

Grafo Integrador de un Mapa Conceptual Hipermedial

Sergio Martig - Perla Señas

Grupo InE

Departamento de Ciencias de la Computación

Universidad Nacional del Sur

Avda. Alem 1253 – (8000) Bahía Blanca – Argentina

smartig@cs.uns.edu.ar

Resumen

El postulado fundamental de las posturas constructivistas declara que aprender significa construir estructuras de conocimiento. Para promover situaciones de aprendizaje constructivista los ambientes deben poseer facilidades para la construcción de conocimiento. Para realizarlo dentro de un Ambiente de Mapas Conceptuales Hipermediales (MCH) resulta de sumo interés poder visualizar todas las vistas del MCH en forma integrada. Para ello se le asocia al mapa un grafo denominado Grafo Integrador (GI_{MCH}). Resolver la visualización de dicho grafo no es un problema trivial, si se tiene en cuenta el elevado número de nodos que tendrá en la mayoría de los casos. Se presenta en este trabajo la definición, justificación desde lo pedagógico y la solución para la visualización de un GI_{MCH} .

Palabras Claves:

Tecnología Avanzada en Educación - Aprendizaje Significativo - Mapas Conceptuales Hipermediales - Visualización de Grafos.

1 Introducción:

Los Mapas Conceptuales Hipermediales (MCH) constituyen un recurso poderoso para la representación de las ideas, su uso es beneficioso para investigadores y profesionales de distintas disciplinas, quienes pueden contar con una herramienta específica para la representación de las ideas. Desde lo didáctico, suman la riqueza de los MC y el poder de la hipermedia, constituyéndose en una moderna tecnología educativa para el trabajo sobre meta-aprendizaje y meta-conocimiento.

Las representaciones pictóricas de grafos han sido una herramienta de modelado tradicional para una variedad de aplicaciones en ingeniería de software, bases de datos, visualización, administración de proyectos, redes de computadoras, etc. Dichas representaciones fueron y son usadas intensivamente, incluso en forma manual. En la actualidad la generación automática de los dibujos de grafos encuentra muchas aplicaciones: diagramas de flujo de datos, jerarquías de clases orientadas a objetos, diagramas de entidad relación, cartas de estructura de sistemas, redes de Petri, diagramas de transición de estados, esquemas de circuitos, representación del conocimiento, entre otras tantas y en particular se los propone en este trabajo para la representación del Grafo Integrador de un MCH (GI_{MCH}).

2 Mapas Conceptuales Hipermediales

Los MCH son esquemas de representación de conocimiento, se basan en los Mapas Conceptuales de Novak (MC) e incorporan la flexibilidad y riqueza que permite la tecnología hipermedial. En el área educativa, ambos esquemas han sido probados con éxito como potentes estructuras capaces de contribuir con la construcción de aprendizajes significativos [Zan98].

En la búsqueda de soluciones a los problemas de aprendizaje, se ha encontrado en los MC una potente herramienta para lograr aprendizajes significativos, para poder asumir la propia elaboración de significados, en definitiva para poder “aprender a aprender”. En los últimos años se ha incrementado su uso en las aulas, no sólo como herramienta de aprendizaje sino también como técnica de evaluación. Sin embargo, se ha detectado que existen aspectos operacionales que dificultan la realización de estos mapas, si se piensa en su construcción mediante elementos tradicionales, especialmente cuando el mapa cuenta con un elevado

número de conceptos, o cuando es necesario hacer refinamientos del mismo o cuando se desea interconectar mapas ya existentes. La aplicación de tecnología informática hipermedial facilita la construcción de los MC, salvando las dificultades antes mencionadas. Además y principalmente, en los MCH, se favorece sustancialmente el trabajo de selección y de jerarquización de conceptos, enriqueciendo de esta manera su valor pedagógico. Siguiendo esta motivación se crearon los MCH, una metodología de desarrollo y una plataforma específica que permite la creación, mantenimiento, lectura e interconexión de estos mapas.

