CAPÍTULO 3

Introducción a la programación lineal

El desarrollo de la programación lineal ha sido clasificado como uno de los avances científicos más importantes de mediados del siglo XX, y estamos de acuerdo con esta aseveración. Su efecto desde 1950 ha sido extraordinario. En la actualidad es una herramienta de uso normal que ha ahorrado miles o millones de dólares a muchas compañías o negocios, incluso empresas medianas, en los distintos países industrializados del mundo; su aplicación a otros sectores de la sociedad se ha ampliado con rapidez. Una proporción muy grande de los programas científicos en computadoras está dedicada al uso de la programación lineal. Se han escrito docenas de libros de texto sobre esta materia y se cuentan por cientos los artículos *publicados* que describen aplicaciones importantes.

¿Cuál es la naturaleza de esta notable herramienta y qué tipos de problemas puede manejar? El lector adquirirá una noción de este tema a medida que trabaje en los ejemplos que se presentarán más adelante. Sin embargo, un resumen verbal puede permitirle elaborar una idea. Expresado en forma breve, el tipo más común de aplicación abarca el problema general de asignar de la mejor manera posible —es decir, de *forma óptima*— *recursos limitados* a *actividades que compiten* entre sí por ellos. Con más precisión, este problema consiste en elegir el nivel de ciertas actividades que compiten por recursos escasos necesarios para realizarlas. Después, los niveles de actividad que se eligen dictan la cantidad de recursos que consumirá cada una de ellas. La variedad de situaciones a las que se puede aplicar esta descripción es sin duda muy grande, ya que abarca desde la asignación de instalaciones de producción a los productos hasta la asignación de los recursos nacionales a las necesidades de un país; desde la selección de una cartera de inversiones hasta la selección de los patrones de envío; desde la planeación agrícola hasta el diseño de una terapia de radiación, etc. No obstante, el ingrediente común de todas estas situaciones es la necesidad de asignar recursos a las actividades mediante la elección de los niveles de éstas.

La programación lineal utiliza un modelo matemático para describir el problema. El adjetivo *lineal* significa que todas las funciones matemáticas del modelo deben ser *funciones lineales*. En este caso, la palabra *programación* no se refiere aquí a términos computacionales; en esencia es sinónimo de *planeación*. Por lo tanto, la programación lineal involucra la *planeación de actividades* para obtener un resultado óptimo; esto es, el resultado que mejor alcance la meta especificada —de acuerdo con el modelo matemático— entre todas las alternativas factibles.

Aunque la asignación de recursos a las actividades es la aplicación más frecuente, la programación lineal tiene muchas otras posibilidades. En realidad, *cualquier* problema cuyo modelo matemático se ajuste al formato general del modelo de programación lineal, es un problema de programación lineal. (Por esta razón, un problema de programación lineal y su modelo se denominan con frecuencia *programa lineal*, o incluso sólo *PL*.) Aún más, se dispone de un procedimiento de solución muy eficiente llamado **método símplex** para resolver estos problemas lineales, incluso los de gran tamaño. Éstas son algunas razones del tremendo efecto de la programación lineal en las décadas recientes.

Debido a su gran importancia hemos dedicado a la programación lineal éste y los siguientes seis capítulos. Después de presentar aquí las características generales de programación lineal, los capítulos 4 y 5 se dedican al método símplex. El capítulo 6 analiza los problemas de programación lineal *después* de la aplicación inicial del método símplex. El capítulo 7 examina varias extensiones muy empleadas de este método e introduce el *algoritmo de punto interior* que en ocasiones se usa para resolver problemas de programación lineal aún más grandes que los que maneja el método símplex. Los capítulos 8 y 9 consideran algunos problemas especiales de programación lineal cuya trascendencia justifica su estudio individual.

Además, en varios de los capítulos posteriores se verán aplicaciones de programación lineal a otras áreas de la investigación de operaciones.

