidencia, se deben definir algunos parámetros adicionales relacionados con los sistemas de ecuaciones. Estos son:

- Cantidad de bloques prohibidos: Mediante un valor aleatorio, se le indica a la plataforma la cantidad de bloques prohibidos o restricciones que acompañarán al sistema de ecuaciones que puede generarse a partir de la matriz de incidencia elaborada. Para mayor detalle acerca de estos bloques, ver Capítulos 2 y 3.
- Grados de no linealidad para ecuaciones y variables: También son definidos de manera aleatoria, y determinarán las condiciones que deben cumplir las ecuaciones y variables del sistema.

Con la generación de toda esta información, se hace posible la evaluación de los métodos de particionamiento desde un punto de vista estadístico. Mediante la prueba y error en el ajuste de los parámetros de acuerdo a modelos matemáticos reales, tal vez se podría llegar a elaborar juicios de valor acerca del desempeño de uno u otro método, lo cual sería, por ejemplo, un soporte para decidir su incorporación en programas de cómputo asociados a la simulación u optimización de procesos.

4.2.1 Diseño algorítmico de la plataforma

Para construir la plataforma deseada, a grandes rasgos, los pasos a seguir dentro de su funcionamiento deberían incluir la generación de la matriz de incidencia, las estructuras de datos asociadas a las restricciones sobre la formación de bloques de asignación y la información del GNL de las ecuaciones y variables del sistema asociado. Luego, la tarea consiste en ejecutar los tres métodos de particionamiento sobre las estructuras generadas y elaborar los reportes correspondientes, que describieran el resultado obtenido al aplicar los particionamientos. En base a los parámetros de formación de matrices descriptos

anteriormente, este proceso puede organizarse para recabar información estadística de numerosos casos generados por la plataforma. El Algoritmo 4-1 expresa en detalle cómo trabaja la plataforma desarrollada.

1. *Datos de entrada: Parámetros de formación de matrices de incidencia,* ***CANT****: cantidad de casos a elaborar*
2. *Repetir* ***CANT*** *veces:*

2a. *Generar aleatoriamente una matriz* ***M*** *en FTiB, en base a los parámetros ingresados al algoritmo*

2b. *Reordenar aleatoriamente las filas y columnas de la matriz* ***M***

2c. *Generar un conjunto* ***Rest*** *de restricciones para la formación de bloques de asignación, en base a los parámetros ingresados al algoritmo*

2d. *Definir* ***GNL aleatorios*** *para ecuaciones y variables*

2e. ***GNL=0*** *para un grupo aleatorio de ecuaciones y variables, en base a los parámetros provistos al algoritmo*

2f. *Ejecutar MD, MDE y MDE Mejorado para la matriz de incidencia y las estructuras de datos obtenidas*

2g. *Guardar los resultados de los particionamientos*

Algoritmo 4-1: Pasos para la generación y evaluación de los casos aleatorios de aplicación.

De acuerdo al estudio que se desee realizar, los datos que se suministran como entrada al Algoritmo 4-1 corresponden a los parámetros que configurarán los casos generados y la cantidad de casos que es necesario construir. Dentro de los parámetros podemos mencionar las dimensiones que tendrán los casos a generar (ecuaciones y variables) y los valores de medias y desvíos estándar para las distribuciones normales que rigen a los tamaños de los diferentes bloques de cada matriz. Un ejemplo de tales valores puede ser el porcentaje promedio μ del tamaño del bloque SR1-SC1 de un conjunto de matrices de incidencia correspondientes a diversos modelos matemáticos, y su desvío estándar σ. Mediante el ajuste de todos estos parámetros pueden generarse muestras significativas para distintas configuraciones de matrices.

En el paso 2, el Algoritmo 4-1 entra en un lazo repetitivo en el cual generará la cantidad de casos de estudio que indica el valor "*CANT*", dado por parámetro. El primer

paso dentro de este ciclo es el armado de una matriz **M** cuya estructura será FTiB, y el procedimiento para su construcción se detalla en el Algoritmo 4-2. Luego, en el paso 2b, se efectúa una permutación de la matriz **M**, cuyo propósito es que los métodos de particionamiento puedan encontrar otra FTiB para **M**, sin información previa de la FTiB inicial.

El paso 2.c del Algoritmo 4-1 genera la estructura de datos que contiene a las restricciones para la formación de bloques de asignación, en la matriz de incidencia actual. El proceso de construcción de esta estructura tiene como parámetro la cantidad máxima de restricciones que pueden generarse, y su funcionamiento consiste en la generación al azar de cada bloque prohibido de variables y ecuaciones, según el tamaño que se desea que tenga cada restricción. Para describir este procedimiento en algunos pasos sencillos, podemos decir que, luego de establecer la cantidad de restricciones que tendrá la estructura, los pasos son los siguientes:

- Establecer la cantidad de bloques prohibidos para cada tamaño de restricción (los bloques prohibidos se entregan a los métodos de particionamiento ordenados de manera ascendente según su tamaño).
- Para cada tamaño de restricción, generar los distintos bloques arrojando números al azar para los identificadores de variables y ecuaciones que formarán parte, controlando que no se repitan estos elementos dentro de un mismo bloque.

Una vez que fueron definidos los bloques prohibidos que acompañarán a la matriz de incidencia en los métodos de particionamiento, se define aleatoriamente el GNL para cada variable y ecuación en el paso 2.d. En principio, esta rutina solamente arroja un valor al azar para cada variable y ecuación, dentro del intervalo [0; 3], que son las cotas entre las que deben estar todos los valores de GNL. El cálculo de los GNL aleatorios podría

vincularse a una generación aleatoria de términos de ecuaciones, lo cual constituye una tarea para el futuro. Para mayor información sobre la tarea propuesta, puede consultarse el Capítulo 2 de esta tesis, en la sección correspondiente al cálculo de valores de GNL.

Una cuestión importante en estos pasos del Algoritmo 4-1 es la formación de ecuaciones y variables lineales. Se debe destacar que para que una ecuación o variable sea lineal su GNL debe ser 0, y en general los números más pequeños que arroja un motor de números aleatorios no llegan a ser 0, sino valores cercanos. Por ello, parte de los parámetros que se incluyen en la ejecución del algoritmo corresponden a la media y desvío estándar de la proporción de ecuaciones y variables lineales que debe generar la plataforma para cada estructura de datos construida. Entonces, el paso 2.e consiste en el cálculo de la proporción de ecuaciones y variables lineales de manera aleatoria y en base a los parámetros provistos, y la determinación también aleatoria de cuáles serán estas ecuaciones y variables.

Concepto	MD	MDE	MDEM
Bloques Lineales	1	2	3
Vars Obs x Bloqs Lins	1	3	5
Bloques No Lineales	3	4	2
Vars Obs x Bloqs No Lins	6	4	2
Total Bloques	4	6	5
Total Variables Observables	7	7	7

Tabla 4-1: Reporte arrojado por la Plataforma para un caso determinado.

Habiendo generado la matriz de incidencia, los bloques prohibidos y los GNL para ecuaciones y variables, el paso 2.f del Algoritmo 4-1 especifica la ejecución de los métodos de particionamiento sobre dicho conjunto de datos. Cada uno de los métodos (MD, MDE y MDE mejorado) es ejecutado por separado, y los resultados obtenidos serán alimentados a la plataforma para la elaboración de los resúmenes correspondientes en el paso 2.g.

Esencialmente, los resultados que serán de utilidad acerca de la ejecución de los métodos son las cantidades y características de los bloques de asignación que cada uno de ellos genera como consecuencia del tratamiento de la información generada por la plataforma. Uno de los reportes que pueden obtenerse directamente del funcionamiento de esta plataforma tiene el formato exhibido en la Tabla 4-1.

1. *Datos de entrada: Params (Parámetros estadísticos para formación de la matriz)*
2. *tamañoBloqueSR1SR2=EnteroAleatorio(Params->SR1SR2)*
3. *tamañoBloqueSR1= EnteroAleatorio(Params->SR1)*
4. *tamañoBloqueSR2= tamañoBloqueSR1SR2- tamañoBloqueSR1*
5. *Diagonal(**BloqueSR1SR2**)=1*

Definición de bloques de asignación

6. ***Repetir***

6a. *Generar al azar un tamaño para bloque de asignación*

7. ***Hasta completar SR1-SC1***
8. ***Repetir***

8a. *Generar al azar un tamaño para bloque de asignación*

9. ***Hasta completar SR2-SC2***
10. ***Para cada bloque de asignación de SR1-SC1 y SR2-SC2***

10a. ***Para cada ecuación (fila)***

i. *10a.i. Determinar la cantidad y posición de las variables que estarán presentes en la ecuación, teniendo en cuenta los límites del bloque*

Bloque de ecuaciones redundantes y Bloque indeterminable

11. *ColumnasBloqIndet=Params.cantVariables-tamañoBloqueSR1SR2*
12. *FilasBloqIndet= EnteroAleatorio (columnasBloqIndet-1)*
13. *ColumnasBloqRedun=tamañoBloqueSR1;*
14. *FilasBloqRedun=Params.cantEcuaciones-tamañoBloqueSR1SR2-FilasBloqIndet;*
15. ***Para cada ecuación (fila) del Bloque Redundante***

15a. *Determinar la cantidad y posición de las variables que estarán presentes en la ecuación, teniendo en cuenta los límites del bloque*

16. ***Para cada ecuación (fila) del Bloque Indeterminable***

16a. *Determinar la cantidad y posición de las variables que estarán presentes en la ecuación, teniendo en cuenta que cada ecuación debe tener por lo menos 2 variables indeterminables*

Algoritmo 4-2: Construcción de una matriz de incidencia en base a parámetros dados.

El procedimiento indicado en el paso 2.a del Algoritmo 4-1, el cual lleva a cabo el armado de la matriz de incidencia, está detallado en el Algoritmo 4-2. Los parámetros de

formación para la matriz de incidencia le son enviados a este algoritmo, para definir las características de la matriz.

Los pasos 2 y 3 del Algoritmo 4-2 determinarán los tamaños del bloque completo SR1-SC1-SR2-SC2, y del bloque SR1-SC1. Para la determinación de ambos valores, se utilizan los parámetros que indican las medias y desvíos estándar para sus proporciones. Después se multiplica el número obtenido para SR1-SC1-SR2-SC2 por el mínimo entre la cantidad de ecuaciones y de variables del sistema para obtener dicho tamaño, y la proporción obtenida para SR1-SC1 por el tamaño total de SR1-SC1-SR2-SC2. Con ello se determina el tamaño de los bloques mencionados, y el tamaño del bloque SR2-SC2 simplemente se obtiene por la diferencia de los dos tamaños anteriores, como se observa en el paso 4.

Una característica fundamental del bloque SR1-SC1-SR2-SC2 es que tiene su diagonal llena. Esto se ve reflejado en el paso 5 del algoritmo, en el cual se establece el valor 1 para todos los elementos de la diagonal de dicho bloque.

La tarea siguiente en el Algoritmo 4-2 es la determinación de los bloques de asignación. Primero, como se indica en los pasos 6 al 9, se define el tamaño que tendrá cada bloque de asignación. Luego, los bloques son armados calculando aleatoriamente la cantidad de variables presentes en cada ecuación de cada bloque (paso 10). Hay que destacar en esta sección del algoritmo que se debe tratar de manera diferenciada a las variables que forman parte del bloque de asignación y aquellas que están en el rectángulo que antecede a dicho bloque, para evitar la posibilidad de generar por accidente bloques de asignación más pequeños.

Para la determinación del bloque redundante y el bloque indeterminable de la matriz de incidencia, se ejecutan los pasos 11 al 16 del Algoritmo 4-2. Lo primero es calcular el

tamaño del bloque indeterminable, ya que este tiene la particularidad de que la cantidad de filas que lo componen no puede ser igual o mayor que la cantidad de variables indeterminables del sistema, porque de lo contrario estaríamos en presencia de otro bloque de asignación, y no existirían variables indeterminables en el sistema. Es por ello que en la línea 11 se calcula la cantidad de columnas asociadas a variables indeterminables como la diferencia entre la cantidad de variables del sistema y la cantidad de variables del bloque completo SR1-SC1-SR2-SC2, y en la línea 12 la cantidad de filas del bloque indeterminable en base a un número aleatorio que no puede ser mayor o igual que la cantidad de variables indeterminables antes calculada.

Después de haber realizado los cálculos anteriores, es posible determinar la cantidad de ecuaciones que formarán parte del bloque redundante como la diferencia entre el total de ecuaciones del sistema y la suma de la cantidad de ecuaciones contenidas en el bloque completo SR1-SC1-SR2-SC2 y el bloque indeterminable. Esto se ve en el paso 14, mientras que en el paso 13 se calcula la cantidad de columnas del bloque redundante, que será igual a la cantidad de columnas del bloque SR1-SC1.

En los pasos 15 y 16 del Algoritmo 4-2 se realiza el cálculo de las cantidades y posiciones de las variables de cada una de las ecuaciones del bloque redundante y el bloque indeterminable, de manera análoga a como se hizo para el bloque SR1-SC1-SR2-SC2. La única salvedad para el bloque redundante es que deben respetarse los límites de dicho bloque, y no incluir ninguna variable que sea ajena al mismo en alguna de sus ecuaciones. Respecto del bloque indeterminable, se debe cuidar que cada ecuación contenga al menos dos variables indeterminables, para que no se formen bloques de asignación de una variable por una ecuación.

4.3 Parámetros estadísticos basados en modelos de plantas químicas

El propósito de la elaboración de la plataforma que se describe en este capítulo es posibilitar la realización de estudios estadísticos sobre los métodos de particionamiento. Un requisito fundamental para lograr este propósito es disponer de información asociada a diversos casos reales y académicos de problemas de simulación y optimización, en forma de parámetros estadísticos que permitan la formación de poblaciones estocásticas generadas con las rutinas de la plataforma.

En esta sección se muestran los parámetros extraídos de algunos casos representativos de la Ingeniería Química y el Diseño de Instrumentación. Luego, en la siguiente sección, se muestra el resultado de incluir algunos de estos parámetros en la plataforma de generación de casos de estudio. Cada parámetro estudiado permite dar la forma deseada a una matriz de incidencia, de acuerdo a las características de un caso real. Por ejemplo, si conocemos en promedio qué densidad tienen las matrices de incidencia de un proceso determinado y cómo se distribuyen las variables a lo largo de las ecuaciones del sistema, tal vez se puedan obtener resultados estadísticos interesantes que permitan determinar el grado de utilidad de uno u otro método de particionamiento. Los casos de estudio a tener en cuenta en esta sección son los siguientes:

- Columna de destilación reactiva: Este caso está asociado al modelo matemático de la sección de purificación mediante destilación de una planta de síntesis de amoníaco. Fue analizado en [45], y consiste en una columna de destilación de dos etapas.
- Planta de Síntesis de Amoníaco: Este modelo fue extraído de [75], y estudiado también en [45].

Los parámetros relativos a la matriz de incidencia y las restricciones que describen el sistema de ecuaciones de cada uno de los modelos mencionados se pueden ver resumidos en la Tabla 4-2.

Parámetro	Columna de destilación	Planta de Amoníaco
Ecuaciones	102	557
Variables	85	513
Cantidad de celdas	8670	285741
Cantidad de unos	265	1991
Densidad	3,06%	0,70%
Cantidad de bloques prohibidos	29	104
Tamaño máximo de bloque prohibido	10	21
Porcentaje de ecuaciones lineales	60%	54%

Tabla 4-2: Parámetros obtenidos de los casos analizados.

La primera columna de la Tabla 4-2 indica qué parámetro se evalúa para cada uno de los modelos matemáticos analizados, los cuales están representados en las dos columnas siguientes. Los primeros dos parámetros indican las cantidades de ecuaciones y variables de los sistemas de ecuaciones, los cuales darán la dimensión de la matriz de incidencia de cada modelo. Luego, se indica la cantidad de celdas de la matriz, que simplemente expresa las posiciones disponibles para almacenar los unos o ceros en ellas, y se obtiene mediante el producto de su cantidad de filas y de columnas. La cantidad de unos en conjunto con el valor anterior nos permiten calcular la densidad de la matriz de incidencia, que refleja la proporción de los elementos presentes en la matriz (unos) frente a la cantidad total de posibles lugares para almacenar valores en ella (total de elementos). Como consecuencia de que los modelos matemáticos tratados aquí producen matrices ralas, podemos ver que la densidad de las matrices de incidencia de los modelos tratados es muy baja: 0,70% para la planta de amoníaco y 3,06% para la columna de destilación. Las dos filas siguientes en esta tabla se refieren a los bloques prohibidos que se deben tener en cuenta para la formación de

bloques de asignación sobre los modelos matemáticos mencionados. El último valor de esta tabla corresponde a la cantidad de ecuaciones lineales del sistema, en proporción a la cantidad total de ecuaciones. Éste último valor brinda información muy interesante, que permitirá analizar más profundamente la no linealidad de los bloques que se obtengan como resultado de los métodos de particionamiento.

4.4 Resultados para una de las configuraciones de plantas químicas

En este apartado se analizan algunas primeras impresiones sobre el desempeño de la plataforma, habiendo incluido algunos de los parámetros desarrollados en la sección anterior de este capítulo. Utilizamos la configuración dada por el modelo matemático de la Columna de Destilación, descripta por los valores de la columna correspondiente de la Tabla 4-2.

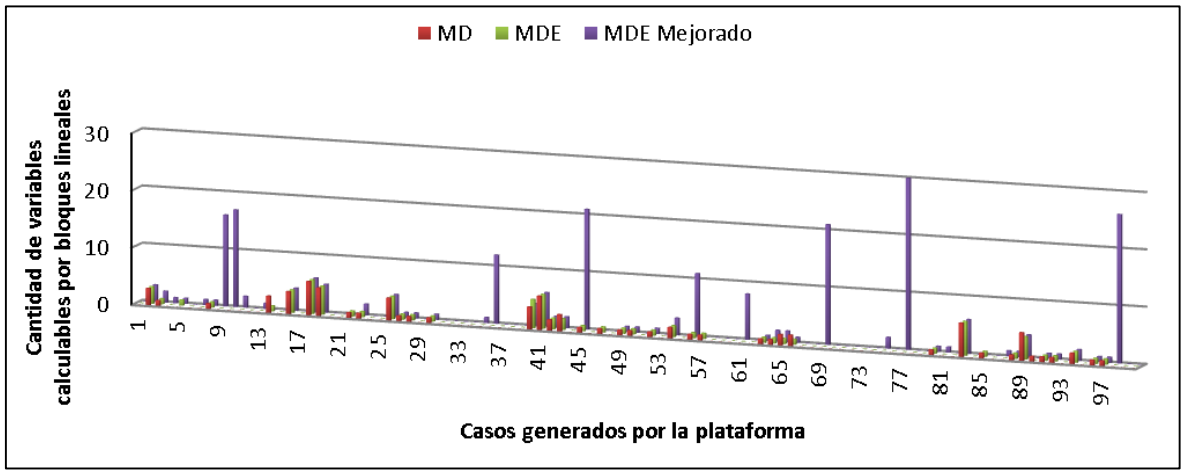


Figura 4-4: Gráfico de barras para 100 casos generados por la plataforma, con la configuración de la Columna de destilación.

En el gráfico de la Figura 4-4 se puede observar algunos valores obtenidos para la configuración mencionada. Para este pequeño estudio fueron generados 1.000 casos aleatorios con esos parámetros, pero por una cuestión de espacio sólo se muestran los primeros 100 en el gráfico de la Figura 4-4. Como se observa en dicho gráfico, en una

considerable cantidad de instancias el número de variables calculables mediante bloques de asignación lineales (eje vertical) es mayor para el particionamiento obtenido con el MDE Mejorado, y esto se observa también a lo largo de los 900 ejemplos no incluidos allí.

Método	Media	Desv Est	IC 95%
MD	0,83	1,44986938	**0,546 - 1,114**
MDE	0,78	1,41121053	**0,503 - 1,057**
MDE Mejorado	2,47	5,50014692	**1,392 - 3,548**

Tabla 4-3: Datos estadísticos de variables lineales para 100 casos generados por la plataforma.

En la Tabla 4-3 se muestran los valores estadísticos correspondientes a los 100 primeros casos generados por la plataforma con los parámetros respectivos a la columna de destilación, en correspondencia con el gráfico de la Figura 4-4. La primera columna de esa tabla indica el método de particionamiento, mientras que la segunda se refiere a la media o promedio de variables calculables por bloques lineales de acuerdo a cada método en los particionamientos efectuados. Las últimas dos columnas corresponden al desvío estándar y los intervalos de confianza al 95% para la cantidad de variables calculables por bloques lineales en la muestra dada. Teniendo en cuenta que los intervalos de confianza para el valor bajo estudio no se encuentran solapados en el caso del MD y el MDE versus el MDE Mejorado, podemos concluir que la diferencia existente entre estos valores para la muestra presentada es significativa desde el punto de vista estadístico, lo cual es un resultado muy favorable para la mejora implementada al MDE.

Método	Media	Desv Est	IC 95%
MD	1,17	2,14763506	1,041 - 1,307
MDE	1,19	2,21315025	1,055 - 1,329
MDE Mejorado	3,57	6,88533099	3,148 - 4,002

Tabla 4-4: Datos estadísticos de variables lineales para 1000 casos generados por la plataforma.

Al observar los resultados de la Tabla 4-4, correspondientes a los 1.000 casos generados con la configuración mencionada, se puede ver que incluso el valor estudiado

tiene una media más alta (3,57 variables calculables por bloques lineales en promedio contra 2,47 en el estudio para los 100 primeros casos). También, debido a que nuevamente el intervalo de confianza para el MDE Mejorado no se encuentra solapado con los otros dos intervalos, podemos decir que la cantidad de variables calculables por bloques lineales al utilizar el MDE Mejorado es significativamente mayor que con los otros dos métodos, para esta configuración de parámetros.

4.5 Proyecto para concluir la validación: Carga de modelos de plantas químicas

Si bien pudieron ser obtenidos resultados preliminares acerca del desempeño de los métodos de particionamiento mediante la utilización de la plataforma de generación de casos, para lograr un ajuste más apropiado de los parámetros que rigen a la construcción de las matrices de incidencia, se hace necesario contar con un gran volumen de datos respecto a modelos y problemas reales. Para ello, y con el objetivo de incluir información verificada por especialistas del área del Diseño de Instrumentación, sería deseable llevar a cabo un proyecto que permita la inclusión de ingenieros químicos en el estudio y la implementación de modelos reales para alimentar la plataforma con parámetros más certeros. Este proyecto se contempla en la sección 8.4 de esta tesis.

El desarrollo de un proyecto como el que se plantea en esta sección también apunta a la formación de recursos humanos en el área respectiva. Como actividad paralela al ajuste de los parámetros de la plataforma, las personas que serían parte de tal proyecto deberían incorporar los conocimientos respectivos a la construcción de los modelos matemáticos asociados.

4.6 Conclusiones

A lo largo de este capítulo fueron expuestas las características del proceso de elaboración de una plataforma de generación de casos aleatorios de matrices de incidencia y otras estructuras de datos, vinculados a sistemas de ecuaciones de modelos matemáticos correspondientes a problemas de simulación y optimización en Ingeniería Química. El objetivo del desarrollo de la misma es la validación estadística de los métodos de particionamiento que son objeto de la primera parte de esta tesis. Es destacable la posibilidad de lograr una vinculación más estrecha entre los casos generados por la plataforma y los problemas reales de Ingeniería Química, particularmente para el área del Diseño de Instrumentación, si se logra un ajuste óptimo de los parámetros que rigen la formación de las estructuras de datos en la plataforma. Este último aspecto también es detallado en una sección del presente capítulo, dando lugar a un proyecto de investigación a tal efecto.

Teniendo en cuenta los métodos de particionamiento empleados aquí y los parámetros que permiten la determinación de la complejidad de las ecuaciones de un sistema, sería deseable también identificar el nivel de impacto que tiene el reajuste de los valores de ponderación para la obtención de los GNL de variables y ecuaciones. En el capítulo 2, en la Tabla 2-1, se puede apreciar el conjunto de valores que se utilizan actualmente para este fin. Como se expresa en [45], un cambio en estos parámetros puede generar valores de GNL más apropiados, según el problema abordado.

Parte II:

Modelos de propagación de relevancia sobre un directorio de internet

Capítulo 5:

Análisis Estructural en Ciencias de la Información

5.1 Introducción

A lo largo de la primera parte de esta tesis fueron desarrollados aspectos teóricos y prácticos concernientes a métodos algorítmicos de particionamiento estructural. Dichos métodos hacen uso de algoritmos sobre grafos, los cuales tienen un área de influencia tan grande que puede abarcar, como en el caso de esta tesis, un espectro muy diversificado de casos de estudio ([10], [68]). La segunda parte de la presente tesis se concentra en otra área diferente de la ciencia, en la que, además del análisis estructural, la utilización de uno de los algoritmos mencionados se torna muy fructífera [19].

En los capítulos 2 a 4 de esta tesis (Parte I), el tema de estudio estuvo constituido por el particionamiento estructural de modelos matemáticos, mientras que en los capítulos 5 a 7 (Parte II), la investigación está volcada a la elaboración y validación de modelos estructurales de propagación de relevancia sobre grandes volúmenes de datos. Si bien existe una clara diferencia entre los ámbitos de aplicación desarrollados en cada una de las dos partes de la tesis, la investigación siempre fue llevada a cabo en torno al concepto de análisis estructural. Partiendo de esta base, el foco de estudio lo constituyen las matrices sobre las que se trabaja en cada caso. En el marco de esta tesis, siempre se ha trabajado con matrices ralas de gran porte. Este es un aspecto fundamental en la representación de la información proveniente de los problemas de la ciencia hoy en día. Tanto en los modelos matemáticos asociados a la ingeniería como en el tratamiento eficiente de grandes volúmenes de información, el análisis estructural puede significar una ventaja decisiva en los desafíos que puedan presentarse.

En las primeras secciones de este capítulo se introduce el concepto de ontologías informáticas y se hace una reseña sobre publicaciones relacionadas con el tratamiento de

grandes corpus de información. Luego se describe el Proyecto de Directorio Abierto (en inglés Open Directory Project, ODP) y una herramienta generada como parte del trabajo de esta tesis para su visualización. Los directorios web como ODP son conocidos también como ontologías de tópicos. Al final del capítulo, se explica la implementación de un algoritmo de detección de componentes fuertemente conexas [20] sobre el grafo de ODP.

5.2 Ontologías informáticas: Análisis Contextual

El concepto de ontología, en el ámbito de la informática, hace referencia a la construcción de un esquema conceptual con un determinado nivel de rigurosidad, con el objetivo de facilitar la comunicación entre diferentes entidades. En una gran cantidad de trabajos como los enumerados aquí, se intenta hacer uso de ontologías de diversas clases para distintos propósitos. Las relaciones entre las entidades que se modelan mediante una ontología pueden ser de diversos tipos. Por ejemplo, en ODP se puede modelar estructuras jerárquicas o relaciones conceptuales entre los tópicos del directorio. Los enlaces del tipo jerárquico pueden ser, a su vez, de dos clases: componentes de la jerarquía principal o clasificaciones alternativas. En la Figura 5-1 se puede apreciar esta situación.

En el siguiente apartado se detallan distintos trabajos publicados por diversos autores, en los cuales es notable la forma en que se manipulan los datos de grandes volúmenes de información representados mediante ontologías, partiendo de un enfoque estructural, y utilizando los conceptos de relevancia y similitud semántica. Para esta recopilación, se tuvo en cuenta aquellos trabajos en los que se hace importante la determinación del grado de relación entre las entidades modeladas en cada ontología.

