

NP-completitud fuerte y débil en problemas de etiquetado¹

M. A. Garrido, A. Márquez, J. R. Portillo y P. Reyes²

Resumen

En los Sistemas de Información Geográfica (SIG), es evidente la necesidad de situar etiquetas de texto asociadas a los distintos componentes del mapa y constituye una de las más importantes tareas a realizar. De hecho, la ACM Computational Geometry Impact Task Report califica al *Etiquetado* como área importante de investigación [1]. Las diferentes condiciones impuestas a los lugares a etiquetar y a las etiquetas en sí determinan distintos problemas de etiquetado, muchos de los cuales resultan ser de naturaleza NP-completa. En nuestro trabajo presentamos dos problemas de etiquetado: $H-4S(\mathcal{P}, \mathcal{L})$ y $3-RVP(1)(\mathcal{T}, \mathcal{ER})$. El primero consiste en situar etiquetas rectangulares asociadas a puntos alineados horizontalmente, de manera que cada punto esté contenido en la frontera de su rectángulo asociado. El segundo problema de etiquetado presenta una formulación distinta, tratándose de dibujar etiquetas de área mínima asociadas a ternas de puntos de forma que éstos estén situados sobre la frontera y cumpliendo unas determinadas condiciones.

En este trabajo presentamos un algoritmo pseudo-polinomial para el problema $H-4S$ y demostramos que $3-RVP(1)$ es fuertemente NP-completo. Por tanto, es importante destacar que dentro de una misma clase de problemas, la de *Etiquetado*, encontramos complejidades computacionales opuestas. Además, debemos hacer notar que son los primeros problemas de etiquetado para los que se obtienen este tipo de resultados.

1 Introducción

Un factor clave para conseguir la efectividad de un mapa como medio de comunicación es la localización de sus componentes. Además de la

cartografía, podríamos citar multitud de aplicaciones en las que la visualización de información es una necesidad primordial: organigramas, bases de datos, trazados de grafos, etc. Los datos a mostrar pueden estar asociados a puntos representando ciudades, países, estaciones de metro o a líneas representando ríos, montañas, etc.

Esta línea de investigación conocida como *Etiquetado* ha adquirido una gran relevancia al ser cada vez más variada y densa la información a mostrar. De hecho la ACM Computational Geometry Impact Task Report califica al *Etiquetado* como área importante de investigación [1]. Podemos encontrar una extensa bibliografía sobre el tema en [7].

Las necesidades sobre el diseño del mapa determinan distintas formulaciones para los problemas de etiquetado. Los diferentes datos de entrada sobre los objetos en el mapa y las etiquetas a situar asociados a ellos dan lugar a una gran variedad de problemas de etiquetado a resolver:

- a) Cada etiqueta puede ir asociada a uno o varios objetos como puntos -alineados o no-, conjuntos de puntos, líneas, etc.
- b) De cada etiqueta, en general rectangular, podemos conocer sus dimensiones exactas, área mínima, etc. También podemos tener información sobre las posibles localizaciones de ellas con respecto a su objeto asociado.

En este trabajo nos centramos en el caso de etiquetas rectangulares asociadas a puntos o a ternas de puntos. En ambas variantes los puntos deben estar contenidos en la frontera de su etiqueta asociada. Estos modelos fueron introducidos en [2, 5, 6], obteniéndose la naturaleza NP-completa para la mayoría de ellos.

Dos son los problemas de etiquetado presentados en este trabajo, los cuales son formulados como

¹Trabajo parcialmente subvencionado por los proyectos BFM2001-2474-ORI y PAI FQM-164.

²Dpto. de Matemática Aplicada I. Universidad de Sevilla. E-mail: {vizuete, almar, josera, preyes}@us.es

sigue:

ETIQUETAS DESLIZANTES SOBRE UNA LÍNEA HORIZONTAL (H-4S(\mathcal{P}, \mathcal{L})):

ENTRADA: Un conjunto $\mathcal{P} = \{P_1, \dots, P_n\}$ de n puntos en el plano situados sobre una línea horizontal l y un conjunto $\mathcal{L} = \{L_1, \dots, L_n\}$ de n rectángulos de lados paralelos a los ejes coordenados.