Este capítulo comienza con el desarrollo de un ejemplo prototípico simplificado de un problema de programación lineal. Este ejemplo es tan pequeño que puede resolverse de manera directa en una gráfica. En las secciones 3.2 y 3.3 se presentan el *modelo general de programación lineal* y sus supuestos básicos. La sección 3.4 proporciona algunos ejemplos adicionales de programación lineal. La sección 3.5 describe cómo pueden establecerse y resolverse problemas de programación lineal de tamaño mediano en una hoja de cálculo. Sin embargo, algunos problemas reales requieren modelos en verdad *masivos*. La sección 3.6 ilustra cómo suelen surgir estos modelos de gran tamaño y cómo se pueden formular de manera correcta con la ayuda de lenguajes especiales de modelado como MPL —su formulación se describe en esta sección— o LINGO (la formulación de este modelo se presenta en el suplemento 2 de este capítulo en el sitio web del libro).

3.1 EJEMPLO PROTOTÍPICO

La WYNDOR GLASS CO. produce artículos de vidrio de alta calidad, entre ellos ventanas y puertas de vidrio. Tiene tres plantas. Los marcos y molduras de aluminio se hacen en la planta 1, los de madera en la planta 2; la 3 produce el vidrio y ensambla los productos.

Debido a una reducción de las ganancias, la alta administración ha decidido reorganizar la línea de producción de la compañía. Se discontinuarán varios productos no rentables y se dejará libre una parte de la capacidad de producción para emprender la fabricación de dos productos nuevos cuyas ventas potenciales son muy prometedoras:

Producto 1: una puerta de vidrio de 8 pies con marco de aluminio
Producto 2: una ventana corrediza con marco de madera de 4 por 6 pies

El producto 1 requiere parte de la capacidad de producción en las plantas 1 y 3 y nada en la planta 2. El producto 2 sólo necesita trabajo en las plantas 2 y 3. La división de comercialización ha concluido que la compañía puede vender todos los productos que se puedan fabricar en las plantas. Sin embargo, como ambos productos competirían por la misma capacidad de producción en la planta 3, no está claro cuál *mezcla* de productos sería la *más rentable*. Por lo tanto, se ha formado un equipo de IO para estudiar este problema.

El grupo comenzó por realizar juntas con la alta administración para identificar los objetivos del estudio. Como consecuencia de ellas se desarrolló la siguiente definición del problema:

> Determinar cuáles *tasas de producción* deben tener los dos productos con el fin de *maximizar las utilidades totales*, sujetas a las restricciones impuestas por las capacidades de producción limitadas disponibles en las tres plantas. (Cada producto se fabricará en lotes de 20 unidades, de manera que la *tasa de producción* está definida como el número de lotes que se producen a la semana.) Se permite *cualquier* combinación de tasas de producción que satisfaga estas restricciones, incluso no fabricar uno de los productos y elaborar todo lo que sea posible del otro.

El equipo de IO también identificó los datos que necesitaba reunir:

1. Número de horas de producción disponibles por semana en cada planta para fabricar estos nuevos productos. (Casi todo el tiempo de estas plantas está comprometido con los productos actuales, lo que limita la capacidad para manufacturar nuevos productos.)
2. Número de horas de fabricación que se emplea para producir cada lote de cada artículo nuevo en cada una de las plantas.
3. La ganancia por lote de cada producto nuevo. (Se escogió la *ganancia por lote producido* como una medida adecuada una vez que el equipo llegó a la conclusión de que la ganancia incremental de cada lote adicional producido sería, en esencia, *constante*, sin que importase el número total de lotes producidos. Debido a que no se incurre en costos sustanciales para iniciar

Recuadro de aplicación

Swift & Company es una empresa diversificada productora de proteína con base en Greeley, Colorado. Con ventas anuales de más de 8 000 millones de dólares, la carne de res y sus productos derivados son, por mucho, la parte más grande del negocio de la compañía.

A fin de mejorar las ventas de la empresa y su desempeño en la manufactura, la alta administración concluyó que necesitaba alcanzar tres objetivos importantes. Uno fue permitir a los representantes de servicio al cliente hablar a sus más de 8 000 clientes para transmitirles información precisa acerca de la disponibilidad de inventario actual y futuro, al mismo tiempo que consideraban fechas de entrega solicitadas y edad máxima del producto en el momento de su entrega. Un segundo objetivo fue producir un programa eficiente de nivel de turno para cada planta en un horizonte de 28 días. El tercer objetivo consistió en determinar de manera exacta si una planta podía embarcar una cantidad solicitada de pedidos-líneas-artículos en la fecha y a la hora requeridas dadas la disponibilidad de ganado y las restricciones impuestas por la capacidad de la planta.

