

Graficación de Politopos en Computador

Jean Pierre Charalambos*, Manuel Guillermo Forero**

RESUMEN

En este artículo se presenta el diseño y análisis de un nuevo algoritmo para la graficación de politopos tridimensionales que representan el espacio de soluciones factibles de cualquier problema de programación lineal en tres variables, incluyendo vértices degenerados.

INTRODUCCIÓN

Existen dos factores importantes que permiten comprender los conceptos fundamentales en programación lineal:

- La visualización de la región de factibilidad de un problema dado en dos o tres dimensiones (véanse figuras 1 y 2).
- La visualización del movimiento, que se realiza por los vértices de la región de factibilidad, en busca de la solución óptima. Este movimiento se determina a partir de algoritmos de solución.

En cuanto a la visualización de la región de factibilidad, las soluciones encontradas en la literatura las ilustran sólo en el espacio bidimensional; además no son interactivas. Para la solución de un problema cualquiera se requiere una elaboración manual; este trabajo no sólo resulta dispendioso e ineficiente, sino que es impracticable en el caso de los problemas tridimensionales. Lo mismo sucede en cuanto a la visualización del movimiento por los vértices de la región de factibilidad de los algoritmos de solución, en los cuales el poder apreciar su desempeño en el espacio tridimensional puede resultar fundamental para validar su corrección.

Con la aparición del computador, la solución a este tipo de problemas puede ser facilitada, permitiendo representar la región de factibilidad en tres dimensiones, así como su rotación y translación en el espacio. Sin embargo, hasta ahora no se ha encontrado ningún trabajo concreto al respecto.

El propósito de este artículo es presentar un procedimiento eficiente para la visualización interactiva de la región de factibilidad de un problema de programación lineal en dos y tres dimensiones acotado en todas las direcciones, que sirva como modelo experimental para probar algoritmos de solución al problema de programación lineal.

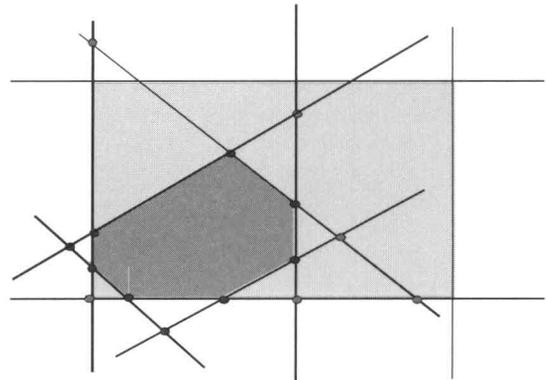


Figura 1. Región de factibilidad de la solución de un programa lineal en dos dimensiones.

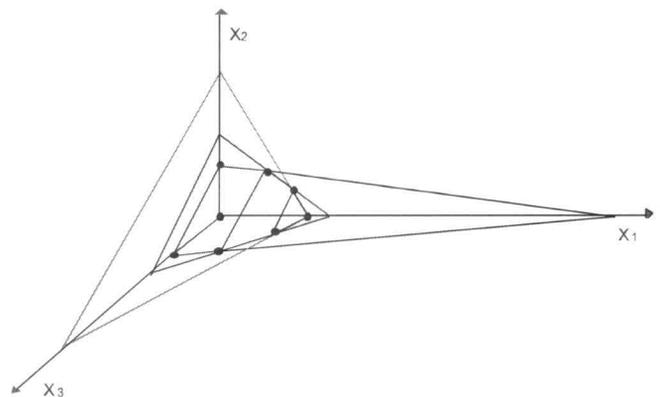


Figura 2. Región de factibilidad de la solución de un programa lineal en tres dimensiones.

I. DESARROLLO

Un politopo está acotado por cada uno de los semiespacios representados por cada una de las restricciones de un problema de programación lineal [1]. En el caso bidimensional, estos semiespacios son rectas (figura 1). Como en el caso tridimensional son planos (figura 2), el politopo tridimensional corresponde al concepto en computación gráfica de malla

* Profesor asistente, M.sc. Ingeniería de Sistemas.