Un MCH es un documento hipermedial; cada nodo de la hipermedia contiene una colección de conceptos relacionados entre sí por palabras enlace. A cada uno de estos nodos se lo denomina *vista* del MCH. Cada vista puede ser visualizada en una ventana y es caracterizada por un nombre y por un color.

Se distinguen dos tipos de conceptos: los *propios* de la vista y los *importados* a la misma. Los primeros corresponden a aquellos que constituyen inicialmente la vista y los segundos son los que se toman desde otra para poder así establecer relaciones entre conceptos de distintas vistas. Las relaciones entre conceptos de una misma vista se denominan *relaciones internas* y las relaciones entre conceptos de distintas vistas se denominan *relaciones externas*.

Gráficamente los conceptos propios se representan mediante elipses rotuladas, los conceptos importados por rectángulos rotulados, ambos con el nombre del concepto que representan, y las relaciones (internas o externas) por arcos etiquetados con palabras enlace.

Para representar las relaciones externas se establece un arco etiquetado entre el concepto propio y el concepto importado. Dicha relación aparece en ambas vistas.

Se dice que un concepto C perteneciente a una Vista1 *explota* en otra Vista2 cuando dicho concepto C se desarrolla en la Vista2. Es decir, C constituye el nodo raíz del MC desarrollado en Vista2. En este caso se dice que el MC representado en la Vista2 es un *submapa* del correspondiente a la Vista1.

La elipse que representa un concepto propio que explota en otra vista es un *botón* que posibilita el acceso directo a esa vista. El rectángulo que representa un concepto importado C es un *botón* que permite acceder directamente a la vista donde C está definido como propio.

Cada vista se identifica con un nombre, el del concepto propio más abarcativo de dicha vista, y con un color, que es usado en todas las elipses que representan conceptos propios. Los conceptos importados mantienen el color de la vista donde fueron definidos originalmente. Es recomendable que una vista cuente entre siete y diez conceptos como máximo; desde un punto de vista psicológico, esta limitación tiene relación con el número máximo de bloques de información que se pueden retener después de un breve intervalo de percepción [Nov84] y desde un punto de vista práctico, se trata de tener una visión panorámica (en una sola pantalla) de los conceptos más relevantes y de sus relaciones. Se debe destacar que si se pretende un MCH con más de siete conceptos, necesariamente se deberá dejar en una primer vista los más relevantes (siete a lo sumo), creando otra vista que explotará a partir de los conceptos seleccionados.

Es posible asociar a un concepto terminal (que no es un botón) distintas apariencias (gráfico, sonido, animación, etc). Todo concepto puede tener asociada una apariencia ya que siempre existirá una vista del MC donde dicho concepto esté definido como terminal.

Con el propósito de poder alcanzar los beneficios formativos del trabajo con MC enunciados por Novak [Nov85] y de poder aprovechar las ventajas de su representación hipermedial, es importante contar con una metodología de desarrollo de MCH y con una plataforma ad-hoc para la creación, lectura e interconexión de estos mapas [Señ96b] [Mor96].

La realización de nuevos aprendizajes, implica en algunos casos, una revisión de los conocimientos anteriormente adquiridos y en todos los casos una integración de lo nuevo con lo previamente incorporado. "Para trabajar en este sentido, resulta de interés, un ambiente para desarrollar interconexiones de MCH. La principal motivación para esta tarea está dada por la necesidad de integrar los nuevos conocimientos con los conocimientos

