



La enseñanza de la investigación de operaciones por medio de acertijos matemáticos

M.A. Murray-Lasso

*Unidad de Enseñanza Auxiliada por Computadora
División de Estudios de Posgrado, Facultad de Ingeniería, UNAM
Email: mamurray@servidor.unam.mx*

(recibido: octubre de 1998; aceptado: diciembre de 1998)

Resumen

Se utiliza el tema llamado “acertijos matemáticos” para introducir conceptos importantes para la enseñanza de técnicas matemáticas útiles en la investigación de operaciones. Por medio de un acertijo se introducen diversos conceptos, métodos de solución y la manera de abordarlos por medio de una computadora digital.

Abstract

In this article, mathematical puzzles are used to introduce important concepts in the teaching of mathematical techniques useful in Operations Research Methods of solution. Algorithms and how to program them in a digital computer are also incorporated in the same manner.

Introducción

La mayor parte de los libros de investigación de operaciones igualan mucho a los textos de matemáticas avanzadas. Frecuentemente, sus temas ofrecen definiciones con poca motivación y luego proceden a demostrar lemas y teoremas haciendo pocos comentarios sobre la importancia y posibles aplicaciones prácticas de los mismos. No solamente se dejan de mencionar las aplicaciones a otras disciplinas, sino también, a las diferentes ramas de las matemáticas. El avance de la tecnología después de la Segunda Guerra Mundial ha sido vertiginoso; las aplicaciones de la energía atómica, la exploración del espacio exterior, el láser, la microelectrónica, la computación, las telecomunicaciones y sus aplicaciones a muchos campos de actividad, así como los avances en biología molecular y en bioingeniería, son unas cuantas de las novedades tecnológicas recientes. Las nuevas tecnologías requieren un conocimiento más profundo de las matemáticas. Sin embargo, por la manera tan árida en que se presenta dicha disciplina, son pocos los estudiantes que se entusiasman con ellas. En las escuelas, en todos los niveles, desde jardín de niños hasta posgrado, las materias asociadas con las matemáticas tienen un alto grado de reprobación; por ello, existe la necesidad de que el personal docente ponga todo su esfuerzo en hacerlas más interesantes y motivantes para los estudiantes. La introducción de la computadora ha

proporcionado esta motivación por su carácter interactivo y sus posibilidades de uso de imágenes, animación, color y sonido, que las hacen semejantes a la televisión y a los juegos electrónicos. Además de la computadora, otro enfoque que puede proporcionar motivación es la aplicación de las matemáticas recreativas en la enseñanza. Aquí se utiliza el tema llamado “acertijos matemáticos para introducir conceptos importantes para la enseñanza de técnicas matemáticas útiles en la investigación de operaciones”. Es decir, por medio de un acertijo, introducimos diversos conceptos, métodos de solución y la manera de abordarlos por medio de una computadora digital.

Un problema sobre cómo pintar eficientemente las líneas de una cancha de tenis

El gran acertijista británico H. Dudeney (1967) creó muchos acertijos muy ingeniosos de carácter matemático, algunos de los cuales pueden servir como introducción a problemas importantes de investigación de operaciones.

Pintando una cancha de tenis

Las líneas de una cancha de tenis están tenues y necesitan repintarse. Mi marcador de líneas es de tal naturaleza que, aunque puedo comenzar y terminar donde yo desee, no puede ser levantado de las líneas sin ensuciar parte de la

cancha. Por lo tanto, debo marcar dos veces algunas líneas. ¿Dónde debo comenzar?, ¿Qué ruta debo tomar sin levantar el marcador de líneas y así repintar la cancha completamente?, ¿Cómo pintar la mínima distancia dos veces? En la figura 1 se muestran las dimensiones de la cancha en pies. ¿Cuál es la mejor ruta?

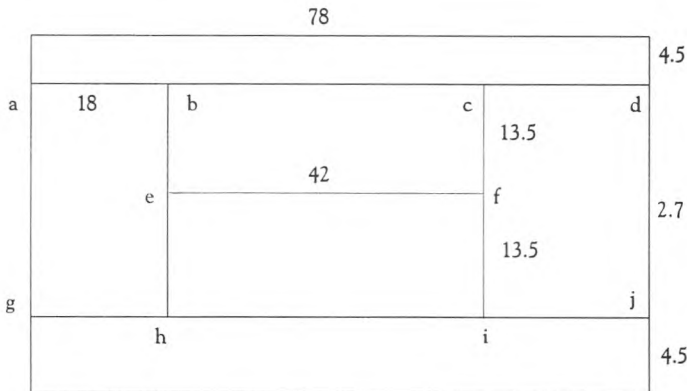


Figura 1

Oportunidades para que el maestro haga interesante la clase

Al plantear este problema, se le presenta al maestro la oportunidad de motivar a sus alumnos haciendo la clase más interesante proponiendo otros acertijos similares y al mismo tiempo explicando un poco de historia de las matemáticas. El problema sugerido se presta para hablar de los acertijos populares sobre cómo dibujar una figura de líneas sin levantar el lápiz. Por ejemplo, Dudeney (1967) reta a los lectores a trazar la figura 2 sin levantar el lápiz (y sin emplear trucos, como doblar el papel y trazar una o más líneas en el reverso del mismo; y, de esa manera, pasar de una parte de la figura a otra, sin levantar el lápiz, y sin que sea visible en el anverso del papel al desdoblarlo).

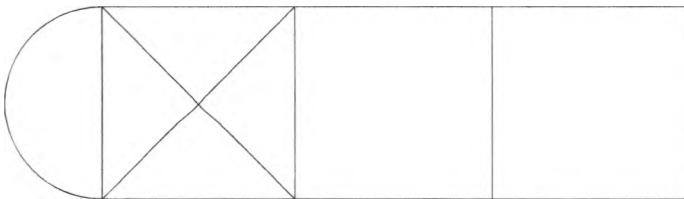


Figura 2

El problema de trazar la figura 2 sin levantar el lápiz es relativamente sencillo de resolver por ensayo y error. Cuando en una figura relativamente simple es posible dibujar sin

levantar el lápiz, generalmente no es muy difícil encontrar una solución. Lo que sí es complicado, es intentar buscarla cuando no existe, pues a veces, por una pequeña decisión equivocada, se queda muy cerca de lograrla, faltando solamente, por ejemplo, una línea. Este caso se aplicaría si a la figura 2 agregásemos, en el extremo derecho, un semicírculo similar al que aparece en el extremo izquierdo.