** Profesor asistente, Universidad Nacional de Colombia, Ing. Electrónico, magíster en Ingeniería Eléctrica, máster en Imágenes Médicas, Dr. en Ingeniería Biomédica, director Grupo de Investigación OHWAHA.

poligonal, concretamente a una malla poligonal convexa, valiéndose la extensión de este último término. Así, el propósito inicial de visualización se puede enunciar como el problema de generar una malla poligonal convexa correspondiente a la región de factibilidad de un problema de programación lineal cualquiera en tres variables, que sirva de modelo experimental para la prueba de algoritmos en la solución del problema de programación lineal.

Para representar una superficie en tercera dimensión en el computador, se emplean librerías gráficas. Estas librerías permiten representar principalmente las mallas poligonales en dos formas: en *forma explícita* y con *apuntadores a una lista de vértices* [3]. A pesar de que la primera resulta ineficiente en cuanto a uso de memoria, ya que cada vértice está presente en al menos tres caras del politopo, es la más apropiada para la adición de nuevos vértices y caras a la malla, como se sugerirá más adelante.

II. ALGORITMO PARA LA GENERACIÓN DE LA MALLA

El primer paso consiste en definir una malla poligonal totalmente acotada, correspondiente a las restricciones de no negatividad $X_1 \geq 0$, $X_2 \geq 0$ y $X_3 \geq 0$, y las tres constantes siguientes: $CotaX_1$, $CotaX_2$ y $CotaX_3$, definidas por el usuario donde $X_1 \leq CotaX_1$, $X_2 \leq CotaX_2$ y $X_3 \leq CotaX_3$. Esta malla inicial corresponde, en el sistema de coordenadas cartesianas, al primer octante acotado por los planos $X_1 = Cota X_1$, $X_2 = Cota X_2$ y $X_3 = Cota X_3$.

En el segundo paso, se hace la búsqueda de la región de factibilidad, definiendo las restricciones del problema y generando iterativamente la malla que resulta de la intersección de la restricción que se está evaluando con respecto a la malla anterior. De este modo, cuando el problema es factible, se genera un conjunto poliédrico acotado (politopo) correspondiente a la región de factibilidad.

Estos pasos pueden ser formalizados en el siguiente algoritmo iterativo presentado en pseudocódigo:

```

function politopo
Leer_restricciones
for i←1 to (m) do % Se tienen m restricciones
  for jj←1 to (i-1) do % Se van a recorrer las caras.
    for k←1 to n do % n es el número de vértices de la cara
      Hallar_cortes % puntos nuevos
      Evaluar_cortes
      Actualizar_cara
    end
  end
end
  Crear_nueva_cara.
end
end
    
```

A medida que se introducen las restricciones en el paso 2, es necesario encontrar los nuevos vértices que definen el politopo solución del problema. Para ello deben tomarse en cuenta las formas en que el plano por evaluar, definido por la restricción, puede cortar los planos del politopo generado por las restricciones anteriores.

III. ANÁLISIS DE LA FORMA EN QUE UN PLANO CORTA LAS CARAS DE UN POLITOPO

En el cuadro 1 se enumeran las maneras en que un plano, determinado por una restricción, puede cortar la cara de un politopo.

Cuadro 1. Relación entre un plano y las caras de un politopo.

CASOS	RELACION
1	La restricción no toca la cara
2	La restricción toca la cara en un punto (necesariamente 1 vértice)
3	La restricción corta la cara en 2 vértices adyacentes
4	La restricción corta la cara en 2 vértices no adyacentes
5	La restricción corta la cara en 2 puntos: uno es un vértice, el otro es un punto nuevo
6	La restricción corta la cara en 2 puntos: los dos son puntos nuevos

Las figuras 3 a 8 presentan un ejemplo de cada uno de los casos mostrados en el cuadro 1. Analizando estos casos se puede determinar la ecuación de los planos que pasan por los vértices definidos por la intersección entre el plano y una cara del politopo. En cada una de las figuras, estas ecuaciones y sus respectivas restricciones, se representan por los números encerrados en círculos. La restricción por calcular está representada en cada figura por la restricción (6). A pesar de que un punto en el espacio tridimensional se define por la intersección de tres planos, se puede notar que sólo se definen explícitamente dos ecuaciones por cada vértice, y la tercera ecuación se define implícitamente, pues corresponde al plano sobre el cual reposa la cara del politopo.