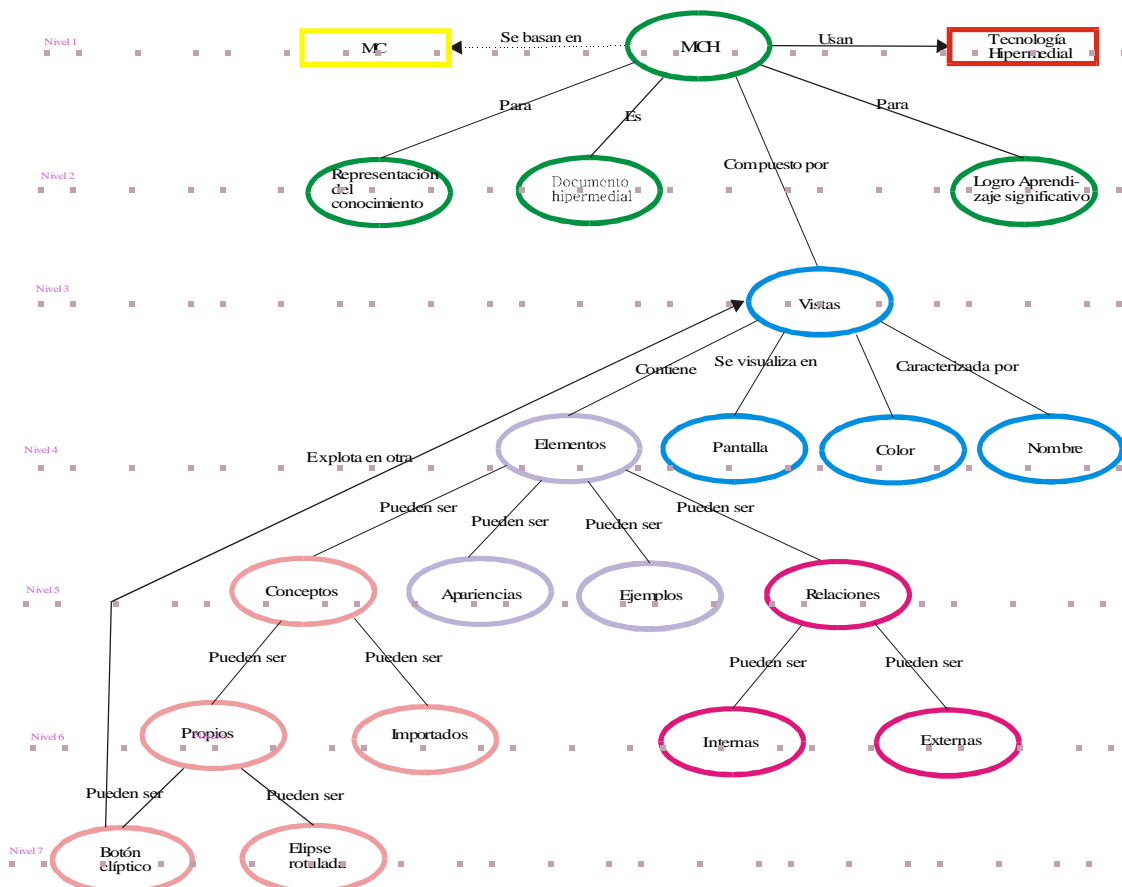
previos y por el deseo de relacionar distintos puntos de vista de un mismo tema...Se trata de una propuesta que trasciende la mera simplificación de las tareas en los procesos de enseñanza-aprendizaje, y que se presenta como una herramienta efectiva para la construcción del conocimiento, de acuerdo con una psicología del aprendizaje constructivista” [Señ99]. El ambiente de aprendizaje que comprende los MCH y la plataforma para su manejo, incluye la posibilidad de la interconexión de mapas, ampliando de este modo, la riqueza de esta técnica como recurso valioso y comprobado para la representación de las ideas [Zan98].

2.1 Grafo Integrador del MCH

Para el trabajo de construcción de conocimiento dentro de un Ambiente MCH resulta de sumo interés poder visualizar todas las vistas de un MCH en forma integrada, como si fuese un MC tradicional al que se le adiciona el código cromático y diferentes apariencias en la representación de los conceptos. Con ese propósito se define el GI_{MCH} , que se obtiene como síntesis de todas las vistas del MCH original (ver figura sobre MCH de siete vistas).

El GI_{MCH} tiene las siguientes características:

- Es un grafo jerárquico. La ubicación de los nodos respeta la jerarquía original del MCH.
- Aparecen todos los conceptos una única vez como nodos del grafo.
- Se respetan las relaciones del MCH original, quedando plasmadas como arcos etiquetados.
- Se mantiene el código cromático y la asociación de apariencias.