Es fácil demostrar que la figura 2 con el semicírculo agregado no puede ser dibujada sin levantar el lápiz, después de que se conocen las ideas principales del primer artículo sobre gráficas (el cual se puede considerar como fundador de la disciplina de teoría de gráficas), escrito por el gran matemático suizo Leonardo Euler —1707- 1782— (1736). Ese artículo fue escrito para resolver un problema que se habían planteado los habitantes de la población de Königsberg, lugar donde se encontraban dos islas en medio del río Pregel; contaba con siete puentes que conectaban las islas entre sí y las riveras del río, como se muestra en la figura 3.

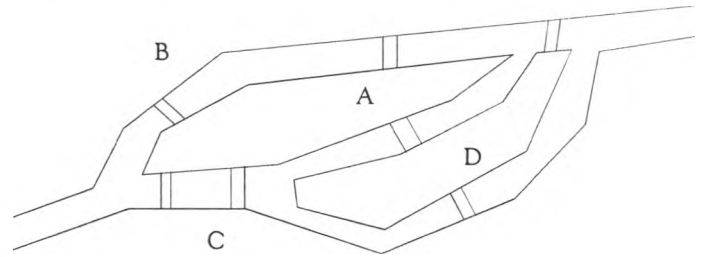


Figura 3

El problema consistía en encontrar una ruta que partiera de cualquier lugar y llegara a cualquier parte del mapa, pasando por cada uno de los puentes una sola vez. Por más que habían intentado muchos habitantes de Königsberg encontrar la ruta adecuada, no lo habían logrado; siempre quedaba un puente sin cruzar o había que cruzarlo más de una vez. Al plantearse al gran matemático Euler, lo convirtió en un problema que consistía en dibujar una gráfica sin levantar el lápiz. Redujo cada una de las regiones marcadas con letras mayúsculas en la figura 3 a un punto, y representó los puentes por medio de aristas, como se muestra en la gráfica de la figura 4.

Es claro que dibujar la figura 4 sin levantar el lápiz y cruzar todos los puentes de la figura 3 son problemas equivalentes. El razonamiento de Euler va más o menos así:

Si contamos el número de líneas incidentes en cada nodo de la figura 4 y llamamos grado del nodo a dichos números, vemos que los nodos B, C y D tienen grado 3 y el A, grado 5. Para dibujar la gráfica sin levantar el lápiz podemos salir de un nodo y dirigirnos a cualquiera de sus vecinos agotando una de las aristas; y si vamos borrando las aristas recorridas, nos queda el nodo inicial con grado

menor en uno del de la gráfica original. Si durante el trazo de la figura llegamos a cualquier nodo y volvemos a salir de él, al borrar las aristas correspondientes se reduce el grado del nodo en 2. Solamente reducimos a un número impar el grado inicial y el final (en caso de que sean distintos), al borrar las aristas recorridas en su grado (pues es posible que hayamos pasado por ellos en el recorrido y, en cada pasada, habremos reducido su grado en 2), y solamente para los nodos al principio y al final habrá una reducción de uno. Por otra parte, si los nodos inicial y final coinciden, entonces en todos los nodos se habrá reducido el grado en un número par. Concluimos entonces que para poder dibujar una gráfica sin levantar el lápiz, es necesario que dicha gráfica tenga todos sus nodos con grado par, excepto posiblemente los nodos inicial y final, que pueden tener grado impar. Si se quiere terminar en el mismo nodo en el que se empezó, entonces es necesario que la gráfica tenga todos sus nodos con grado par. En vista de que la gráfica de la figura 4, correspondiente al problema de los puentes de Königsberg, no cumple estas condiciones porque sus 4 nodos tienen grado impar, es imposible hacer el recorrido pasando por todos los puentes una sola vez, o equivalentemente, trazar la gráfica de la figura 4 sin levantar el lápiz y sin retrazar ninguna de las líneas.

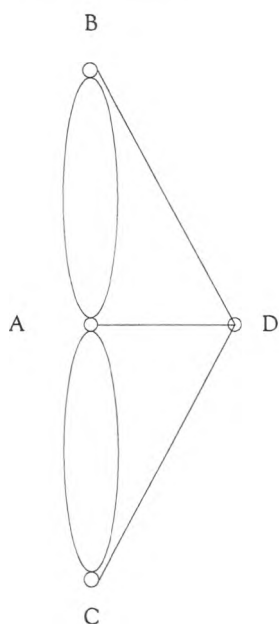


Figura 4

Para dibujar, sin levantar el lápiz, una gráfica conectada con dos nodos de grado impar y todos los demás, de grado par, se hace lo siguiente: se comienza el trazo en uno de los nodos de grado impar y se recorren arbitrariamente las aristas hasta llegar al otro nodo de grado impar. En ese momento, o se han recorrido todas las aristas y se ha resuelto el problema, o algunas aristas no han sido

recorridas porque todos sus nodos tienen grado par, por lo que se puede partir de un nodo y regresar a él sin levantar el lápiz, ya que todo nodo por el que se pasa (incluyendo aquel por el cual se sale), por tener grado par, tiene un par de aristas, lo que hace posible entrar por una y salir por otra. Cuando ya no es posible recorrer más aristas nuevas, se tiene que estar en el nodo inicial. Dado que la gráfica está conectada, este ciclo cerrado tiene que poseer algún nodo en común con el camino recorrido en la primera etapa, el cual se puede alargar comenzando en el nodo inicial y, cuando se llegue al primer nodo incidente con el ciclo de la segunda etapa, se recorre dicho ciclo y, al regresar al primer nodo del ciclo, se puede continuar recorriendo el camino de la primera etapa. Cuando este recorrido más largo es borrado, la gráfica restante (si no se agota la gráfica original) tendrá nuevamente todos sus nodos con grado par y por lo tanto, se podrá repetir el proceso de la segunda etapa. Y con el procedimiento anteriormente descrito, alargar el camino del nodo inicial al final. Repitiendo la rutina las veces que sea necesario, eventualmente se agotará la gráfica y se tendrá la solución del problema.