En la figura 3 se ilustra el caso 1. En la parte **a** se aprecia el caso en el cual la cara del politopo no se encuentra dentro de la zona de factibilidad determinada por la restricción (6), y por tanto, la cara es descartada. En la parte **b**, por el contrario, la cara está incluida dentro de la zona de factibilidad, y en consecuencia, la cara es conservada.

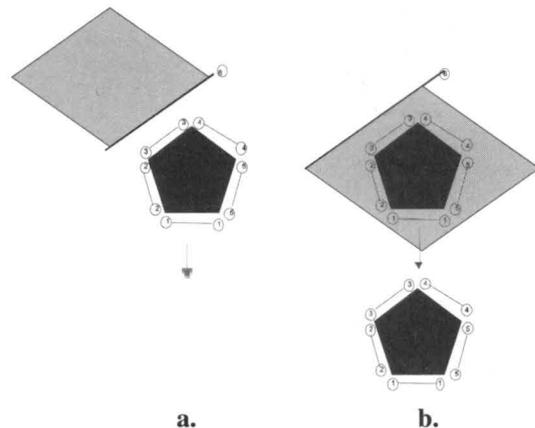


Figura 3. Caso 1: El plano definido por la ecuación, representada por la recta con el número 6 encerrado en un círculo, no interseca la cara.

En la figura 4 se ilustra el segundo caso. El plano definido por la restricción corta la cara del politopo en un vértice. En la parte **a**, la cara del politopo no se encuentra dentro de la zona de factibilidad, determinada por la restricción (6), y por tanto, la cara es descartada. En la parte **b**, ésta está incluida dentro de la zona de factibilidad, y la cara es conservada.

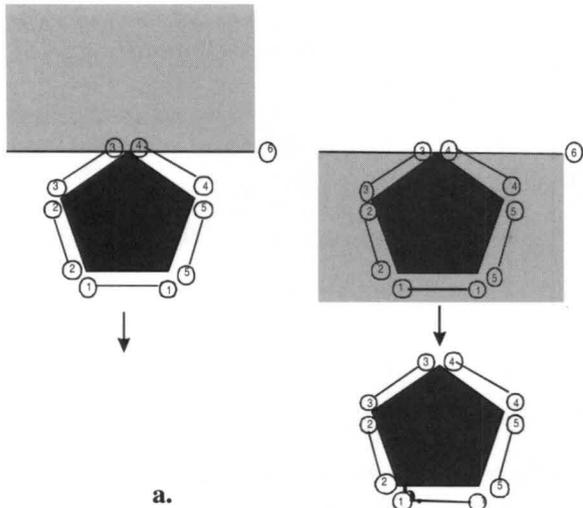


Figura 4. Caso 2: El plano definido por la ecuación, representada por la recta con el número 6 interseca la cara en un vértice.

En la figura 5 se muestra el tercer caso. Aquí, el plano definido por la restricción corta la cara del politopo sobre dos vértices adyacentes. En la parte **a**, la cara del politopo está fuera de la zona de factibilidad, dada por la restricción (6), y la cara es descartada. En la parte **b**, la cara se halla en la zona de factibilidad y es conservada, en este caso. Si el nuevo plano (6) no corta el interior del politopo, entonces tanto la restricción (3) como la (6) pueden representar los extremos de la arista de la cara, pero si el nuevo plano corta el interior del politopo, sólo la restricción (6) podrá representar los extremos de la arista de la cara.