Para enfrentar estos tres desafíos, un equipo de IO desarrolló un *sistema integrado de 45 modelos de programación lineal* basado en tres formulaciones de modelo para programar de manera dinámica sus operaciones de fabricación de carne en cinco plantas en tiempo real cuando recibe los pedidos. *Los beneficios totales auditados que se observaron en el primer año* de operación de este sistema fueron de **12.74 millones de dólares**, de los cuales 12 millones correspondieron a la *optimización de la mezcla de productos*. Entre otros beneficios se destacan la disminución de las órdenes perdidas, la reducción de los descuentos de precio y la mejora de las entregas a tiempo.

Fuente: A. Bixby, B. Downs y M. Self, "A Scheduling and Capable-to-Promise Application for Swift & Company", en *Interfaces*, **36**(1): 39-50, enero-febrero de 2006. (En nuestro sitio web, www.mhhe.com/hillier, se proporciona un vínculo con este artículo.)

la producción y la comercialización de estos nuevos productos, la ganancia total de cada uno es aproximadamente la *ganancia por lote que se produce* multiplicada por el *número de lotes*.)

La obtención de estimaciones razonables de estas cantidades requirió del apoyo de personal clave en varias unidades de la compañía. El personal de la división de manufactura proporcionó los datos de la primera categoría mencionada. En la segunda categoría, el desarrollo de estimaciones requirió un análisis de los ingenieros de manufactura involucrados en el diseño de los procesos de producción para elaborar los nuevos artículos. Al analizar los datos de costos que se obtuvieron, junto con la decisión sobre los precios de la división de marketing, el departamento de contabilidad calculó las estimaciones para la tercera categoría.

La tabla 3.1 resume los datos reunidos.

De inmediato, el equipo de IO reconoció que se trataba de un problema de programación lineal del tipo clásico de **mezcla de productos** y procedió a la formulación del modelo matemático correspondiente.

Formulación como un problema de programación lineal

La definición del problema planteado indica que las decisiones que deben tomarse son el número de lotes de los productos que se fabricarán semanalmente, de manera que se maximice su ganancia total.

Para formular el modelo matemático de programación lineal de este problema se define

x_1 = número de lotes del producto 1 que se fabrican por semana
x_2 = número de lotes del producto 2 que se fabrican por semana
Z = ganancia semanal total (en miles de dólares) que generan estos dos productos

TABLA 3.1 Datos del problema de la Wyndor Glass Co.

	Tiempo de producción por lote, horas		
	Producto		
Planta	1	2	Tiempo de producción disponible a la semana, horas
1	1	0	4
2	0	2	12
3	3	2	18
Ganancia por lote	$3 000	$5 000	

Por lo tanto, x_1 y x_2 son las *variables de decisión* del modelo. Si se usa el último renglón de la tabla 3.1 se obtiene

$$Z = 3x_1 + 5x_2.$$

El objetivo es elegir los valores de x_1 y x_2 que *maximice* $Z = 3x_1 + 5x_2$, sujeta a las restricciones impuestas sobre sus valores por las capacidades de producción limitadas de las cuales se disponen en las tres plantas. La tabla 3.1 indica que cada lote del producto 1 que se produce por semana emplea una hora de producción en la planta 1, y sólo se dispone de 4 horas semanales. En términos matemáticos, esta restricción se expresa mediante la desigualdad $x_1 \leq 4$. De igual manera, la planta 2 impone la restricción $2x_2 \leq 12$. El número de horas de producción usadas a la semana en la planta 3 que se consume al elegir x_1 y x_2 como las tasas de producción de los nuevos productos sería $3x_1 + 2x_2$. En consecuencia, la expresión matemática de la restricción de la planta 3 es $3x_1 + 2x_2 \leq 18$. Por último, como las tasas de producción no pueden ser negativas, es necesario restringir las variables de decisión a valores no negativos: $x_1 \geq 0$ y $x_2 \geq 0$.