A la luz de lo anterior, vamos a analizar el problema de marcar la cancha de tenis. Si consideramos como nodos a los puntos donde se unen más de dos líneas (las esquinas donde la misma línea se quiebra en ángulo recto las podemos considerar parte de una arista y no un nodo), la gráfica tiene 10 nodos con grado 3, por lo que no es posible marcar la cancha de tenis sin tener que marcar algunas líneas más de una vez. Si a la gráfica le agregáramos algunas aristas paralelas de igual longitud que las existentes y solamente dejáramos dos nodos con grado par, la gráfica resultante se podría dibujar a un solo trazo, con la salvedad de que las aristas paralelas se dibujarían dos veces. De lo que se trata, entonces, es de buscar la manera de que la suma de las longitudes de las aristas paralelas sea lo más pequeña posible.

Al agregar una arista paralela entre dos nodos, se le aumenta uno a grados de los nodos extremos de la línea, por lo tanto, ambos grados son par. Tenemos entonces que buscar cuatro conexiones de ese tipo con la característica de que no se puede usar un nodo más de una vez, pues si lo hiciéramos, nuevamente el grado del nodo resultaría impar. Observando las longitudes de los lados y teniendo en cuenta la restricción recientemente mencionada, lo más conveniente es seleccionar dos de las líneas verticales que van del centro de la cancha a las líneas que marcan el límite de la misma para jugar “sencillos” —no podemos escoger las cuatro, pues violan la condición de usar cada nodo una sola vez—, y dos de las líneas horizontales que van del límite del área de “saque legal” al extremo de la cancha. Las líneas mencionadas aparecen más gruesas en la figura 5. Habiendo otras posibilidades simétricas obvias.

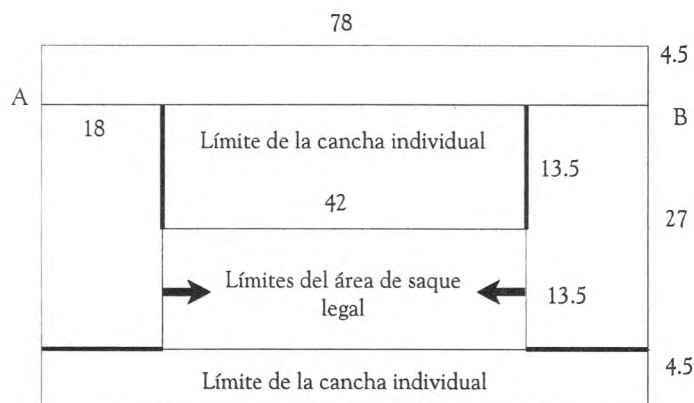


Figura 5

La deducción de cómo se hace el trazado puede hacerse de un golpe o aplicando el razonamiento de Euler. Se debe comenzar y terminar en nodos con grado impar (A y B); por ejemplo, si comenzamos en A y nos referimos a los puntos marcados con letras minúsculas en la figura 1, el trazo de un solo golpe queda descrito por la secuencia de nodos: a-b-e-b-c-f-c-d-a-g-h-g-j-i-h-e-f-i-j-d, y así terminamos en el punto B de la figura 5, donde se termina el trazo completo de la cancha habiendo trazado dos veces las cuatro líneas gruesas que aparecen en esta última figura.

Generalización del problema resuelto. El problema del cartero chino

El acertijo que se resolvió es un caso particular, muy cercano a un modelo de investigación de operaciones conocido con el nombre de *Problema del Cartero Chino*, mencionado en la literatura respectiva por Kwan (1962). El problema recibe el nombre del Cartero Chino—lo de chino es por su inventor—, ya que se le presenta a un cartero que tiene que realizar su repartición a pie cubriendo varias cuadras. Es necesario que pase por todas las calles de su zona—en muchos países los carteros no solamente entregan cartas sino también recogen las que se quieren enviar, por lo tanto, se ven forzados a recorrer toda su ruta, independientemente de que en todas las cuadras vayan a entregar cartas o no— recorriendo cada cuadra por lo menos una vez. Si para lograr esto tiene que pasar nuevamente por algunas cuadras, es conveniente que éstas sumen la menor distancia posible—para minimizar el cansancio—.

Para resolver el problema del cartero chino, Edmonds y Johnson (1965; 1973) han dado el siguiente algoritmo:

S0. Dada una gráfica conectada G , con pesos asociados con las aristas, se requiere una ruta cerrada óptima de cartero chino.

S1. Identifique todos los nodos de G con grado impar. Por un teorema de teoría de gráficas, el número de estos nodos debe ser par (Thulasiraman y Swamy, 1992). Llámoslos $v_1, v_2, \dots, v_{2k}, k \geq 1$. En el caso de que no haya nodos con grado impar en G , haga $G^* = G$ y siga en el punto S5.

S2. Calcule los caminos más cortos entre todos los pares de nodos con grado impar. Llame $d(v_i, v_j)$ a la distancia entre los nodos v_i y v_j .

S3. Construya una gráfica bipartita G' con la bipartición como sigue:

$$\begin{aligned} X &= \{x_1, x_2, \dots, x_{2k}\}, \\ Y &= \{y_1, y_2, \dots, y_{2k}\}, \\ w(x_i, y_i) &= M, \\ w(x_i, y_j) &= d(v_i, v_j) \text{ para } i \neq j, \end{aligned}$$

donde M es un número grande.

S4. Encuentre una asignación perfecta de peso mínimo de G' . Para cada (x_i, y_i) en esta asignación perfecta, agregue pseudoaristas paralelas a G en el camino más corto entre v_i y v_j . Llame G^* a la gráfica así obtenida.

S5. Construya un camino cerrado Euleriano en G^* . Este camino cerrado define un camino óptimo del cartero chino. DETÉNGASE.