PREGUNTA: ¿Se puede colocar cada rectángulo L_i asociado al punto P_i de forma que dicho punto quede situado sobre alguno de los lados del rectángulo y sin que se produzcan intersecciones entre las etiquetas?

Notemos que aunque el planteamiento general permite que cada punto pueda estar contenido en cualquiera de los lados de su etiqueta asociada, la naturaleza del problema nos indica que nos podemos restringir a los lados horizontales de cada rectángulo. Un problema equivalente a H-4S(\mathcal{P}, \mathcal{L}) sería aquel en el que los puntos están alineados verticalmente y cada uno de ellos debe estar contenido en los lados verticales de su etiqueta asociada.

El segundo problema de etiquetado abordado en este trabajo presenta un planteamiento diferente y podemos formularlo como sigue:

RECTÁNGULOS CON VÉRTICES PREFIJADOS Y ÁREA MÍNIMA (RVP(1)(\mathcal{T})):

ENTRADA: Un conjunto $\mathcal{T} = \{T_1, \dots, T_n\}$ de n ternas de puntos en el plano.

PREGUNTA: ¿Es posible situar una etiqueta rectangular de lados paralelos a los ejes coordenados, asociada a cada terna T_i , de manera que los tres puntos estén contenidos en la frontera de dicho rectángulo, al menos uno de ellos quede situado sobre un vértice y sin que se produzcan intersecciones entre las etiquetas?

Notemos que según las restricciones del problema, la envolvente rectangular de cada trío de puntos debe estar contenida en la etiqueta a situar, por lo que determina un área mínima para ésta. En función de la posición relativa de los tres puntos de cada terna, pueden existir infinitas localizaciones para la etiqueta. Por ello, nos vamos a restringir a un subproblema que denotamos 3-RVP(1)($\mathcal{T}, \mathcal{ER}$), en el que la entrada está compuesta por n ternas de puntos en el plano y para cada una de ellas, a lo más, tres posibles localizaciones para su etiqueta cumpliendo las condiciones expuestas en el problema general.

En trabajos anteriores hemos demostrado la

naturaleza NP-completa de ambos problemas. Las referencias a citar son [3, 6] para H-4S(\mathcal{P}, \mathcal{L}) y 3-RVP(1)($\mathcal{T}, \mathcal{ER}$), respectivamente. En este trabajo probamos que, aún presentando ambos casos la misma naturaleza intratable, existe una gran diferencia que los distingue desde un punto de vista computacional. Daremos en la sección 2 un algoritmo pseudo-polinomial que resuelve el problema H-4S(\mathcal{P}, \mathcal{L}), mientras que en la sección 3 demostramos que 3-RVP(1)($\mathcal{T}, \mathcal{ER}$) es fuertemente NP-completo.

Debemos resaltar el hecho de que dentro de una misma clase de problemas, *Etiquetado*, encontramos naturalezas computacionales muy distintas. Además, ambos casos son los primeros problemas de etiquetado conocidos pertenecientes a dichas clases: débilmente NP-completo para H-4S(\mathcal{P}, \mathcal{L}) y fuertemente NP-completo para 3-RVP(1)($\mathcal{T}, \mathcal{ER}$).

2 Algoritmo pseudo-polinomial para H-4S(\mathcal{P}, \mathcal{L})

Presentamos en esta sección un algoritmo pseudo-polinomial para el problema H-4S(\mathcal{P}, \mathcal{L}) cuando los datos de la entrada son números enteros, siendo así el primer problema de etiquetado para el cual se obtiene dicho resultado. En [3] se prueba su naturaleza NP-completa, construyendo para ello entradas del problema con etiquetas desmesuradamente grandes y pequeñas. Este hecho nos ha llevado a diseñar un algoritmo cuyo tiempo de ejecución depende tanto del tamaño de la entrada, como de las etiquetas.

Para ello hemos de encontrar un algoritmo de forma que para una instancia I del problema nos aporte la respuesta, afirmativa o negativa, en tiempo $p(\text{Length}[I], \text{Max}[I])$ siendo p un polinomio y $\text{Length}[I]$ y $\text{Max}[I]$ dos elementos que medirán la extensión de la entrada y el tamaño de la misma.