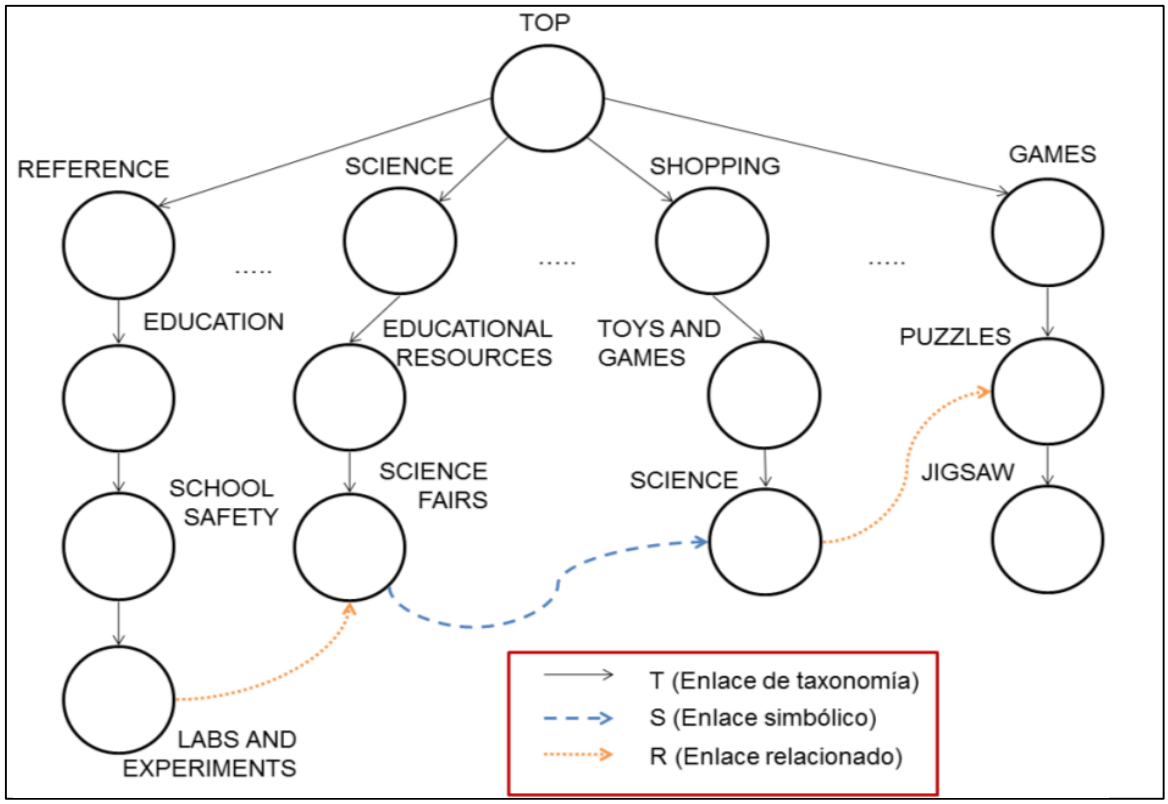


Figura 5-1: Porción del grafo de la ontología de ODP.

5.2.1 Relevancia y Similitud Semántica: Distintos ámbitos de aplicación

La similitud semántica constituye una medida muy confiable acerca de cuan relevante es un objeto de información respecto a otro. Hoy en día, podemos encontrar diversas aplicaciones de esta medida sobre ontologías, como por ejemplo la evaluación automática de estudiantes en exámenes de respuestas cortas [76] o la desambiguación del sentido de las palabras mediante el uso de un enfoque basado en el conocimiento [77]. Dentro de áreas más específicas del conocimiento, tales como la biología o la medicina, técnicas de cálculo de valores de similitud semántica y herramientas de minería de texto pueden ser útiles para la determinación de patrones de interacción entre entidades, como por ejemplo Interacciones Proteína-Proteína (PPI) ([78], [79]), relaciones y tratamientos de enfermedades humanas [80], como pueden ser los desórdenes del habla como la afasia

anómica [81], y la generación automática de anotaciones en ontologías como GO [82]. Otros enfoques para el cómputo de medidas de similitud semántica en ontologías biológicas fueron propuestos en [35] y [83].

En otro ámbito de aplicación, algunos lenguajes para ontologías como OWL[2] han sido utilizados para clasificar servicios web. Tomando ventaja de estas estructuras, algunos autores han propuesto implementaciones de esquemas semánticos [84] y pareamiento de servicios (service matching) [85], como también técnicas para cómputo de similitud entre documentos XML [86]. Las medidas de similitud semántica también han sido propuestas en el área de procesos de negocio, con aplicaciones a problemas específicos como se describe en [87] y [88].

Existen otras ontologías pensadas para diferentes propósitos, como búsqueda geográfica [89], búsqueda sobre bibliotecas digitales [90] y recomendación [91]. Los modelos de propagación de relevancia que son propuestos en el próximo capítulo y el cálculo de medidas de similitud semántica pueden ser naturalmente aplicados a estas ontologías. También Wikipedia[3] puede ser útil como una fuente para derivar relaciones semánticas, como se propone en [92] y [93]. Otro ejemplo de una medida de similitud semántica es SimRank [94], la cual es implementada y mejorada en trabajos subsiguientes ([95], [96], [97]).

Como no podía ser de otra manera, las redes sociales también han motivado diversos estudios basados en medidas de similitud semántica, lo cual puede verse en los trabajos de [98], [99], y [100]. Estudios adicionales de similitud semántica en ontologías pueden ser encontrados en [101], [102], [103] [104] [105], [106], [107], [108] y [22].

[2] http://www.w3.org/2001/sw/wiki/OWL
[3] http://es.wikipedia.org/

Un tema que no escapa del ámbito de la relevancia y la similitud semántica es la Recuperación de Información. Mediante la implementación de medidas de similitud o esquemas de relevancia es posible definir métricas más adecuadas de valores de *precisión* y *exhaustividad* (*recall*), los cuales reflejan el desempeño general de un entorno de Recuperación de Información [109]. También los procesos de Minería de Datos pueden ser mejorados mediante la aplicación de estos conceptos [110].

5.2.2 Tratamiento de Grandes Volúmenes de Información mediante grafos

La representación de corpus de documentos por medio de grafos puede ser muy adecuada para muchos propósitos. Por ejemplo, en [111] se efectúa el cálculo de similitud semántica mediante la utilización de este concepto. También, en [112] se lleva a cabo un trabajo sobre compresión y representación de grandes conjuntos de información en la estructura de la web, y en [10] se realizan variados análisis respecto a la organización de internet y su macroestructura de conectividad.

La identificación de comunidades temáticas en grandes volúmenes de información puede ser un punto de partida para trabajos de investigación de distintas clases. Los sistemas de recomendación, por ejemplo, pueden beneficiarse del uso extensivo que se puede hacer de estas comunidades formadas en los corpus ([113], [114], [115]). También hay variados casos de aplicación que pueden beneficiarse de la creación de tales comunidades, como por ejemplo sistemas exploradores o "*crawlers*" de la web ([116], [117], [118]) y redes sociales ([98], [100]).

Otros casos interesantes en los que es muy importante el tratamiento de estructuras de redes de información pueden ser el trabajo detallado en [119], en el cual se exploran las propiedades de grandes redes; también [120], donde se muestran redes de colaboración

sobre prácticas en temas particulares; adicionalmente [121], que explica el proceso de elaboración de una plataforma de búsqueda de archivos similares en un gran almacén de datos referentes a servicios a clientes; y por último [122], que destaca la búsqueda de código fuente en grandes repositorios, utilizando conceptos de la web semántica.

5.3 El Proyecto de Directorio Abierto

Existen innumerables sitios de internet que son visitados diariamente por millones de personas para incontables propósitos. Las necesidades de información de estos usuarios son muy variadas, y llevar a cabo la búsqueda de información sobre un tópico sería una tarea muy dificultosa si no existieran herramientas tales como los motores de búsqueda en internet o los directorios web, que ayudan a los usuarios en estos asuntos. Uno de estos directorios es el Proyecto de Directorio Abierto (ODP), el cual es ampliamente explotado por usuarios y proyectos de investigación. Algunos de los proyectos de investigación mencionados utilizan ODP para entrenar y evaluar clasificadores automáticos ([123], [124]), como un punto de partida para recolectar material temático en crawlers por tópicos ([118], [125]), como un marco de trabajo para entender la estructura de comunidades basadas en contenido en la web [126], para implementar plataformas de evaluación de Recuperación de Información ([127], [128]), para entender la evolución de comunidades en búsquedas punto a punto [115], para definir esquemas de propagación de pesos de palabras claves informados jerárquicamente [129], y para evaluar semánticas emergentes del etiquetado social en la web [130], entre otras aplicaciones.

5.3.1 Organización y representación del directorio

ODP es un proyecto abierto mantenido por miles de usuarios alrededor del mundo, que catalogan páginas en diferentes áreas de interés. Cada una de estas áreas identifica un

tópico, y estos tópicos están organizados dentro de una estructura de grafo. Los nodos de este grafo son los distintos tópicos de ODP, y las aristas son relaciones entre estos tópicos y, por consiguiente, entre las páginas que contienen. Estas relaciones pueden ser de tres tipos:

- Jerárquicas: Corresponden a la jerarquía principal del directorio, a través de la cual las categorías de sitios son organizadas.
- Simbólicas: Aristas del mismo nivel que las jerárquicas, concebidas para expresar las relaciones taxonómicas de algunas organizaciones alternativas del directorio. Esto sucede porque muchas veces podemos caracterizar más de un tópico como el antecesor inmediato del mismo subtópico, lo que podría ser imposible de lograr en una estructura puramente jerárquica.
- "Vease También": Aristas que conectan tópicos relacionados. Tienen un nivel de significación menor que las aristas de las otras dos clases.

En la Figura 5-2 puede observarse la estructura del sitio web de ODP. Por otro lado, en la Figura 5-1 se puede ver la estructura de una porción del grafo de ODP. En ese grafo, los enlaces jerárquicos son representados mediante una línea sólida negra, los simbólicos con líneas a rayas azules, y los "véase también" con líneas punteadas color naranja.

Para identificar unívocamente un tópico determinado en la estructura de ODP, se utiliza la rama completa que lo contiene en la jerarquía principal, sin incluir el tópico TOP, ya que este contiene a todos los demás. Así, si quisiéramos representar el tópico "Ferias de ciencias" (Science Fairs) dentro de la rama de "Recursos educacionales" (Educational Resources), y dado que este último está dentro del tópico "Ciencia" (Science), la forma de indicarlo sería SCIENCE / EDUCATIONAL RESOURCES / SCIENCE FAIRS. Como se

ve en esta nomenclatura, se utilizan barras para separar los tópicos en la ruta correspondiente.

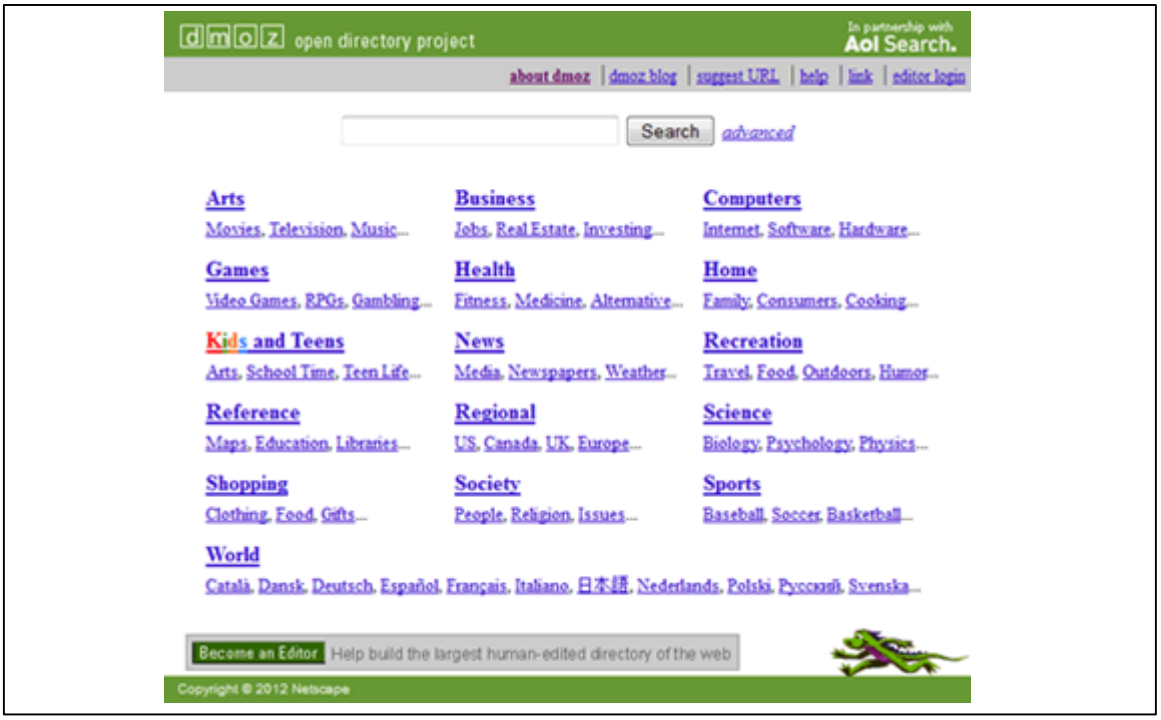


Figura 5-2: Sitio web de ODP.

Debido a la cantidad de páginas web que existen en el directorio, y como consecuencia la cantidad de tópicos presentes, el grafo y la matriz correspondiente a su representación son extremadamente grandes. Para representar una versión de ODP, fueron utilizadas matrices con una dimensión cercana a las 600.000 filas por 600.000 columnas.

5.3.2 Representación de la Estructura de Grafo de un Directorio Web

Un grafo de un directorio web es un grafo dirigido de nodos que representan tópicos. Cada nodo contiene objetos que representan documentos -páginas web-. Tiene una componente jerárquica -árbol- compuesta por enlaces del tipo *"es un"* y componentes no jerárquicas con aristas de cruce de distintas clases.

Por ejemplo, la ontología de ODP es un grafo dirigido *G=(V,E)*, donde *V* es un conjunto de nodos que representan tópicos que contienen documentos, y *E* es un conjunto de aristas entre los nodos de *V*, particionado en tres subconjuntos, *T*, *S* y *R*, tales que *T* corresponde a la componente jerárquica de la ontología, *S* corresponde a la componente no jerárquica y está formada por enlaces del tipo *"simbólicos"*, y *R* corresponde a la componente no jerárquica formada por enlaces del tipo *"relacionados"*.

La Figura 5-1 muestra un ejemplo simple de un grafo de un directorio web extraído de ODP. En este grafo, el conjunto *V* contiene nodos de tópicos tales como *REFERENCE*, *EDUCATION*, *SCHOOL_SAFETY*, *LABS_AND_EXPERIMENTS*, y demás tópicos. El subconjunto *T* correspondiente a la componente jerárquica del grafo del directorio web contiene aristas tales como (*TOP*, *REFERENCE*), (*REFERENCE*, *EDUCATION*), (*EDUCATION*, *SCHOOL_SAFETY*), y demás. En este ejemplo, hay una arista correspondiente a un enlace *"simbólico"* (*SCIENCE_FAIRS*, *SCIENCE*) y dos aristas de enlaces *"relacionados"*, (*LABS_AND_EXPERIMENTS*, *SCIENCE_FAIRS*) y (*SCIENCE*, *PUZZLES*).

5.3.3 Una Herramienta de Visualización de la Estructura Jerárquica de ODP

Dentro del marco de trabajo presentado aquí, fue desarrollada una herramienta de software que permite la visualización de la estructura jerárquica de ODP. La necesidad de su desarrollo surge del estudio de los volúmenes de datos que deben ser manipulados para generar los modelos de propagación de relevancia que serán descriptos en el capítulo 6 de esta tesis. Las diferentes clases de relaciones entre los tópicos de ODP se registran en matrices binarias o reales, las cuales indican la existencia de una relación entre tópicos mediante un valor distinto de 0 en la celda de la matriz que tiene como coordenadas a los

números identificadores de los tópicos en cuestión (para mayor detalle, véase sección 6.5). Como consecuencia de la utilización de tales matrices, frecuentemente se hace necesario determinar la ruta completa o el nombre de un tópico del cual solamente sabemos su número de identificación, luego de encontrar una relación. Es por este motivo que fue desarrollada la herramienta mencionada, la cual permite hallar rápidamente un tópico y todos los tópicos de la rama que lo contiene en la jerarquía, solamente ingresando su número identificador.

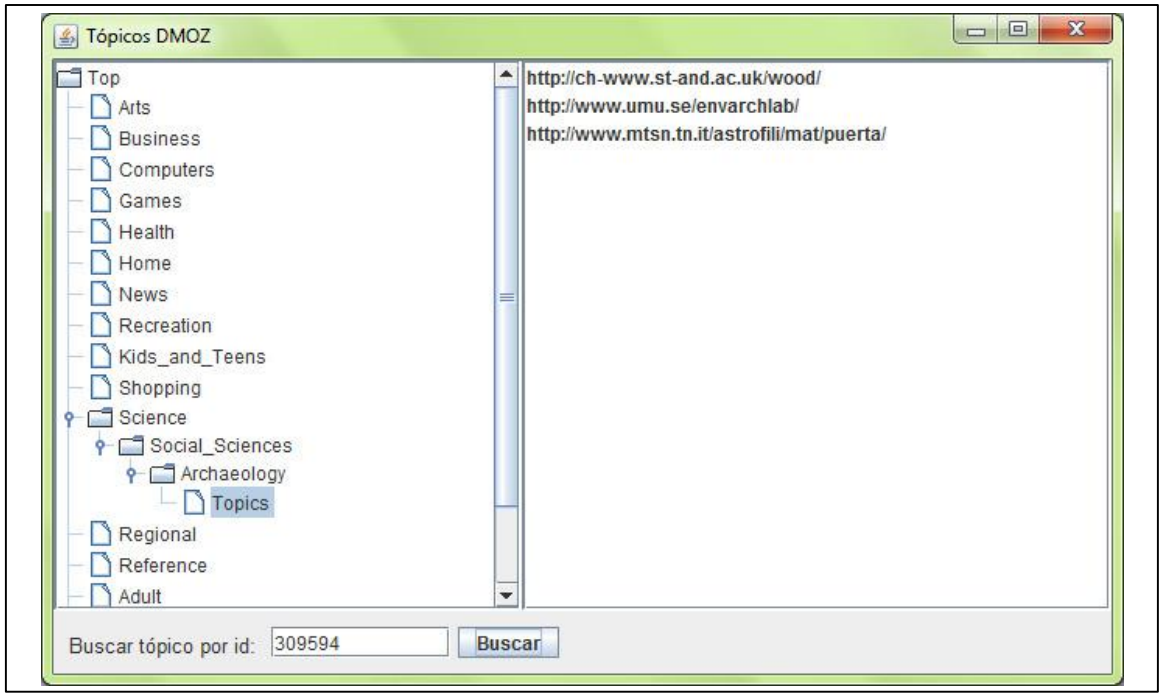


Figura 5-3: Herramienta de visualización de la estructura jerárquica de ODP.

Como puede observarse en la Figura 5-3, la herramienta nos ofrece una sección en la cual se visualiza la estructura jerárquica de los tópicos del directorio en el lado izquierdo, y del lado derecho se detallan los enlaces a sitios web contenidos en el tópico seleccionado de la jerarquía. Además, la característica más importante y útil para los fines de este trabajo se encuentra en la parte inferior de la ventana. Allí se observa un cuadro de texto en el que se puede ingresar el número identificador de algún tópico. Al presionar el botón que se

encuentra al lado de este cuadro, la herramienta efectúa la búsqueda del tópico correspondiente y sus sitios web asociados y expande la rama completa del mismo, en el árbol de la izquierda de la ventana. También, los enlaces correspondientes al tópico son listados en el sector derecho de la ventana. Estas características permitieron llevar a cabo de forma más eficiente las selecciones de tópicos candidatos para los experimentos explicados en el capítulo 7 de esta tesis.

Un aspecto importante en la construcción de esta herramienta fue el armado de una base de datos relacional, para alojar la información de los tópicos y sus enlaces. La fuente de información primaria consistía en archivos de texto con los números identificadores de los tópicos y su ruta, y también los enlaces con su tópico asociado y otro número identificador propio. También fue necesario automatizar la generación de esta base de datos mediante la programación de rutinas para tal fin. Con esta base disponible, se construyeron índices sobre las tablas de la misma para que las consultas fueran más eficientes. La herramienta de visualización fue desarrollada utilizando la base de datos construida como fuente de información para su funcionamiento.

En el futuro, se podría también desarrollar algunas características adicionales para esta herramienta, como por ejemplo la posibilidad de explorar jerarquías alternativas o la búsqueda textual de tópicos o enlaces.

5.4 Implementación de un algoritmo de grafos sobre la estructura de ODP

Existen diferentes formas para identificar grupos de objetos relacionados en un gran volumen de datos. Si logramos obtener una representación mediante grafos dirigidos para estas grandes estructuras, una manera muy común para detectar agrupaciones de objetos de información relacionados es la detección de componentes fuertemente conexas. Por

ejemplo, mediante la utilización del algoritmo definido en [20] y explicado en la sección 2.2.3 de esta tesis, fueron llevados a cabo diversos estudios con excelentes resultados sobre la estructura de internet en general en [10]. El objetivo del trabajo explicado en esta sección es la aplicación del mismo algoritmo para hallar comunidades temáticas sobre el grafo de ODP, entendiendo las mismas como grupos de tópicos relacionados entre sí no sólo por la existencia de aristas que los unen directamente, y con ello lograr modelos de propagación de relevancia adicionales a los obtenidos en [18].

5.4.1 Nueva implementación del algoritmo de detección de componentes fuertes

La librería utilizada para la detección de componentes fuertes sobre ODP está basada en la misma librería que se utiliza en los métodos de particionamiento descriptos en la Parte I de esta tesis. Debido a que dicha implementación no permitía el tratamiento de estructuras de datos muy grandes, fue necesario reescribir el algoritmo utilizando nuevas estructuras. La cantidad de nodos de los grafos dirigidos que fueron objeto de las secciones correspondientes del MD, el MDE y el MDE mejorado no sobrepasaba los 2000, mientras que el grafo de ODP contiene aproximadamente 600.000 nodos sólo en la versión analizada en este trabajo.

Ya que la limitación de la rutina original de detección de componentes fuertes existía en la cantidad de memoria RAM disponible, en la nueva rutina se crearon nuevas estructuras de datos que utilizaban al disco duro como medio de almacenamiento temporal. Si bien al tratar volúmenes de datos muy extensos esto podría comprometer el rendimiento del programa por la utilización de ese medio de almacenamiento, el límite para el tamaño de las estructuras es simplemente la capacidad del mismo disco duro. De esta forma, se

logró superar la dificultad planteada y efectuar la detección de componentes fuertes sobre la estructura de grafo de ODP.

5.4.2 Componentes fuertes encontradas para el grafo de ODP

El algoritmo de detección de componentes fuertes implementado en este trabajo fue ejecutado para la versión del grafo de ODP que también fue objeto de estudio de los capítulos 5, 6 y 7 de esta tesis. Los resultados de esta sección fueron publicados en [19]. En la Tabla 5-1 se resumen los resultados obtenidos. Se puede ver allí el número de componentes fuertes halladas agrupadas por la cantidad de nodos que las componen. Por ejemplo, existen 12 componentes fuertes de tamaño 22, y 8 de tamaño 25, mientras que no se halló ninguna con 35 nodos exactamente.

Tamaño	Cantidad	Tamaño	Cantidad	Tamaño	Cantidad	Tamaño	Cantidad
1	260702	16	18	31	4	60	1
2	5305	17	12	32	2	64	1
3	1524	18	12	33	2	70	1
4	632	19	6	34	1	78	1
5	381	20	8	37	1	81	1
6	225	21	8	41	2	83	2
7	167	22	12	42	1	85	1
8	98	23	5	43	1	86	1
9	70	24	7	44	2	89	2
10	77	25	8	45	1	98	1
11	43	26	1	46	1	101	1
12	48	27	3	48	1	123	1
13	28	28	3	49	1	279519	1
14	31	29	4	51	1		
15	25	30	4	54	1		

Tabla 5-1: Cantidad de componentes fuertes halladas para cada tamaño.

Para ampliar la información contenida en la Tabla 5-1, el gráfico de la Figura 5-4 muestra la relación existente entre el tamaño de componente expresado en cantidad de nodos que incluye, y la cantidad de componentes encontradas para cada tamaño. Dicho

gráfico está expresado en escala doble logarítmica, y muestra una posible relación funcional entre la frecuencia y el tamaño de las componentes fuertes. Para evaluar si realmente puede existir, por ejemplo, una distribución potencial de frecuencias por tamaños, deberían ser llevados a cabo diversos estudios estadísticos.

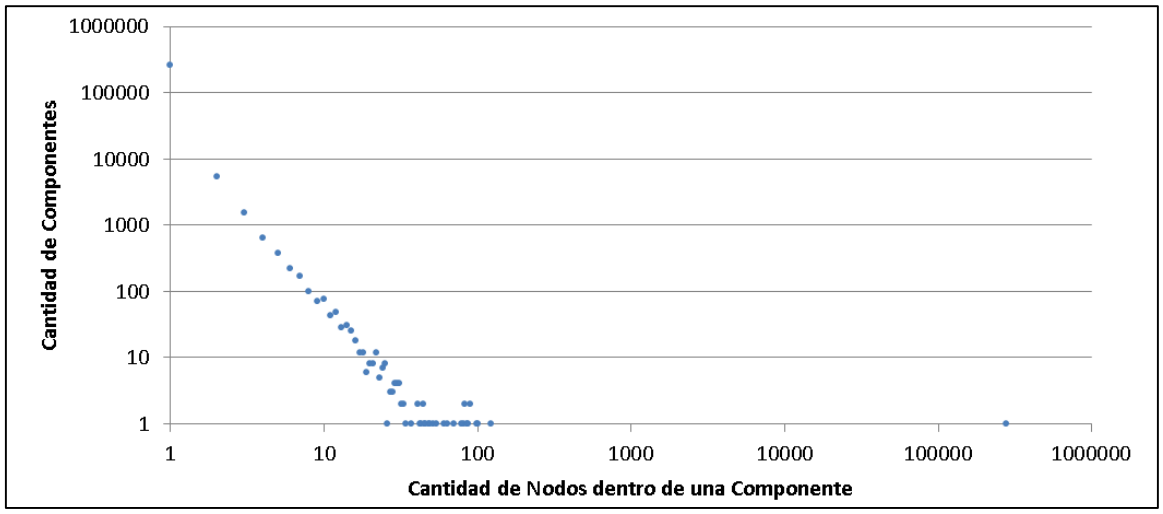


Figura 5-4: Gráfico de cantidades de componentes fuertes por tamaño, en escala doble logarítmica.

Otro resultado importante es la existencia de una gran componente de 279.519 nodos, cerca de la mitad de todos los nodos del volumen. Es muy probable que algunas de las relaciones inducidas por esta componente no sean coherentes, ya que las relaciones se establecen vinculando a cada nodo con todos los demás dentro de cada componente.

En contraste con la gran componente descripta se encuentran numerosos nodos aislados, es decir, componentes fuertes de un solo nodo, totalizando las 260.702. Al igual que para la gran componente, sería de interés efectuar algún análisis sobre las posibles relaciones existentes entre cada uno de estos nodos aislados y otros nodos correspondientes a diferentes tópicos. La única razón por la que estos nodos no están dentro de otras componentes es que el algoritmo no encontró un ciclo que los contenga.

Tal vez con la simple incorporación de algún enlace adicional, los nodos aislados pueden incorporarse a alguna componente fuerte, y si quitamos alguna arista dentro de la

componente fuerte más grande hallada, se pueda dividir esta en componentes más pequeñas con relaciones más significativas. El criterio utilizado para agregar o eliminar una arista podría estar asociado con el nivel de importancia de los distintos tipos de arista, como por ejemplo dar mayor importancia a enlaces de la jerarquía y menor importancia a los enlaces del tipo "*véase también*".

Id	Tópico
502266	Top/Computers/Software/Typesetting/TeX/Plain_TeX/Macros
155677	Top/Science/Math/Publications/Style_Files
300975	Top/Computers/Software/Typesetting/TeX/Macros
4795	Top/Computers/Software/Typesetting/TeX/LaTeX/Macros

Tabla 5-2: Tópicos de una componente fuerte de ODP.

Para evaluar algunos ejemplos de las relaciones entre tópicos surgidas por la detección de componentes fuertes, la Tabla 5-2 y la Tabla 5-3 muestran la información de los tópicos contenidos en dos de las componentes fuertes halladas. La Tabla 5-2 describe un conjunto de tópicos que en principio podrían estar muy relacionados entre sí. El descubrimiento de la relación entre los tópicos COMPUTERS / SOFTWARE / TYPESETTING / TEX / PLAIN_TEX / MACROS y SCIENCE / MATH / PUBLICATIONS / STYLE_FILES, proveniente de la componente fuerte mencionada, agrega información significativa al modelo del grafo de ODP, porque la relación es coherente y no estaba contemplada en el modelo original de ODP. Por otro lado, podemos ver en la Tabla 5-3 dos tópicos que parecerían no estar relacionados como son LOUISIANA / TRAVEL_AND_TOURISM / TRAVEL_SERVICES / TOUR_OPERATORS y LOUISIANA / RECREATION_AND_SPORTS / FISHING_AND_HUNTING / GUIDES_AND_CHARTERS, pero que de acuerdo a la componente que los contiene tendrían una relación, lo cual aportaría información inadecuada al modelo.