El algoritmo de Edmonds y Johnson requiere que se resuelvan varios subproblemas, como la determinación de todas las rutas más cortas entre cierto conjunto de nodos en una red, la solución de un problema de asignación perfecta de mínimo peso en una gráfica bipartita con aristas y peso, así como la búsqueda de un camino cerrado Euleriano. Para el último subproblema ya hemos dado un algoritmo descubierto por el propio Euler; para la solución de los otros subproblemas existen algoritmos eficientes. Para el primero se menciona más adelante un algoritmo de Floyd (1962), que encuentra en una sola pasada las rutas más cortas entre cada par de nodos de una red. Para el problema de asignación, se explica el llamado Método Húngaro (Kuhn, 1955). A continuación se muestran detalles de algunos de estos algoritmos.

El algoritmo de Floyd para todas las distancias mínimas entre los nodos de una red

Se tiene una gráfica en la cual cada una de las aristas tiene asignado un número positivo que denota la distancia directa entre sus nodos extremos. Algunos nodos pueden no estar conectados directamente—si se trata de una

gráfica dirigida, algunos nodos pueden estar conectados en una dirección, pero no en la contraria, o estar conectados en las dos direcciones, pero tener distancias diferentes. De esta manera, se podría modelar un sistema de calles con un solo sentido, y la diferencia de distancias en direcciones distintas se podría deber a que en realidad los números, en vez de representar distancias, representen tiempos para recorrer las calles. Por la diferencia de volumen de tráfico, el tiempo puede ser diferente en un sentido de la calle que en otro—. Para el algoritmo que vamos a presentar, supondremos gráficas dirigidas—en el caso de gráficas no dirigidas se puede recurrir al subterfugio de sustituir por cada arista dos aristas dirigidas en paralelo con igual distancia. Para distancias mínimas, esto no representa ningún peligro pues la propia minimización eliminará la posibilidad de que una rama sea recorrida en ambas direcciones—. Si numeramos los nodos $1, 2, \dots, n$, las distancias directas pueden, entonces, representarse por medio de una matriz $n \times n$ W , cuyo componente w_{ij} representa la distancia directa del nodo i al nodo j . Dada la interpretación de los componentes de W , los componentes w_{ii} son todos cero, mientras que los que corresponden a pares de nodos que no están conectados directamente, son infinitas. También utilizaremos una matriz $n \times n$ Z que nos ayudará, al final, a determinar la secuencia de nodos que determinan las rutas de mínima distancia.

El algoritmo es el siguiente:

S1. Pónganse en las componentes de W las distancias directas entre los nodos como se indicó arriba. En la matriz Z , para $i, j = 1, 2, \dots, n$, hágase z_{ij} igual a cero si w_{ij} no es infinita o igual a j si w_{ij} es infinita.

S2. Hágase $k = 0$

S3. Hágase $k = k + 1$. Para toda $i \neq k$ tal que $w_{ik} \neq \infty$ y para toda $j \neq k$ tal que $w_{kj} \neq \infty$

Hágase lo siguiente:

1. $M = \min \{ w_{ij}, w_{ik} + w_{kj} \}$

2. Si $M < w_{ij}$, entonces ponga $z_{ij} = z_{ik}$, $w_{ij} = M$

S4. Si $k = n$, entonces la componente w_{ij} de la matriz W nos da la longitud de la ruta más corta que sale del nodo i y termina en el nodo j , y la componente z_{ij} nos da el primer nodo en dicha ruta. ALTO. Si $k < n$, entonces regrese al paso **S3**.

Sabiendo el primer nodo en la ruta más corta entre un nodo inicial y uno final, el siguiente se averigua viendo cuál es el primero en la ruta más corta, entre el nodo primero y el final. Aplicando reiteradamente este proceso, eventualmente se llega al nodo final y la secuencia de primeros

nodos, nos da la secuencia de la ruta más corta entre el nodo inicial y final originales.

Programa en LogoWriter para el algoritmo de Floyd

Mostramos procedimientos en LogoWriter y una sesión que los utiliza para encontrar las rutas más cortas entre todos los pares de nodos de una red, usando el algoritmo de Floyd. En la rutina **Floyd** que se muestra después de la sesión, los parámetros significan:

n - número de nodos de la red

w - matriz $n \times n$ en la que se almacenan las distancias directas entre pares de nodos

z - matriz $n \times n$ en la que se almacena la información para determinar la sucesión de nodos que define las rutas más cortas.

Utilizaremos la red que se muestra en la figura 6 para ilustrar el uso de los procedimientos.

Para operar los procedimientos se teclea `floyd 5 "w "z` y el programa muestra en el área de trabajo un cursor parpadeante que espera que se tecleen por filas, separando cada componente con un espacio vacío las matrices W y Z , una tras la otra. Se utilizará como número grande M la cantidad 9999. Mostramos la sesión; lo tecleado por el usuario está subrayado, lo no subrayado lo imprime el programa

```
floyd 5 "w "z
0 1 5 3 9999
1 0 1 4 9999
5 1 0 2 7
3 4 2 0 1
9999 9999 7 1 0
```

```
1 2 3 4 0
1 2 3 4 0
1 2 3 4 5
1 2 3 4 5
0 0 3 4 5
```

Con el procedimiento **escribemat** escribimos las matrices W y Z que nos dan las longitudes de las rutas más cortas y las secuencias de nodos que definen las rutas.

```
escribemat 5 "w
0 1 2 3 4
1 0 1 3 4
2 1 0 2 3
3 3 2 0 1
4 4 3 1 0
```


escribemat 5 "z

```
1 2 2 4 4
1 2 3 3 3
2 2 3 4 4
1 3 3 4 5
4 4 4 4 5
```

En las dos últimas matrices podemos consultar, por ejemplo, la ruta más corta entre los nodos 2 y 5. En la primera matriz W , consultamos el componente w_{25} en la fila 2 y columna 5, donde encontramos que la longitud es 4. Para determinar los nodos por los que pasa tras comenzar en el nodo 2, consultamos el componente z_{25} de la segunda matriz y encontramos 3; después, el componente z_{35} y encontramos 4; consultamos ahora el componente z_{45} y encontramos 5, con lo que llegamos al destino. Por lo tanto, la ruta más corta entre 2 y 5 es la definida por los nodos 2-3-4-5. Analizando la figura 6, verificamos que efectivamente ésa es la ruta más corta y que su longitud coincide con el componente w_{25} . De la misma manera, se pueden revisar todas las demás rutas.