Ejemplo de Grafo Integrador de un MCH

3 Visualización de Grafos

Los grafos son estructuras abstractas que pueden ser utilizadas para representar información que pueda ser modelada como objetos y conexiones entre los mismos. Estas

características y el hecho de ser fáciles de leer y entender hacen que sean una herramienta muy útil en diferentes dominios, en particular en el área de representación de conocimiento. La visualización de grafos encara el problema de construir representaciones geométricas de los mismos respetando determinados criterios estéticos y restricciones impuestas por el usuario. Hay varios criterios estándar que se utilizan en la graficación de grafos. Usualmente los vértices son representados por símbolos tales como cajas o puntos, y las aristas por curvas abiertas de Jordan conectando los símbolos que representan a los vértices asociados. Sin embargo esos estándares pueden variar según la aplicación. Además un grafo tiene infinitud de formas de ser dibujado. La utilidad de una representación dada depende de su legibilidad, es decir de la capacidad de transmitir el significado de ese grafo en forma clara y rápida. Los aspectos de legibilidad pueden ser expresados por medio de los criterios estéticos, tales como la minimización de cruces entre aristas o la muestra de simetrías. En este marco varios problemas de visualización de grafos pueden ser formalizados como problemas de optimización de objetivos múltiples, por ejemplo obtener una representación con un mínimo número de cruces y puntos de inflexión y mínima área.

3.1 Técnicas Generales para el dibujo de grafos:

Un algoritmo de visualización de grafos recibirá como dato de entrada el propio grafo G que se quiere visualizar y determinadas restricciones que deben especificarse adecuadamente. Dado que lo percibido por el ser humano varía de una persona a otra, y que aplicaciones diferentes requieren sin duda tipos de representaciones diferentes, el dominio de aplicación surge también como un parámetro esencial en el dibujo de grafos.

Podemos decir entonces que los requerimientos que contribuirán a caracterizar el dibujo a obtener pueden expresarse en términos de los siguientes aspectos:

- a) *Convenciones del dibujo*: Son las reglas básicas que el dibujo debe respetar para ser admisible..
- b) *Estética*: Los criterios estéticos especifican un conjunto de propiedades del gráfico que deberemos aplicar, tanto como sea posible, para mejorar su legibilidad.
- c) *Restricciones*: Mientras que tanto las convenciones como los criterios estéticos se aplican a todo el grafo, las restricciones comúnmente se refieren a subgrafos específicos.

La investigación en el dibujo de grafos se ha enfocado casi exclusivamente en el aspecto algorítmico, en el que el dibujo del grafo es generado de acuerdo a un conjunto de criterios estéticos y convenciones prefijadas.

El enfoque de formalizar los criterios estéticos para la visualización de grafos como metas de optimización es eficiente desde el punto de vista computacional, pero no es muy flexible a la adopción de restricciones impuestas por el usuario ya que las restricciones definidas por éste no son naturalmente soportadas.

El enfoque declarativo formaliza los criterios estéticos especificando la conectividad del grafo mediante un conjunto de restricciones especificadas por el usuario. La conectividad deseada se obtiene entonces como la solución del sistema de restricciones especificadas por el usuario. La ventaja de este enfoque es su poder expresivo. Como desventaja, debemos notar que algunos criterios estéticos, como por ejemplo la planaridad, deben especificarse con restricciones muy complicadas; además, los sistemas generales para resolver restricciones son ineficientes computacionalmente. Por otro lado, se destaca la falta de un lenguaje poderoso en cuanto a su expresividad para la especificación de restricciones. Esto determina que los algoritmos existentes no tengan una gran capacidad para satisfacer restricciones y que la modificación de los mismos para lograrlo sea muy difícil ([Tam88]).

En los últimos años se han realizado varios intentos en el desarrollo de lenguajes para la especificación de restricciones y en el diseño de técnicas para dibujar grafos basadas en sistemas de restricciones ([Kam89],[Mar91],[Den93]). Edades y Lin ([Lin95]) combinan métodos algorítmicos y declarativos en el dibujo de árboles y Brandenburg ([Bra95]) presenta un intento basado en una gramática para grafos.