Sea una entrada (\mathcal{P}, \mathcal{L}) del problema H-4S, formada por un conjunto $\mathcal{P} = \{P_1, \dots, P_n\}$ de puntos sobre una línea horizontal l (supuestos ordenados de izquierda a derecha) y para cada punto P_i una etiqueta rectangular L_i de lados paralelos a los ejes coordenados y de dimensiones $l_i \times 1$ (hemos supuesto, sin pérdida de generalidad que la altura de las etiquetas es unitaria). Los elementos de medida de la instancia (\mathcal{P}, \mathcal{L}) serán el número de pun-

tos $\text{Length}[I] = n$ y la anchura de la mayor etiqueta $\text{Max}[I] = \max_{i=1, \dots, n} l_i = l_{\max}$.

A continuación damos una serie de conceptos necesarios para la demostración del resultado. Se expondrán de manera intuitiva pudiendo encontrar más detalles en [3]. Llamamos *k-realización* a una solución parcial para el problema correspondiente a los k primeros puntos de la entrada. Si t_k es la abscisa del lado derecho de la etiqueta situada más a la derecha en la parte superior de la línea y b_k es la abscisa análoga para la parte inferior, la *sombra* de una k -realización es la región del plano formada por: el semiplano delimitado por la recta $x = t_k$ (en la parte superior de la línea horizontal) y el semiplano delimitado por la recta $x = b_k$ (en la parte inferior de la línea horizontal) (ver el ejemplo mostrado en la Figura 1).

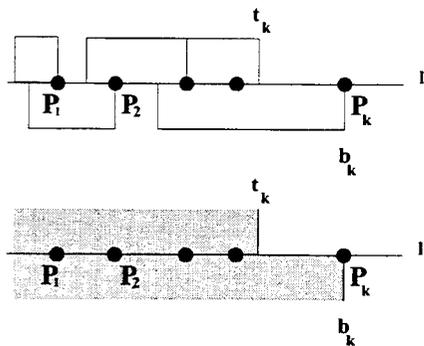


Figura 1: Sombra de una k -realización.

Diremos que una k -realización es una *k-realización optimizada* si todas sus etiquetas están colocadas lo más a la izquierda posible. Notemos que la k -realización mostrada en la Figura 1 es optimizada.

Tras estas definiciones, pasamos a diseñar el algoritmo pseudo-polinomial que resuelve el problema H-4S(\mathcal{P}, \mathcal{L}) cuando la entrada está formada por valores enteros. Como hemos comentado anteriormente, podemos suponer que los puntos de entrada están ordenados en orden creciente según su abscisa.

Teorema 1 *El problema H-4S(\mathcal{P}, \mathcal{L}) para entradas formadas por valores enteros se puede resolver en tiempo y espacio $O(n l_{\max}^2)$ donde n es el número de*

puntos y l_{\max} es el tamaño (anchura) de la mayor etiqueta.

Demostración: La programación dinámica será el método a utilizar en el diseño del algoritmo. Definiremos una matriz T de dimensión $(n+1) \times (2l_{\max}^2 + 1) \times (2l_{\max}^2 + 1)$ y cada elemento tomará un valor lógico: verdadero o falso.

Sea una entrada cualquiera del problema, formada por un conjunto de puntos de abscisas $\{x_1, \dots, x_n\}$ y un conjunto de etiquetas asociadas a cada punto cuyas anchuras son $\{l_1, \dots, l_n\}$.

Sean $k \in \{0, \dots, n\}$, $t, b \in \{-l_{\max}, \dots, l_{\max}\}$. Definimos los elementos $T[k, t, b]$ de la siguiente forma: $T[0, t, b]$ es falso para todo t y b excepto $T[0, -l_{\max}, -l_{\max}]$ que será verdadero; para $k \geq 1$, $T[k, t, b]$ será la respuesta a la siguiente pregunta: "¿Es (t, b) la sombra de alguna k -realización optimizada?" donde t, b se consideran medidos relativos a la abscisa del punto siguiente x_{k+1} (considerando $x_{n+1} = x_n + l_n$), es decir, los valores t y b serían las coordenadas que tendría la sombra si el origen estuviera en P_{k+1} .