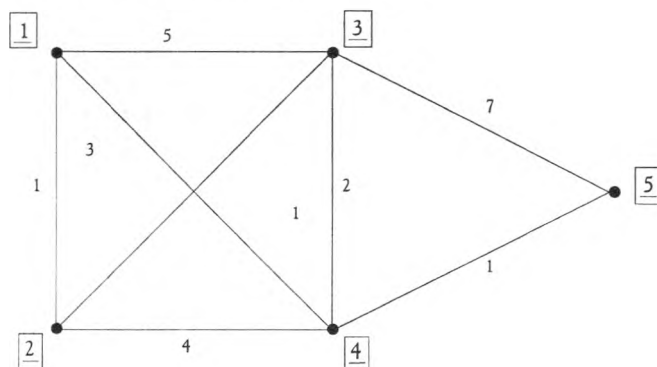


Figura 6

Para la cancha de tenis, el diálogo con la computadora es el que sigue:

floyd 10 "w" "z

```
0 18 9999 87 9999 9999 27 9999 9999 9999
18 0 42 9999 13.5 9999 9999 9999 9999 9999
9999 42 0 18 9999 13.5 9999 9999 9999 9999
87 9999 18 0 9999 9999 9999 9999 9999 27
9999 13.5 9999 9999 0 42 9999 13.5 9999 9999
9999 9999 13.5 9999 42 0 9999 9999 13.5 9999
27 9999 9999 9999 9999 9999 0 18 9999 87
9999 9999 9999 9999 13.5 9999 18 0 42 9999
9999 9999 9999 9999 9999 13.5 9999 42 0 18
```

```
9999 9999 9999 27 9999 9999 87 9999 18 0
1 2 0 4 0 0 7 0 0 0
1 2 3 0 5 0 0 0 0 0
```

```
0 2 3 4 0 6 0 0 0 0
1 0 3 4 0 0 0 0 0 10
0 2 0 0 5 6 0 8 0 0
0 0 3 0 5 6 0 0 9 0
1 0 0 0 0 0 7 8 0 10
0 0 0 0 5 0 7 8 9 0
0 0 0 0 0 6 0 8 9 10
0 0 0 4 0 0 7 0 9 10
```

escribemat 10 "w

```
0 18 60 78 31.5 73.5 27 45 87 105
18 0 42 60 13.5 55.5 45 27 69 87
60 42 0 18 55.5 13.5 87 69 27 45
78 60 18 0 73.5 31.5 105 87 45 27
31.5 13.5 55.5 73.5 0 42 31.5 13.5 55.5 73.5
73.5 55.5 13.5 31.5 42 0 73.5 55.5 13.5 31.5
27 45 87 105 31.5 73.5 0 18 60 78
45 27 69 87 13.5 55.5 18 0 42 60
87 69 27 45 55.5 13.5 60 42 0 18
105 87 45 27 73.5 31.5 78 60 18 0
```

escribemat 10 "z

```
1 2 2 2 2 2 7 2 2 2
1 2 3 3 5 3 1 5 3 3
2 2 3 4 2 6 2 2 6 4
3 3 3 4 3 3 3 3 3 10
2 2 2 2 5 6 8 8 6 6
3 3 3 3 5 6 5 5 9 9
1 1 1 1 8 8 7 8 8 8
5 5 5 5 5 5 7 8 9 9
6 6 6 6 6 6 8 8 9 10
4 4 4 4 9 9 9 9 9 10
```

El lector puede revisar los datos de las matrices de longitudes de rutas más cortas y de los primeros nodos; asimismo, confrontarlos con la figura 3 para convencerse de que el programa produce resultados correctos. Por tratarse de puras líneas en ángulos rectos (geometría Manhattan), es sumamente fácil hacer el trabajo manualmente y el uso de la computadora no es estrictamente necesario.

En el listado del procedimiento **Floyd** se invocan los procedimientos **leemat** y **min** para introducir las matrices W y Z , así como para encontrar el mínimo elemento de una lista. Los listados de dichos procedimientos aparecen después del de **Floyd**.

```
para floyd :n :w :z
leemat :n :w
leemat :n :z
da "k 0
repite :n
[da "k :k + 1 da "i 0
```

```

repite :n
[da "i :i + 1 da "j 0
repite :n
[da "j :j + 1
si (y (no :i = :k) ((cosa (palabra :w :i :k)) < 9999)
(no :j = :k) ((cosa (palabra :w :k :j)) < 9999))
[da "m min (lista (cosa (palabra :w :i :j)) (cosa
(palabra :w :i :k)) + (cosa (palabra :w :k :j)))
si :m < (cosa (palabra :w :i :j))
[da (palabra :z :i :j) (cosa (palabra :z :i :k))
da (palabra :w :i :j) :m
]
]
]
]
]
Fin

```

```

para leemat :n :w
da "h 1
repite :n
[da "fila leelista da "j 1
repite :n
[da (palabra :w :h :j) pr :fila da "j:j + 1 da "fila mpr
:fila
] da "h :h + 1
]
Fin

```

```

para min :lista
si vacia? :lista
[re []]
si (cuenta :lista) = 1
[re pr :lista]
si (cuenta :lista) = 2
[re minaux :lista]
re minaux (frase (pr :lista) (min mpr :lista))
Fin

```

```

para minaux :lista
sio (pr :lista) < (ul :lista)
[re pr :lista]
[re ul :lista]
Fin

```

```

para escribemat :n :mat
da "i 0 da "j 0
repite :n
[da "i :i + 1 da "j 0
repite :n
[da "j :j + 1 inserta cosa (palabra :mat :i :j)
inserta " | |

```

```

] es "
]
Fin

```

El procedimiento **min** invoca al procedimiento auxiliar **minaux**. Su listado aparece después del de **min**.

El problema de la asignación perfecta de peso mínimo

El algoritmo llamado "húngaro" para la solución de la asignación perfecta de peso mínimo (Kuhn, 1955; Munkres, 1957) está claramente descrito por el método de Kaufmann y Faure (1967), el cual se tomó para este artículo.