Id	Tópico
91990	Top/Regional/North_America/United_States/Louisiana/ Travel_and_Tourism/Travel_Services/Tour_Operators
380010	Top/Recreation/Outdoors/Guides_and_Outfitters/ North_America/United_States/Louisiana
221405	Top/Recreation/Outdoors/Hunting/Guides_and_Outfitters/ North_America/United_States/Louisiana
34931	Top/Regional/North_America/United_States/Louisiana/ Recreation_and_Sports/Fishing_and_Hunting/Guides_and_Charters

Tabla 5-3: Tópicos de una componente fuerte de ODP.

5.5 Conclusiones

Este capítulo introduce el tema del análisis estructural en Ciencias de la Información, teniendo en cuenta para ello al gran directorio del proyecto ODP. Primero se explica el concepto de ontología informática, y se enumeran trabajos de investigación que hacen uso de este tipo de representación, enfocándose sobre problemas relacionados con la relevancia entre objetos de un corpus de datos o el cálculo de distintas medidas de similitud entre los mismos, con el objetivo de hallar información valiosa para un usuario. Luego, se muestra una descripción del proyecto ODP y una herramienta de visualización de su estructura, para terminar con la implementación de un algoritmo que permite obtener información valiosa sobre las relaciones existentes entre sus tópicos.

Tanto la información contenida en el grafo de ODP como las herramientas desarrolladas en esta sección constituyen una base fundamental para el trabajo que se describe en los siguientes dos capítulos de esta tesis. Las representaciones del directorio ODP y las componentes fuertes halladas constituyen el material de trabajo que permitió llevar a cabo distintos experimentos para determinar la validez de los modelos de propagación de relevancia que se enumeran en el capítulo 6 del presente trabajo.

Capítulo 6:

Propagación de Relevancia

6.1 Introducción

Luego de definir en el capítulo anterior las cuestiones fundamentales acerca del funcionamiento y la representación del proyecto ODP, los aspectos relacionados a la relevancia entre tópicos y su propagación son definidos aquí. El concepto de relevancia es muy útil para cuantificar medidas de relación entre distintas entidades de un gran corpus de datos. Por ejemplo, la similitud semántica entre documentos de un directorio puede ser definida en base a esquemas de propagación de relevancia.

La información que se obtiene mediante el uso de los esquemas mencionados y las medidas que de esto se generan, pueden derivar en resultados más precisos sobre búsquedas de información o consultas de usuarios en distintos contextos. Para lograr este objetivo, en este capítulo se describe la construcción de numerosos modelos de propagación de relevancia sobre ODP, los cuales hacen uso de distintas operaciones matriciales y algoritmos sobre los grafos que representan la estructura de este directorio.

La primera sección del capítulo traza los objetivos perseguidos por la determinación de la propagación de relevancia. Después se define más profundamente el concepto de relevancia, asociando este concepto con la teoría de probabilidad. Luego se enumeran trabajos relacionados, se explica el mecanismo de construcción de los distintos modelos de propagación de relevancia, y por último se expresan las conclusiones del capítulo.

6.2 Relevancia, búsqueda de información y directorios web

Un directorio web es un conjunto de páginas web clasificadas por tópicos o categorías temáticas. Esta clase de directorios consisten en ontologías de tópicos para discriminar las categorías en que serán incluidas las páginas. Ejemplos de estos directorios pueden ser el Directorio Yahoo! y el Proyecto de Directorio Abierto (Open Directory

Project, ODP)[4]. Aun cuando una búsqueda regular en la web es la manera más común adoptada por los usuarios para encontrar información relacionada a un tópico específico, los directorios web son particularmente útiles para la navegación entre tópicos relacionados, o cuando los usuarios no están seguros de cómo se puede reducir el espectro de búsqueda dentro de una categoría muy amplia. Las ontologías de tópicos pueden ayudar a los usuarios a entender cómo están relacionados los tópicos dentro de un área específica y también sugerir términos que sean útiles para conducir una búsqueda. Además de estar organizadas por tópicos, las páginas web clasificadas en estas ontologías tienen las ventajas de tener anotaciones -que funcionan como una descripción- y haber sido evaluadas por un editor. ODP, por ejemplo, tiene cerca de 90.000 editores voluntarios, los cuales llevan a cabo revisiones de sitios y los clasifican por tópico.

Aunque los directorios web fueron originalmente concebidos como una forma de organizar páginas web para facilitar la navegación por usuarios humanos, el contenido y la estructura de estos directorios están siendo usados cada vez más frecuentemente para otros propósitos. Por ejemplo, los resultados de búsquedas web regulares en Google son mejorados con información del directorio web de Google. Algunos usos del directorio ODP fueron enumerados en la sección 5.3. Muchos de esos usos se sustentan en la identificación de relaciones de relevancia o similitud semántica entre páginas web clasificadas en ODP.

Un análisis inicial del problema de la definición de la relevancia entre documentos clasificados en una ontología de tópicos sugiere que es necesario abordar el problema de identificar relaciones no obvias en la estructura de la ontología. El hallazgo de estas relaciones en ontologías de tópicos es un problema desafiante. La estructura de las ontologías es típicamente no plana, ya que los conceptos o tópicos pueden estar clasificados

[4] http://www.dmoz.org/

de acuerdo a algún esquema taxonómico. Las taxonomías tópicas contienen relaciones padre-hijo entre tópicos y sus subtópicos. De todas formas, también son muy comunes las relaciones que van más allá de las jerarquías padre-hijo. Por ejemplo, la ontología de ODP es más compleja que un simple árbol. Algunos tópicos tienen múltiples criterios para clasificar subtópicos. La categoría "Negocios", por ejemplo, está subdividida por tipos de organizaciones (cooperativas, pequeños negocios, grandes compañías, etc.) así como por áreas (automotores, cuidado de la salud, telecomunicaciones, etc.). Más aun, ODP tiene varios tipos de enlaces de referencia cruzada entre categorías, por lo que un nodo puede tener muchos nodos padres, e incluso existen ciclos.

La combinación de diferentes tipos de enlaces da origen a relaciones intrincadas entre tópicos. Aunque algunas de estas relaciones son observables explícitamente por la existencia de los enlaces correspondientes, la mayoría de ellas permanecen implícitas. Actualmente, ODP contiene más de un millón de categorías, haciendo el problema de derivar automáticamente relaciones entre tópicos implícitas computacionalmente muy dificultoso.

Es posible definir diferentes mecanismos para derivar relaciones de relevancia implícitas, dando origen a modelos computacionales de propagación de relevancia múltiples. Una vez que las relaciones de relevancia son derivadas, pueden ser definidos otros conceptos importantes, tales como medidas de similitud semántica entre tópicos o documentos en una ontología, el grado de utilidad de un documento para un contexto temático, o relaciones de incumbencia entre consultas y tópicos. En particular, algunas medidas de desempeño para Recuperación de Información ampliamente adoptadas, como *precisión* y *recall* [131], son definidas en términos de la relevancia.

6.3 Contexto

Tradicionalmente, la noción de relevancia ha sido estudiada en el contexto de la Teoría de Probabilidad. En los primeros intentos de formalizar la relevancia, tal noción fue tomada como equivalente a la noción de dependencia condicional, y luego esto fue refinado principalmente por Keynes (1921), Carnap (1950) y Gärdenfors (1978) (citados en [132]). Una definición formal de relevancia basada en el uso de una medida probabilística puede ser expresada del siguiente modo:

Definición 6-1: Una fórmula α es relevante para una fórmula β dada una base de conocimiento K sí y solo sí

$$P_{\mathrm{K}}(\beta|\alpha) > P_{\mathrm{K}}(\beta)\ siempre\ que\ \ P_{\mathrm{K}}(\alpha) \neq 0$$

Donde P_K representa una medida de probabilidad para la base de conocimiento K.

En principio, adaptar esta definición para determinar cuándo un tópico t_i es relevante para un tópico t_j en una taxonomía tópica T parece ser muy sencillo. La reformulación de esta definición simplemente involucrará determinar si la probabilidad de clasificar un documento bajo el tópico t_j crece si aprendemos que el documento pertenece al tópico t_i.

Definición 6-2: Un tópico t_i es relevante para un tópico t_j dada una taxonomía tópica T sí y solo sí:

$$P_{\mathrm{T}}(t_j|t_i) > P_{\mathrm{T}}(t_j)\ siempre\ que\ \ P_{\mathrm{T}}(t_i) \neq 0$$

Donde P_T representa una medida de probabilidad para la taxonomía T.

Dada una taxonomía tópica T, podemos asumir que P_T (t_j) representa la probabilidad anterior de que cualquier documento sea clasificado bajo el tópico t_j. En la práctica, $P_T(t_j)$ puede ser computada para cada tópico t_j en una taxonomía de tipo "es un" mediante la

cuenta de la fracción de documentos almacenados en el nodo t_j y sus descendientes respecto de todos los documentos dentro de la taxonomía. La probabilidad condicional $P_T(t_j|t_i)$ representa la probabilidad de que cualquier documento sea clasificado bajo el tópico t_j dado que ya fue clasificado bajo el tópico t_i, y se computa contando la fracción de documentos almacenados en el nodo t_j y sus descendientes respecto de todos los documentos almacenados en el tópico t_i y sus descendientes. En otras palabras, $P_T(t_j|t_i)$ es la fracción de documentos en el subárbol con raíz en t_i que pertenece al subárbol con raíz en t_j. Por ejemplo, si el tópico *BONSAI_AND_SUISEKI* es un subtópico del tópico *GARDENS* (Figura 6-1), entonces la probabilidad de clasificar un documento *d* bajo el tópico *BONSAI_AND_SUISEKI* es más alta si sabemos que *d* está clasificado bajo el tópico más general *GARDENS* que si no fuera adelantada ninguna evidencia.

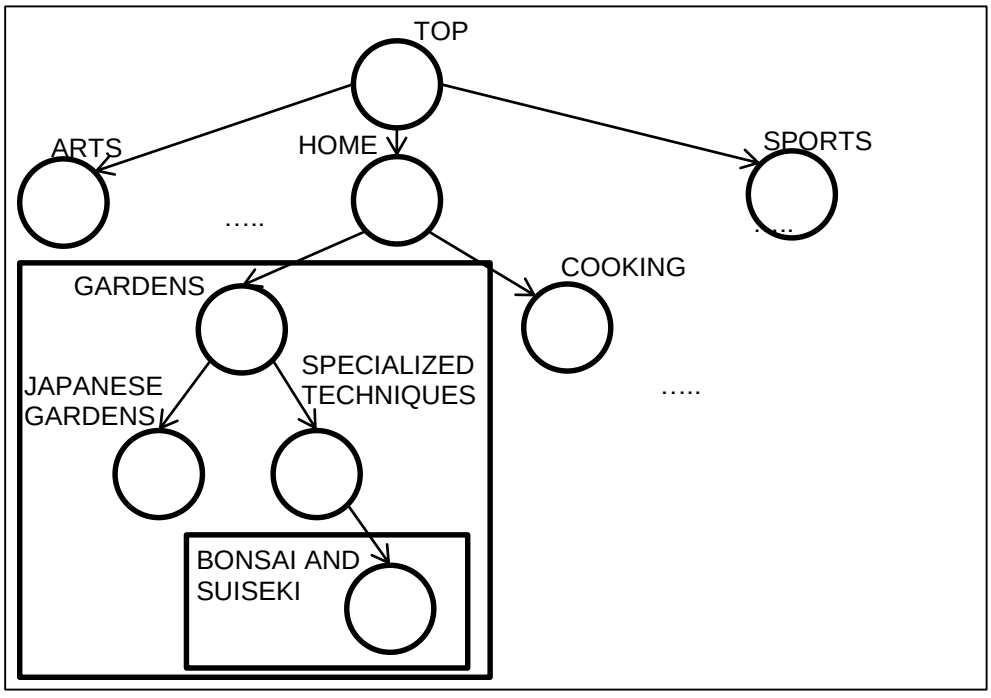


Figura 6-1: Ilustración de una porción de una taxonomía tópica.

Una limitación importante de la Definición 6-2 es que no se puede aplicar directamente a ontologías tópicas en general, tales como ODP, que son más complejas que

un simple árbol. Dada una ontología tópica φ, la principal dificultad para aplicar esta definición subyace en el cómputo de $P_\varphi(t_j)$ y $P_\varphi(t_j|t_i)$, ya que no es suficiente la cuenta del número de documentos almacenados en los subárboles con raíz en t_i y t_j para estimar estas probabilidades. Para ilustrar este problema, tomemos por ejemplo los tópicos *TOYS_AND_GAMES* y *PUZZLES* en la ontología de la Figura 6-2. Aunque existe una clara relación de relevancia entre estos dos tópicos, sus subárboles correspondientes son independientes.

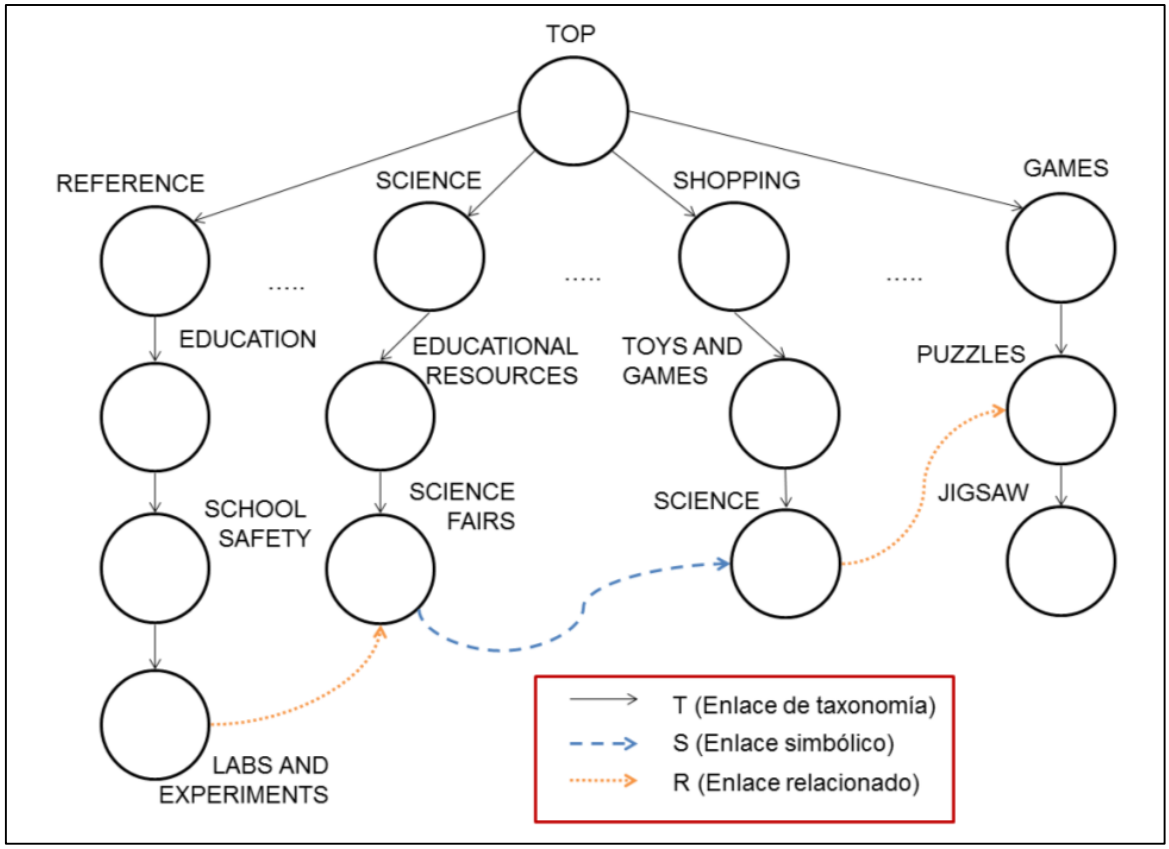


Figura 6-2: Ilustración del grafo de un directorio web extraído de ODP.

En una ontología general de tópicos, computar $P_\varphi(t_j|t_i)$ no solamente involucraría reconocer si existe una relación de "descendiente" o "ancestro" entre t_i y t_j, sino que también significaría determinar si el hecho de que un documento esté relacionado al tópico t_i tendría un impacto en la determinación de si el documento es relativo al tópico t_j. En otras

palabras, necesitamos establecer en primer lugar si t_i es relevante para t_j para computar $P_\varphi(t_j|t_i)$. Por lo tanto, para el caso de una ontología general, la definición tradicional de relación de relevancia se vuelve circular.

La discusión anterior apunta a la idea de que la definición de estas probabilidades en términos de la relevancia es más natural que la definición de relevancia en términos de medidas de probabilidad. Desde una perspectiva cognitiva, normalmente es más fácil arribar a una relación de relevancia que estimar valores de probabilidad. Además, aun si los valores de probabilidad son provistos con anterioridad, es posible llegar a conclusiones erróneas debido a "errores puramente numéricos" (R. von Mises, 1963, citado en [133]).

Para superar las dificultades anteriormente mencionadas, asumiremos que *la relevancia es una noción conceptual primitiva*. Esta noción no solamente capturará las relaciones del tipo "es un" derivadas de una ontología jerárquica, sino que también tomará en consideración las componentes no jerárquicas. La extensión de la noción de relevancia de taxonomías a grafos de ontologías da origen a la pregunta de cómo extender la definición de subárbol con raíz en un tópico para el caso de un grafo.

Un "enfoque audaz" significaría formular que t_i es relevante para t_j si existe un camino dirigido en el grafo de la ontología desde t_i a t_j. De todas formas, como se analizará luego, esta formulación de relevancia de tópicos es inexacta ya que la introducción de muchos enlaces cruzados en un camino puede llevar a una pérdida de significado. En suma, admitir múltiples enlaces cruzados es infactible porque llevaría a una relación de relevancia muy densa, lo cual quiere decir que cada tópico se convertiría en relevante para casi todos los demás tópicos. Esto tampoco es robusto debido a que unos pocos enlaces cruzados no fiables provocarían cambios globales muy significativos en un esquema de propagación de

relevancia de estas características. Este capítulo está dedicado al análisis de las fortalezas y limitaciones de enfoques "más cautelosos" para establecer la propagación de relevancia.

6.4 Trabajos relacionados

La relevancia es un concepto muy poderoso utilizado en varias subdisciplinas dentro de las ciencias de la computación, especialmente en inteligencia artificial y ciencias de la información. En esta sección se realiza una revisión de diferentes enfoques para caracterizar y aplicar la relevancia, y más específicamente la propagación de relevancia en el ámbito del manejo del conocimiento, la minería de datos, y la recuperación de información.

6.4.1 El estudio de la Relevancia como un Aspecto Clave en Ciencias de la Información

Ha habido una gran diversidad de esfuerzos por estudiar y caracterizar la noción de relevancia en ciencias de la información. La mayoría del trabajo de investigación se centra en la definición de relevancia entre tópicos, con el propósito final de formular métricas para evaluar la efectividad de sistemas de recuperación de información. Un antiguo trabajo [134] define relevancia como la medida de información transmitida por un documento para una consulta.

Aunque la estructura de tópicos ha sido la base para las afirmaciones sobre relevancia en la mayoría de las propuestas existentes -al igual que en esta tesis-, algunos estudios han destacado que puede ser inadecuado tomar esta estructura como único ingrediente para elaborar juicios de valor al respecto. Por ejemplo, en [135] replican que la definición de relevancia debería tomar en cuenta conceptos como la información provista por un documento, el conocimiento previo del usuario, y la utilidad real de la información

para el usuario. Siguiendo esta postura, en [136] se resaltan muchos criterios centrados en el usuario que afectan a las opiniones sobre relevancia. Estos criterios incluyen el contenido de información del documento, el conocimiento previo del usuario, las preferencias del usuario, otra información y otras fuentes dentro del ambiente, las fuentes del documento, el documento como una entidad física, y la circunstancia del usuario. Un trabajo más reciente [137] propone una discusión sobre cinco factores que afectan a la relevancia: "topicalidad", novedad, confiabilidad, "entendibilidad", y ámbito. Después de completar un estudio de usuarios, los autores notaron que la topicalidad y la novedad resultaron ser los criterios más importantes en lo que respecta a la relevancia.

Una revisión más extensiva de la literatura existente sobre el concepto de relevancia está fuera del alcance de este trabajo. Más información puede hallarse en [138], que brinda un panorama de la historia de la relevancia en el campo de las ciencias de la información desde la década de 1930 hasta 1997. Revisiones más recientes pueden ser encontradas en [139], [140] y [141].

Aunque la noción de relevancia ha sido el foco de muchos estudios en ciencias de la información, la noción de propagación de relevancia sólo ha sido parcialmente estudiada. La propagación de relevancia se torna fundamental en la presencia de estructuras interconectadas tales como subgrafos de la web, ontologías, grafos de citas bibliográficas, y redes sociales en general. En particular, la noción de propagación de relevancia es esencial para computar relaciones semánticas entre nodos ordenados en cualquier clase de red. Las secciones siguientes muestran una revisión de trabajos de investigación que conducen estos aspectos.

6.4.2 Similitud Semántica en Ontologías

Aun manteniendo la idea de que la noción de relevancia es más primitiva que la noción de similitud semántica y que la última puede ser definida en términos de la primera, ambas nociones son utilizadas de forma alternada a menudo en la literatura, bajo el concepto general de "relación semántica". Algunos enfoques orientados al cómputo de medidas de similitud semántica entre nodos en una ontología toman una representación de red ignorando la estructura taxonómica de la correspondiente ontología. Propuestas anteriores han usado la distancia de caminos entre nodos de la red (por ejemplo, [142]). Estos marcos de trabajo están basados en la premisa de que mientras más fuerte sea la relación semántica entre dos objetos, más cerca estarán estos dos objetos entre sí en la representación de red. Sin embargo, como ha sido discutido por diversos autores, los problemas surgen cuando se intenta aplicar esquemas basados en distancia para medir la similitud entre objetos en ciertas clases de redes donde los enlaces pueden no representar distancias uniformes ([143], [144] y [145]). Además, algunos autores han argumentado que el uso de distancias no es apto para computar similitud o relevancia. Esto es debido principalmente al hecho de que algunas propiedades que deberían mantenerse en espacios métricos no son válidas para medidas de similitud o relevancia. Tomemos, por ejemplo, la *desigualdad del triángulo*, que es una propiedad definitoria de los espacios métricos. La desigualdad del triángulo implica que si *a* es muy similar a *b*, y *b* es muy similar a *c*, entonces *a* y *c* no pueden ser muy distintos -o no similares- uno del otro. El siguiente ejemplo (basado en William James, citado en [146], p. 329) ilustra la inadecuación de este supuesto: *"Jamaica es similar a Cuba (por su proximidad geográfica); Cuba es similar a Rusia (por su afinidad política); pero Jamaica y Rusia no son similares en nada"*. Este

ejemplo encaja con el caso de las páginas web y sus tópicos, sugiriendo que la desigualdad del triángulo no debería ser aceptada como una piedra angular en modelos de similitud o relevancia.

Otro problema asociado con la aplicación de enfoques basados en la distancia para computar relevancia o similitud es que en ontologías jerárquicas, tales como ODP, ciertos enlaces conectan categorías muy densas y generales, mientras que otros conectan algunas más específicas. Para conducir este problema, algunas propuestas estiman la similitud semántica en una taxonomía basándose en la noción del contenido de información ([147], [145]). En estos enfoques, el significado compartido por dos objetos puede ser medido por la cantidad de información necesaria para determinar lo que tienen en común esos objetos. Estas propuestas, sin embargo, están limitadas a taxonomías y, como consecuencia, no abordan la cuestión de cómo estimar relevancia y similitud semántica en ontologías generales.

El problema general de computar similitud semántica en ontologías generales tales como el grafo de ODP ha sido conducido primero por [148]. La medida de similitud semántica propuesta en ese trabajo toma ventaja tanto de los componentes jerárquicos (enlaces tipo “es un”) como de los no jerárquicos (enlaces cruzados) de la ontología. De todas maneras, se tomó un enfoque simplista de la propagación de relevancia, omitiendo un análisis profundo de la noción de relevancia y enfocándose solamente en la noción de similitud.

Los modelos computacionales de similitud semántica no necesitan estar limitados a ontologías tópicas y búsquedas web. La identificación de relaciones de distinto grado de afinidad entre nodos en otras ontologías requiere de mecanismos apropiados para modelar diferentes clases de componentes ontológicos y sus interacciones. Por ejemplo, la

Ontología Génica (Gene Ontology, GO) [149] tiene dos clases de enlaces jerárquicos ("es un" y "es parte de"). En contraste, la ontología *WordNet* [150] tiene una tipología de relaciones mucho más numerosa. Esta incluye relaciones semánticas entre *conjuntos de sinónimos* (*synsets*) tales como *hiperónimo*, *merónimo*, y *holónimo*, así como relaciones léxicas entre sentidos de palabras (*miembros de synsets*) tales como *antónimos*, *"véase también"*, *formas derivadas*, y *participio*.

6.4.3 Propagación de Relevancia para identificar Fuentes Autoridades de Tópicos

Una amplia variedad de modelos de propagación de relevancia ha sido aplicada para identificar fuentes autoridades en representaciones mediante grafos para diferentes dominios, donde los grafos podrían representar una red social de expertos, una porción de la web, una red de citas bibliográficas o cualquier clase de colección de documentos interconectados.

En el campo de la búsqueda de expertos (*expert finding*), se ha llevado a cabo la elaboración de un modelo de propagación de relevancia que consiste en construir una red social *ad-hoc* para una consulta dada [151]. El marco de trabajo referido propaga la relevancia a través de la red construida para identificar autoridades en los campos de experticia requeridos. En [49] se presenta una propuesta similar, en la que se usa un grafo hecho tanto de nodos de documentos como de expertos para identificar expertos por dominios. Esto se logra reconociendo nodos autoridades por medio de un modelo de propagación de relevancia.

Muchos enfoques aplican modelos de propagación de relevancia para identificar páginas web autoridades sobre algún tópico, un área de investigación conocida como

"depuración temática" (*topic distillation*). Por ejemplo, en [152] se usa un modelo tradicional de recuperación de información basado en similitud de enlaces y contenido para propagar la relevancia a lo largo de los hiperenlaces. De una forma similar, en [153] proponen un modelo de propagación de relevancia basado en contenido -y enlaces-, que es enriquecido iterativamente por información del comportamiento del usuario. Otro esquema para computar la autoridad temática de una página web es presentada en [154]. Este esquema utiliza el directorio ODP para construir un clasificador para páginas web arbitrarias, dando origen a un nuevo método para la propagación de autoridad dependiente de la relevancia temática entre las páginas conectadas.

Un método alternativo de depuración de tópicos que se sustenta tanto en información del contenido como de los enlaces es descripto en [155] y subsecuentemente refinado en [156]. En el último trabajo, la propagación de relevancia a través de los enlaces se basa en el agrupamiento de vecinos en clases. Un método similar se presenta en [157], en donde, en lugar de limitar el análisis a los hiperenlaces de un subgrafo de la web, se toma en cuenta toda la estructura de los mapas de sitios involucrados en el subgrafo.

6.4.4 Propagación de Relevancia en Ontologías

Más cercano a esta sección están aquellos marcos de trabajo que intentan propagar la relevancia entre ontologías de tópicos. Un modelo de propagación de relevancia en ontologías de tópicos que toma en consideración el contenido de los documentos se puede consultar en [158]. En dicho trabajo, se construye una ontología basada en la noción de la relevancia temática. La ontología resultante es utilizada luego para guiar un *crawler* -robot que recaba información de sitios- enfocado. La ontología evoluciona iterativamente, basada en una función de relevancia que trata de mapear el contenido de cada página web

descubierta a una clase en la ontología. La propagación de relevancia es llevada a cabo mediante la evolución de las clases que están en el vecindario de aquellas clases que han sido actualizadas.

Otro modelo de propagación de relevancia en ontologías tópicas puede verse en [129]. Allí se propone un algoritmo de propagación de palabras claves -*keywords*- para aumentar la descripción de las entradas en una jerarquía de navegación, mediante el agregado de información semántica suplementaria a dichas entradas. Para el caso particular de taxonomías tópicas, esta información es derivada de los nombres y descripciones de los ancestros y descendientes de los tópicos. El enfoque es generalizado luego de tal forma que las palabras clave pueden ser propagadas a lo largo de estructuras más complejas.

Los dos esquemas de propagación anteriores se relacionan con el enfoque adoptado en esta tesis en tanto que intentan modelar la propagación de relevancia dentro de ontologías temáticas. Sin embargo, a diferencia de este marco de trabajo, esas propuestas propagan contenido -por ejemplo, palabras clave o peso de las palabras clave- entre pares de entradas vecinas más que propagar relaciones de relevancia entre tópicos a lo largo de una ontología. Como será detallado en la sección de Discusión del capítulo siguiente, se afirma que la propuesta elaborada aquí podría ser utilizada para mejorar plataformas de propagación de contenido para ontologías tópicas como las revisadas en esta sección.