Cada elemento del nivel k se obtendrá del nivel $k-1$ en la forma siguiente. Sea $f_k(x) = \max\{-l_{\max}, x - \Delta x_{k, k+1}\}$ la función que a cada punto x medido relativo a x_k le asocia el punto x_{k+1} , con la limitación inferior de $-l_{\max}$. Si $T[k-1, t, b]$ es verdadero entonces señalaremos como verdaderas a lo sumo dos entradas en el nivel k : si $t \leq 0$ generamos una k -realización optimizada colocando en la correspondiente $(k-1)$ -realización optimizada la etiqueta L_k lo más a la izquierda posible, encima de la línea de entrada. Es decir si consideramos $t' = \max\{t, -l_k\}$ haremos $T[k, f_k(t'), f_k(b)]$ verdadero. Similarmente si $b \leq 0$ colocamos la etiqueta L_k lo más a la izquierda posible, por debajo de la línea de entrada. Es decir si consideramos $b' = \max\{b, -l_k\}$ haremos $T[k, f_k(t), f_k(b' + l_k)]$ verdadero. El algoritmo devolverá respuesta afirmativa si alguno de los elementos del nivel n es verdadero y negativa si en alguno de los niveles no existe ningún valor verdadero.

Teniendo en cuenta que en cada nivel añadiremos por cada valor verdadero del nivel anterior dos nuevos valores verdaderos, este cálculo se puede computar en tiempo constante, por lo que el algoritmo corre en tiempo y espacio $O(n l_{\max}^2)$. ■

Hemos obtenido por tanto un algoritmo

pseudo-polinomial que resuelve el problema NP-completo H-4S(\mathcal{P}, \mathcal{L}). Obviamente este algoritmo es más ineficaz en la medida en que aumente el tamaño de las etiquetas, pero también es cierto que cuando tratamos de etiquetar puntos sobre una línea horizontal (como por ejemplo una línea de metro) las etiquetas son generalmente pequeñas.

3 NP-completitud fuerte de 3-RVP(1)($\mathcal{T}, \mathcal{ER}$)

Dedicamos esta sección a demostrar que el problema 3-RVP(1)($\mathcal{T}, \mathcal{ER}$) es fuertemente NP-completo. Este resultado unido al de la sección anterior establece una situación especial en el área del etiquetado: dos problemas de etiquetado NP-completos presentan naturalezas opuestas, esto es, damos un algoritmo pseudo-polinomial para un caso y demostramos la NP-completitud fuerte para el otro.

Centrándonos en la NP-completitud fuerte de 3-RVP(1)($\mathcal{T}, \mathcal{ER}$), vamos a demostrar que un subproblema de él definido polinomialmente es NP-completo. Dicho subproblema verifica que el tamaño métrico de sus entradas (espacio que ocupan en el plano los n tríos de puntos y las tres posibles etiquetas para cada uno de ellos) está acotado linealmente por n . De esta forma obtendríamos el resultado buscado.

Puesto que la prueba es similar a la realizada en [6] para la NP-completitud de 3-RVP(1)($\mathcal{T}, \mathcal{ER}$), omitiremos algunos detalles en este trabajo.

Teorema 2 *El problema 3-RVP(1)($\mathcal{T}, \mathcal{ER}$) es fuertemente NP-completo.*

Demostración: Es fácil ver que 3-RVP(1)($\mathcal{T}, \mathcal{ER}$) pertenece a NP y por lo tanto cualquier subproblema suyo.

Para probar que 3-RVP(1)($\mathcal{T}, \mathcal{ER}$) es fuertemente NP-completo transformaremos de manera polinomial cada entrada UC del problema NP-completo 3-SAT PLANO en una instancia ($\mathcal{T}, \mathcal{ER}$) de nuestro problema, perteneciente a un subconjunto polinomial de entradas. Esta transformación debe conservar la equivalencia entre las respuestas afirmativas de ambos problemas.

Sea UC una entrada arbitraria de 3-SAT PLANO, formada por un conjunto de variables $U = \{u_1, \dots, u_p\}$ y un conjunto de cláusulas $C = \{c_1, \dots, c_q\}$, donde cada cláusula contiene exactamente tres literales (sin pérdida de generalidad, podemos suponer que éstos corresponden a variables distintas). Sea \mathcal{G}_{UC} el grafo bipartito plano que se obtiene de la entrada UC y que contiene un vértice por cada variable u_i (que denotaremos vértice-variable y que tiene a lo más valencia q) y cada cláusula c_j (que denotaremos vértice-cláusula y que tiene valencia 3). El par (u_i, c_j) será una arista del grafo si la cláusula c_j contiene a algún literal de la variable u_i .