Se cuenta con una matriz cuadrada de pesos W , cuyos componentes w_{ij} representan los costos de asignar al trabajador i y al trabajo j . Queremos asignar a cada trabajador una tarea, de tal manera que la suma de los costos de asignación (la suma de las w_{ij} asignadas) sea mínima. —Nótese que en el algoritmo de Edmonds-Johnson se habla de pesos y ahora se habla de costos; eso se hace para que el problema se entienda intuitivamente—. El adjetivo "perfecto" se refiere al hecho de que esperamos que cada trabajador quede asignado y cada trabajo (cuya cantidad es igual a la cantidad de trabajadores) también quede asignado a un trabajador. En las asignaciones que no son perfectas, puede haber exceso de trabajadores o de tareas, y alguno de los dos conjuntos puede tener elementos que no quedaron asignados. Aunque el algoritmo puede describirse en forma abstracta con símbolos, es más entendible si, al mismo tiempo que se describe, se trabaja un ejemplo sencillo. Usaremos el ejemplo que utilizan Kaufmann y Faure (1967).

Para el ejemplo ilustrativo la matriz de costos (o de pesos) es la siguiente:

	1	2	3	4	5
1	17.5	15	9	5.9	12
2	16	16.5	10.5	5	10.5
3	12	15.5	14.5	11	5.5
4	4.5	8	14	17.5	13
5	13	9.5	8.5	12	17.5

La primera fase de generación de ceros del algoritmo húngaro consiste en restar a todos los elementos de cada columna el número más pequeño de ésta. Así, por ejemplo, para la W dada, a la primera columna se le resta 4.5; a la segunda, 8; a la tercera, 8.5; a la cuarta, 5; y a la quinta, 5.5. La nueva matriz es

		1	2	3	4	5
	1	13	7	0.5	0.5	6.5
	2	11.5	8.5	2	0	5
$W_1 =$	3	7.5	7.5	6	6	0
	4	0	0	5.5	12.5	7.5
	5	8.5	1.5	0	7	12

La misma operación que se hizo con las columnas se hace con las filas, es decir, se resta a cada fila el número más pequeño. Solamente la primera fila cambia, pues en las demás, el menor número es cero. La nueva matriz es

	1	2	3	4	5	
$W_2=$	1	12.5	6.5	0	0	6
	2	11.5	8.5	2	0	5
	3	7.5	7.5	6	6	0
	4	0	0	5.5	12.5	7.5
	5	8.5	1.5	0	7	12

Estamos listos para la segunda fase de asignación inicial. Busquemos ceros en la matriz con el objeto de asignar la mayor parte de ellos. Lo ideal sería encontrar un conjunto de ceros tal, que cada uno estuviera en filas y columnas diferentes —algo así como si fuera uno de los elementos de la expansión de un determinante—. Dado que hay distintas maneras de escoger los ceros, mostraremos un método que asegure que se asignó el máximo número de ellos. Comiencese asignando (por ejemplo, encerrando en un cuadro) los ceros de aquellas líneas que tienen menor número de ceros; éstos son los elementos 2,4; 3,5 y 5,3. Al encuadrar los ceros seleccionados se elimina tanto la fila como la columna correspondiente; al arreglo restante se le vuelve a aplicar el proceso, buscando en el arreglo reducido las líneas que tienen menos ceros. Cuando ya no haya ceros, se asigna el menor número que aparece en una fila. Este proceso nos da una asignación inicial. Si se tiene la suerte de poder asignar puros ceros, el proceso termina y sus posiciones nos indican una asignación mínima. En la siguiente tabla se han encuadrado los ceros asignados —posiciones (2,4); (3,5); (5,3); (4,1)— y para completar la asignación con un elemento en cada fila y columna se tendría que escoger la cantidad no nula 6.5 en la posición (1,2).

		1	2	3	4	5
$W_3 =$	1	12.5	6.5	0	0	6
	2	11.5	8.5	2	0	5
	3	7.5	7.5	6	6	0
	4	0	0	5.5	12.5	7.5
	5	8.5	1.5	0	7	12

Pasamos a la tercera fase de búsqueda del mínimo número de filas y columnas que contengan todos los ceros. El proceso de asignación inicial no necesariamente asigna el máximo número de ceros posible, por lo que se puede aplicar el siguiente subproceso para lograr dicho objetivo:

Una vez que se tiene la asignación inicial:

1) Se marcan con x las filas que no contengan ningún cero enmarcado —se marcaría con x la primera fila—.

2) Se marcan con x las columnas que tienen ceros no enmarcados en filas que están marcadas con x —se marcan con x las columnas 3 y 4—.

3) Se marcan con x las filas que tengan un cero enmarcado en una columna marcada —se marcan con x las filas 2 y 5—.

4) Se repiten los pasos 2 y 3 hasta que ya no existan más filas o columnas que marcar —en el ejemplo ya no aparecen nuevas marcas—.

La fase 4 es la terminación de la fase 3. Se dibuja una línea sobre toda fila no marcada y sobre toda columna marcada. Estas líneas nos dan aquéllas que en número mínimo contienen todos los ceros —esto se muestra para nuestro ejemplo en la siguiente matriz:

		1	2	3	4	5	
$W_4 =$	1	12.5	6.5	0	0	6	x
	2	11.5	8.5	2	0	5	x
	3	7.5	7.5	6	6	0	
	4	0	0	5.5	12.5	7.5	
	5	8.5	1.5	0	7	12	x
		x	x				

La fase 5 consiste en hacer desplazamientos de algunos ceros. Considerando la tabla restante e ignorando las líneas tachadas, se toma el menor número de esta tabla parcial, se le resta a los elementos de las columnas no atravesadas por una recta y se le suma a los elementos de los renglones atravesados por una recta —esto es equivalente a restar el mínimo de la tabla no tachada, a la parte de la tabla no rayada y sumarlo a los elementos que hayan sido rayados dos veces, en renglón y columna—. El resultado de esta operación aplicada a nuestro ejemplo es la tabla siguiente:

		1	2	3	4	5
W ₅ =	1	11	5	0	0	4.5
	2	10	7	2	0	3.5
	3	7.5	7.5	7.5	7.5	0
	4	0	0	7	14	7.5
	5	7	0	0	7	10.5

La fase 6 busca un nuevo óptimo o punto de partida para un nuevo ciclo regresando a la fase 2. En la nueva tabla se

hacen asignaciones de los ceros. Se comienza por asignar los ceros en las posiciones (2,4), (3,5); seguido de las posiciones (1,3), (5,2), finalmente la correspondiente a (4,1). Notamos que todas las asignaciones fueron ceros, quedando en diferentes filas y columnas, por lo que la asignación es óptima. En caso de que no se lograra el óptimo, se deben repetir las fases 2 a 6 hasta obtenerlo (Kuhn demostró que el proceso converge). Con las asignaciones óptimas indicadas y consultando las cantidades correspondientes de la matriz **W** original, se obtiene el valor mínimo de la asignación, la cual es, para este caso: $5 + 5.5 + 9 + 9.5 + 4.5 = 33.5$.