4 Dibujo de un MCH integrador: Una aplicación del método Jerárquico

De las tres grandes familias de algoritmos utilizados para el dibujado de grafos (Paso de Planarización, Jerárquica y Sistemas de Fuerzas), teniendo en cuenta las características del GI_{MCH} se opta por la segunda.

La metodología de dibujo en capas o jerárquica es altamente intuitiva y tiene el atractivo que puede ser aplicada a cualquier dígrafo, independientemente de sus características teóricas.

Esta estrategia consta de los siguientes tres pasos:

- ¾ *Asignación de capas o niveles:* Asigna los vértices a capas horizontales, es decir en un sistema de ejes cartesianos $x:y$ les asigna una coordenada y .
- ¾ *Reducción de cruces:* Ordena los vértices dentro de cada capa para minimizar el número de cruces entre los arcos.
- ¾ *Asignación de la coordenada horizontal:* Determina para cada vértice una coordenada x .

En caso de que el dígrafo de entrada contenga ciclos, es necesario realizar un paso previo de remoción de ciclos, invirtiendo temporalmente las aristas necesarias para que el dígrafo sea acíclico.

4.1 Asignación de Capas

El objetivo de este paso es asignar una coordenada- y a cada vértice. En otras palabras es transformar un dígrafo acíclico en otro por niveles. Esa transformación debe considerar los siguientes puntos:

- ¾ El dígrafo por niveles debe ser *compacto*. Esto es, su ancho y su alto deben ser lo más pequeños posible. Dado que la distancia entre capas es constante, el límite inferior de la altura está dado por el número de vértices en un paso desde un vértice fuente, a un sumidero.
- ¾ El dígrafo por niveles debe ser *propio*, esto es fácilmente logrado insertando 'vértices ficticios' a lo largo de las aristas largas de la siguiente manera: Se reemplaza cada arista (u,v) que se expanda a más de un nivel con el paso $(v_1, v_2, \dots, v_{k-1}, v_k)$ con $v_1 = u$ y $v_k = v$, siendo v_2, \dots, v_{k-1} vértices ficticios. La necesidad de esto está planteada por el paso de reducción de cruces, que se complica demasiado si tratamos con aristas que se expanden en más de un nivel.
- ¾ El número de vértices ficticios debe ser pequeño por varias razones, entre otras podemos citar:
 - El tiempo que insumen los pasos siguientes es proporcional a la cantidad total de vértices.
 - Los puntos de inflexión en las aristas ocurrirán en las coordenadas asociadas a este tipo de vértices. Dado que las inflexiones de las aristas disminuyen la legibilidad del dibujo es conveniente mantener su número bajo.
 - La cantidad de vértices ficticios en un paso tiene relación directa con su longitud en término de las coordenadas- y . Esta probado que es más claro seguir visualmente el trazado de las aristas cortas que el de aquellas que superen una cierta longitud.

Para el caso particular del grafo integrador de un MCH la pertenencia de los vértices a un determinado nivel está dada por la jerarquía de los conceptos que representa cada vértice dentro del mapa. Por lo tanto el dígrafo de entrada ya tiene asignado a cada vértice un nivel o coordenada- y obtenido durante la etapa de construcción del mismo. Es importante notar que el nivel de un nodo (concepto) está dado por la ubicación física que el autor del MCH le dió a ese concepto dentro de la vista correspondiente

Como el dígrafo obtenido de esta manera ya tiene una asignación de niveles, lo que resta hacer en este paso es llevarlo a la forma de un dígrafo por niveles propio. Para lograr este objetivo debemos analizar cada arista y reemplazarla temporalmente por el paso conteniendo la cantidad de vértices ficticios necesarios para limitar las expansiones a 1.

4.2 Reducción de cruces:

En esta instancia lo que deseamos es construir un dibujo del dígrafo por niveles propio

generado en el paso anterior con un número de cruces aceptable.