Para resumir, en el lenguaje matemático de programación lineal, el problema consiste en seleccionar valores de x_1 y x_2 para

$$\text{Maximizar} \quad Z = 3x_1 + 5x_2,$$

sujeta a las restricciones

$$\begin{aligned} x_1 \qquad &\leq 4 \\ 2x_2 &\leq 12 \\ 3x_1 + 2x_2 &\leq 18 \end{aligned}$$

y

$$x_1 \geq 0, \quad x_2 \geq 0.$$

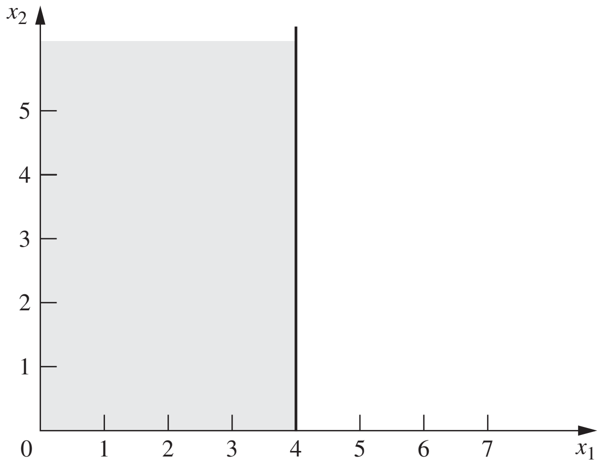
(Observe cómo la información de la tabla 3.1 en esencia se duplica en la distribución de los coeficientes de x_1 y x_2 en el modelo de programación lineal.)

Solución gráfica

Este pequeño problema tiene sólo dos variables de decisión, esto es, sólo dos dimensiones, así que se puede usar un procedimiento gráfico para resolverlo. Este procedimiento incluye la construcción de una gráfica de dos dimensiones con x_1 y x_2 como los ejes. El primer paso es identificar los valores de (x_1, x_2) permitidos por las restricciones. Este objetivo se logra dibujando cada una de las rectas que limitan los valores permitidos por una restricción. Para comenzar, observe que las restricciones de no negatividad $x_1 \geq 0$ y $x_2 \geq 0$ exigen que el punto (x_1, x_2) se encuentre en el lado *positivo* de los ejes (incluso sobre *cualquiera* de los dos ejes), es decir, en el primer cuadrante. Después, debe observarse que la restricción $x_1 \leq 4$ significa que (x_1, x_2) no puede estar a la derecha de la recta $x_1 = 4$. Estos resultados se muestran en la figura 3.1, en la que el área sombreada contiene los únicos valores de (x_1, x_2) permitidos.

De manera parecida, la restricción $2x_2 \leq 12$ (o de modo equivalente, $x_2 \leq 6$) implica que la recta $2x_2 = 12$ debe agregarse a la frontera de la región permisible. La última restricción, $3x_1 + 2x_2 \leq 18$, se encuentra al graficar los puntos (x_1, x_2) tales que $3x_1 + 2x_2 = 18$ (otra recta) para completar la frontera. (Observe que los puntos que cumplen $3x_1 + 2x_2 \leq 18$ son aquellos que están sobre o por debajo de la recta $3x_1 + 2x_2 = 18$, por lo que ésta es la recta que limita, y más allá de ella, la desigualdad no se satisface.) En la figura 3.2 se muestra la región de valores permisibles de (x_1, x_2), llamada **región factible**. (La demostración llamada *Graphical Method* —método gráfico— en el OR Tutor proporciona un ejemplo detallado de la construcción de la región factible.)

El paso final es seleccionar, dentro de esta región factible, el punto que maximiza el valor de $Z = 3x_1 + 5x_2$. Para descubrir cómo realizar este paso de manera eficiente se pueden intentar algunos valores por prueba y error. Por ejemplo, probar, $Z = 10 = 3x_1 + 5x_2$ para ver si existe algún valor de (x_1, x_2) dentro de la región permisible que dé un valor de 10 para Z. Si se dibuja la recta $3x_1 + 5x_2 = 10$ se puede ver que existen muchos puntos sobre esta recta que están dentro de la región (vea la figura 3.3). Después de intentar este valor arbitrario de $Z = 10$ se tiene una mejor



■ **FIGURA 3.1**
El área sombreada muestra los valores de (x_1, x_2) permitidos por $x_1 \geq 0$, $x_2 \geq 0$, $x_1 \leq 4$.