Algunas modificaciones para resolver el marcado de la cancha de tenis

El problema del cartero chino supone un camino cerrado que comienza y termina en el mismo punto. El problema del marcado de la cancha de tenis nos permite empezar y terminar en cualquier punto. Para convertir un problema en el otro, solamente hay que agregar un camino virtual de longitud cero entre el punto inicial y el final; de esa manera, los dos nodos con grado impar se convierten en nodos con grado par. Una asignación de mínimo costo con un camino de costo cero incluirá a dicho camino en la asignación; sin embargo, por tratarse de un camino virtual, no hay necesidad de recorrerlo en la solución final real del problema. Para no eliminar con el camino virtual alguno de los caminos más cortos (los cuales conviene escogerlos para retrazarlos), se debe escoger, como los dos nodos con grado impar, a otro par de nodos con distancia más corta entre ellos, de las más grandes, siendo la política más segura escoger los dos nodos de grado impar que estén a máxima distancia.

Aunque por falta de espacio ya no damos programas de computadora para resolver el problema de asignación, no es muy difícil automatizar el algoritmo descrito en un lenguaje que sea capaz de manejar arreglos bidimensionales. Dejamos como un ejercicio al lector el hacerlo y aplicarlo al acertijo. En problemas sencillos, como el de la cancha de tenis, el problema de asignación se puede resolver por inspección sin aplicar algoritmos formales.

También es posible conseguir programas de biblioteca para resolver problemas de investigación de operaciones. A continuación vamos a exhibir la solución del problema de asignación perfecta de peso mínimo correspondiente a la cancha de tenis. Utilizaremos el programa QSB+ una de cuyas opciones es el problema de asignación.

Un hombre que comienza en la población A tiene que inspeccionar todos los caminos que conectan a las poblaciones que se muestran en la figura 7. Las longitudes respectivas en millas se muestran junto a los caminos.

¿Cuál es la ruta más corta que puede adoptar terminando la jornada ? (Dudeney, 1967).

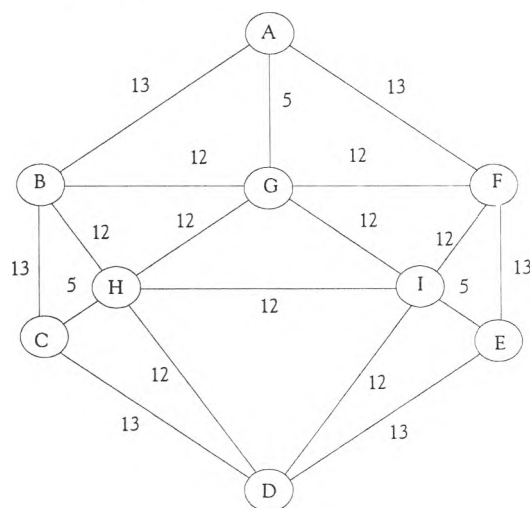


Figura 7

La opción de asignación de QSB+ le permite al usuario escoger si se trata de una maximización o de una minimización. En nuestro caso escogemos minimización. Luego hay que introducir la matriz de costos de asignación, que es la matriz final de distancias más cortas entre los puntos con grados impares de la gráfica de la cancha de tenis, con leves modificaciones: en los elementos de la diagonal principal, en lugar de ceros les damos un valor grande —en nuestro caso se les dio el valor 999999 que la máquina redondeó a $1E+06$ —, esto para que no queden asignados los puntos a sí mismos. De acuerdo con el comentario hecho anteriormente, a la distancia entre los puntos más lejanos se le da el valor 0. A continuación se muestran los datos introducidos al programa bajo el nombre “asigten” (todos los números han sido multiplicados por 10 para evitar fracciones decimales, la numeración de los nodos corresponde al orden alfabético de los puntos de la figura 1, es decir, a = 1, b = 2, c = 3, ..., i = 9, j = 10):

Cost/Profit/Distance Coefficients for asigten Page 1

Assignment Problem : Minimization

Objects Tasks

(Los resultados producidos por el programa son los mostrados en la página siguiente).

La interpretación de los resultados arrojados por el programa, es que las líneas de la figura 1 que hay que trazar dos veces son: *be* con longitud 13.5, *cd* con longitud 18, *fi* con longitud 13.5 y *gh* con longitud 18 —se debe recordar que se habían multiplicado los números por 10 para evitar fracciones, por lo que de la última tabla hay que dividir los números entre 10—. La solución obtenida por el programa

01	T1:	1E+06	T2:	+180.0	T3:	+600.0	T4:	+780.0	T5:	+315.0
	T6:	+735.0	T7:	+270.0	T8:	+450.0	T9:	+870.0	T10:	0
02	T1:	+180.0	T2:	1E+06	T3:	+420.0	T4:	+600.0	T5:	+135.0
	T6:	+555.0	T7:	+450.0	T8:	+270.0	T9:	+690.0	T10:	+870.0
03	T1:	+600.0	T2:	+420.0	T3:	1E+06	T4:	+180.0	T5:	+555.0
	T6:	+135.0	T7:	+870.0	T8:	+690.0	T9:	+270.0	T10:	+450.0
04	T1:	+780.0	T2:	+600.0	T3:	+180.0	T4:	1E+06	T5:	+735.0
	T6:	+315.0	T7:	+1050	T8:	+870.0	T9:	+450.0	T10:	+270.0
05	T1:	+315.0	T2:	+135.0	T3:	+555.0	T4:	+735.0	T5:	1E+06
	T6:	+420.0	T7:	+315.0	T8:	+135.0	T9:	+555.0	T10:	+735.0
06	T1:	+735.0	T2:	+555.0	T3:	+135.0	T4:	+315.0	T5:	+420.0
	T6:	1E+06	T7:	+735.0	T8:	+555.0	T9:	+135.0	T10:	+315.0
07	T1:	+270.0	T2:	+450.0	T3:	+870.0	T4:	+1050	T5:	+315.0
	T6:	+735.0	T7:	+1E+06	T8:	+180.0	T9:	+600	T10:	+780.0
08	T1:	+450.0	T2:	+270.0	T3:	+690.0	T4:	+870.0	T5:	+135.0
	T6:	+555.0	T7:	+180.0	T8:	1E+06	T9:	+420.0	T10:	+600.0
09	T1:	+870.0	T2:	+690.0	T3:	+270.0	T4:	+450.0	T5:	+555.0
	T6:	+135.0	T7:	+600.0	T8:	+420.0	T9:	1E+06	T10:	+180.0
010	T1:	0	T2:	+870.0	T3:	+450.0	T4:	+270.0	T5:	+735.0
	T6:	+315.0	T7:	+780.0	T8:	+600.0	T9:	+180.0	T10:	1E+06