6.5 Representación de las relaciones de relevancia en el grafo de ODP

Teniendo en cuenta la representación mediante grafo del directorio ODP explicada en la sección 5.3.1, como un punto de partida, decimos que el tópico t_i es relevante al tópico t_j si existe una arista de algún tipo desde el tópico t_i al tópico t_j dentro del grafo respectivo. En el grafo del directorio web de la Figura 6-2 podemos decir que el tópico *EDUCATION*

es relevante para el tópico *SCHOOL_SAFETY*, o que el tópico *LABS_AND_EXPERIMENTS* es relevante para el tópico *SCIENCE_FAIRS*, entre otros ejemplos.

De todas maneras, para derivar relaciones de relevancia entre tópicos implícitas -o indirectas-, deberían ser consideradas también relaciones transitivas entre aristas del grafo. El análisis de algunos ejemplos permitió concluir que aunque las relaciones de relevancia se preservan consistentemente a través de enlaces jerárquicos, es necesario imponer ciertas restricciones en la forma en que los enlaces no jerárquicos pueden participar en las relaciones transitivas. Permitir un número arbitrario de enlaces cruzados no es factible porque relacionaría cada tópico con casi todos los demás tópicos. Tomemos, por ejemplo, la porción de ODP mostrada en la Figura 6-2. En este ejemplo, existe un camino que involucra tres aristas entre los tópicos *REFERENCE / EDUCATION / SCHOOL_SAFETY / LABS_AND_EXPERIMENTS* y *GAMES / PUZZLES*, pero la relevancia del primer tópico para el segundo es cuestionable. Por otro lado, existen otros caminos indirectos que preservan la relevancia, como es el caso del camino de tres aristas también entre *SHOPPING / TOYS_AND_GAMES* y *GAMES / PUZZLES / JIGSAWS*.

El problema abordado aquí es el siguiente: ¿Podemos derivar relaciones de relevancia no obvias entre tópicos automáticamente? Nuestro objetivo es imponer ciertas restricciones sobre cómo los enlaces de cruce pueden participar en cada camino de tal forma que se capturen los enlaces no jerárquicos del grafo de un directorio web sin dejar de preservar el sentido y el significado de la relevancia entre los tópicos.

Para construir nuestros modelos computacionales de propagación de relevancia, empezamos numerando los tópicos en V como t_1, t_2,…, t_n, y representando la estructura del grafo del directorio web por medio de matrices de adyacencia. Las matrices *booleanas* T, S,

y R son utilizadas para codificar las relaciones de relevancia explícitas como se describe a continuación. La matriz T se usa para representar la estructura jerárquica de la ontología. Dicha matriz codifica las aristas en T y se define como $T_{ij}=1$ si $(t_i,t_j) \in T$ y $T_{ij}=0$ sino. Las componentes no jerárquicas correspondientes a las aristas de enlaces *"simbólicos"* y *"relacionados"* del grafo de ODP son representadas por las matrices S y R respectivamente. La matriz S se define de tal manera que $S_{ij}=1$ si $(s_i,s_j) \in S$ y $S_{ij}=0$ de otra manera. La matriz R está definida análogamente, como $R_{ij}=1$ si $(r_i,r_j) \in R$ y $R_{ij}=0$ sino.

6.6 Modelos de Propagación de Relevancia

Luego de haber codificado las diferentes componentes del grafo de ODP en las matrices T, S y R, se debe encarar la cuestión de cómo pueden ser utilizadas estas matrices para capturar la noción de relevancia. Antes de presentar los distintos modelos de propagación de relevancia, se muestra una revisión sobre los conceptos de unión, intersección y composición de relaciones binarias, y cómo aquellas operaciones pueden ser implementadas como operaciones booleanas sobre matrices.

6.6.1 Operaciones Booleanas sobre Matrices

Ya fue establecido que las relaciones de relevancia serán codificadas como matrices booleanas. Para poder computar efectivamente nuevas relaciones a partir de las existentes, debemos tomar ventaja de la teoría disponible que conecta las operaciones sobre relaciones con las operaciones sobre matrices. A continuación, son revisadas brevemente estas ideas:

- **Unión de relaciones binarias:** Dadas las relaciones binarias ρ_A y ρ_B, la unión $\rho_A \cup \rho_B$ puede ser computada como:

$$\mathbf{A} \vee \mathbf{B},$$

Donde A y B corresponden a las representaciones matriciales de ρ_A y ρ_B, respectivamente. La operación de adición booleana $\vee$ en matrices se define como:

$$[\mathbf{A} \vee \mathbf{B}]_{ij} = \mathbf{A}_{ij} \vee \mathbf{B}_{ij}$$

- **Intersección de relaciones binarias:** Dadas las relaciones binarias ρ_A y ρ_B, la intersección $\rho_A \cap \rho_B$ puede ser computada como:

$$\mathbf{A} \wedge \mathbf{B},$$

Donde A y B corresponden a las representaciones matriciales de ρ_A y ρ_B, respectivamente. La operación de intersección booleana $\wedge$ en matrices se define como:

$$[\mathbf{A} \wedge \mathbf{B}]_{ij} = \mathbf{A}_{ij} \wedge \mathbf{B}_{ij}$$

- **Composición de relaciones binarias:** Dadas las relaciones binarias ρ_A y ρ_B, la composición $\rho_A \circ \rho_B$ se puede computar como:

$$\mathbf{A} \otimes \mathbf{B},$$

Donde **A** y **B** son las representaciones matriciales de ρ_A y ρ_B, respectivamente. La operación producto booleano $\otimes$ en matrices se define como:

$$[\mathbf{A} \otimes \mathbf{B}]_{ij} = \bigvee_k (\mathbf{A}_{ik} \wedge \mathbf{B}_{kj})$$

6.6.2 Un Modelo Inducido por Relaciones de Relevancia Explícitas

Consideremos la operación lógica $\vee$ sobre matrices, y sea $\mathbf{M_1}$ computada de la siguiente manera:

$$\mathbf{M_1} = \mathbf{T} \vee \mathbf{S} \vee \mathbf{R} \vee \mathbf{I},$$

Donde **I** es la matriz identidad. $\mathbf{M_1}$ es la matriz de adyacencia para el grafo **G** aumentada con unos (1) en la diagonal. Aunque la matriz $\mathbf{M_1}$ abarca todas las relaciones de relevancia explícitas existentes en ODP, la misma falla en la captura de muchas relaciones de relevancia indirectas que resultan de aplicar clausuras transitivas o combinar relaciones

de diferentes tipos. El modelo $\mathbf{M_1}$ será el más conservador de todos los modelos propuestos.

6.6.3 Modelos Inducidos por la Clausura Transitiva sobre la Componente Jerárquica

Aquí utilizamos el producto booleano de matrices para definir recursivamente $\mathbf{T}^{(r)}$ de la siguiente forma: Sea $\mathbf{T}^{(0)} = \mathbf{I}$ y $\mathbf{T}^{(r+1)} = \mathbf{T} \otimes \mathbf{T}^{(r)}$.

La matriz $\mathbf{T}^{(r)}$ codifica todos los caminos de tamaño *r* entre tópicos. Se define la clausura reflexiva y transitiva de **T**, denotada como $\mathbf{T}^*$, de la siguiente manera:

$$T^* = \bigvee_{r=0}^{\infty} T^{(r)}$$

La matriz $\mathbf{T}^*$ codifica todos los caminos –de cualquier tamaño- que existen entre pares de tópicos mediante enlaces de tipo *"es un"*. Debido a que existe un número finito de tópicos, la matriz $\mathbf{T}^*$ puede ser computada en un número finito de pasos. En esta matriz, $\mathbf{T}^*_{ij} = 1$ indica que t_j pertenece al subárbol de tópicos con raíz en t_i, y $\mathbf{T}^*_{ij} = 0$ significa lo contrario.

Teniendo en cuenta que se ha observado que las relaciones de relevancia son preservadas consistentemente a través de enlaces de tipo *"es un"*, se torna razonable computar la clausura $\mathbf{T}^*$ y aumentarla con las matrices que representan los enlaces de tipo *"simbólico"* y *"relacionado"*. Esto da origen al segundo modelo de propagación de relevancia propuesto en esta sección:

$$\mathbf{M_2 = T^* \vee S \vee R.}$$

En este nuevo modelo, un tópico t_i es relevante para otro tópico t_j si:

a. Existe un camino desde el tópico t_i al tópico t_j incluyendo solamente enlaces del tipo *"es un"* (de la componente jerárquica del grafo) ó

b. Existe un enlace del tipo *"simbólico"* o *"relacionado"* del tópico t_i al tópico t_j.

El modelo $\mathbf{M_2}$ es un modelo conservador en el sentido de que propaga relevancia sólo a través de la componente jerárquica del grafo de ODP, mientras que la participación de los enlaces de cruce está restringida a relaciones de relevancia explícitas o directas.

Una cuestión que surge como consecuencia del último modelo descripto es si los enlaces de cruce pueden ser incluidos en caminos indirectos mientras se sigue preservando el significado en las relaciones establecidas entre los tópicos. Se ha observado anteriormente (Figura 6-2) que la relevancia a menudo se pierde si se suma al camino un número arbitrario de enlaces de cruce. Por lo tanto, para que los modelos de relevancia sean admisibles, deberían imponerse ciertas restricciones.

En las próximas subsecciones se formula una familia de modelos de propagación de relevancia plausibles, que resultan de extender los modelos previos.

6.6.4 Modelos Inducidos por la Propagación de Enlaces de Cruce a lo largo de la Taxonomía

Una manera simple de incorporar enlaces cruzados dentro del modelo es la propagación de estos hacia arriba o hacia abajo en la taxonomía. Si queremos propagar relaciones de relevancia inducidas por enlaces de cruce hacia la raíz, obtendremos el siguiente modelo de relevancia:

$$\mathbf{M_3 = T^* \otimes (S \vee R \vee I).}$$

Alternativamente, si propagamos relaciones de relevancia inducidas por enlaces cruzados hacia las hojas de la taxonomía (es decir, partiendo de un nodo origen, siguiendo

primero con un enlace cruzado y finalmente recorriendo la taxonomía), obtenemos el siguiente modelo:

$$\mathbf{M_4 = (S \vee R \vee I) \otimes T^*.}$$

Finalmente, podemos propagar relaciones de relevancia inducidas por enlaces cruzados a través de toda la taxonomía, pero admitiendo un solo enlace de cruce en cada camino. Esto resulta en el siguiente modelo:

$$\mathbf{M_5 = T^* \otimes (S \vee R \vee I) \otimes T^*.}$$

El modelo $\mathbf{M_5}$ es equivalente al utilizado en [148] como modelo de propagación de relevancia. El mismo fue aplicado en el cómputo de medidas de similitud semántica con buenos resultados.

Otra cuestión surgida de los modelos elaborados es si las relaciones de relevancia deberían ser simétricas. La componente jerárquica del grafo de ODP –por consiguiente, los enlaces del tipo *"es un"*- codifican relaciones de relevancia desde un tópico padre a un tópico hijo que en la mayoría de los casos son no simétricas. Mientras tanto, debido a que la duplicación de URLs no está permitida, los enlaces *"simbólicos"* son una manera de representar la pertenencia de páginas web a múltiples tópicos. Por ejemplo, las páginas del tópico *SHOPPING / TOYS_AND_GAMES / SCIENCE* también pertenecen al tópico *SCIENCE / EDUCATIONAL_RESOURCES / SCIENCE_FAIRS*. Por lo tanto, los enlaces *"simbólicos"* también codifican relaciones padre – hijo que, como es el caso de los enlaces de tipo *"es un"*, son generalmente no simétricos. Por otro lado, los enlaces de tipo *"relacionados"* parecen codificar relaciones de relevancia simétricas. Consecuentemente, un modelo nuevo de relevancia puede ser formulado si se incluye la simetría de los enlaces de este tipo, haciéndolos bidireccionales. Esto se logra extendiendo el conjunto de las

matrices de enlaces de cruce con $\mathbf{R}^{\mathbf{T}}$, esto es, la traspuesta de **R**, y obteniendo de esa forma el modelo de propagación de relevancia expresado a continuación:

$$\mathbf{M_6 = T^* \otimes (S \vee R \vee R^T \vee I) \otimes T^*.}$$

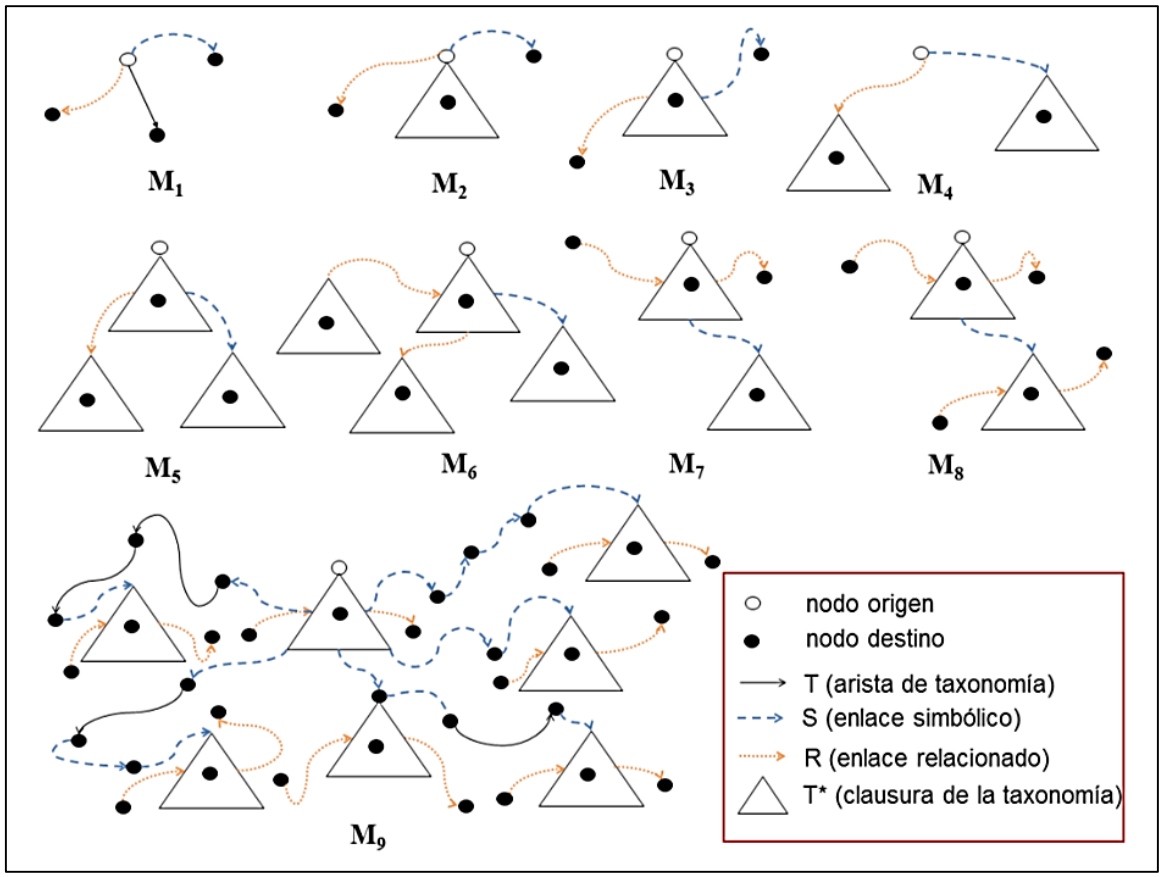


Figura 6-3: Caminos posibles desde un nodo origen a un nodo destino en los diferentes modelos de propagación de relevancia.

También pueden obtenerse modelos alternativos mediante la imposición de restricciones adicionales o la relajación de algunas. En general, los enlaces de tipo *"relacionado"* aparentan ser más débiles que los otros tipos de enlaces. Esto puede ser reflejado en un nuevo modelo que resulta de no permitir la propagación por debajo de los enlaces *"relacionados"*:

$$\mathbf{M_7 = [T^* \otimes (S \vee I) \otimes T^*] \vee [T^* \otimes (R \vee R^T \vee I)].}$$

A continuación, se muestra una generalización de $\mathbf{M_6}$ y $\mathbf{M_7}$ en la que tanto los enlaces *"simbólicos"* como los *"relacionados"* son admitidos para participar simultáneamente en un mismo camino:

$$\mathbf{M_8 = T^* \otimes (S \vee I) \otimes T^* \otimes (R \vee R^T \vee I).}$$

Existen infinidad de formas en que estos modelos pueden ser restringidos o amplificados. Por ejemplo, podríamos admitir hasta *n* enlaces *"simbólicos"*, como se muestra en la siguiente generalización de $\mathbf{M_8}$.

$$\mathbf{M_9 = T^* \otimes (T \vee S \vee I)^n \otimes T^* \otimes (R \vee R^T \vee I).}$$

La Figura 6-3 muestra posibles caminos de relevancia desde un nodo origen hacia un nodo destino, de acuerdo a los diferentes modelos. Diversos modelos han sido considerados, pero aquellos que fueron discutidos antes capturan los aspectos más interesantes o destacables de la noción de propagación de relevancia analizada en este capítulo.

6.6.5 Modelos inducidos por la detección de comunidades temáticas

Se pueden obtener nuevos modelos de propagación de relevancia si es incluida la información provista por la determinación de las componentes fuertemente conexas del grafo de ODP descripta en la sección 5.4 de esta tesis, de la cual surgen nuevas relaciones entre los tópicos del directorio. Dichas relaciones provienen de enlazar cada uno de los tópicos de una componente fuerte con los demás tópicos de la misma componente. Además, al combinar esas relaciones entre tópicos con algunos de los modelos de propagación de relevancia calculados hasta aquí, pueden obtenerse resultados diferentes con mayor o menor coherencia en sus relaciones.

El primer modelo propuesto, $\mathbf{M_{10}}$, consiste simplemente en los enlaces pertenecientes a las relaciones determinadas por las componentes fuertemente conexas (*Strongly Connected Components*, SCC) del grafo de ODP, teniendo en cuenta los enlaces de tipo "*jerárquicos*", "*simbólicos*" y "*relacionados*", estos últimos en forma simétrica:

$$\mathbf{M_{10} = SCC\ (T \vee S \vee R \vee R^T).}$$

Si combinamos los enlaces surgidos de las SCC con la información provista por otros modelos, el resultado podría ser aún más preciso. Para ello, dos de los modelos descriptos en las secciones anteriores son utilizados. La forma de obtener dichas combinaciones consiste en determinar la intersección entre la matriz de relaciones de las componentes fuertes con cada una de las matrices de estos modelos.

En primera instancia, el modelo $\mathbf{M_{11}}$ surge de intersectar las relaciones de $\mathbf{M_{10}}$ con el modelo $\mathbf{M_6 = T^* \otimes (S \vee R \vee R^T \vee I) \otimes T^*}$. Entonces, la fórmula del modelo queda así:

$$\mathbf{M_{11} = SCC\ (T \vee S \vee R \vee R^T) \wedge (T^* \otimes (S \vee R \vee R^T \vee I) \otimes T^*).}$$

Luego, $\mathbf{M_{12}}$ se deriva de la intersección entre las relaciones de $\mathbf{M_{10}}$ con las relaciones contenidas en $\mathbf{M_8 = T^* \otimes (S \vee I) \otimes T^* \otimes (R \vee R^T \vee I)}$:

$$\mathbf{M_{12} = SCC\ (T \vee S \vee R \vee R^T) \wedge (T^* \otimes (S \vee I) \otimes T^* \otimes (R \vee R^T \vee I)).}$$

Al igual que todos los modelos anteriores, los nuevos modelos $\mathbf{M_{10}}$, $\mathbf{M_{11}}$ y $\mathbf{M_{12}}$ deben ser evaluados para determinar la coherencia de las relaciones que contienen. En el capítulo siguiente de esta tesis se muestra el resultado de validaciones efectuadas en este sentido.

6.7 Conclusiones

A lo largo de este capítulo, ha sido abordado el problema de inferir relaciones de relevancia entre tópicos en un directorio web representado mediante un grafo, teniendo en

cuenta solamente aspectos estructurales de dicho grafo. Con este objetivo, fueron propuestos doce modelos diferentes de propagación de relevancia. Dichos modelos también fueron computados para un grafo de gran tamaño, que consistía en más de medio millón de nodos. Esto resultó en una tarea computacional muy desafiante, para la cual fueron implementados algoritmos eficientes específicos. Los modelos que constituyen el resultado de esta tarea son comparados cualitativamente y cuantitativamente en el capítulo siguiente de esta tesis, en el cual también se detallan experimentos con usuarios humanos para validar estadísticamente la significación de algunos de ellos.

Es destacable la participación de un algoritmo conocido sobre grafos para la formación de modelos adicionales de propagación de relevancia, el cual en principio podría aumentar la significación de algunas relaciones de relevancia entre los tópicos del directorio inferidas por los primeros modelos. Dicho algoritmo permite la identificación de comunidades de tópicos, mediante la búsqueda de grupos fuertemente conectados de nodos en el grafo del directorio. Para determinar si realmente la incorporación de tal algoritmo puede mejorar los modelos previamente elaborados, los nuevos modelos resultantes deberían ser objeto de distintos análisis.

Para nuestro conocimiento, este es el primer intento de modelar el problema de la propagación de relaciones de relevancia en el grafo de un directorio web. La aplicabilidad de los modelos de propagación de relevancia propuestos dentro del área de inteligencia artificial y ciencias de la información es extensiva y múltiple. Debido a que gran parte del conocimiento de cualquier ente con capacidad de razonar puede ser expresado en términos de relaciones de relevancia, un modelo computacional de propagación de relevancia es una herramienta útil para el diseño de sistemas de razonamiento y búsqueda de información con base en el sentido común.

Capítulo 7:

Análisis y Validación de los Modelos de Propagación de Relevancia

7.1 Introducción

Luego de elaborar un conjunto de modelos de propagación de relevancia sobre ontologías temáticas -como se detalla en el capítulo anterior de esta tesis- e implementarlos sobre la base de un conocido directorio web como es el de ODP, surgió la necesidad de validar las relaciones inferidas por tales modelos, lo cual determinaría finalmente si vale la pena o no tenerlos en cuenta para distintos fines. Dicha validación consistía en la confrontación de las relaciones de relevancia surgidas de la elaboración de los modelos propuestos, con el criterio de usuarios humanos.

En las siguientes secciones se detalla la elaboración de una plataforma para llevar a cabo dos experimentos tendientes a recabar datos estadísticos sobre el criterio de los usuarios, en base a las relaciones de relevancia entre tópicos de un directorio web que los nuevos modelos sugieren. La tarea de implementación de tal plataforma requirió la ejecución de trabajos de distintas clases, desde el tratamiento de las grandes matrices que representan a los modelos obtenidos hasta el diseño y la programación de una plataforma de visualización para la jerarquía de ODP, y también la elaboración en todos sus aspectos del sitio web a través del cual se podían llevar a cabo los correspondientes experimentos con usuarios humanos.

Respecto de los conceptos estadísticos utilizados en esta sección, los parámetros que se obtuvieron para determinar la validez de los modelos evaluados consisten en intervalos de confianza, dentro de los cuales se espera que estén incluidos los valores de determinadas muestras. Para el caso de los experimentos descriptos aquí, la muestra está constituida por las respuestas brindadas por los usuarios para cada relación de relevancia que les fue propuesta, y dichas relaciones de relevancia fueron obtenidas por una selección al azar

dentro de los modelos. A su vez, los intervalos de confianza comprenden el porcentaje esperado de respuestas de los usuarios, para las distintas posibilidades de respuesta que ellos tenían. Los tipos de respuestas de los usuarios indicaban si algún modelo en particular daba una información más precisa sobre las relaciones de relevancia, o si en realidad la relación de relevancia respectiva carecía de sentido de acuerdo al criterio del correspondiente usuario.

Las conclusiones estadísticas aquí presentadas se basan en las características de los intervalos de confianza que surgieron para los valores estudiados. El diseño de los experimentos fue llevado a cabo de manera tal que se mantuviera la aleatoriedad, en pos de evitar favorecer la información provista por algún modelo de propagación de relevancia, o por los modelos en su conjunto.

7.2 Comparación Cuantitativa

Los modelos propuestos fueron computados para la ontología de ODP. La porción del grafo de ODP que fue utilizada para el análisis consiste en 571.148 nodos de tópicos (solamente las categorías *WORLD* y *REGIONAL* no fueron tenidas en cuenta). La Tabla 7-1 muestra el tamaño de las componentes del grafo utilizadas en el análisis:

COMPONENTE	TAMAÑO
V	571.148 nodos
T	571.147 aristas
S	545.805 aristas
R	380.264 aristas

Tabla 7-1: Tamaño de cada componente para el grafo de ODP.

Para comparar cuantitativamente los diferentes modelos, se observa el número de relaciones de relevancia entre pares de tópicos inducidas por cada uno de ellos. Esta comparación es mostrada en la Tabla 7-2. La misma revela una amplia variación en el

número de relaciones de relevancia inducidas por cada modelo. Además, fue computado el número de diferencias entre los modelos, las cuales indican los pares de tópicos para los que un modelo indica la existencia de una relación y el otro no, y se observó que para algunos pares de modelos, tales como $\mathbf{M_6}$ y $\mathbf{M_9}$, el número de diferencias es tan grande como 177.799.003. Podemos inferir por esto último que los dos modelos comparados son muy diferentes entre sí, ya que el número de diferencias es mayor que el tamaño del modelo más grande.

MODELO	RELACIONES
$M_1 = T \vee S \vee R \vee I$	**2.068.364**
$M_2 = T^* \vee S \vee R$	**5.502.581**
$M_3 = T^* \otimes (S \vee R \vee I)$	**7.072.930**
$M_4 = (S \vee R \vee I) \otimes T^*$	**71.443.444**
$M_5 = T^* \otimes (S \vee R \vee I) \otimes T^*$	**170.573.370**
$M_6 = T^* \otimes (S \vee R \vee R^T \vee I) \otimes T^*$	**174.534.253**
$M_7 = [T^* \otimes (S \vee I) \otimes T^*] \vee [T^* \otimes (R \vee R^T \vee I)]$	**14.177.359**
$M_8 = T^* \otimes (S \vee I) \otimes T^* \otimes (R \vee R^T \vee I)$	**16.915.322**
$M_9 = T^* \otimes (T \vee S \vee I)^n \otimes T^* \otimes (R \vee R^T \vee I)$, n=4	**37.609.462**
$M_{10} = SCC\ (T \vee S \vee R \vee R^T)$	**78.131.181.150**
$M_{11} = SCC\ (T \vee S \vee R \vee R^T) \wedge (T^* \otimes (S \vee R \vee R^T \vee I) \otimes T^*)$	**91.223.971**
$M_{12} = SCC\ (T \vee S \vee R \vee R^T) \wedge (T^* \otimes (S \vee I) \otimes T^* \otimes (R \vee R^T \vee I))$	**10.169.346**

Tabla 7-2: Comparación cuantitativa de los modelos.

7.3 Análisis Cualitativo

Habiendo observado que los modelos produjeron caracterizaciones diferentes en términos cuantitativos de la noción de relevancia, fue llevado a cabo un análisis de la calidad de las relaciones inducidas por cada uno de ellos.

Una observación teórica importante respecto del conjunto de modelos es que estos forman un orden parcial bajo la relación *"$\boldsymbol{M_m \leq M_n}$ sí y solo sí $\boldsymbol{[M_m]_{ij} = 1}$ implica que $\boldsymbol{[M_n]_{ij} = 1}$ para todo **i, j**"*. El orden parcial resultante está representado en la Figura 7-1, y puede ser fácilmente verificado mediante el análisis de la definición de cada modelo, como

también por las definiciones de los operadores $\vee$, $\wedge$ y $\otimes$. Más aun, esto es consistente con los modelos elaborados utilizando el grafo de ODP.

Para profundizar un poco más dentro de los aspectos cualitativos de cada modelo, se hizo uso de la herramienta de visualización desarrollada en el marco de este trabajo y descripta en la sección 5.3.3 de esta tesis. Dicha herramienta fue utilizada en combinación con las matrices computadas para identificar casos en que los modelos no coincidieran respecto a la existencia o no de una relación de relevancia entre pares de tópicos. Una vez que los tópicos en conflicto eran identificados en los modelos, la herramienta de visualización permitió observar estos tópicos y el conjunto de páginas asociadas a ellos. Esto brindó una ayuda muy importante para sobrellevar el problema de determinar cuáles modelos producían las caracterizaciones más precisas de la noción de relevancia.