El número de cruces de las aristas entre dos niveles consecutivos, no depende de la ubicación precisa de cada vértice dentro de ese nivel, sino del orden relativo de los vértices dentro del mismo. Es decir es un problema de combinatoria (decidir el ordenamiento de los vértices en una capa) y no de geometría (asignarle la coordenada-x a cada vértice). Aún así, sigue siendo un problema extremadamente complejo. El problema de minimizar el cruce de aristas en un dígrafo por niveles es NP-Completo, incluso si hay solamente dos capas [Ead94]. El problema de reducir el número de cruces entre las distintas capas o niveles un dígrafo por niveles, es una generalización del *problema de reducir el número de cruces entre dos capas* L_i y L_{i+1} . Este problema ha sido denominado como el problema de cruces entre dos capas (two layer crossing problem) y se ha trabajado en extenso sobre el mismo [Ead86, Cat88, Jün97].

Para facilitar el abordaje del problema trabajaremos con un dígrafo de dos capas únicamente (two-layered digraph), $D=\{C_1, C_2, A\}$ donde los conjuntos C_1 y C_2 de vértices son disjuntos y el conjunto de aristas $A \in C_1 \times C_2$. Notaremos con x_i , y lo llamaremos ordenamiento de la capa i , al conjunto de coordenadas x asignadas a los vértices de la capa i . Además notaremos con $cruces(D, x_1, x_2)$ al número de cruces que aparecen en un dibujo del dígrafo D si a los vértices de las capas C_1 y C_2 se les asignan las coordenadas x según x_1 y x_2 respectivamente.

Podemos expresar al número de cruces en un dibujo de un dígrafo con dos capas, ordenadas según x_1 y x_2 respectivamente, de la siguiente manera:

$$cruces(D, x_1, x_2) = \sum_{x_2(u) < x_2(v)} c_{uv}$$

donde c_{uv} es el número de cruces que se producen entre las aristas incidentes en u , con las aristas incidentes en v , cuando $x_2(u) < x_2(v)$ con $u, v \in C_2$.

4.2.1 Algoritmos para reducir el número de cruces

Dado que el problema del número de cruces es NP mucho se ha hecho para su resolución. Se han desarrollado algoritmos que garantizan encontrar la solución óptima, que si bien no se asegura que terminen en tiempo polinomial han demostrado una performance aceptable en grafos de tamaño pequeño o mediano, considerando como tales a los que presenten hasta 15 nodos en su capa más chica [Jün97] [Mut97]. Otro aspecto a tener en cuenta es realmente cuál es el objetivo, obtener un dibujo con un mínimo número de cruces o la obtención de un dibujo legible, con algunos cruces más.

En respuesta a lo anterior se han desarrollado varios métodos heurísticos para resolver el problema en cuestión, entre los cuales encontramos [Dib99]:

4.2.1.1 Métodos de ordenamiento

Una característica común a este tipo de algoritmos es el cálculo previo de los números de cruces de cada capa. Dentro de ésta categoría podemos nombrar a:

- **Intercambio de Adyacentes:** El cual partiendo de un ordenamiento inicial para los vértices de una capa, los recorre intercambiando aquellos pares de vértices adyacentes (u, v) tal que $c_{uv} > c_{vu}$. Repite el proceso hasta que el número de cruces no varía.
- **Partir:** La idea aplicada en éste método es análoga a la usada en el QuickSort, se selecciona un vértice p que actúa como pivote dentro de cada capa y se clasifican los restantes a la izquierda o derecha de p según sea su número de cruces. Luego se invoca recursivamente con cada uno de los conjuntos de vértices obtenidos.

4.2.1.2 El método del Baricentro y el de la Mediana

Uno de los métodos más comunes es el denominado Baricentro, en el cual la posición de cada vértice u es calculada como el promedio de las posiciones de sus vecinos (en la capa C_1)

$$Prom(u) = 1 / grado(u) \sum_{v \in N_u} x_1(v)$$

En el caso de que dos vértices tengan el mismo valor Prom(u) se les asigna arbitrariamente algún valor próximo. No debemos olvidar que no se calcula un valor exacto de coordenada, sino que las posiciones relativas de los vértices dentro de cada capa.

El método de la Mediana es muy similar al anterior, la diferencia radica en que la posición asignada a un vértice u es determinada por la posición del vecino central de u. En el caso de haber dos vértices con mismo valor med(u)=med(v) se ubica a la izquierda el que tenga grado par, si los dos tuviesen grado par, se resuelve arbitrariamente.