Summary of Assignments for asigten Page:1					
Object	Task	Cost/Prof	Object	Task	Cost/Prof
01	T10	0	06	T9	+135.0
02	T5	+135.0	07	T8	+180.0
03	T4	+180.0	08	T7	+180.0
04	T3	+180.0	09	T6	+135.0
05	T2	+135.0	010	T1	0
Minimum OBJ = 1260 Iteration = 1 CPU second = .3789063					

es diferente de la obtenida anteriormente, la cual se muestra en la figura 5, pero es de la misma longitud total y se trata de una de las otras posibilidades simétricas obvias, mencionadas al presentar la figura 5. Una posible manera de trazar la cancha en forma continua (con los cuatro segmentos mencionados trazados dos veces) está dada por la secuencia aghgihebifgcdabcd.

Conclusiones

El problema de la figura 7 sería un problema laborioso si se intentara resolver manualmente por ensayo y error. Con las técnicas explicadas aquí se debe resolver manualmente en un par de minutos por inspección. Todo lo que hay que hacer es contar grados en los nodos y observar las longitudes de los caminos. La respuesta que da Dudeney es: "La

ruta más corta es como sigue: ABCHCDEIEFGBHDIH-GIFAG"; de esta manera, ha viajado 211 millas, y pasado por los dos caminos cortos CH y EI dos veces.

En muchas operaciones de inspección, vigilancia, pintado de carreteras, reparto de mercancías, y similares, se presenta el problema del cartero chino. Cuando las gráficas correspondientes son sencillas, la solución óptima es más fácil de obtener por inspección; cuando se trata de operaciones mayores con muchos tramos y geometría irregular, se requiere el uso de la computadora. En este artículo se planteó el problema por medio de acertijos matemáticos, posteriormente, se dieron algoritmos formales de solución, así como listados de programas de computadora que los automatizan o la manera de usar programas de biblioteca. La tesis principal del artículo es que se pueden enseñar las matemáticas aplicadas utilizando las matemáticas recreativas y que, en dicha enseñanza, es posible llegar a aplicaciones prácticas introduciendo la computadora digital. Estos dos ingredientes introducen factores importantes de motivación a los alumnos que actualmente brillan por su ausencia en la enseñanza de la investigación de operaciones, una disciplina cada vez más importante para la vida moderna.

Referencias

H.E. Dudeney (1967). *Puzzles & Curious Problems*. M. Gardner, Editor del Departamento de Juegos

- Matemáticos de Scientific American, Charles Scribner's Sons, New York.
- L. Euler (1736). *Solutio Problematis ad geometriam situs pertinentis*. *Academiae Petropolitanae*, Vol. 8, 128-140. Una traducción al inglés aparece en L. Euler (1953). The Königsberg Bridges. *Scientific American*, Vol. 189, No. 1, 66-70.
- Meo-Ko Kwan (1962). Graphic Programming Using Odd and Even Points. *Chinese Math*, Vol. 1, 273-277.
- J. Edmonds (1965). The Chinese Postman Problem. *Operations Research*, Vol. 13, Suppl. 1, 373.
- J. Edmonds y E.L. Johnson (1973). Matching Euler Tours and the Chinese Postman. *Math. Programming*, Vol. 5, 88-124.
- R.W. Floyd (1962). Algorithm 97: Shortest Path. *Communications of the ACM*, Vol. 5, 345.
- K. Thulasiraman y M.N.S. Swamy (1992). *Graphs: Theory and Algorithms*, John Wiley & Sons, Inc., New York.
- H.W. Kuhn (1955). The Hungarian Method for the Assignment Problem. *Naval Research Logistics Quarterly*, Vol. 2, 83-87.
- J. Munkres (1957). Algorithms for the Assignment and Transportation Problems. *Journal of SIAM*, Vol. 5, 32 - 38.
- A. Kaufmann y R. Faure (1967). *Invitación a la Investigación de Operaciones*. C.E.C.S.A., México.

Semblanza del autor

Marco Antonio Murray-Lasso. Realizó la licenciatura en ingeniería mecánica-eléctrica en la Facultad de Ingeniería de la UNAM. El Instituto de Tecnología de Massachussetts (MIT) le otorgó los grados de Maestro en Ciencias en Ingeniería Eléctrica y Doctor en Ciencias Cibernéticas. En México, ha laborado como investigador en el Instituto de Ingeniería y como profesor en la Facultad de Ingeniería, UNAM durante 36 años; en el extranjero, ha sido asesor de la NASA en diseño de circuitos por computadora para aplicaciones espaciales, investigador en los Laboratorios Bell, así como profesor de la Universidad Case Western Reserve y Newak College of Engineering, en los Estados Unidos. Además, ha ocupado diversos puestos en sociedades profesionales relativas a la ingeniería, tanto nacionales como internacionales. Es coautor de libros y artículos técnicos, memorias de conferencias y revistas de investigación y divulgación. Actualmente, es jefe de la Unidad de Enseñanza Auxiliada por Computadora de la División de Estudios de Posgrado de la Facultad de Ingeniería, UNAM, investigador nacional y consultor de la UNESCO.