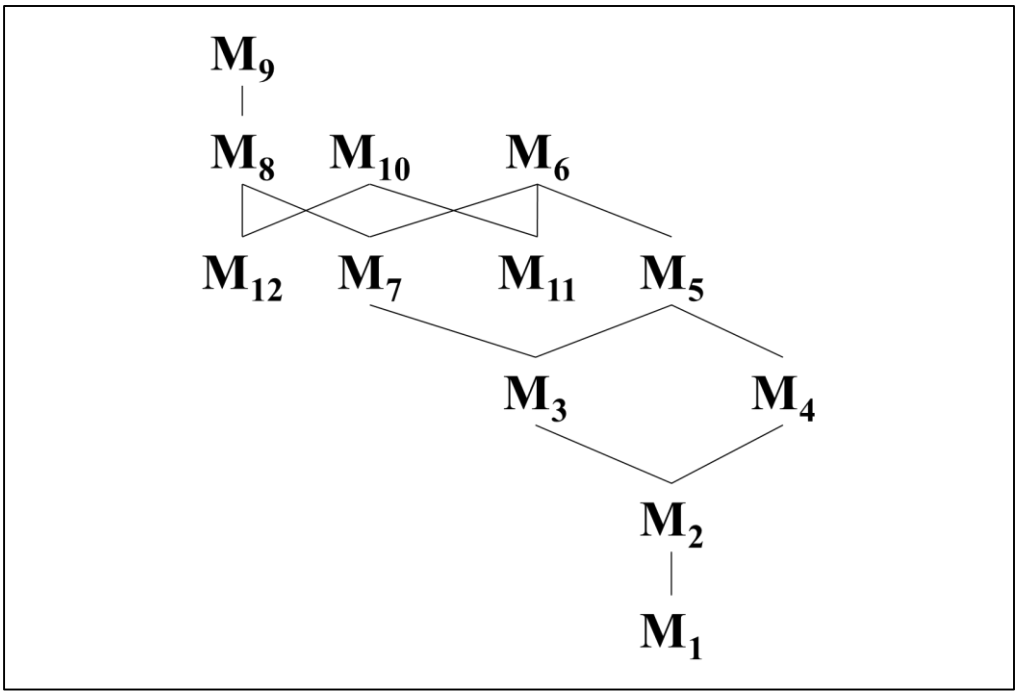


Figura 7-1: Orden parcial en el conjunto de modelos de propagación de relevancia.

El concepto de relevancia es altamente subjetivo ([159], [160]). Luego de realizar un experimento piloto, se llegó a observar bajos niveles de acuerdo en las opiniones

respecto de la relevancia entre evaluadores humanos. Para complicar aún más la tarea de evaluación de los diferentes modelos, notamos que incluso para la misma opinión, una relación de relevancia que existía en un cierto instante de tiempo puede desaparecer después, o viceversa. A pesar de estas discrepancias, para un gran número de pares de tópicos, hubo un claro acuerdo respecto a la existencia o ausencia de una relación de relevancia implícita. Por ejemplo, en la Figura 7-2, la existencia de una relación de relevancia implícita entre el tópico *SHOPPING / TOYS_AND_GAMES* y el tópico *GAMES / PUZZLES / JIGSAW* es incuestionable, siendo que solamente los modelos **M_5** y **M_6** capturan esta relación. En contraste, no hay una relación notable de relevancia entre los tópicos *SOCIETY / ORGANIZATIONS / STUDENT* y *ARTS / ART_HISTORY / MOVEMENTS / IMPRESSIONISM* de la Figura 7-3, a pesar del hecho de que modelos menos conservadores (**M_5**, **M_6**, **M_7**, **M_8** y **M_9**) indicarían la existencia de tal relación.

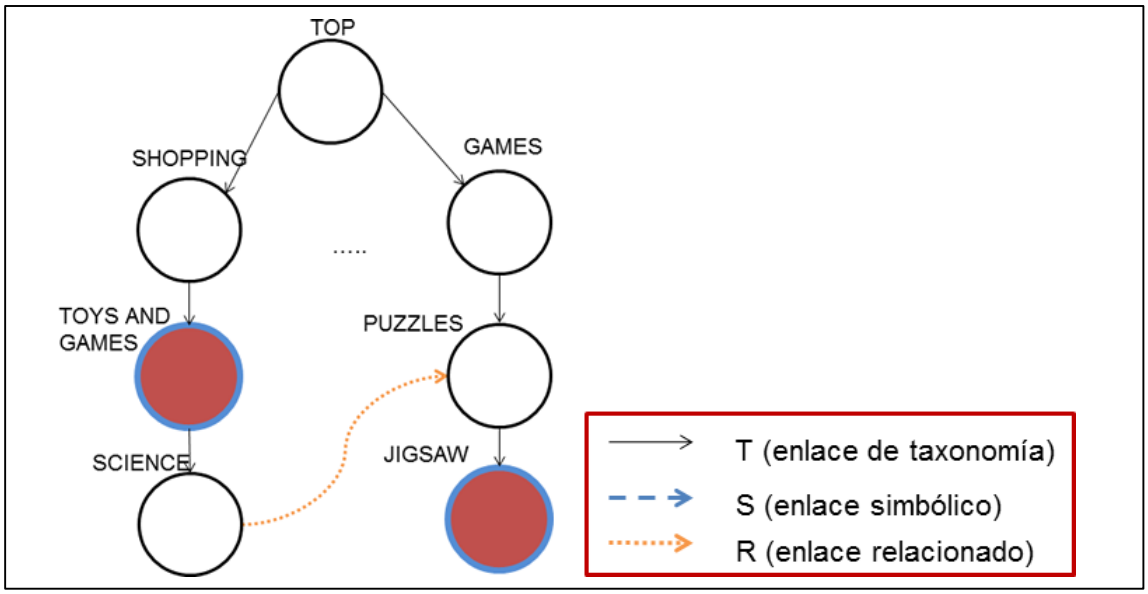


Figura 7-2: Ejemplo de una relación de relevancia implícita incuestionable dentro de los modelos propuestos.

Ejemplos similares al de la Figura 7-3 son muy generalizados en ODP. Esto resalta el hecho de que los esquemas menos conservadores de propagación de relevancia no son robustos, debido a que unos pocos enlaces cruzados no confiables provocan cambios

globales significativos en los modelos de propagación de relevancia. Por el otro lado, los modelos más conservadores están incompletos, y por lo tanto son incapaces de derivar muchas relaciones de relevancia útiles inducidas por los menos conservadores.

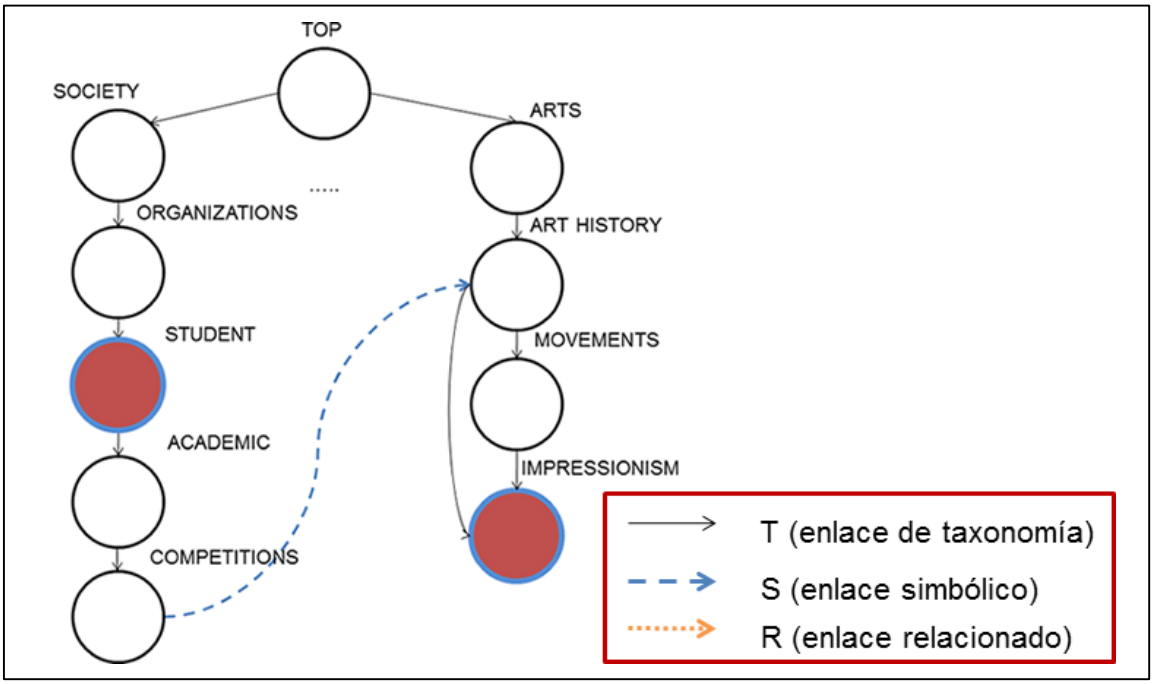


Figura 7-3: Ejemplo de una relación de relevancia implícita cuestionable dentro de los modelos propuestos.

7.4 Validación de los modelos mediante Estudios con Usuarios

Para evaluar la precisión de algunos de los modelos propuestos, fueron llevados a cabo dos experimentos para comparar algunos de los más promisorios. El primero de estos experimentos fue llevado a cabo sobre dos modelos básicos menos conservadores de propagación de relevancia propuestos en este trabajo. El segundo involucró a dos modelos resultantes de aumentar los primeros con la detección de componentes fuertemente conexas dentro del grafo de ODP, lo cual se puede ver en detalle en el capítulo anterior de esta tesis.

Los propósitos de la realización de estos experimentos fueron los siguientes:

1. Determinar cuál de los modelos analizados es más preciso que el otro, en cada uno de los experimentos.

2. Destacar la importancia de incorporar relaciones de relevancia que van más allá de los modelos básicos más conservadores.

7.4.1 Descripción del Estudio con Usuarios

A cada participante le fueron mostradas en secuencia 30 triplas de sitios web pertenecientes a un tópico principal y dos potenciales tópicos relacionados, de acuerdo con los modelos de propagación de relevancia evaluados. La selección de estos tópicos se explica más adelante en esta sección. Para cada tripla mostrada, se presentaba una imagen asociada con el tópico principal en la parte superior de la pantalla y dos imágenes correspondientes con los tópicos potencialmente relacionados en la parte inferior. Las dos imágenes eran mostradas aleatoriamente una a la izquierda y otra a la derecha de la pantalla, con el propósito de evitar favorecer a algunos de los modelos en particular. Solamente las imágenes de los sitios web seleccionados fueron mostradas, sin dar ninguna información respecto de los tópicos correspondientes a los sitios. Se les daba a los participantes la posibilidad de navegar por los sitios, proveyéndoles el enlace correspondiente en un lado de la imagen. Para cada tripla mostrada, se les solicitó a los usuarios decidir cuál de las páginas web candidatas estaba más relacionada a la del tópico principal, mediante la selección de una de las siguientes opciones:

- La página de la izquierda está más relacionada que la de la derecha con la página principal.
- La página de la derecha está más relacionada que la de la izquierda con la página principal.
- Ambas están igualmente relacionadas.
- Ninguna está relacionada.

El lenguaje de los sitios web seleccionados para el experimento fue restringido al inglés. Por lo tanto, se requirió a los usuarios que tuvieran conocimientos en dicho lenguaje. Un ejemplo de una tripla presentada en los experimentos puede verse en la Figura 7-4.

7.4.2 Primer experimento: Modelos básicos de propagación de relevancia

En una primera instancia, se llevó a cabo un estudio con usuarios incluyendo relaciones de relevancia provenientes de modelos básicos poco conservadores de propagación de relevancia. Esta evaluación involucró a 32 voluntarios humanos. En las secciones siguientes se muestran los aspectos particulares y resultados del primer experimento con usuarios llevado a cabo para determinar la validez de los modelos de propagación de relevancia elaborados.

7.4.2.1 Selección de los Modelos a evaluar

La selección de los modelos más prometedores fue llevada a cabo considerando aquellos que eran menos conservadores, sin llegar a modelos muy audaces, en cuanto a la significación de las relaciones que estos derivan entre los tópicos. El objetivo era resaltar las relaciones transitivas entre tópicos, evitando muchos pasos que involucraran aristas de enlaces de cruce, como es el caso de $\mathbf{M_9}$.

Otro aspecto importante que fue considerado al momento de la selección fue que la mayoría de los modelos restantes deberían estar incluidos dentro de los seleccionados, para que estos fueran lo más representativos posible. Por ejemplo, $\mathbf{M_7}$ está contenido tanto en $\mathbf{M_6}$ como en $\mathbf{M_8}$, mientras que $\mathbf{M_5}$ está contenido en $\mathbf{M_6}$ (ver Figura 7-1).

El experimento piloto llevado a cabo (mencionado en la sección Análisis cualitativo de este capítulo) ayudó en el proceso de selección, llevando a la identificación de relaciones

de relevancia útiles que estaban presentes en modelos menos conservadores pero ausentes en los modelos más básicos. En la Figura 7-5, al igual que en la Figura 7-6, se ilustran instancias de tales relaciones. Por ejemplo, $\mathbf{M_6}$ induce una relación de relevancia entre los tópicos *SCIENCE / PHYSICS / INSTRUMENTS_AND_SUPPLIES* y *SCIENCE / INSTRUMENTS_AND_SUPPLIES / LABORATORY_EQUIPMENT / GLASS_PRODUCTS_AND_ACCESORIES*. De todos modos, la mayoría de los modelos propuestos son incapaces de identificar esta relación. De manera similar, $\mathbf{M_8}$ infiere una relación de relevancia entre los tópicos *BUSINESS / ENERGY_AND_ENVIRONMENT / OIL_AND_GAS* y *SCIENCE / EARTH_SCIENCES / PRODUCTS_AND_SERVICES / CONSULTING*, que no es identificada por el resto de los modelos computados.

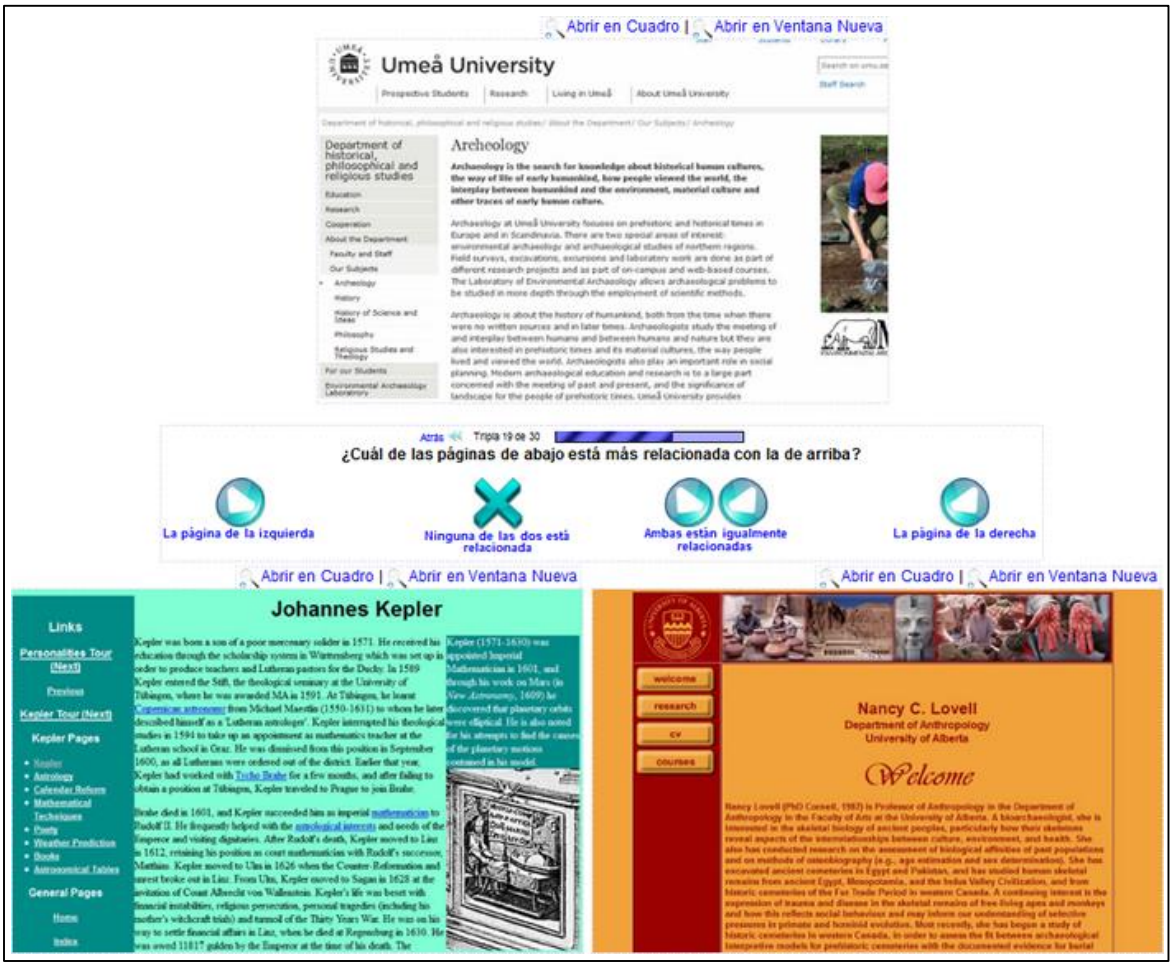


Figura 7-4: Ejemplo de una tripla mostrada a los usuarios en los experimentos.

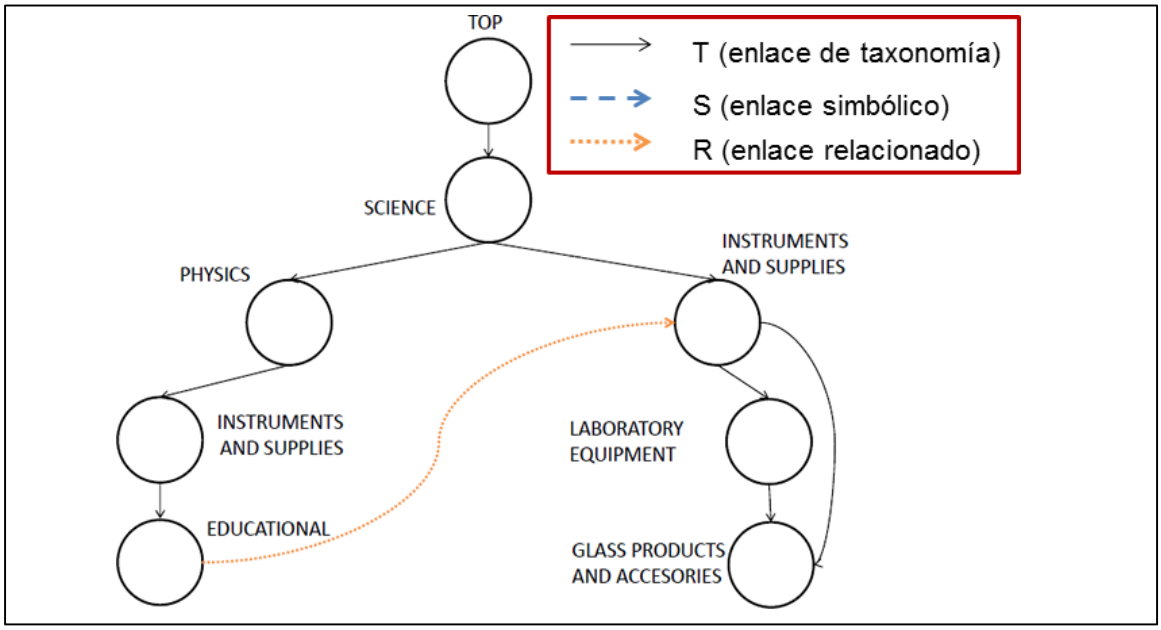


Figura 7-5: Ejemplo de una relación de relevancia útil existente en $\mathbf{M_6}$ pero ausente en los otros modelos.

Tomando en cuenta las consideraciones antes mencionadas, los modelos seleccionados fueron los siguientes:

- $\mathbf{M_6} = \mathbf{T}^* \otimes (\mathbf{S} \vee \mathbf{R} \vee \mathbf{R}^T \vee \mathbf{I}) \otimes \mathbf{T}^*$
- $\mathbf{M_8} = \mathbf{T}^* \otimes (\mathbf{S} \vee \mathbf{I}) \otimes \mathbf{T}^* \otimes (\mathbf{R} \vee \mathbf{R}^T \vee \mathbf{I})$

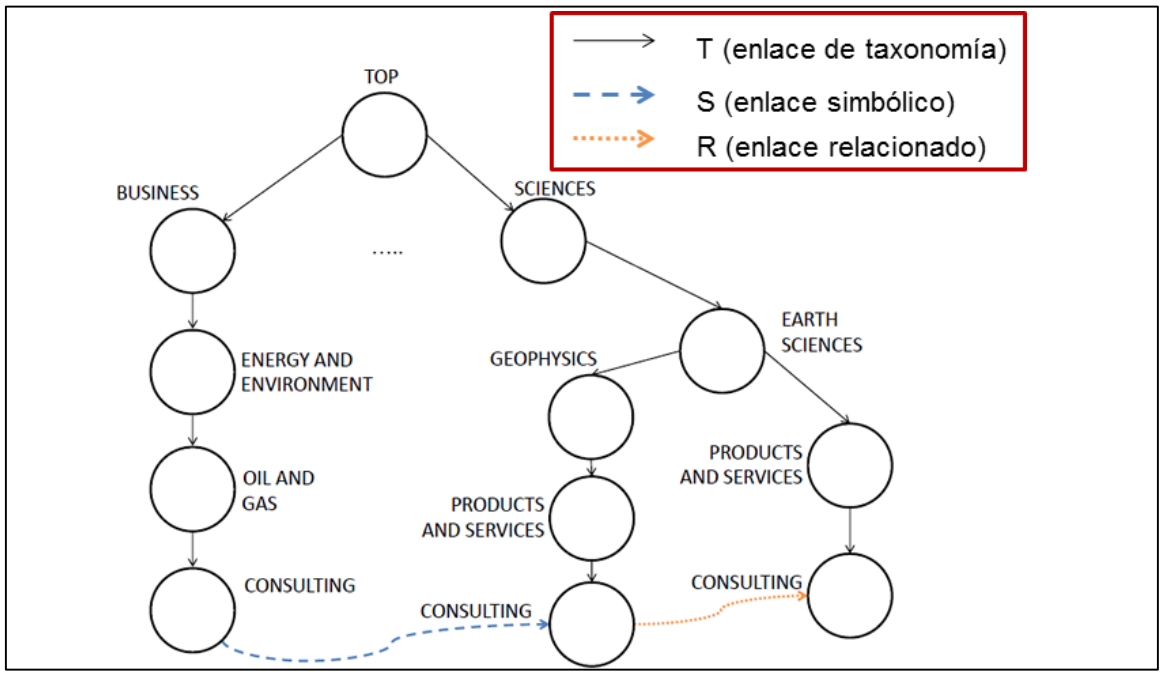


Figura 7-6: Ejemplo de una relación de relevancia útil existente en $\mathbf{M_8}$ pero ausente en los otros modelos.

7.4.2.2 Puesta a punto del experimento

Una vez que fueron seleccionados $\mathbf{M_6}$ y $\mathbf{M_8}$ como los modelos candidatos más promisorios, la tarea siguiente fue aislar triplas de tópicos (t_1, t_2, t_3) que pudieran satisfacer las siguientes condiciones:

- El tópico principal t_1 debía tener por lo menos un tópico relacionado de acuerdo a $\mathbf{M_6}$ y otro tópico relacionado de acuerdo a $\mathbf{M_8}$.
- El tópico t_2 debía estar relacionado a t_1 de acuerdo a $\mathbf{M_6}$ pero no de acuerdo a $\mathbf{M_8}$.
- El tópico t_3 debía estar relacionado a t_1 de acuerdo a $\mathbf{M_8}$ pero no de acuerdo a $\mathbf{M_6}$.

Un ejemplo de una tripla que satisface estas condiciones puede verse en la Tabla 7-3, y la visualización de dicha tripla en el experimento se corresponde con la Figura 7-4.

URL	**Tópico**
www.idesam.umu.se/english/about/subjects/archeology/?languageId=1	***t₁***: SCIENCE / SOCIAL_SCIENCES / ARCHAEOLOGY / TOPICS: **Tópico Principal**
www.hps.cam.ac.uk/starry/kepler.html	***t₂***: SCIENCE / ASTRONOMY / HISTORY / PEOPLE / KEPLER,_JOHANNES: **Relacionado con *t₁* de acuerdo a M₆:** ***M₆ [t₁,t₂]=1; M₈[t₁,t₂]=0***
www.ualberta.ca/~nlovell/index.html	***t₃:*** SCIENCE / SOCIAL_SCIENCES / ARCHAEOLOGY / ARCHAEOLOGISTS / BIOARCHAEOLOGISTS: **Relacionado con *t₁* de acuerdo a M₈:** ***M₆ [t₁,t₂]=0; M₈[t₁,t₂]=1***

Tabla 7-3: Ejemplo de una tripla utilizada en la evaluación.

Mediante el uso de la notación matricial, se puede decir que las triplas de tópicos *(t_1, t_2, t_3)* deben cumplir la siguiente condición:

$$M_6[t_1, t_2] \wedge \neg M_8[t_1, t_2] \wedge \neg M_6[t_1, t_3] \wedge M_8[t_1, t_3] \quad (1)$$

El primer paso para identificar estas triplas fue aislar aquellas relaciones que estaban presentes en uno de los modelos pero no en el otro. Para lograr esto, se aplicó el operador lógico de diferencia en las matrices de los modelos, como sigue:

- Relaciones candidatas de **M₆**: **M₆** \ **M₈** = **M₆** ∧ ¬ **M₈**
- Relaciones candidatas de **M₈**: **M₈** \ **M₆** = **M₈** ∧ ¬ **M₆**

Esto permitió identificar conjuntos de relaciones candidatas de cada modelo. La cantidad de relaciones candidatas en **M₆** \ **M₈** fue 159.926.121 (en efecto, elementos no cero en la matriz resultante), mientras que el número de relaciones candidatas en **M₈** \ **M₆** fue 2.307.190. Las matrices de relaciones candidatas resultantes permitieron aislar una secuencia de triplas *(t_1, t_2, t_3)* que cumplían con la condición (1). La forma de obtener esta secuencia consistió en buscar índices de filas t_1 e índices de columnas t_2 y t_3 tales que para

cada posición (t_1, t_2) de la matriz resultante $\mathbf{M_6} \setminus \mathbf{M_8}$ y para cada posición (t_1, t_3) de la matriz obtenida $\mathbf{M_8} \setminus \mathbf{M_6}$ los valores fueran no ceros.

Con el propósito de lograr un experimento más preciso, fueron seleccionados 5 (cinco) tópicos principales, asociando cada uno con 6 (seis) triplas, lo que dio como resultado las 30 (treinta) triplas que fueron necesarias para que el experimento tuviera validez estadística. También, de esta forma, se evitaba imponer un esfuerzo cognitivo excesivo en los usuarios humanos que llevaron a cabo este experimento, ya que solamente debían volver a asimilar la página web mostrada para el tópico principal luego de seis triplas mostradas y evaluadas. Esta implementación fue muy similar a la adoptada en [148] para validar estadísticamente un modelo de cómputo de similitud semántica. Además, se requirió que cada una de las triplas tuviera asociadas páginas web activas al momento de realizar el experimento, y que representaran adecuadamente los contenidos de cada tópico. El hecho de que las páginas estuvieran activas facilitaba a los usuarios disipar cualquier duda que surgiera acerca del tema que se trataba en cada una de ellas, ya que, como fue mencionado antes, se facilitaba enlaces a dichas páginas en todo momento durante el experimento, para que los usuarios pudieran visitarlas y sacar sus propias conclusiones.

Por todo lo anterior, se puede afirmar que la selección de las triplas de tópicos no fue una tarea trivial. Una cuestión importante que da soporte a esa afirmación es la desaparición de algunos sitios web durante la ejecución del experimento, lo cual generaba la necesidad de un monitoreo permanente de todos los sitios, debiendo alternar algunos de ellos dentro de un mismo tópico cuando dejaban de ser accesibles. Cabe aclarar que, si bien se contaba con información significativa del sitio web desaparecido (imágenes descriptivas de la página y textos asociados), era importante que los usuarios tuvieran siempre acceso a los sitios para tener muy en claro el ámbito de los mismos, y por ello también los tópicos

seleccionados debían tener más de un sitio asociado en funcionamiento al momento de llevar a cabo el experimento.