Es fácil probar que ambos métodos dan ordenamientos con cero cruces cuando éste es posible. Sin embargo cuando una solución de cero cruces no existe, ninguno de los dos devuelve el ordenamiento óptimo. [Dib99]

4.2.1.3 Asignación de la coordenada horizontal

En esta instancia ya tenemos un grafo por niveles propio, y dentro de cada nivel un orden relativo de los vértices pertenecientes al mismo.

El objetivo de este paso es asignar a cada vértice un valor para su coordenada horizontal x, manteniendo el ordenamiento dentro de cada capa, pero también logrando que las aristas sean lo más rectas posibles. Es decir, se trata de asignar a cada nodo una coordenada x que permita reducir los ángulos de quiebre de las aristas a valores aceptables. Dado que los puntos de inflexión ocurren en los vértices ficticios introducidos para transformar el GI_{MCH} en un dígrafo por niveles propio, el objetivo de este paso se puede establecer como el siguiente problema de optimización: Dado un paso dirigido $p=(v_1, v_2, \dots, v_{k-1}, v_k)$ siendo v_2, \dots, v_{k-1} vértices ficticios. Si el paso fuera dibujado con una línea recta, entonces para $2 \leq i \leq k-1$, la coordenada x de un vértice v_i , $x(v_i)$, debería satisfacer

$$x(v_i) - x(v_1) = i - 1 / k - 1 (x(v_k) - x(v_1))$$

De donde el alejamiento de la coordenada $x(v_i)$ del ideal que lo colocaría sobre la recta $x'(v_i)$ lo podemos expresar de la siguiente manera:

$$d(v_i) = (x(v_i) - x'(v_i) = x(v_i) - [i - 1 / k - 1 (x(v_k) - x(v_1)) + x(v_1)]$$

Luego para lograr que la arista correspondiente a un paso p sea lo más recta posible deberíamos minimizar

$$\sum_{i=2}^{k-1} d(v_i) = \sum_{i=2}^{k-1} (x(v_i) - [i - 1 / k - 1 (x(v_k) - x(v_1)) + x(v_1)])^2$$

Esto lo podemos generalizar a todos los pasos que involucren vértices ficticios. Una cuestión importante a tener en cuenta es que para asegurar que no se perturbará el orden relativo obtenido en el paso de minimización de cruces de los vértices dentro de un mismo nivel, se deberá imponer algún tipo de restricción que deberán satisfacer todo par de vértices v, w con w a la derecha de v

$$x(w) - x(v) \leq \delta$$

Estamos frente a un problema de programación cuadrática que puede ser resuelto por métodos estándar.

5 Conclusiones

Esta presentación se trata de una propuesta de Informática Educativa que trasciende la mera simplificación de las tareas en los procesos de enseñanza-aprendizaje, y se presenta como una herramienta efectiva para la construcción del conocimiento, de acuerdo con un enfoque constructivista.

La creación del GI_{MCH} que se propone, amplía la riqueza del trabajo con los MCH como recurso valioso y comprobado para la representación de las ideas. Extiende su potencial al permitir trabajar expresamente sobre la relación correcta entre conocimientos recientemente aprendidos y los conocimientos previos o entre diversos enfoques de un mismo tema y fundamentalmente la posibilidad de poder visualizar con claridad el marco contextual de un determinado conocimiento.

Actualmente está en desarrollo la ampliación de la plataforma existente para el trabajo con MCH para incluir la generación automática del grafo integrador y su visualización. El diseño contempla la actualización automática y permanente del GI_{MCH} durante la etapa de autoría.