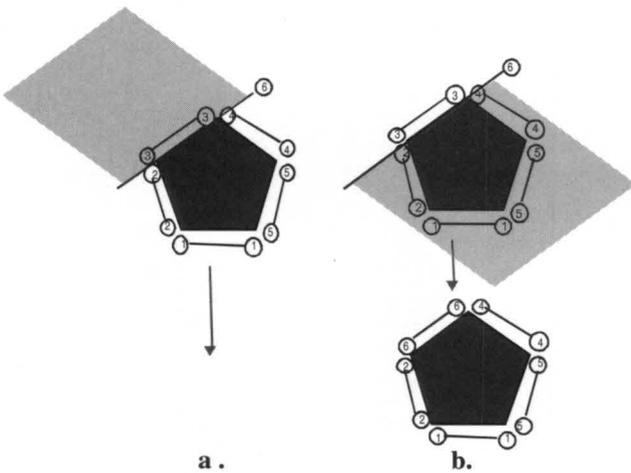


Figura 5. Caso 3: El plano definido por la ecuación, representada por la recta número 6, interseca la cara del politopo en dos vértices adyacentes.

En la figura 6 se muestra el cuarto caso. Aquí, el plano definido por la restricción corta la cara del politopo sobre dos de sus vértices no adyacentes. Tanto en la parte **a** como en la **b**, la cara en consideración se modifica, siendo ahora limitada por la recta que resulta de la intersección entre la cara y el plano de la restricción. De esta manera, la cara resultante se mantiene

dentro de la nueva región de factibilidad. Esta modificación se ve reflejada en el cambio de representación de la cara resultante.

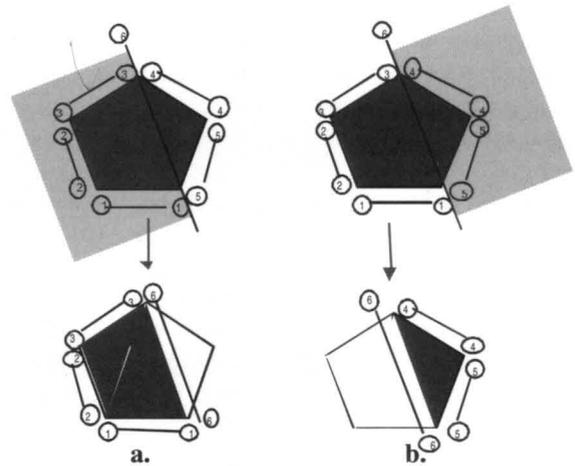


Figura 6. Caso 4: El plano definido por la ecuación, representada por la recta número 6 corta la cara del politopo en dos vértices no adyacentes
a. La nueva cara es limitada ahora por la recta (6) que reemplaza a las rectas (4) y (5).
b. La nueva cara es limitada por la recta (6) y reemplaza a las rectas (1), (2) y (3).

La figura 7 ilustra el quinto caso. En éste el plano definido por la restricción corta la cara del politopo en dos puntos: el primero corresponde a uno de sus vértices, y el otro a un nuevo punto de intersección, resaltado en la figura. Tanto en la parte **a** como en la **b**, la cara del politopo varía, y al igual que en el caso 4, la recta definida por la intersección de la cara del politopo con el plano de la restricción limita ahora la cara resultante, que se conserva dentro de la nueva región de factibilidad.