Una pieza fundamental para el proceso de selección de triplas fue la herramienta de visualización de la jerarquía de tópicos de ODP descripta anteriormente (ver sección 5.3.3), la cual permitió verificar rápidamente la existencia y funcionamiento de los sitios web asociados a los tópicos seleccionados.

7.4.2.3 Resultados del Primer Experimento

Un dato muy importante para tener en cuenta respecto de la implementación del experimento es el tiempo promedio para la ejecución completa del mismo, que en este caso fue de 20 minutos aproximadamente. De esta forma, si se necesita llevar a cabo futuras implementaciones para experimentos similares, se cuenta con una base de información que puede ser útil para captar mayor cantidad de voluntarios, y así darle mayor precisión a los resultados finales de una validación estadística.

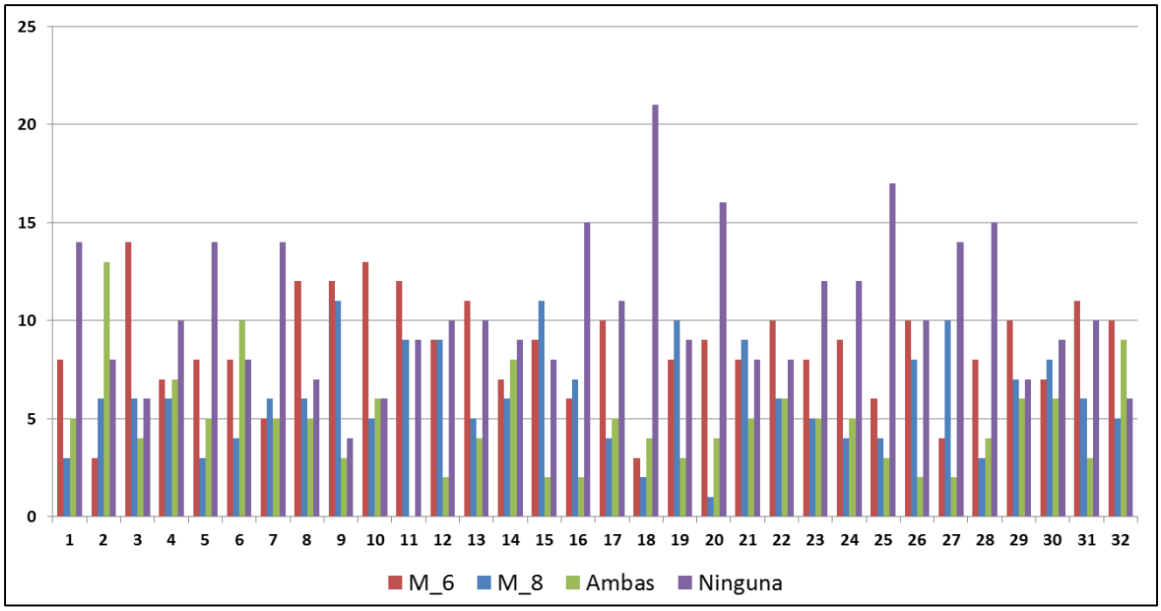


Figura 7-7: Número de respuestas para cada opción agrupadas por usuario.

Otro resultado obtenido en base a este experimento fue el número de respuestas para cada una de las cuatro posibles opciones mostradas con cada tripla. La Figura 7-7 muestra estos resultados agrupados por usuario, y la Figura 7-8 ilustra los mismos resultados pero agrupados por tripla. El volumen de datos utilizado para llevar a cabo este experimento, así como también las respuestas individuales dadas por cada usuario están disponibles en línea, en el siguiente enlace:

http://ir.cs.uns.edu.ar/downloads/relevance_propagation_experiment_dataset.xls

La Tabla 7-4 muestra el primer análisis realizado, agrupando las respuestas por usuario. A partir de dicho análisis, se puede ver que los intervalos de confianza (ICs) para el número medio de respuestas asociadas a $\mathbf{M_6}$ y $\mathbf{M_8}$ no se solapan. A primera vista, se podría sugerir que el análisis apunta a $\mathbf{M_6}$ como un mejor esquema de propagación de relevancia. De todas maneras, si se observa el solapado de los ICs para las respuestas asociadas a cada una de las cuatro opciones, no se puede decir que exista una diferencia estadísticamente significativa. Por lo tanto, aun cuando las medias de las respuestas para $\mathbf{M_6}$ y $\mathbf{M_8}$ son diferentes, no hay una diferencia estadísticamente significativa que justifique la elección de un modelo sobre el otro debido a la poca significación de las diferencias con las otras respuestas.

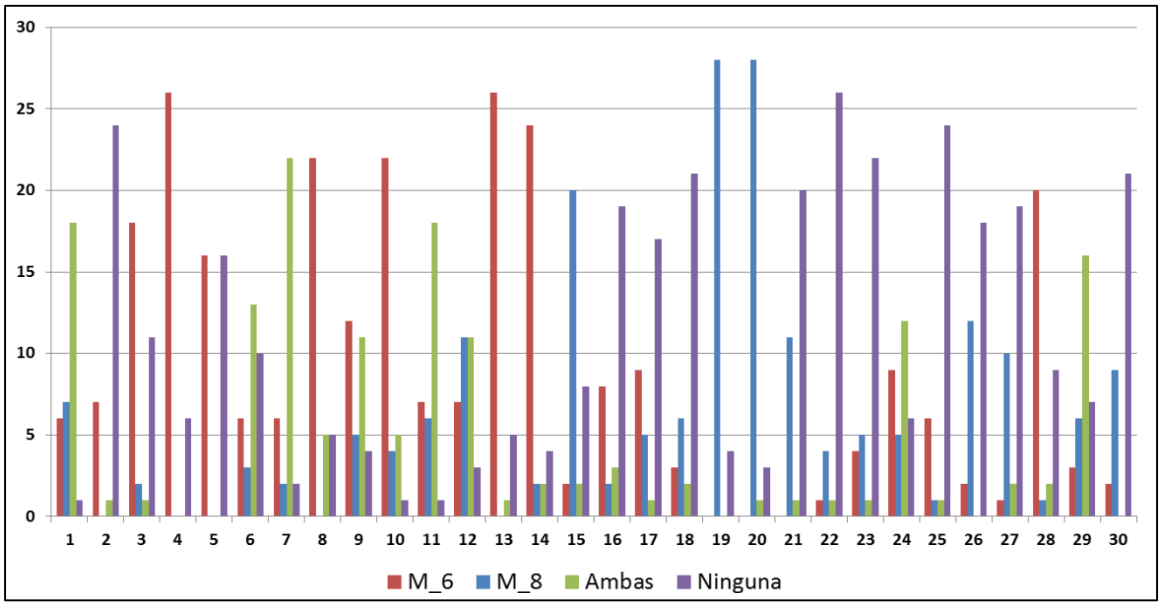


Figura 7-8: Número de respuestas para cada opción agrupadas por tripla.

Respuesta	*n* (usuarios)	Media	Desvío Estándar	IC 95%
M_6	32	28,65%	8,99%	25,53-31,76
M_8	32	20,31%	8,61%	17,33-23,29
Ambas	32	15,94%	8,71%	12,92-18,95
Ninguna	32	35,10%	12,64%	30,72-39,48

Tabla 7-4: Primer análisis sobre los datos del experimento.

Si consideramos solamente la existencia o ausencia de una relación para cada respuesta de acuerdo al criterio del usuario, los resultados son bastante diferentes. Esto se realizó agrupando las respuestas que indicaban la existencia de alguna relación de relevancia entre el tópico principal y cualquiera de los tópicos de los modelos evaluados para cada usuario. Dichas respuestas son las tres primeras opciones dentro de cada tripla mostrada a los usuarios: *"La página de la izquierda"*, *"La página de la derecha"* y *"Ambas están igualmente relacionadas"* (ver Figura 7-4). Luego, se calculó el porcentaje de respuestas que reflejan la existencia de una relación de relevancia y este fue comparado con el porcentaje de respuestas que reflejan la no existencia de tal relación (para las triplas mostradas, esta respuesta correspondería a la cuarta opción: *"Ninguna de las dos está relacionada"*). Esta comparación se puede observar en la Tabla 7-5. El gráfico que ilustra

los porcentajes totales de respuestas para cada una de las cuatro opciones se encuentra en la Figura 7-9, mientras que el gráfico con los porcentajes para las opciones agrupadas de acuerdo al segundo análisis es mostrado en la Figura 7-10.

Respuesta	*n* (usuarios)	Media	Desvío Estándar	IC 95%
M_6**,** M_8 **o ambas**	32	64,90%	12,64%	60,52-69,28
Ninguna	32	35,10%	12,64%	30,72-39,48

Tabla 7-5: Segundo análisis sobre los datos del experimento.

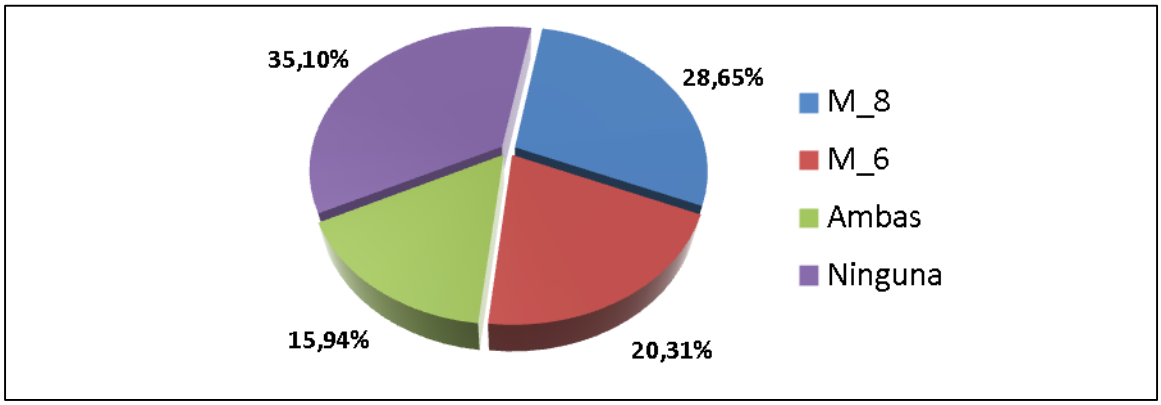


Figura 7-9: Porcentaje de respuestas para cada opción.

Estos resultados indican que existe una diferencia estadísticamente significativa entre las medias de los dos grupos, para el caso de las respuestas agrupadas por existencia o no de una relación de relevancia, dado que los ICs no se solapan, con un nivel de significación del 5% (95% de nivel de confianza). Como consecuencia de ello, tenemos suficiente evidencia estadística como para concluir que las relaciones de relevancia determinadas por los modelos evaluados son consistentes en muchos casos de acuerdo al criterio de los usuarios y pueden ser tenidas en cuenta, entre otras cosas, para el cómputo de medidas de similitud semántica entre sitios web. En otras palabras, los modelos básicos son insuficientes para reflejar relaciones de relevancia útiles que podrían ser provistas por algunos otros modelos menos conservadores.

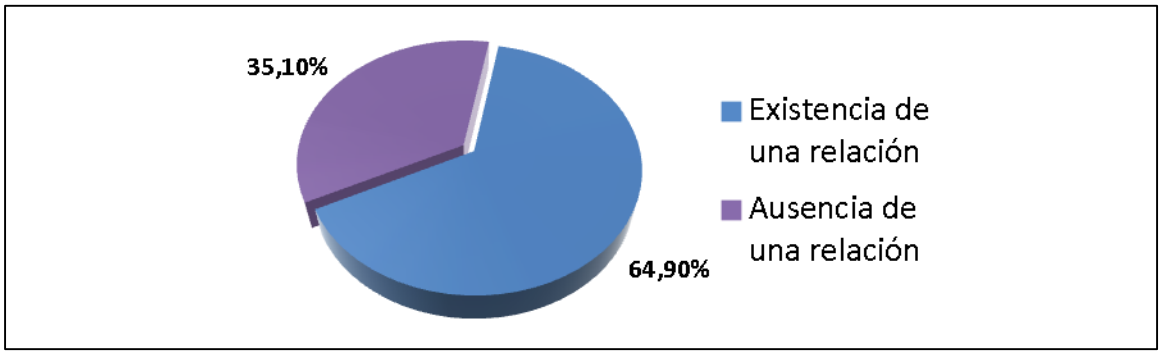


Figura 7-10: Porcentaje de respuestas de existencia y no existencia de una relación de relevancia.

7.4.3 Segundo experimento: Modelos aumentados con la detección de comunidades temáticas

Respetando los lineamientos establecidos como marco de trabajo en el primer experimento, fueron evaluados los modelos $\mathbf{M_{11}}$ y $\mathbf{M_{12}}$ de la misma manera. El propósito de la elección de estos dos modelos se fundamenta en la intención de aumentar la precisión de los modelos evaluados en el primer experimento, mediante la incorporación de información provista por las componentes fuertes del grafo de ODP. Para resumir la sección, se destacan los aspectos fundamentales del experimento llevado a cabo:

- Modelos de propagación de relevancia evaluados:
  - $\mathbf{M_{11} = SCC\ (T \vee S \vee R \vee R^T) \wedge (T^* \otimes (S \vee R \vee R^T \vee I) \otimes T^*)}$
  - $\mathbf{M_{12} = SCC\ (T \vee S \vee R \vee R^T) \wedge (T^* \otimes (S \vee I) \otimes T^* \otimes (R \vee R^T \vee I))}$
- El experimento fue llevado a cabo por 34 voluntarios humanos.
- 30 triplas de tópicos fueron seleccionadas, con las mismas cuatro opciones del primer experimento para los usuarios, en cada tripla mostrada.
- Cantidad de diferencias en las relaciones dadas por los modelos:
  - Elementos en $\mathbf{M_{11} \backslash M_{12}}$: 83.244.916
  - Elementos en $\mathbf{M_{12} \backslash M_{11}}$: 2.190.291

- La selección de las triplas de tópicos con sus correspondientes sitios web fue llevada a cabo de la misma manera que para el experimento anterior, teniendo en cuenta los nuevos modelos empleados y las matrices que los representan.

7.4.3.1 Resultados del Segundo Experimento

Efectuando los mismos análisis que en el experimento anterior, en esta sección se muestran los gráficos y tablas que describen los resultados finales del segundo experimento. En primer lugar, en las Figuras Figura 7-11 y Figura 7-12 se especifica la cantidad de respuestas por usuario y por pregunta respectivamente, para cada una de las opciones presentadas en cada una de las triplas mostradas durante el experimento.

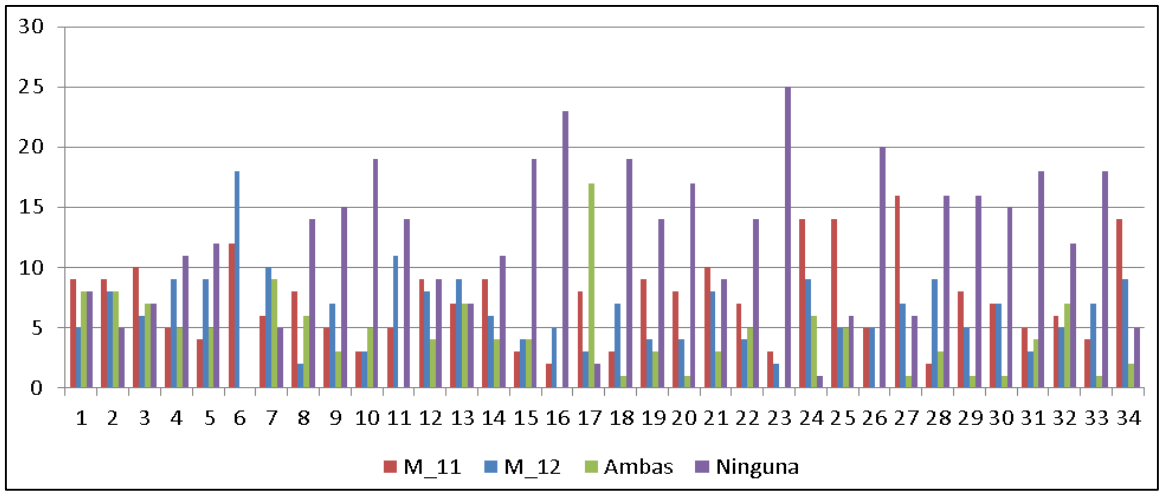


Figura 7-11: Cantidad de respuestas por usuario para cada opción.

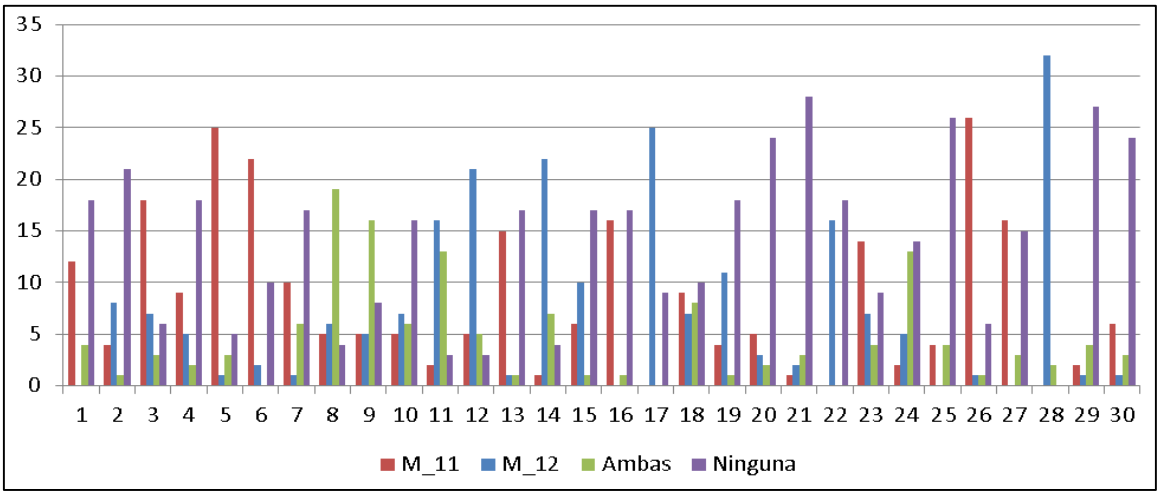


Figura 7-12: Cantidad de respuestas por pregunta para cada opción.

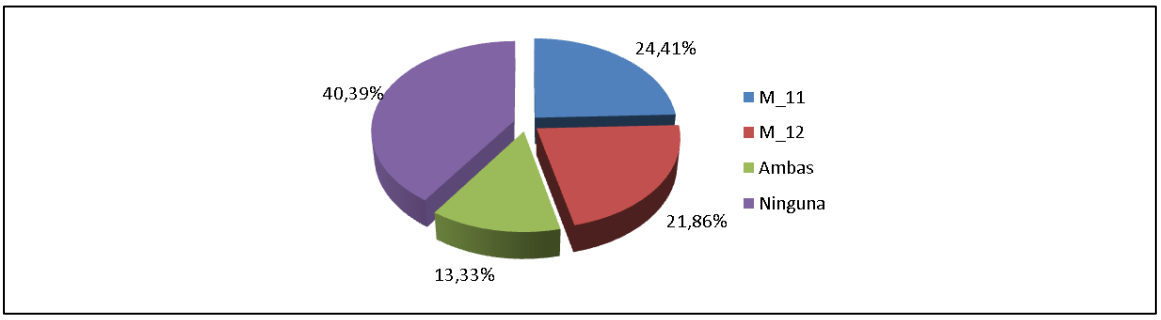


Figura 7-13: Porcentaje de respuestas para cada opción.

Respuesta	*n* (usuarios)	Media	Desvío Estándar	IC 95%
M_{11}	34	24,41%	12,25%	20,30-28,53
M_{12}	34	21,86%	10,48%	18,34-25,39
Ambas	34	13,33%	11,72%	9,39-17,27
Ninguna	34	40,39%	21,19%	33,27-47,51

Tabla 7-6: Primer análisis sobre los datos del experimento.

Luego, en la Figura 7-13 se muestra un gráfico circular describiendo los porcentajes totales para cada tipo de respuesta. Al igual que en el experimento anterior, el segmento más significativo es aquel que corresponde a la opción "Ninguna de las páginas está relacionada". Además, se repite la situación del solapamiento de los intervalos de confianza para las medias de las respuestas asociadas a $\mathbf{M_{11}}$ y $\mathbf{M_{12}}$ con las demás, como puede verse en la Tabla 7-6, la cual exhibe los resultados del análisis estadístico por intervalos de confianza de los datos. Más aun, los mismos intervalos de confianza para las Medias de los

porcentajes de respuesta para las opciones correspondientes a **M_{11}** y **M_{12}** se solapan, como lo indica la columna de los intervalos de confianza de la Tabla 7-6.

Si utilizamos el mismo recurso que en el experimento anterior y asociamos las respuestas que indican la existencia de una relación, y las contrastamos con la opción que refleja la ausencia de relación entre las páginas mostradas en cada tripla del experimento, nuevamente los resultados se vuelven más auspiciosos. La Figura 7-14 detalla la proporción de cada uno de estos grupos. Se observa allí que, agrupando los resultados de este modo, la proporción de respuestas que indican la existencia de una relación entre las páginas supera ampliamente a la proporción que indica lo contrario. Este resultado también se puede observar en el análisis que muestra la Tabla 7-7. El resultado que se observa allí muestra que los intervalos de confianza no se solapan, por lo que es de esperar que en general las relaciones arrojadas por los modelos evaluados tengan un nivel de significación considerable.

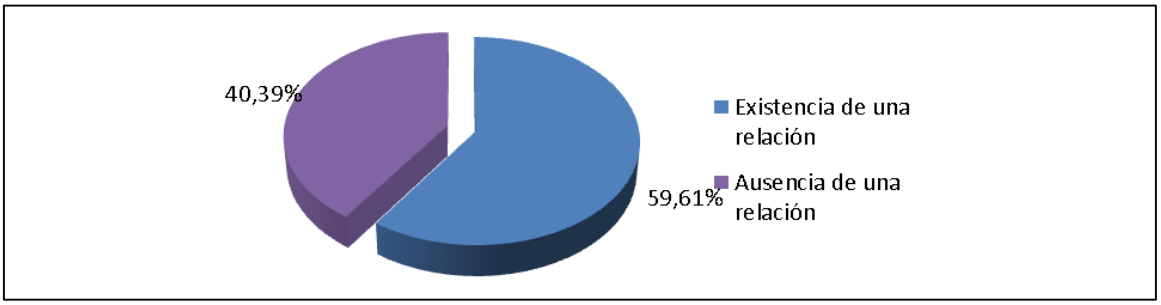


Figura 7-14: Porcentajes de respuestas agrupadas por existencia o ausencia de una relación.

Respuesta	*n* (usuarios)	Media	Desvío Estándar	IC 95%
M_{11}, M_{12} **o Ambas**	34	59,61%	21,19%	52,49-66,73
Ninguna	34	40,39%	21,19%	33,27-47,51

Tabla 7-7: Segundo análisis sobre los datos del experimento.

7.5 Discusión

Los análisis anteriores permiten arribar a la conclusión de que, aunque algunos modelos son mejores predictores que otros de la existencia o ausencia de relaciones de

relevancia, ninguno de ellos es infalible. Esto señala que, a pesar de ser un concepto clave en la inteligencia artificial y las ciencias de la información, la relevancia es una noción difusa y sutil, difícil, si no imposible, de formalizar utilizando solamente aspectos estructurales.

A pesar de estas limitaciones, los mencionados análisis indican que existe un claro incremento en la cantidad de información útil inferida cuando los modelos menos conservadores (tales como $\mathbf{M_6}$ o $\mathbf{M_8}$) son utilizados para identificar relaciones de relevancia implícitas. Este estudio provee una nueva percepción dentro del problema del cómputo de medidas de similitud semántica para ontologías generales, resaltando los beneficios de tomar ventaja tanto de las componentes jerárquicas como de las no jerárquicas de estas ontologías.

Como fue propuesto en [148], la similitud semántica entre dos tópicos t_i y t_j, en un grafo proveniente de una ontología, puede ser calculada utilizando un enfoque de teoría de la información como sigue:

$$\sigma^{G}(t_i, t_j) = \max_k \frac{2 * \left(M[t_k, t_i] \wedge M[t_k, t_j]\right) * \log P_\varphi(t_k)}{\log(P_\varphi(t_i|t_k) * P_\varphi(t_k)) + \log(P_\varphi(t_j|t_k) * P_\varphi(t_k))}$$

La probabilidad $P_\varphi(t_k)$ representa la probabilidad anticipada de que cualquier documento sea clasificado bajo el tópico t_k. Una vez que un modelo de propagación de relevancia ha sido computado, $P_\varphi(t_k)$ puede ser estimado naturalmente en términos del modelo **M** como:

$$P_\varphi(t_k) = \frac{\sum_{t_j \in V}(M[t_k, t_j] * |t_j|)}{|U|} \quad (2)$$

Donde $|t_j|$ es la cantidad de documentos directamente asociados al tópico t_j, y $|U|$ es el número total de documentos en la ontología. La probabilidad condicional $P_\varphi(t_i|t_k)$

representa la probabilidad de que cualquier documento sea clasificado bajo el tópico t_i dado que ya fue clasificado bajo el tópico t_k, y puede ser también estimada en términos del modelo **M** como sigue:

$$P_{\varphi}(t_i|t_k) = \frac{\sum_{t_j \epsilon V}[(M[t_i, t_j] \wedge M[t_k, t_j]) * |t_j|]}{\sum_{t_j \epsilon V}(M[t_k, t_j] * |t_j|)} \quad (3)$$

Las ecuaciones (2) y (3) tienen plena concordancia con los argumentos presentados en sección 6.3, donde se establece que *la relevancia es una noción conceptual primitiva* y se sugiere que la definición de $P_{\varphi}(t_j)$ y $P_{\varphi}(t_j|t_i)$ en términos de relevancia es más natural que la definición de relevancia en términos de estas medidas de probabilidad.

Existe una gran cantidad de formas en que los modelos de relevancia propuestos pueden ser mejorados. Por ejemplo, los modelos menos conservadores podrían ser combinados con mecanismos que los prevengan de derivar relaciones de relevancia entre tópicos a menos que un análisis del contenido de los tópicos sugiera una conexión entre ellos. Este análisis podría basarse en el texto que describe los tópicos, el cual está disponible en ODP. Otra fuente de contenidos son las características de los sitios web asociados a los tópicos, tales como el texto, los enlaces salientes, los enlaces entrantes, o una combinación de todo lo anterior.

Otra mejora posible es la extensión de los modelos propuestos a modelos más intrincados de propagación de relevancia. Las aristas de distintas clases tienen diferentes roles en el grafo de la ontología, y una manera de distinguir estos roles es la asignación de pesos a esas aristas. Entonces, el peso $w_{ij} \epsilon [0,1]$ para una arista entre el tópico t_i y el tópico t_j puede ser interpretado como una medida explícita del grado de "membrecía" de t_j en la familia de tópicos con raíz en t_i. El producto booleano de matrices $\otimes$ debería ser reemplazado por algún operador más adecuado para la propagación de relevancia. Por

ejemplo, podríamos utilizar el operador de composición difusa *MaxProduct* [161] definido sobre matrices, como sigue:

$$[A \odot B]_{ij} = \max_{k}(A_{ik} * B_{kj})$$

El elemento $\mathbf{M}[t_i,t_j]$ resultante de propagar la relevancia en los nuevos modelos difusos será interpretado como una nueva relación de relevancia difusa del tópico t_i al tópico t_j. Para ciertos esquemas de ponderación, la distancia entre dos tópicos en el directorio tendrá un impacto en su valor de relevancia.

Finalmente, es importante distinguir la propagación de relaciones de relevancia de la propagación de palabras clave y de pesos de palabras clave, a través de una estructura de tópicos. En la sección 6.4 fueron revisados dos enfoques para la propagación de palabras clave ([129], [158]). En estos enfoques, las palabras clave son propagadas a través de los tópicos -siguiendo la componente jerárquica de un directorio de tópicos- o a tópicos vecinos. En esta sección también se afirma que el mecanismo de propagación propuesto en los enfoques mencionados podría ser extendido, guiado por los modelos de propagación de relevancia descriptos aquí. En otras palabras, pueden implementarse esquemas de propagación más complejos si el contenido es propagado del tópico t_i al tópico t_j siempre que $\mathbf{M}[t_i,t_j]\neq 0$ para un modelo dado **M**.