Para obtener una inmersión ortogonal plana de \mathcal{G}_{UC} , vamos a sustituir cada vértice-variable u_i por un ciclo de longitud su valencia, de forma que sus vértices adyacentes lo serán, tras la transformación, a un único vértice del ciclo. De esta forma trabajamos con un 3-grafo plano conexo con a lo más $pq + q$ vértices y en virtud de [4] podemos obtener en tiempo lineal en el número de vértices, una inmersión ortogonal plana Γ_{UC} sobre una malla de tamaño $\lfloor \frac{pq+q}{2} \rfloor \times \lfloor \frac{pq+q}{2} \rfloor$ y con, a lo más, $\lfloor \frac{pq+q}{2} \rfloor + 1$ codos.

A partir de esta inmersión ortogonal plana vamos a obtener la entrada de 3-RVP(1)($\mathcal{T}, \mathcal{ER}$), asignando a sus vértices y aristas ternas de puntos y las tres etiquetas rectangulares asociadas a éstas. Para ello necesitamos definir los siguientes elementos: configuraciones de cláusula, configuraciones de variable, cadenas de arista ortogonal, conexiones entre arista y configuración de cláusula, conexiones entre arista y configuración de variable.

Configuración de cláusula: Cada cláusula de UC viene representada en Γ_{UC} por un vértice de valencia tres, el cual, para definir ($\mathcal{T}, \mathcal{ER}$), será sustituido por una terna de puntos y tres posibles etiquetas, cada una de ellas asociadas a cada arista incidente en dicho vértice, es decir, cada etiqueta representa a un literal de la cláusula. La Figura 2 muestra el vértice mediante un círculo blanco, sus aristas incidentes, el trío de puntos y los tres rectángulos.

Conjuntos de variable: Cada variable de UC viene representada en Γ_{UC} por un ciclo ortogonal de, a lo más, q vértices de valencia tres. La arista incidente a cada uno de ellos que no pertenece al ciclo puede estar trazada en el in-

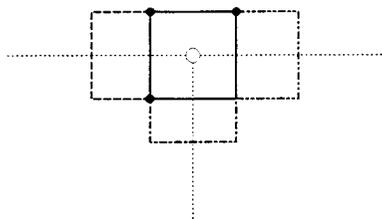


Figura 2: CONFIGURACIÓN DE CLÁUSULA. El círculo blanco y las líneas punteadas representan el vértice-cláusula y sus aristas incidentes. Se muestra la terna de puntos y sus tres etiquetas asociadas.

terior o exterior de éste, aunque se puede definir la misma estructura para ambos casos. Los cambios de dirección que presente el ciclo ortogonal corresponderán, o bien a un vértice o a un codo. Por cada vértice y por cada codo vamos a situar dos ternas de puntos, cada una de ellas con dos posibles localizaciones para la etiqueta. Posteriormente, utilizaremos estas dos ternas para efectuar la conexión con la arista proveniente de la configuración de cláusula. En total, entre todos los conjuntos de variable, estaremos añadiendo $O(pq)$ ternas de puntos a $(\mathcal{T}, \mathcal{ER})$. La Figura 3 muestra algunos casos, en función del tipo de cambio de dirección, que aparecen en los conjuntos de variable. Es importante que se produzcan exclusivamente las intersecciones mostradas entre las distintas etiquetas. Cada una de las dos localizaciones de cada etiqueta corresponderá a uno de los dos valores (verdadero o falso, 0 ó 1) de la variable, de manera que la fijación de uno de ellos fuerza el trazado de los otros para que no se produzcan intersecciones.

Cadena de conjuntos de literal: Una vez diseñadas las configuraciones para los vértices-variable y cláusula hemos de diseñar configuraciones asociadas a las aristas ortogonales de la inmersión Γ_{UC} , que representan la aparición de la variable (en sentido afirmado o negado) en la cláusula correspondiente. Para cada uno de los segmentos rectilíneos de estas aristas utilizaremos una configuración que llamamos *conjunto de literal* y estará formado por 5 ternas de puntos, cada uno de ellos con dos localizaciones distintas para sus etiquetas asociadas y enlazados entre sí como muestra la Figura 4.