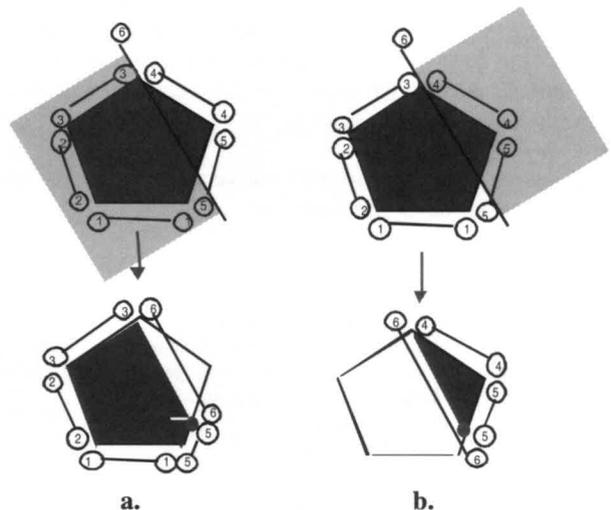


Figura 7. Caso 5: El plano definido por la ecuación, representada por la recta número 6, corta la cara del politopo en dos puntos: un vértice y un punto entre dos vértices adyacentes del cualquier otro lado.
a. La nueva cara se ve limitada por la recta (6) y reemplaza a la recta (4). **b.** La nueva cara es limitada por la recta (6) y reemplaza a las rectas (1), (2) y (3).

En la figura 8 se ilustra el sexto y último caso. Aquí, el plano definido por la restricción corta la cara del politopo en dos puntos sin que ninguno de ellos sea un vértice. Tanto en la

parte **a** como en la **b**, la cara del polítopo cambia, y la cara resultante es aquella que se encuentra dentro de la nueva región de factibilidad.

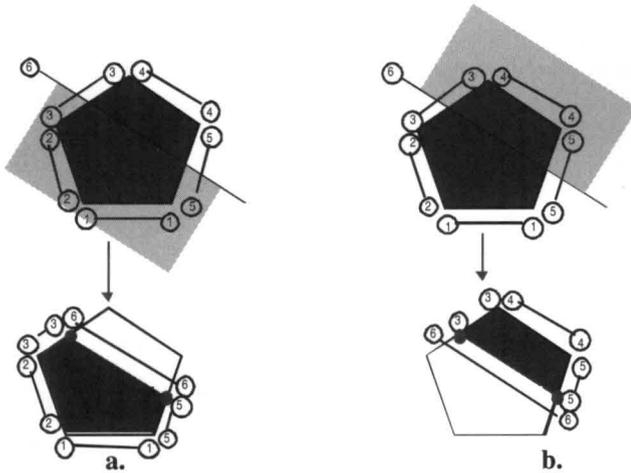


Figura 8. Caso 6: El plano definido por la ecuación, representada por la recta número 6, corta la cara del polítopo en dos puntos, ninguno de ellos vértice.

- a. La nueva cara se encuentra ahora limitada por la recta (6) que reemplaza a la recta (4)
- b. La nueva cara es limitada por la recta (6) y reemplaza a las rectas (1) y (2).

Es de anotar que en caso de que se presente degeneramiento en algún vértice, es decir, cuando por un vértice se intersequen más de tres planos, como ocurre en los casos 2, 3, 4 y 5; el vértice puede ser definido únicamente por tres de estas ecuaciones. Esto nos permite enunciar el siguiente principio, cuya veracidad fue validada experimentalmente.

IV. PRINCIPIO DE CONTINUIDAD

Cuando por un vértice pasen al menos tres planos, para preservar la coherencia de la representaciones suficiente, mantener sólo tres de ellos, siempre y cuando en cada cara del polítopo, entre un vértice y cualquiera adyacente, exista una y sólo una ecuación en común.

Este principio se logra sin ambigüedad, obedeciendo al conjunto de reglas definidas en la sección anterior.

V. ANÁLISIS COMPUTACIONAL

La eficiencia del algoritmo está fundamentada en el crecimiento lineal que existe entre el número de vértices del polítopo y el número de caras, comportamiento que sin duda está relacionado con la fórmula de Euler [4].

Tomando como barómetro [2] la función Hallar_cortes, puesto que es la función que más veces se ejecuta dentro del algoritmo, se tiene que:

$$m \ i \ n$$

$$\sum_{i=1} \sum_{j=1} \sum_{k=1} (Hallar_cortes) \quad (1)$$

define el número de veces que se ejecuta el barómetro.