7.6 Conclusiones

Para llevar a cabo la validación estadística de los modelos de propagación de relevancia propuestos en el capítulo anterior, fue necesario llevar a cabo distintas tareas que permitieron obtener los parámetros estadísticos que se explican en este capítulo. En primera instancia, se efectuaron análisis cualitativos y cuantitativos para tener una idea preliminar acerca de la calidad de las relaciones inferidas por los modelos elaborados. Luego, fue

realizado el diseño del experimento que serviría de base para obtener los datos necesarios respecto del criterio de los usuarios. A continuación, el trabajo consistió en la selección de los tópicos -y sus correspondientes páginas web- que pudieran ser incorporados en el experimento, siendo esta una tarea muy compleja debido a las condiciones que dichos tópicos debían cumplir. Finalmente, debía coordinarse y ejecutarse el experimento con usuarios, para poder analizar y evaluar estadísticamente los modelos, a la luz de los resultados obtenidos. Si bien ninguno de los dos modelos trabajados en el primer y el segundo experimento parecía tener una significación estadística más alta que el otro, al cambiar la perspectiva del análisis se pudo determinar que existía un aporte fundamental en la determinación del concepto de relevancia por parte de los modelos propuestos, como consecuencia de las respuestas de los usuarios en el experimento realizado.

Aunque algunos modelos parecen aproximar de mejor manera la noción de relevancia que otros, la posibilidad de definir modelos precisos de propagación de relevancia considerando solamente aspectos estructurales parece ser nula, debido a ciertas dificultades generales. Este resultado tiene consecuencias prácticas y teóricas interesantes, ya que muchos métodos existentes intentan identificar relaciones semánticas implícitas en representaciones de redes, solamente observando la estructura o topología de la red (por ejemplo, [162]; [142]). Esto promueve la investigación y el desarrollo de mecanismos que integren aspectos estructurales con otras características -tales como contenido u otros aspectos conceptuales- para derivar modelos mejorados de propagación de relevancia.

En este sentido, el análisis de la estructura y el contenido puede ser provechosamente integrado de dos formas. Primero, los modelos estructurales de propagación de relevancia propuestos pueden ser mejorados tomando en consideración el contenido. Segundo, los modelos existentes de propagación de contenido tales como los

propuestos en [158] y [129] -discutidos en la sección de trabajos relacionados del capítulo anterior- pueden ser reformulados para propagar palabras clave y sus valores de ponderación a través de nuevos caminos inducidos por los modelos de propagación de relevancia.

Capítulo 8:

Conclusiones Generales y Trabajo Futuro

8.1 Introducción

En esta tesis se llevó a cabo la explotación de diversas características estructurales de modelos ingenieriles y de grandes conjuntos de información. Una forma muy útil de representar muchos aspectos de las estructuras mencionadas es la utilización de matrices. Para el caso particular de la Ingeniería, las representaciones matriciales que fueron objeto de este trabajo corresponden a los sistemas de ecuaciones que describen ciertas características de los respectivos modelos matemáticos estudiados. Dichas matrices indican la incidencia de las correspondientes variables en cada ecuación del modelo. Diversos esquemas de Diseño de Instrumentación, Simulación y Optimización en Ingeniería Química fueron abordados para analizar y validar los métodos de particionamiento estructural de matrices de incidencia propuestos. Por el lado de las Ciencias de la Información, el trabajo giró en torno a representaciones matriciales de la estructura de un directorio de sitios de internet. En este caso, cada matriz representa las relaciones existentes entre distintos tópicos pertenecientes al directorio, de acuerdo a un modelo de relaciones de relevancia dado. Tanto para Ingeniería como para Ciencias de la Información, las matrices trabajadas son binarias, es decir, sus elementos son unos y ceros únicamente, e indican la presencia o no de una relación en el modelo.

En líneas generales, el punto en común entre los trabajos realizados en el marco de esta tesis ha sido el análisis estructural. Mediante la representación matricial como pilar fundamental, se logró exponer diversas mejoras en la eficiencia y el manejo de numerosos modelos matemáticos. En los distintos casos de aplicación, la tarea consistió en llevar a cabo operaciones y algoritmos sobre las matrices y los grafos que estas representan, para obtener estructuras equivalentes que provean una mejora en la organización y eficiencia del

modelo, o también un incremento de la cantidad de información útil obtenida del mismo. Además, fueron llevados a cabo distintos procesos de validación empírica y estadística sobre algunos modelos nuevos de propagación de relevancia sobre grandes estructuras de datos vinculadas al ámbito de las Ciencias de la Información, y se elaboró el marco de trabajo necesario para la validación estadística de los métodos de particionamiento de matrices de incidencia trabajados en la disciplina de la Ingeniería de Sistemas de Proceso.

8.2 Resultados del trabajo

A continuación se enumeran y detallan las distintas contribuciones científicas que fueron fruto de esta tesis, según las correspondientes disciplinas.

8.2.1 Contribuciones en la disciplina de Ingeniería

8.2.1.1 Incorporación de una variante en el MDE

Fue llevada a cabo la programación e implementación de un ordenamiento intermedio de las ecuaciones del sistema que se envía como entrada al MDE [50]. Este ordenamiento es llevado a cabo en el algoritmo del MDE antes de realizar la tarea de obtención de un pareamiento maximal, durante la fase de descomposición gruesa. Las ecuaciones son ordenadas de menor a mayor de acuerdo a su Grado de No-Linealidad, con el objetivo de formar los pares ecuación-variable privilegiando para la elección a las ecuaciones lineales.

En algunos de los casos de aplicación preliminares, los resultados mostraron una clara ventaja del MDE con la variante implementada sobre sus predecesores, el MDE sin dicha variante y el MD. El siguiente paso en este sentido es la verificación exhaustiva de esta variante implementada en el algoritmo de particionamiento original, para chequear si

existe una mejora significativa y considerable del MDE con la variante de ordenamiento de ecuaciones respecto del MDE original y su antecesor el MD.

8.2.1.2 Estudio del desempeño de los métodos de particionamiento en casos de estudio de Diseño de Instrumentación y Simulación/Optimización, en Ingeniería Química

Se llevó a cabo la evaluación de los tres métodos de particionamiento que fueron objeto de trabajo de esta tesis, en distintos casos de estudio de las correspondientes áreas de Ingeniería Química [51]. Para cada caso se verificó si existían mejoras en la eficiencia por resolución en bloques de los problemas originales, y si existía algún aporte respecto a la cantidad relativa de variables observables como resultado de los particionamientos. También la evaluación se realizó respecto de las mejoras que pudiera ofrecer un método de particionamiento sobre los otros.

Se observó que el tiempo de resolución de los modelos completos como una unidad es, en la mayoría de los casos, mucho más eficiente que la resolución de los bloques en forma secuencial, luego de realizar el particionamiento del modelo. Aun así, esto abre la posibilidad de un estudio más profundo sobre la forma de utilizar los particionamientos, ya que la pérdida de eficiencia se debe principalmente a la necesidad de inicializar y resolver cada bloque por separado. Posiblemente, si se pudiera incorporar alguno de los métodos de particionamiento dentro de una plataforma de resolución de modelos, la eficiencia de resolución se vería enormemente beneficiada.

8.2.1.3 Elaboración de una plataforma para generación de casos prácticos de aplicación y validación estadística de los métodos de particionamiento

Dada la dificultad de generar modelos de casos reales o académicos adecuados para determinar el grado de utilidad real de los métodos de particionamiento, se construyó una plataforma de software para automatizar esta tarea. Esta plataforma permite generar un número arbitrario de casos de estudio aleatorios según sea necesario. Los casos generados responden a un conjunto de parámetros estadísticos que pueden ser ingresados a la plataforma por el usuario. Estos parámetros determinan distintas características de las matrices de incidencia que se construyen, como por ejemplo la densidad, tamaño y cantidad de los bloques cuadrados para la Forma Triangular Inferior en Bloques de dicha matriz.

El propósito de la elaboración de esta plataforma es la validación estadística de los distintos métodos de particionamiento estudiados. Con ella se podrá determinar con criterios estadísticos si existen mejoras sustanciales al utilizar particionamientos estructurales, como los desarrollados en el grupo de investigación, para resolver los modelos matemáticos observados. Una tarea adicional que se requiere para la implementación correcta de esta plataforma es el ajuste de los parámetros estadísticos que servirán para la generación de los casos de estudio. Para realizar dicha tarea, se debe cargar la información correspondiente a un número considerable de casos de estudio reales o académicos. Dicha tarea constituye un proyecto en marcha que dará el soporte necesario para obtener los resultados buscados.

8.2.2 Contribuciones en la disciplina de Ciencias de la Información

8.2.2.1 Construcción de nuevos Modelos de Propagación de Relevancia en un directorio de internet

Para abordar el tema del tratamiento estructural de grandes volúmenes de datos mediante representaciones matriciales, se llevó a cabo un trabajo sobre modelos de propagación de relevancia temática provenientes del Proyecto de Directorio Abierto (Open Directory Project, ODP) [52] [18]. En las representaciones matriciales de los modelos mencionados se llevaron a cabo operaciones de distintas clases, las cuales permitieron el descubrimiento de relaciones de relevancia no observadas hasta ese momento entre los tópicos del directorio. Este hallazgo es muy importante, ya que la mecánica de trabajo utilizada puede ser extendida a distintos directorios de información y con diversas clases de operaciones aplicadas a las matrices respectivas.

Dentro de este marco de trabajo, también fue desarrollada una herramienta de software para la visualización de la estructura del directorio bajo estudio, que facilitó la observación de los tópicos intervinientes en cada una de las relaciones de relevancia modeladas. A lo largo de todo el trabajo llevado a cabo sobre los modelos de relevancia, esta herramienta fue clave para los diferentes análisis realizados.

8.2.2.2 Validación empírica y estadística de los modelos generados

Luego de elaborar los modelos de propagación de relevancia mencionados, siguiendo con el trabajo sobre ODP, se llevó a cabo una validación empírica de dichos modelos, con la intervención de usuarios humanos [18]. Para ello se diseñó un experimento, a través del cual estos usuarios decidían si las relaciones de relevancia indicadas por los

nuevos modelos realmente eran válidas o no. Cuatro de los modelos, que inicialmente parecían ser mejores que los demás, fueron seleccionados para el experimento.

Los datos arrojados por este experimento fueron analizados mediante el uso de técnicas estadísticas. Luego del análisis, se pudo concluir que existía un aporte significativo de los modelos en su conjunto. Si bien las estadísticas finales no permitían determinar si alguno de los modelos analizados experimentalmente era mejor que los otros, la cantidad de respuestas de los usuarios que expresaban la existencia de una relación entre los tópicos mostrados era estadísticamente más significativa que la cantidad de respuestas que expresaban lo contrario. Esto último dio un valor estadístico muy importante a los modelos generados.

8.2.2.3 Detección de comunidades temáticas en el directorio web analizado mediante algoritmos sobre grafos

Otra forma de trabajo propuesta en esta tesis fue la implementación de un algoritmo de detección de componentes fuertemente conexas sobre grafos dirigidos, en la estructura de uno de los modelos más conservadores de propagación de relevancia del directorio web. Para ello, se elaboró una nueva implementación del algoritmo mencionado, para poder ejecutarlo sobre la estructura básica de ODP.

Los hallazgos remarcables de este trabajo consisten en los datos obtenidos de las componentes fuertes que se encontraron. Se pudo ver que existe una gran cantidad de nodos aislados, mientras que también se observó una gran componente que contiene cerca del 48% de los tópicos del directorio. Queda pendiente un trabajo de análisis más profundo sobre las comunidades temáticas encontradas, para ver posibles usos de las mismas en otras investigaciones.

8.3 Conclusiones generales

Respecto del tema central de la tesis, Análisis Estructural, los trabajos desarrollados a lo largo de todos los capítulos se enfocan en el tratamiento de matrices que reflejan la estructura de una variedad de modelos matemáticos. Un concepto central dentro del Análisis Estructural es el de "Particionamiento Estructural", abordado en la primera parte de esta tesis. Allí se muestran explicaciones detalladas de los distintos métodos de particionamiento estructural estudiados, los cuales tienen como objetivo el hallazgo de bloques de asignación con características particulares, dentro de las matrices de incidencia correspondientes a los sistemas de ecuaciones de los modelos abarcados. Una cuestión esencial para el abordaje de dichos métodos de particionamiento estructural fue la apertura de los mismos a nuevos campos de aplicación. Tal vez sería muy interesante la inclusión de alguno de estos métodos en diversos programas de software asociados a la simulación y optimización en las distintas áreas de la ciencia. Sin embargo, el trabajo desarrollado aquí puso en evidencia la necesidad de un contacto permanente y exhaustivo con los especialistas de las diferentes áreas que se desea explorar. Si bien los principios matemáticos que rigen a los modelos que se pretende estudiar pueden ser bastante similares, como por ejemplo el tratamiento de ecuaciones diferenciales, la ganancia que pueda obtenerse por la incorporación del particionamiento de las correspondientes matrices de incidencia queda supeditada al área de la ciencia respectiva. Esto quiere decir que la significación del aumento en la eficiencia de la resolución de un problema queda a criterio de los usuarios de cada modelo y sus necesidades.

En la Ingeniería en general, permanentemente deben encararse problemas de optimización con una gran diversidad de patrones estructurales. Como se vio en el Capítulo

3 de esta tesis, un problema de optimización puede tener restricciones de desigualdad. En esa sección se explicó el inconveniente presentado cuando se particionan estos modelos. Es por ello que una profundización del estudio sobre este aspecto, incluyendo por ejemplo técnicas metaheurísticas combinadas con los métodos de particionamiento, podría lograr métodos más robustos para la resolución de esta clase de problemas.

Debe hacerse una mención especial también a las tareas de programación llevadas a cabo a lo largo de este trabajo. Se desarrollaron distintas herramientas que permitieron lograr los objetivos planteados a cada momento. Un ejemplo de esto lo constituye la herramienta de visualización desarrollada para la jerarquía de ODP, que permitió observar la estructura de este directorio y hallar la información correspondiente a cada tópico requerido. También, para la realización de los experimentos sobre los modelos de propagación de relevancia, se llevaron a cabo diversas tareas de programación para la localización exhaustiva de tópicos y enlaces candidatos para cada tripla, y por supuesto para la puesta en marcha del experimento en lo que a diseño y programación web se refiere. Sobre este ámbito también corresponde mencionar que se extendió el código fuente correspondiente a los métodos de particionamiento estudiados, pero preservando siempre la eficiencia de los mismos, sin comprometer su rendimiento general.

En otro orden de cosas, un eje central para este trabajo estuvo constituido por la estadística y sus herramientas para determinar la validez de una hipótesis. Por el lado de los métodos de particionamiento, se ha iniciado la elaboración de una plataforma que permita la generación y evaluación de casos de matrices de incidencia, lo cual constituye el puntapié inicial para la validación empírica y estadística de la ganancia en eficiencia que puede surgir del uso de esos métodos. Respecto de los modelos de propagación de relevancia temática sobre ODP, fueron diseñados y ejecutados dos experimentos, sobre los

cuales se obtuvieron importantes resultados mediante estudios estadísticos que permitieron afirmar la significación de las relaciones entre tópicos derivadas por dichos modelos. Es sabido que la estadística brinda las herramientas necesarias para demostrar la validez de una hipótesis científica, y esto fue utilizado de manera recurrente en los distintos capítulos de esta tesis.

Como conclusión final, los dos grandes mundos que fueron explorados aquí, con todas sus diferencias y similitudes desde el punto de vista científico, representan ámbitos muy interesantes y extensos para la investigación. En los albores del presente trabajo, se vislumbraba una conexión más extensa entre los métodos de particionamiento en el área de la Ingeniería de Sistemas de Proceso y el trabajo estructural a realizar sobre los modelos de datos correspondientes a las Ciencias de la Información. Tal vez, se esperaba tener la posibilidad de evaluar la implementación de alguno de los métodos de particionamiento sobre los volúmenes de datos correspondientes al directorio web utilizado, pero esto no fue posible debido a los objetivos sustancialmente diferentes de las dos áreas de la ciencia exploradas. Aun así, los objetivos alcanzados en cada una de las áreas por separado realmente son muy alentadores, y generaron una vastedad notable de proyectos que pueden ser explotados luego.

8.4 Trabajo futuro

De acuerdo a las líneas de investigación cubiertas en esta tesis, se desprenden diversos estudios que pueden ser llevados a cabo sobre las mismas bases. Teniendo en cuenta las dos grandes áreas exploradas, se detallan a continuación las perspectivas de trabajo futuro.

8.4.1 Perspectivas de trabajo futuro en el área de Ingeniería

8.4.1.1 Explorar nuevas posibilidades de mejora para el MDE

Además del ordenamiento de ecuaciones incluido en el algoritmo original, sería conveniente investigar otras técnicas, herramientas y enfoques para incluir en dicho algoritmo. Por ejemplo, llevar a cabo un estudio exhaustivo sobre los valores de GNL que se asignan a cada tipo de término, y también trabajar sobre distintos modelos matemáticos provenientes de diversos ámbitos de la ciencia, para determinar los valores óptimos de los parámetros según el caso. El objetivo de esta tarea sería lograr una herramienta que permita aumentar la cantidad de variables observables que se obtengan del nuevo algoritmo de particionamiento, o disminuir la no linealidad en los bloques formados. En otras palabras, el objetivo sería aumentar la calidad de los bloques de asignación obtenidos como resultado de aplicar los particionamientos a los sistemas de ecuaciones correspondientes.

8.4.1.2 Llevar a cabo la aplicación y evaluación de los métodos de particionamiento sobre una gran cantidad de casos reales y académicos de Ingeniería Química de probado diseño

Si bien los casos sobre los que se trabajó en esta tesis son bien representativos del Diseño de Instrumentación y la Simulación/Optimización en Ingeniería Química, será muy útil contar con una cantidad importante de casos prácticos provenientes de la bibliografía y de la realidad [163]. Para completar la colección de un adecuado conjunto de casos de estudio, es necesario llevar a cabo un trabajo de relevamiento de modelos matemáticos. Con toda esta información, se podría realizar un estudio de desempeño (benchmarking). Sería posible también visualizar de manera más clara cuáles pueden ser las mejoras a

aplicar sobre los métodos, y también sacar conclusiones con valor estadístico respecto de su desempeño.

8.4.1.3 Identificar nuevas áreas de aplicación para los métodos de particionamiento estructural

Existen modelos matemáticos de diversas áreas de la ciencia que tienen características muy similares a los modelos de Diseño de Instrumentación de Plantas Químicas. Se debería indagar sobre la posibilidad de incluir los métodos de particionamiento como preprocesadores de la estructura de los modelos de dichas áreas, con el propósito de evaluar las mejoras que pudieran producirse en los mismos. Una característica sobresaliente de los modelos matemáticos en general es su crecimiento en tamaño y complejidad, a medida que la representación se acerca a la realidad. Es sobre este aspecto que se puede sacar provecho de un reordenamiento estructural.

8.4.1.4 Investigar las posibilidades de implementación de los particionamientos para preprocesamiento de Sistemas de Ecuaciones Diferenciales

Al modelar matemáticamente ciertos sistemas teniendo en cuenta el tiempo como factor variable, surgen dependencias funcionales que solamente se pueden expresar mediante ecuaciones diferenciales de diversas complejidades. Un procesamiento previo de los sistemas de ecuaciones que surgen de estos modelos tal vez podría resultar muy útil para la resolución de los mismos.

8.4.1.5 Evaluar las posibilidades de paralelización dentro de los métodos de particionamiento

Según la estructura de los bloques de asignación generados para los casos de estudio abarcados, podrían generarse bloques que no tengan dependencias entre sí. Esto significaría que dichos bloques se podrían abordar en paralelo, con el consecuente impacto en la eficiencia que eso podría involucrar. Por lo tanto, sería necesario llevar a cabo las investigaciones correspondientes acerca de la conveniencia de paralelizar o no, teniendo en cuenta las características estructurales que pueden surgir de los modelos matemáticos de las diferentes órbitas de la Ingeniería.

8.4.1.6 Estudiar la incorporación de los métodos de particionamiento en paquetes de software conocidos y difundidos en la comunidad científica

A lo largo del trabajo realizado en esta tesis, se pudo observar que los particionamientos obtenidos podrían optimizar la resolución de los sistemas de ecuaciones asociados. Sin embargo, para que estos estudios sean fructíferos y los métodos de particionamiento estructural estudiados puedan ser utilizados ampliamente, deberían incorporarse ciertas características especiales dentro de algún paquete de software de utilización masiva. Para el trabajo en esta tesis, la plataforma utilizada para implementar los distintos problemas de simulación y optimización fue GAMS®. Debido a que cada bloque de asignación de un problema es tratado como un problema independiente, surge la necesidad de reinicializar los datos de la plataforma para resolver cada uno de estos bloques generados. Esto involucra un tiempo adicional por la finalización de cada bloque y el inicio del siguiente. Esta dificultad podría ser abordada en futuras investigaciones.

8.4.2 Perspectivas de trabajo futuro en Ciencias de la Información

8.4.2.1 Profundizar el estudio sobre los modelos de propagación de relevancia elaborados

Una posible mejora de los mismos podría ser el refinamiento de las relaciones establecidas entre los tópicos del directorio. Este refinamiento puede realizarse mediante la incorporación de información relacionada con los textos incluidos en las descripciones de los tópicos y las páginas catalogadas en cada uno de ellos, o también por la utilización de matrices reales en lugar de binarias para establecer el grado de relación entre tópicos de acuerdo a la distancia entre los mismos. También se podría emplear un esquema de ponderación para evaluar las relaciones que surgieran entre los tópicos.

8.4.2.2 Llevar a cabo la elaboración de modelos de propagación de relevancia en otros grandes corpus de información científica

Al igual que los modelos generados para ODP en esta tesis, se podría efectuar la construcción de modelos de propagación de relevancia para conjuntos de datos como por ejemplo WordNet [150] o GeneOntology [149]. Estas grandes bases de datos tienen ontologías asociadas que permitirían realizar estudios de gran valor para obtener información de interés científico. Además, los trabajos que se realizaran sobre estos corpus podrían servir de base para su implementación en los innumerables repositorios de datos científicos de libre acceso que existen en la web.

8.4.2.3 Extender los resultados de los modelos generados a la obtención de valores de similitud semántica más precisos

Siguiendo el procedimiento descripto en [3], serían utilizados los nuevos modelos de propagación de relevancia para llevar a cabo el procedimiento de cálculo de los valores de similitud semántica sobre los documentos contenidos en los tópicos del directorio. Posiblemente con este trabajo, se obtendría un aumento en la precisión de los resultados por la incorporación de los modelos generados a lo largo de este trabajo de tesis. Para este fin, se cuenta con una gran cantidad de información relacionada con búsquedas en el popular buscador de internet Google, que permitiría llevar a cabo evaluaciones más profundas y detalladas sobre el desempeño de los modelos como clasificadores para la obtención de valores de similitud semántica.

8.4.2.4 Explorar las posibles aplicaciones de los nuevos modelos en casos diversos de aplicación

Además de su conocida aplicación en la determinación de valores de similitud semántica sobre documentos [3], sería interesante verificar la utilidad de estos modelos de propagación de relevancia en otros casos de aplicación, como el particionamiento en clústeres de grandes estructuras de información o la minería de datos sobre grandes corpus de documentos.

8.4.2.5 Verificar posibles usos en la estructura de las redes sociales actuales para los modelos de propagación de relevancia

Es sabido que las redes sociales han sido y seguirán siendo objeto de estudios científicos para muy diversos propósitos ([66], [164], [130], [98], [100]). En gran medida pueden representar la opinión y los sentimientos de grandes grupos sociales, y además son

una fuente inagotable de contenidos de información actual y sumamente útil para muchísimas personas y organizaciones. Por ello, todos los estudios que puedan hacerse sobre las estructuras de datos que se generan sobre sus flujos de información son proclives a tener un valor muy considerable. Como consecuencia de todo esto, sería de gran interés la generación de modelos de propagación de relevancia sobre ontologías construidas para bloques de contenidos extraídos de redes sociales ampliamente utilizadas.

Referencias Bibliográficas

[1] L. Cucek, M. Martín, I. E. Grossmann and Z. Kravanja, "Energy, water and process technologies integration for the simultaneous production of ethanol and food from the entire corn plant", *Computers & Chemical Engineering,* vol. 35, no. 8, pp. 1547-1557, 2011.

[2] M. Carnero, J. Hernandez and M. Sanchez, "Comparación de Estrategias para el Diseño Óptimo de Instrumentación en Plantas de Proceso", *Información tecnológica,* vol. 16, pp. 57-63, 00 2005.

[3] A. Maguitman, F. Menczer, F. Ferdinc, H. Roinestad and A. Vespignani, Algorithmic Computation and Approximation of Semantic Similarity, WWW Journal. Springer Science+Business Media B.V, 2006.

[4] V. M. Carvalho, "INPUT-OUTPUT NETWORKS: A SURVEY", *Complexity Research Initiative for Systemic InstabilitieS,* 2012.

[5] D. Hughes, "Generalized incidence matrices over group algebras", *Illinois Journal of Mathematics,* vol. 1, no. 4, pp. 545-551, 1957.

[6] D. R. Fulkerson and O. A. Gross, "Incidence matrices and interval graphs", *Pacific J. Math,* vol. 15, no. 3, pp. 835-855, 1965.

[7] D. G. Kendall, "Incidence matrices, interval graphs and seriation in archaeology", *Pacific J. Math,* vol. 28, no. 3, pp. 565-570, 1969.

[8] W. M. Kantor, "On incidence matrices of finite projective and affine spaces", *Mathematische Zeitschrift,* vol. 124, no. 4, pp. 315-318, 1972.

[9] P. Sin, "The Elementary Divisors of the Incidence Matrices of Points and Linear Subspaces in $P^n(F_p)$", *Journal of Algebra,* vol. 232, no. 1, pp. 76-85, 2000.

[10] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins and J. Wiener, "Graph structure in the Web", *Computer Networks,* vol. 33, no. 1–6, pp. 309-320, 2000.

[11] C. Balbuena, "Incidence matrices of projective planes and of some regular bipartite graphs of girth 6 with few vertices", *SIAM Journal on Discrete Mathematics,* vol. 22, no. 4, pp. 1351-1363, 2008.

[12] T. Drummond, I. S. Duff, R. Guivarch, D. Ruiz and M. Zenadi, "Partitioning strategies for the Block Cimmino algorithm", *Journal of Engineering Mathematics,* pp. 1-19, 2013.

[13] D. S. Kershaw, "The incomplete Cholesky—conjugate gradient method for the iterative solution of systems of linear equations", *Journal of Computational Physics,* vol. 26, no. 1, pp. 43-65, 1978.

[14] J. Unger, A. Kroner and W. Marquardt, "Structural analysis of differential-algebraic equation systems—theory and applications", *Computers & chemical engineering,* vol. 19, no. 8, pp. 867-882, 1995.

[15] D. Henrion and J.-B. Lasserre, "Convergent relaxations of polynomial matrix inequalities and static output feedback", *Automatic Control, IEEE Transactions on,* vol. 51, no. 2, pp. 192-202, 2006.

[16] I. S. Duff and J. Koster, "The design and use of algorithms for permuting large entries to the diagonal of sparse matrices", *SIAM Journal on Matrix Analysis and Applications,* vol. 20, no. 4, pp. 889-901, 1999.

[17] I. B. Tjoa and L. T. Biegler, "Simultaneous solution and optimization strategies for parameter estimation of differential-algebraic equation systems", *Industrial & Engineering Chemistry Research,* vol. 30, no. 2, pp. 376-385, 1991.

[18] E. Xamena, N. B. Brignole and A. G. Maguitman, "A study of relevance propagation in large topic ontologies", *Journal of the American Society for Information Science and Technology; Wiley Online Library,* vol. 64, no. 11, pp. 2238-2255, 2013.

[19] E. Xamena, N. B. Brignole and A. G. Maguitman, "Strongly Connected Components Detection in Open Directory Project Graphs", in *Mecánica Computacional*, Salta, Salta, Argentina, 2012.

[20] R. Tarjan, "Depth-First Search and Linear Graph Algorithms", *SIAM J. Comput.*, no. 1, pp. 146-160, 1972.

[21] P. Berkhin, "A Survey of Clustering Data Mining Techniques", in *Grouping Multidimensional Data*, Springer Berlin Heidelberg, 2006, pp. 25-71.

[22] D. Bollegala, Y. Matsuo and M. Ishizuka, "A relational model of semantic similarity between words using automatically extracted lexical pattern clusters from the web", in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2*, Stroudsburg, PA, USA, 2009.

[23] C. Zha, Y. Dou, M. Guo and Y. Dong, "A New Hybrid Clustering Algorithm Based on Simulated Annealing", in *Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2013 5th International Conference on*, 2013.