Los conjuntos de literal se unirán entre sí me-

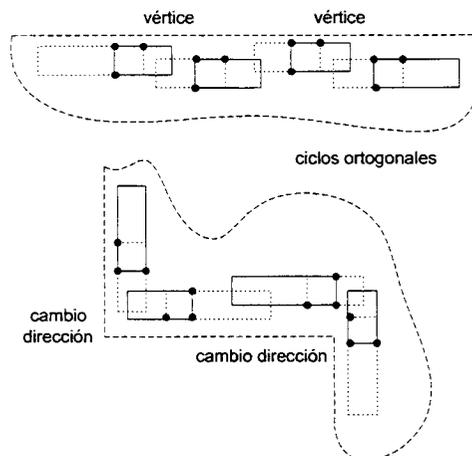


Figura 3: Configuración de variable.

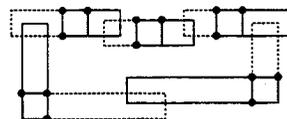


Figura 4: Conjunto de literal.

dante un nuevo trío de puntos para obtener la configuración correspondiente a toda la arista ortogonal, llamada *cadena de conjuntos de literal*. La Figura 5 muestra un ejemplo de una cadena de tres conjuntos de literal correspondiente a una arista con tres segmentos rectilíneos. El resto de posibilidades se obtiene realizando simetrías con las configuraciones de la figura.

De la misma manera que ocurría en la configuración de variable, una vez fijada una de las posibilidades para la etiqueta (trazado continuo o trazado discontinuo) de cualquiera de las ternas de puntos de la cadena de conjuntos de literal, el trazado de las restantes etiquetas queda fijado.

Conexiones entre arista y configuración de cláusula: La Figura 6 muestra la conexión entre la arista ortogonal y la configuración de cláusula, la cual se realiza intersectando una posición de la etiqueta de ésta (la correspondiente a la arista en cuestión) con otra etiqueta del conjunto de literal.

Conexiones entre arista y configuración de variable: Por último conectamos la arista

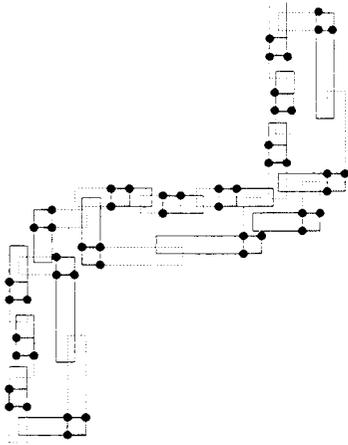


Figura 5: Cadena de conjuntos de literal.

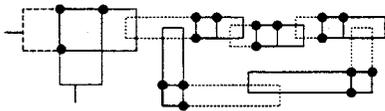


Figura 6: Conexión de una cadena de conjuntos de literal a una configuración de cláusula.

por el otro extremo a las dos ternas de puntos pertenecientes a la configuración de variable correspondiente. Esta conexión se realiza añadiendo dos nuevos tríos de puntos con dos posibles localizaciones para su etiqueta. La Figura 7 muestra el caso de dos vértices consecutivos (sin cambio de dirección) en la configuración de variable. Los restantes casos se obtendrían de manera similar.

De esta manera concluimos la construcción en tiempo polinomial en el número de vértices y cláusulas, de la entrada $(\mathcal{T}, \mathcal{ER})$ del problema 3-RVP(1)($\mathcal{T}, \mathcal{ER}$), la cual contiene $O(pq)$ ternas de puntos en el plano y cada una de ellas tiene asociadas, a lo más, tres posibles situaciones para la localización de su etiqueta. Puesto que la inmersión ortogonal plana Γ_{UC} estaba contenida en una malla de tamaño $\lfloor \frac{pq+q}{2} \rfloor \times \lfloor \frac{pq+q}{2} \rfloor$, la entrada $(\mathcal{T}, \mathcal{ER})$ construida pertenece al subconjunto de instancias de 3-RVP(1)($\mathcal{T}, \mathcal{ER}$) que ocupan un área $O(n) \times O(n)$, siendo n el tamaño de la entrada.