La solución de la expresión 1 permite determinar que el orden del algoritmo pertenece a $\Theta(m^2*n)$, siendo m el número de restricciones y n el número de vértices promedio por cara.

El número máximo de vértices promedio por cara en el caso estudiado (polítopo tridimensional) se debe determinar a partir del número máximo de soluciones básicas factibles o vértices en un problema de m restricciones y 3 variables. Al dibujar un polítopo tridimensional cualquiera, como el mostrado en la figura 9, se puede observar que el número máximo de puntos críticos que surgen al calcular la primera restricción es de 4 y el número máximo de puntos críticos que aparecen por cada nueva restricción es de 2. Por tanto, el número máximo de soluciones básicas factibles de un problema en tres dimensiones está dado por:

$$2*(m-1) + 4 = 2*m + 2 \quad (2)$$

Así pues, en el caso en que la primera restricción corta los tres ejes, el número máximo de puntos críticos está dado por los tres puntos de corte y el vértice en el origen. Ahora, el número máximo de puntos que surge al calcular el resto de restricciones es de dos, dado que en el peor caso (mayor número posible de soluciones factibles) debe necesariamente suceder que todo vértice sea producido por el mínimo número posible de restricciones, que es de tres en el caso tridimensional. El número máximo de vértices adicionales que aparecen al ingresar una nueva restricción es de dos. Esto se debe a que al definir una nueva restricción, el número máximo de nuevos vértices es de tres, cuando la nueva restricción corta los tres planos que como mínimo forman un vértice, y cuando menos sustituye a un antiguo vértice.

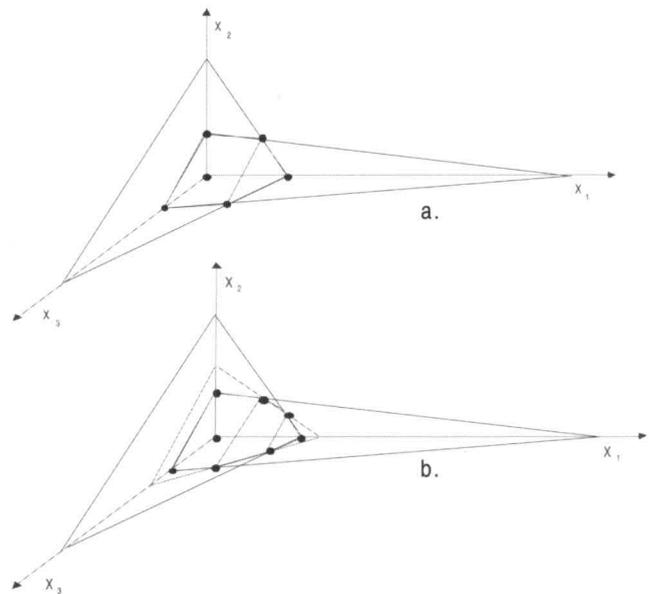


Figura 9. Corte de un polítopo tridimensional. En la parte a se muestra el polítopo definido por cinco planos (los planos $x_1=0$, $x_2=0$, $x_3=0$ y los dos planos visualizados como triángulos). En la parte b se visualiza el polítopo resultante al aparecer una nueva restricción.

Por tanto, se puede afirmar, sabiendo que cada vértice está definido por el cruce de tres planos, que el máximo número promedio de vértices por cada cara está dado por el valor constante:

$$n = [(2 * m + 2) * 3] / m \quad (3)$$

Lo anterior permite concluir que:

$$T(\text{politopo}) \in \Theta(m^2) \quad (4)$$

Es decir, el tiempo de ejecución del algoritmo crece proporcionalmente con el cuadrado del número de restricciones.