[24] S. Kirkpatrick and M. Vecchi, "Optimization by simmulated annealing", *science,* vol. 220, no. 4598, pp. 671-680, 1983.

[25] N. Castet, "A Parallel Graph Partitioner for STAPL", 2013.

[26] A. Buss, I. Papadopoulos, O. Pearce, T. Smith, G. Tanase, N. Thomas, X. Xu, M. Bianco, N. M. Amato, L. Rauchwerger and others, "STAPL: standard template adaptive parallel library", in *Proceedings of the 3rd Annual Haifa Experimental Systems Conference*, 2010.

[27] Y. Tiany, A. Balminx, S. A. Corsten, S. Tatikonday and J. McPherson, "From "Think Like a Vertex" to "Think Like a Graph"", *IBM Almaden Research Center,* 2013.

[28] R. L. Bowerman, P. H. Calamai and G. Brent Hall, "The spacefilling curve with optimal partitioning heuristic for the vehicle routing problem", *European Journal of Operational Research,* vol. 76, no. 1, pp. 128-142, 1994.

[29] S. A. Jacobs, N. Stradford, C. Rodriguez, S. Thomas and N. M. Amato, "A scalable distributed RRT for motion planning", in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[30] A. Ben-Dor, R. Shamir and Z. Yakhini, "Clustering gene expression patterns", *Journal of computational biology,* vol. 6, no. 3-4, pp. 281-297, 1999.

[31] P. D'haeseleer, "How does gene expression clustering work?", *Nature biotechnology,* vol. 23, no. 12, pp. 1499-1502, 2005.

[32] R. Sharan and R. Shamir, "CLICK: a clustering algorithm with applications to gene expression analysis", in *Proc Int Conf Intell Syst Mol Biol*, 2000.

[33] E. Segal, H. Wang and D. Koller, "Discovering molecular pathways from protein interaction and gene expression data", *Bioinformatics,* vol. 19, no. suppl 1, pp. i264-

i272, 2003.

[34] P. Claes, D. K. Liberton, K. Daniels, K. M. Rosana, E. E. Quillen, L. N. Pearson, B. McEvoy, M. Bauchet, A. Z. A., W. Yao, H. Tang, G. S. Barsh, D. M. Absher, D. A. Puts, J. Rocha, S. Beleza, R. W. Pereira, G. Baynam, P. Suetens, D. Vandermeulen, J. K. Wagner, J. S. Boster and M. D. Shriver, "Modeling 3D Facial Shape from DNA", *PLoS Genet,* vol. 10, no. 3, p. e1004224, 03 2014.

[35] M. Schulz, F. Krause, N. L. Novere, E. Klipp and W. Liebermeister, "Retrieval, alignment, and clustering of computational models based on semantic annotations", *Molecular Systems Biology,* vol. 7, 2011.

[36] V. Fionda and L. Palopoli, "Biological network querying techniques: analysis and comparison", *Journal of Computational Biology,* vol. 18, no. 4, pp. 595-625, 2011.

[37] M. Emmerich, M. Grotzner and M. Schutz, "Design of graph-based evolutionary algorithms: A case study for chemical process networks", *Evolutionary Computation,* vol. 9, no. 3, pp. 329-354, 2001.

[38] J. Moreno and D. Dochain, "Global observability and detectability analysis of uncertain reaction systems", in *IFAC World Congress, Prague,* 2005.

[39] M. R. Maurya, R. Rengaswamy and V. Venkatasubramanian, "Application of signed digraphs-based analysis for fault diagnosis of chemical process flowsheets", *Engineering Applications of Artificial Intelligence,* vol. 17, no. 5, pp. 501-518, 2004.

[40] M. R. Maurya, R. Rengaswamy and V. Venkatasubramanian, "A signed directed graph and qualitative trend analysis-based framework for incipient fault diagnosis", *Chemical Engineering Research and Design,* vol. 85, no. 10, pp. 1407-1422, 2007.

[41] I. Ponzoni, M. C. Sánchez and N. B. Brignole, "Direct Method for Structural Observability Analysis", *Industrial & Engineering Chemistry Research,* vol. 43, no. 2, pp. 577-588, 2004.

[42] I. Ponzoni, "Aplicación de Teoría de Grafos al Desarrollo de Algoritmos para Clasificación de Variables", Tesis Doctoral en Ciencias de la Computación, Directores: Simari G., Brignole N.B., UNS, Bahía Blanca, 2001.

[43] I. Ponzoni, G. E. Vazquez, M. C. Sánchez and N. B. Brignole, "A Computer-Aided DSS for Observability Analysis", *Signal Processing, Communications and Computer Science,* pp. 222-227, 2000.

[44] A. O. Domancich, Nuevas Estrategias de Particionamiento para Matrices Ralas Generales: Aplicaciones Matemáticas y Tecnológicas, Tesis Doctoral en Ingeniería Química, Directores: Brignole N.B., Hoch P.A., UNS, Bahía Blanca, 2009.

[45] A. O. Domancich, M. Durante, S. Ferraro, P. Hoch, N. B. Brignole and I. Ponzoni, "How to Improve the Model Partitioning in a DSS for Instrumentation Design", *Industrial & Engineering Chemistry Research,* vol. 48, no. 7, pp. 3513-3525, 2009.

[46] B. Park, Y. Won, J. Chung, M.-s. Kim and J. W.-K. Hong, "Fine-grained traffic classification based on functional separation", *International Journal of Network Management,* vol. 23, no. 5, pp. 350-381, 2013.

[47] M. W. Berry, S. T. Dumais and G. W. O'Brien, "Using linear algebra for intelligent information retrieval", *SIAM review,* vol. 37, no. 4, pp. 573-595, 1995.

[48] P. Husbands, H. Simon and C. Ding, "On the use of the singular value decomposition

for text retrieval", *Computational information retrieval,* pp. 145-156, 2001.

[49] P. Serdyukov, H. Rode and D. Hiemstra, "Modeling multi-step relevance propagation for expert finding", in *Proceedings of the 17th ACM conference on Information and knowledge management*, 2008.

[50] E. Xamena, A. G. Maguitman and N. B. Brignole, "Optimized resolution of systems of equations", in *XVIII Congreso Argentino de Ciencias de la Computación*, 2012.

[51] E. Xamena, B. Cañete, A. G. Maguitman and N. B. Brignole, "Particionamiento estructural de modelos de plantas de procesos", in *11ª Congreso Interamericano de Computación Aplicada a la Industria de Procesos*, 2013.

[52] E. Xamena, N. B. Brignole and A. G. Maguitman, "Computational Models of Relevance Propagation in Web Directories", Córdoba, Córdoba, Argentina - 29/8-2/9/2011, 2011.

[53] B. E. Borders, "Systems of Equations in Forest Stand Modeling", *Forest Science,* vol. 35, no. 2, pp. 548-556, 1989.

[54] A. Bruckstein, D. Donoho and M. Elad, "From Sparse Solutions of Systems of Equations to Sparse Modeling of Signals and Images", *SIAM Rev.*, vol. 51, no. 1, p. 34–81, 2009.

[55] T. B. Benjamin, J. L. Bona and J. J. Mahony, "Model Equations for Long Waves in Nonlinear Dispersive Systems", *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences,* vol. 272, no. 1220, pp. 47-78, 1972.

[56] J. Wright, M. Yi, J. Mairal, G. Sapiro, T. S. Huang and Y. Shuicheng, "Sparse Representation for Computer Vision and Pattern Recognition", *Proceedings of the IEEE,* vol. 98, no. 6, pp. 1031 -1044, june 2010.

[57] L. Grigori, P.-Y. David, J. W. Demmel and S. Peyronnet, "Brief announcement: Lower bounds on communication for sparse Cholesky factorization of a model problem", in *Proceedings of the 22nd ACM symposium on Parallelism in algorithms and architectures*, Thira, Santorini, Greece, ACM, 2010, pp. 79-81.

[58] D. A. Spielman and S.-H. Teng, "Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems", in *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, Chicago, IL, USA, ACM, 2004, pp. 81-90.

[59] S. Abbasbandy, "Improving Newton–Raphson method for nonlinear equations by modified Adomian decomposition method", *Applied Mathematics and Computation*, vol. 145, no. 2-3, pp. 887 - 893, 2003.

[60] T. J. Ypma, "Historical Development of the Newton-Raphson Method", *SIAM Review,* vol. 37, no. 4, pp. 531-551, 1995.

[61] J. Hopcroft and R. Karp, "An $n^{5/2}$ Algorithm for Maximum Matchings in Bipartite Graphs", *SIAM Journal on Computing,* vol. 2, no. 4, pp. 225-231, 1973.

[62] A. Gibbons, Algorithmic graph theory, Cambridge University Press, 1985.

[63] D. Cvetkovic, P. Rowlinson and S. Simic, "An introduction to the theory of graph spectra", *Cambridge-New York,* 2010.

[64] M. Schubert and E. Steffen, "The set of circular flow numbers of regular graphs",

Journal of Graph Theory, vol. 76, no. 4, pp. 297-308, 2014.

[65] P. Harish and P. Narayanan, "Accelerating large graph algorithms on the GPU using CUDA", in *High performance computing--HiPC 2007*, Springer, 2007, pp. 197-208.

[66] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel and B. Bhattacharjee, "Measurement and analysis of online social networks", in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, 2007.

[67] J. A. Romagnoli and M. C. Sanchez, Data processing and reconciliation for chemical process operations, vol. 2, Academic Press, 1999.

[68] V. Melkonian, "The Paired Assignment Problem", *Open Journal of Discrete Mathematics*, vol. 2014, 2014.

[69] C. A. R. Hoare, "Quicksort", *The Computer Journal*, vol. 5, no. 1, pp. 10-16, 1962.

[70] A. O. Domancich, M. Maidana, P. M. Hoch, N. B. Brignole and I. Ponzoni, "MP4SO: A Model-Partitioning Software for Simulation and Optimization", *Computer Aided Chemical Engineering*, vol. 27, pp. 471-476, 2009.

[71] A. Brooke, D. Kendrick and A. Meeraus, "GAMS Release 2.25", *A User's Guide*, *boyd & fraser publishing company*, 1992.

[72] A. Domancich, N. Brignole and P. Hoch, "Structural analysis of reactive distillation columns", *VirtualPro*, no. 84, p. 18, 2009.

[73] H. Nilsson, J. Peterson and P. Hudak, "Functional Hybrid Modeling from an Object-Oriented Perspective.", in *EOOLT*, 2007.

[74] E. Carpanzano and C. Maffezzoni, "Symbolic manipulation techniques for model simplification in object-oriented modelling of large scale continuous systems", *Mathematics and Computers in Simulation*, vol. 48, no. 2, pp. 133-150, 1998.

[75] S. Bike, "Design of an ammonia synthesis plant", *CACHE case study*, 1985.

[76] M. Mohler, R. Bunescu and R. Mihalcea, "Learning to grade short answer questions using semantic similarity measures and dependency graph alignments", in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, Stroudsburg, PA, USA, 2011.

[77] R. Navigli, "Word sense disambiguation: A survey", *ACM Comput. Surv.*, vol. 41, no. 2, pp. 10:1--10:69, 2009.

[78] J. Reimand, S. Hui, S. Jain, B. Law and G. D. Bader, "Domain-mediated protein interaction prediction: From genome to network", *FEBS Letters*, 2012.

[79] H. H. H. B. M. van Haagen, P. A. C. 't Hoen, A. de Morrae, W. M. C. van Roon-Mom, D. J. M. Peters, M. Roos, B. Mons, G.-J. van Ommen and M. J. Schuemie, "In silico discovery and experimental validation of new protein-protein interactions", *PROTEOMICS*, vol. 11, no. 5, pp. 843-853, 2011.

[80] G. Yu and L.-G. Wang, "Disease Ontology Semantic and Enrichment", 2012.

[81] S. Nikolova, J. Boyd-Graber and C. Fellbaum, "Collecting semantic similarity ratings to connect concepts in assistive communication tools", in *Modeling, Learning, and Processing of Text Technological Data Structures*, Springer, 2012, pp. 81-93.

[82] C. A. Joslyn, J. D. Cohn, K. M. Verspoor and S. M. Mniszewski, "Automating

Ontological Function Annotation: Towards a Common Methodological Framework", *Bio-Ontologies,* 2005.

[83] C. Pesquita, D. Faria, A. O. Falcao, P. Lord and F. M. Couto, "Semantic similarity in biomedical ontologies", *PLoS computational biology,* vol. 5, no. 7, p. e1000443, 2009.

[84] C. Joslyn, K. Verspoor and D. Gessler, "Knowledge Integration in OpenWorlds: Utilizing the Mathematics of Hierarchical Structure", in *International Conference on Semantic Computing*, 2007.

[85] A. Fern, A. Polleres and S. Ossowski, *Towards Fine-grained Service Matchmaking by Using Concept Similarity,* 2007.

[86] J. Tekli, "Semantic and Structure Based XML Similarity: An integrated Approach", in *In proc. of the 13th International Conference on Management of Data (COMAD)*, 2006.

[87] M. Dumas, L. G. Bañuelos and R. Dijkman, "Similarity Search of Business Process Models", *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering,* 2009.

[88] R. Dijkman, M. Dumas, B. van Dongen, R. Karik and J. Mendling, "Similarity of business process models: Metrics and evaluation", *Information Systems,* vol. 36, no. 2, pp. 498-516, 2011.

[89] F. Ferri, A. Formica, P. Grifoni and M. Rafanelli, "Query Approximation by Semantic Similarity in GeoPQL", in *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops*, vol. 4278, R. Meersman, Z. Tari and P. Herrero, Eds., Springer Berlin / Heidelberg, 2006, pp. 1670-1680.

[90] J. Diederich and W.-T. Balke, "Topic-Based User Models: Design & Comparison", in *Proceedings 10th Delos Workshop on Personalised Access, Profile Management and Context Awareness in DL (PersDL),* 2007.

[91] P. chia Chang and L. M. Quiroga, *Using Wikipedia Content to Derive an Ontology for Modeling and Recommending Web Pages across Systems,* 2004.

[92] S. P. Ponzetto and M. Strube, "Knowledge derived from Wikipedia for computing semantic relatedness", *Journal of Artificial Intelligence Research,* p. 212, 2007.

[93] S. P. Ponzetto and M. Strube, "Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution", in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, Stroudsburg, PA, USA, 2006.

[94] G. Jeh and J. Widom, "SimRank: a measure of structural-context similarity", in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, 2002.

[95] C. Li, J. Han, G. He, X. Jin, Y. Sun, Y. Yu and T. Wu, "Fast computation of SimRank for static and dynamic information networks", in *Proceedings of the 13th International Conference on Extending Database Technology*, New York, NY, USA, 2010.

[96] D. Lizorkin, P. Velikhov, M. Grinev and D. Turdakov, "Accuracy estimate and optimization techniques for SimRank computation", *The VLDB Journal,* vol. 19, no.

1, pp. 45-66, 2010.

[97] P. Zhao, J. Han and Y. Sun, "P-Rank: a comprehensive structural similarity measure over information networks", in *Proceedings of the 18th ACM conference on Information and knowledge management*, New York, NY, USA, 2009.

[98] R. Schifanella, A. Barrat, C. Cattuto, B. Markines and F. Menczer, "Folks in Folksonomies: social link prediction from shared metadata", in *Proceedings of the third ACM international conference on Web search and data mining*, New York, NY, USA, 2010.

[99] N. H. Phan, V. D. T. Hoang and H. Shin, "Adaptive combination of tag and link-based user similarity in flickr", in *Proceedings of the international conference on Multimedia*, New York, NY, USA, 2010.

[100] H. Lin, J. Davis and Y. Zhou, "An Integrated Approach to Extracting Ontological Structures from Folksonomies", in *The Semantic Web: Research and Applications*, vol. 5554, L. Aroyo, P. Traverso, F. Ciravegna, P. Cimiano, T. Heath, E. Hyvanen, R. Mizoguchi, E. Oren, M. Sabou and E. Simperl, Eds., Springer Berlin / Heidelberg, 2009, pp. 654-668.

[101] A. Maedche and S. Staab, "Ontology learning for the Semantic Web", *Intelligent Systems, IEEE*, vol. 16, no. 2, pp. 72-79, mar-apr 2001.

[102] M. H. Seddiqui and M. Aono, "Metric of intrinsic information content for measuring semantic similarity in an ontology", in *Proceedings of the Seventh Asia-Pacific Conference on Conceptual Modelling - Volume 110*, Darlinghurst, Australia, Australia, 2010.

[103] C. d'Amato, N. Fanizzi and F. Esposito, "A Semantic Similarity Measure for Expressive Description Logics", *CoRR*, vol. abs/0911.5043, 2009.

[104] P. Ziegler, C. Kiefer, C. Sturm, K. Dittrich and A. Bernstein, "Detecting Similarities in Ontologies with the SOQA-SimPack Toolkit", in *Advances in Database Technology - EDBT 2006*, vol. 3896, Y. Ioannidis, M. Scholl, J. Schmidt, F. Matthes, M. Hatzopoulos, K. Boehm, A. Kemper, T. Grust and C. Boehm, Eds., Springer Berlin / Heidelberg, 2006, pp. 59-76.

[105] A. Bernstein, E. Kaufmann, C. Barki and M. Klein, "How Similar Is It? Towards Personalized Similarity Measures in Ontologies", in *In 7. Internationale Tagung Wirtschaftsinformatik*, 2005.

[106] A. Bernstein, E. Kaufmann, C. Kiefer and C. Barki, "SimPack: A Generic Java Library for Similarity Measures in Ontologies", 2005.

[107] D. Sanchez, M. Batet, D. Isern and A. Valls, "Ontology-based semantic similarity: A new feature-based approach", *Expert Systems with Applications*, vol. 39, no. 9, pp. 7718-7728, 2012.

[108] F. Ferri, A. Formica, P. Grifoni and M. Rafanelli, "Evaluating Semantic Similarity Using GML in Geographic Information Systems", in *On the Move to Meaningful Internet Systems 2005: OTM 2005 Workshops*, vol. 3762, R. Meersman, Z. Tari and P. Herrero, Eds., Springer Berlin / Heidelberg, 2005, pp. 1009-1019.

[109] W. B. Croft, D. Metzler and T. Strohman, Search engines: Information retrieval in practice, Addison-Wesley Reading, 2010.

[110] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, "The WEKA data mining software: an update", *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10-18, 2009.

[111] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Pacsca and A. Soroa, "A study on similarity and relatedness using distributional and WordNet-based approaches", in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Stroudsburg, PA, USA, 2009.

[112] P. Boldi and S. Vigna, "The WebGraph framework I: Compression techniques", in *Proceedings of the 13th international conference on World Wide Web*, New York, NY, USA, 2004.

[113] N. J. Belkin, "Helping people find what they don't know", *Communications of the ACM*, vol. 43, no. 8, pp. 58-61, 2000.

[114] M. Balabanovic and Y. Shoham, "Fab: content-based, collaborative recommendation", *Communications of the ACM*, vol. 40, no. 3, pp. 66-72, 1997.

[115] R. Akavipat, L.-S. Wu, F. Menczer and A. Maguitman, "Emerging Semantic Communities in Peer Web Search", *P2PIR 2006: International Workshop on Information Retrieval in Peer-to-Peer, ACM Press, Arlington, USA,* 2006.

[116] M. Kumar and R. Vig, "Design of CORE: context ontology rule enhanced focused web crawler", in *Proceedings of the International Conference on Advances in Computing, Communication and Control*, New York, NY, USA, 2009.

[117] C. C. Aggarwal, F. Al-Garawi and P. S. Yu, "Intelligent crawling on the World Wide Web with arbitrary predicates", in *Proceedings of the 10th international conference on World Wide Web*, New York, NY, USA, 2001.

[118] S. Chakrabarti, M. van den Berg and B. E. Dom, "Focused crawling: a new approach to topic-specific Web resource discovery", *Computer Networks*, vol. 31, pp. 1623-1640, 1999.

[119] M. E. Newman, "The structure and function of complex networks", *SIAM review*, vol. 45, no. 2, pp. 167-256, 2003.

[120] M. M. Wasko and S. Faraj, "Why should I share? Examining social capital and knowledge contribution in electronic networks of practice", *MIS quarterly*, pp. 35-57, 2005.

[121] G. Forman, K. Eshghi and S. Chiocchetti, "Finding similar files in large document repositories", in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, 2005.

[122] I. Keivanloo, L. Roostapour, P. Schugerl and J. Rilling, "Semantic web-based source code search", in *Proc. 6th Intl. Workshop on Semantic Web Enabled Software Engineering*, 2010.

[123] I. Biro, A. Benczúr, J. Szabó and A. G. Maguitman, "A Comparative Analysis of Latent Variable Models for Web Page Classification", *LA-WEB 2008* , 2008.

[124] S. Gauch, A. Chandramouli and S. Ranganathan, "Training a hierarchical classifier using inter document relationships", *J. Am. Soc. Inf. Sci. Technol.*, vol. 60, pp. 47-58, January 2009.

[125] F. Menczer, G. Pant and P. Srinivasan, "Topical web crawlers: Evaluating adaptive algorithms", *ACM Transactions on Internet Technology (TOIT),* vol. 4, no. 4, pp. 378-419, November 2004.

[126] S. Chakrabarti, M. M. Joshi, K. Punera and D. M. Pennock, "The structure of broad topics on the Web", in *Proceedings of the 11th international conference on World Wide Web*, New York, NY, USA, 2002.

[127] S. M. Beitzel, E. C. Jensen, A. Chowdhury and D. Grossman, "Using titles and category names from editor-driven taxonomies for automatic evaluation", in *Proceedings of the twelfth international conference on Information and knowledge management*, New York, NY, USA, 2003.

[128] A. G. Maguitman, R. L. Cecchini, C. M. Lorenzetti and F. Menczer, "Using Topic Ontologies and Semantic Similarity Data to Evaluate Topical Search", in *XXXVI Conferencia Latinoamericana de Informática*, Asuncion, Paraguay, 2010.

[129] J. W. Kim and K. S. Candan, "Leveraging structural knowledge for hierarchically-informed keyword weight propagation in the web", in *Proceedings of the 8th Knowledge discovery on the web international conference on Advances in web mining and web usage analysis*, Berlin, Heidelberg, 2007.

[130] B. Markines, C. Cattuto, F. Menczer, D. Benz, A. Hotho and G. Stumme, "Evaluating similarity measures for emergent semantics of social tagging", *Proceedings of the 18th international conference on World wide web,* pp. 641-650, 2009.

[131] J. A. Swets, "Information retrieval systems", *Science,* vol. 141, no. 3577, pp. 245-250, 1963.

[132] P. Gardenfors, "On the Logic of Relevance", *Syntheses,* vol. 37, no. 3, pp. 351-367, 1978.

[133] L. F. Del Cerro and A. Herzig, "Belief change and dependence", in *Proceedings of the 6th conference on Theoretical aspects of rationality and knowledge*, San Francisco, CA, USA, 1996.

[134] W. Goffman, "On relevance as a measure", *Information Storage and Retrieval,* vol. 2, no. 3, pp. 201-203, 1964.

[135] A. M. Rees and T. Saracevic, The measurability of relevance, Center for Documentation & Communication Research, Western Reserve University, 1966.

[136] C. L. Barry, "User-defined relevance criteria: an exploratory study", *JASIS,* vol. 45, no. 3, pp. 149-159, 1994.

[137] Y. C. Xu and Z. Chen, "Relevance judgment: What do information users consider beyond topicality?", *Journal of the American Society for Information Science and Technology,* vol. 57, no. 7, pp. 961-973, 2006.

[138] S. Mizarro, "How many relevances in information retrieval?", *Interacting with Computers,* vol. 10, pp. 305-302, 1998.

[139] B. Hjorland, "The foundation of the concept of relevance", *Journal of the American Society for Information Science and Technology,* vol. 61, no. 2, pp. 217-237, 2010.

[140] T. Saracevic, "Relevance: A review of the literature and a framework for thinking on the notion in information science. Part II: The nature of relevance", *Journal of the American Society for Information Science and Technology*, vol. 58, no. 13, pp. 2126-

2144, 2007.

[141] T. Saracevic, "Relevance: A review of the literature and a framework for thinking on the notion in information science. Part III: Behavior and effects of relevance", *Journal of the American Society for Information Science and Technology,* vol. 58, no. 13, pp. 2126-2144, 2007.

[142] R. R. Rada, H. Mili, E. Bicknell and M. Blettner, "Development and application of a metric on Semantic Nets", *IEEE Transactions on Systems, Man and Cybernetics,* vol. 19, no. 1, pp. 17-30, 1989.

[143] J. J. Jiang and D. W. Conrath, "Semantic similarity based on corpus statistics and lexical taxonomy", in *Proceedings of the International Conference on Research in Computational Linguistics (ROCLING X)*, Taiwan, 1998.

[144] C. Joslyn and W. J. Bruno, "Weighted Pseudo-Distances for Categorization in Semantic Hierarchies", in *International Conference on Conceptual Structures. Lecture Notes in Computer Science 3956*, 2005.

[145] P. Resnik, "Using Information Content to Evaluate Semantic Similarity in a Taxonomy", in *IJCAI'95: Proceedings of the 14th international joint conference on Artificial intelligence*, San Francisco, CA, USA, 1995.

[146] A. Tversky, "Features of similarity", *Psychological Review,* vol. 84, no. 4, pp. 327-352, 1977.

[147] D. Lin, "An Information-Theoretic Definition of Similarity", in *Proceedings of the Fifteenth International Conference on Machine Learning*, San Franciso, CA, USA, 1998.

[148] A. Maguitman, F. Menczer, H. Roinestad and A. Vespignani, Algorithmic Detection of Semantic Similarity. Proceedings of WWW 2005, Chiba, Japan, 2005.

[149] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig and others, "Gene Ontology: tool for the unification of biology", *Nature genetics,* vol. 25, no. 1, pp. 25-29, 2000.

[150] G. A. Miller, "Wordnet: a dictionary browser", *Information in Data*, pp. 25-28, 1985.

[151] H. Rode, P. Serdyukov, D. Hiemstra and H. Zaragoza, "Entity ranking on graphs: Studies on expert finding", 2007.

[152] I. Chibane and B.-L. Doan, "Relevance propagation model for large hypertext document collections", in *Large Scale Semantic Access to Content (Text, Image, Video, and Sound)*, Paris, France, France, 2007.

[153] A. M. Zareh Bidoki, P. Ghodsnia, N. Yazdani and F. Oroumchian, "A3CRank: An adaptive ranking method based on connectivity, content and click-through data", *Information processing & management,* vol. 46, no. 2, pp. 159-169, 2010.

[154] N. Dai, B. D. Davison and Y. Wang, "Mining neighbors' topicality to better control authority flow", in *Advances in Information Retrieval*, Springer, 2010, pp. 653-657.

[155] A. Shakery and C. Zhai, "Relevance Propagation for Topic Distillation UIUC TREC 2003 Web Track Experiments", in *TREC*, 2003.

[156] A. Shakery and C. Zhai, "A probabilistic relevance propagation model for hypertext retrieval", in *Proceedings of the 15th ACM international conference on Information and knowledge management*, New York, NY, USA, 2006.

[157] T. Qin, T.-Y. Liu, X.-D. Zhang, G. Feng, D.-S. Wang and W.-Y. Ma, "Topic distillation via sub-site retrieval", *Information Processing & Management,* vol. 43, no. 2, pp. 445-460, 2007.

[158] C. Su, Y. Gao, J. Yang and B. Luo, "An efficient adaptive focused crawler based on ontology learning", in *Fifth International Conference on Hybrid Intelligent Systems, 2005. HIS'05.*, 2005.

[159] P. Bailey, N. Craswell, I. Soboroff, P. Thomas, A. P. Vries and E. Yilmaz, "Relevance assessment: are judges exchangeable and does it matter", in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, New York, NY, USA, 2008.

[160] R. Burgin, "Variations in relevance judgments and the evaluation of retrieval performance", *Information Processing & Management,* vol. 28, pp. 619-627, July 1992.

[161] A. Kandel, Fuzzy Mathematical Techniques with Applications, Boston, MA, USA: Addison-Wesley, 1986, p. 274.

[162] T. Pedersen, S. Patwardhan and J. Michelizzi, "WordNet::Similarity: measuring the relatedness of concepts", in *Demonstration Papers at HLT-NAACL 2004*, Stroudsburg, PA, USA, 2004.

[163] J. Kroschwitz and A. Seidel, Kirk-Othmer Encyclopedia of Chemical Technology, Wiley, 2006.

[164] B. Markines, H. Roinestad and F. Menczer, "Efficient assembly of social semantic networks", in *HT '08: Proceedings of the nineteenth ACM conference on Hypertext and hypermedia*, New York, NY, USA, 2008.