Bibliografía

- [Aus78] Ausubel, D. P., Novak J. D. *Educational Psychology: A Cognitive View*. 2nd Ed. New York: Holt, Rinehart and Winston. 1978.
- [Bra95] Brandenburg, F. J. *Designing Graph Drawings by Layout Graph Grammars*, *Graph Drawing (Proc. GD '94)*, vol. 894 of *Lecture Notes Comput. Sci.*, Springer-Verlag, 1995.
- [Bru84] Bruner, J. *Acción, pensamiento y lenguaje*. Madrid. Alianza. 1984.
- [Bru94] Bruner, J. *Realidad mental y mundos posibles*. Barcelona. Gedisa. 1994.
- [Cat88] Catarci, T. *The Assignment Heuristic for Crossing Reduction in Bipartite Graphs*, *Proc. 26th. Allerton Conference Commun. Control Comp.*, 1988
- [Coll89] Coll, C. *Aprendizaje escolar y construcción del conocimiento* Ed. Paidós. 1989.
- [Ded94] Dede, C. *Making the most of multimedia. Multimedia and Learning: A school leader's guide*. Alexandria, VA. NSBA.
- [Den93] Dengler, E. M. Friedell, and J. Marks. *Constraint-driven diagram layout*. *Proc. IEEE Sympos. On Visual Languages (VL '93)*, 1993.
- [Dib99] Di Battista, P. Eades, G. Tamassia, R. y Tollis, I. *Graph Drawing> algorithms for the visualization of graphs*, Prentice Hall, 1999.
- [Ead86] Eades, P. and D. Kelly. *Heuristic for reducing Crossing in Two-Layer Networks*. *Ars. Combin.* 1986
- [Ead94] Eades, P. and Wormald, N. *Edge crossings in drawings of bipartite graphs*. *Algorithmica*, 11, 1994.
- [Jün97] Jünger, M. and Mutzel, P. *2-Layer Straightline Crossing Minimization: Performance of exact and heuristics algorithms*. *JGAA*, 1, n. 1, 1997.
- [Kam89] T. Kamada. *Visualizing Abstract Objects and Relations*. *World Scientific Series in Computer Science*, 1989.
- [Lin95] Lin, T. y Eades, P. *Integration of declarative and algorithmic approaches for layout creation*. *Graph Drawing (Proc. GD '94)*, vol. 894, *Lectures notes in Computer Science*, Springer-Verlag, 1995.
- [Laj93] Lajoie, S. *Computer Environments as Cognitive Tools for Enhancing Learning*. 1993. McGill University.
- [Leh93] Lehrer, R. *Authors of knowledge: Patterns of Hypermedia Design*. 1993. University of Wisconsin-Madison.
- [Mar91] Marks, J. *A formal specification for network diagrams that facilitates automatic design*. *Journal of Visual Languages and Computing*, 1991
- [Mor96] Moroni, N. - Vitturini, M. - Zanconi, M. - Señas, P. *Una plataforma para el desarrollo de mapas conceptuales hipermediales*. *Taller de Software Educativo - IV Jornadas Chilenas de Computación*. Valdivia. 1996.
- [Mut97] Mutzel, P. *An Alternative Method to Crossing Minimization on Hierarchical Graphs*.
- [Nov84] Novak, Joseph - Gowin, D. Bob. *Learning how to learn*. Cambridge University Press. 1984.
- [Nov85] Novak, J. *Metalearning and metaknowledge strategies to help students learn how to learn*. *Cognitive Structure and Conceptual Change*. New York. Academic Press. 1985.
- [Señ96a] Señas, P., Moroni, N., Vitturini, M. y Zanconi, M.: *Hypermedial Conceptual Mapping: A Development Methodology*. *13th International Conference on Technology and Education*. University of Texas at Arlington, Department of Computer Science an Engineering. New Orleans 1996.
- [Señ96b] Señas, P., Moroni, N., Vitturini, M. y Zanconi, M.: *Combining Conceptual Mappings and Hypermedia*. *ED-MEDIA96*. Boston. 1996.
- [Tam88] Tamassia, R., Di Battista, G. y Batín, C. *Automatic Graph Drawing and Readability of diagrams*. *IEEE Trans. Syst. Man Cybern.*, SMC-18(1), 1988.
- [Zan98] Zanconi, M., Moroni, N., Vitturini, M., Malet, A., Borel, C. y Señas, P. *Tecnología computacional y meta-aprendizajes*. *RIBIE-98*. 1998.