VI. MEJORAMIENTO ADICIONAL DEL DESEMPEÑO

El enfoque más simple, que puede ser empleado para resolver el problema de hallar las coordenadas de los puntos de corte entre un plano definido por una restricción y una cara, consta de una rutina que recorre cada cara del politopo por aristas adyacentes, y resuelve para cada arista el sistema de ecuaciones que surge entre la ecuación del plano de la cara, la de la arista y la ecuación de la restricción que se está calculando; y luego verifica si el punto solución corta la arista. Una solución más eficiente se consigue al resolver el sistema de ecuaciones, sólo cuando se sabe que la restricción corta la arista. Esto se consigue determinando para el primer vértice una satisfactibilidad inicial—es decir, si éste satisface o no la restricción que se está analizando— y luego definiendo del mismo modo, en orden adyacente, la satisfactibilidad de los demás vértices. Cuando ésta cambia, se resuelve el sistema de ecuaciones.

VII. APLICACIÓN

Con base en el estudio anterior, se desarrolló una aplicación en Matlab con el fin de verificar los razonamientos alcanzados. La figura 10 muestra la aplicación. Con la aplicación se generaron 20 ejemplos, en los cuales se presentaban cada uno de los casos analizados anteriormente. Los resultados obtenidos coincidieron con la respuesta esperada. Actualmente esta aplicación está en uso en el curso de Programación Lineal y Grafos de las carreras de Ingeniería en la Universidad Nacional de Colombia donde el programa ha sido igualmente probado.

La experiencia que se ha tenido con la aplicación es que facilita al estudiante la comprensión de los conceptos de programación lineal.

CONCLUSIONES

Se ha presentado un nuevo algoritmo para la visualización interactiva de la región de factibilidad de un problema de programación lineal en dos y tres dimensiones acotado en todas las direcciones. Este algoritmo presenta características de eficiencia comparado con el enfoque más simple que puede pensarse en la solución de este problema. Esta solución puede

ser empleada también en el modelamiento de algunos sistemas dinámicos, entre otras posibles aplicaciones.

Igualmente ha servido como base para la implementación de una aplicación, la cual ha sido probada, y permite la visualización de la solución de problemas de programación lineal; además, facilita la comprensión de este tipo de problemas por parte del estudiante.

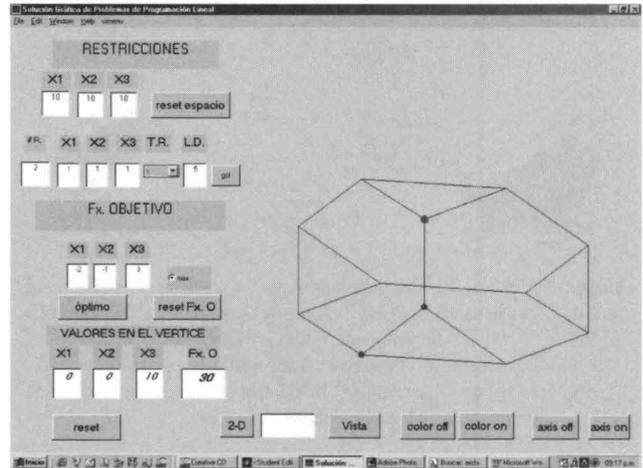


Figura 10. Se presenta una instantánea de la aplicación durante ejecución para la resolución del siguiente programa lineal:

$$\begin{aligned} \text{Max } Z &= -2X_1 - X_2 + 3X_3 \\ \text{Sujeta a:} \\ X_1 + X_2 + X_3 &\leq 15 \\ X_1 + X_2 + X_3 &\geq 5 \\ X_1 &\leq 10 \\ X_2 &\leq 10 \\ X_3 &\leq 10 \\ X_i &\geq 0 \end{aligned}$$

BIBLIOGRAFÍA

1. BAZAARA, M. S., *et al.*, *Programación Lineal y Flujo en Redes*. 2a. ed., Limusa, 1998.
2. BRASSARD, G., BRATLEY, P., *Fundamentals of Algorithmics*. Prentice Hall, 1996.
3. FOLEY, *et al.*, *Introducción a la graficación por computador*. 2a. ed., Addison-Wesley, 1994.
4. WILSON, R. J., WATKINS, J. J. *Graphs An Introductory Approach*. Wiley, 1990.