Para finalizar la prueba debemos demostrar que la entrada $(\mathcal{T}, \mathcal{ER})$ del problema 3-RVP(1)($\mathcal{T}, \mathcal{ER}$) es realizable si, y sólo si, la

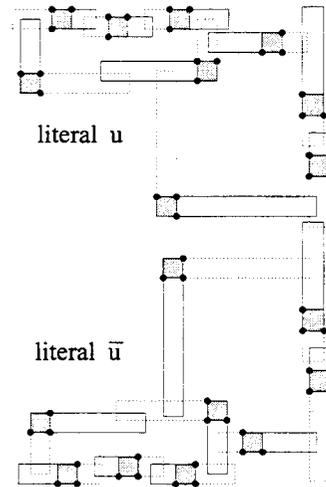


Figura 7: Conexión de una cadena de literal a una configuración de variable. La conexión superior corresponde a una arista que viene de un literal afirmado y la inferior de un literal negado.

entrada UC del problema 3-SAT PLANO es satisficible. La idea principal de este hecho consiste en asociar las localizaciones de las etiquetas mostradas en *trazo continuo* al valor de verdad *verdadero* de las variables. De igual forma el *trazo discontinuo* correspondería al valor falso. Se pueden encontrar todos los detalles de esta parte de la prueba en [6].

■

De esta forma, hemos concluido la demostración de la NP-completitud fuerte del problema 3-RVP(1)($\mathcal{T}, \mathcal{ER}$).

4 Conclusiones y problemas abiertos

El área de *Etiquetado* constituye una de las líneas de investigación de más interés y fructífera hoy en día. Son numerosos y diversos los resultados que podemos encontrar sobre el tema.

Dentro de las múltiples variantes, nos hemos centrados en el caso de etiquetas rectangulares asociadas a puntos o a ternas de puntos, los cuales deben estar contenidos en la frontera de aquellas. Se ha probado anteriormente la naturaleza NP-

completa de los dos problemas abordados y hemos dedicado este trabajo a mostrar la gran diferencia computacional existente entre ellos: el problema $H-4S(\mathcal{P}, \mathcal{L})$ es débilmente NP-completo mientras que $3-RVP(1)(\mathcal{T}, \mathcal{ER})$ es fuertemente NP-completo.

Encontramos, por tanto, dentro de una misma clase de problemas, naturalezas computacionales opuestas. Además, ambos casos son los primeros problemas de etiquetado conocidos pertenecientes a dichas clases.

Como proyectos inmediatos, nos podemos plantear la aplicación de métodos de aproximación para ambos problemas, sobre todo para $3-RVP(1)(\mathcal{T}, \mathcal{ER})$, ya que el algoritmo pseudo-polinomial expuesto para $H-4S(\mathcal{P}, \mathcal{L})$ es en sí una manera de "resolver" el problema. También nos podemos formular variantes de los casos tratados, que sean interesantes desde un punto de vista práctico y estudiar su naturaleza computacional.

Referencias

- [1] B. Chazelle y 36 co-autores. The computational geometry impact task force report. En B. Chazelle, J. E. Goodman y R. Pollack, editores, *Advances in Discrete and Computational Geometry*, vol. 223, pp. 407-463. American Mathematical Society, Providence, 1999.
- [2] M. Formann y F. Wagner. A packing problem with applications to lettering of maps. En *Proc. 7th Annu. ACM Symposium on Computational Geometry (SoCG'91)*, pp. 281-288, 1991.
- [3] M. A. Garrido, C. Iturriaga, A. Márquez, J. R. Portillo, P. Reyes y A. Wolff. ISAAC Labeling Subway Lines, *Lecture Notes in Computer Science*, Springer-Verlag, vol. 2223, pp. 649-659, 2001.
- [4] G. Kant. Drawing planar graphs using the canonical ordering, *Algorithmica*, vol. 16, n.1, pp. 4-32, 1996.
- [5] M. van Kreveld, T. Strijk y A. Wolff. Point labeling with sliding labels, *Computational Geometry: Theory and Applications*, vol. 47, pp. 13:21, 1999.
- [6] J. R. Portillo. *Problemas de conexiones ortogonales*. Tesis doctoral, Universidad de Sevilla, Dpto. Matemática Aplicada I, 2001.
- [7] A. Wolff y T. Strijk. The Map-Labeling Bibliography. <http://www.math-inf.uni-greifswald.de/map-labeling/bibliography/>, 1996.