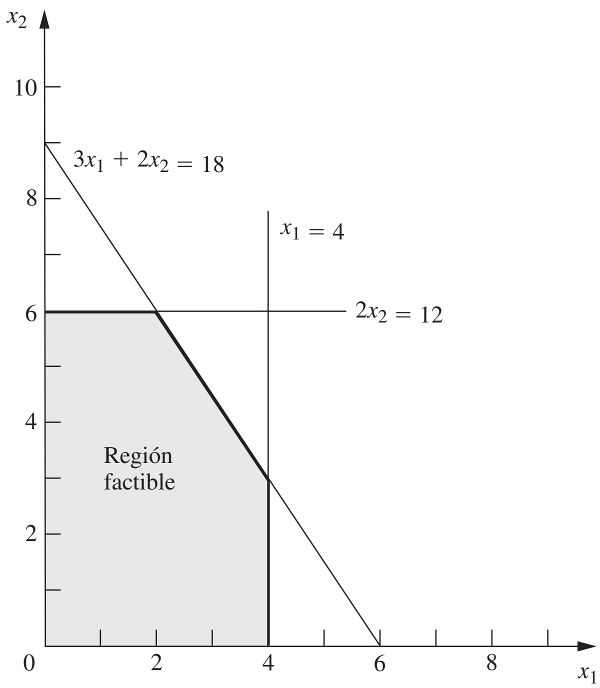
perspectiva, debe intentarse ahora un valor arbitrario más grande, por ejemplo, $Z = 20 = 3x_1 + 5x_2$. De nuevo, la figura 3.3 revela que un segmento de la recta $3x_1 + 5x_2 = 20$ se encuentra dentro de la región, de manera que el máximo valor permisible de Z debe ser, por lo menos, 20.

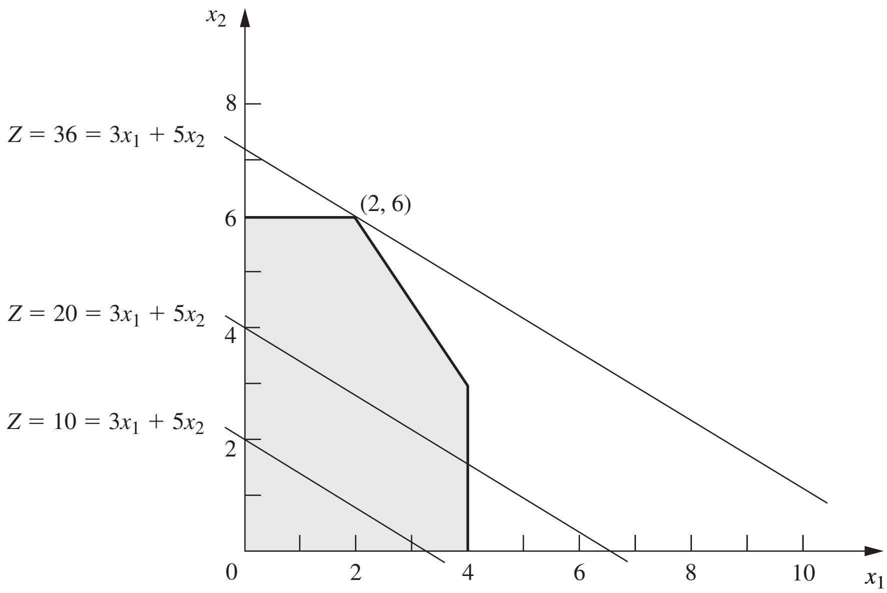
Observe ahora en la figura 3.3 que las dos rectas que se acaban de graficar son paralelas. Esto no es coincidencia, ya que *cualquier* recta construida de esta manera tiene la forma $Z = 3x_1 + 5x_2$ para el valor seleccionado de Z, lo que implica que $5x_2 = -3x_1 + Z$ o, en forma equivalente,

$$x_2 = -\frac{3}{5}x_1 + \frac{1}{5}Z$$

Esta última ecuación, llamada forma de **pendiente-ordenada al origen** de la función objetivo, demuestra que la *pendiente* de las rectas es $-\frac{3}{5}$ (ya que cada incremento de una unidad en x_1 hace que x_2 cambie en $-\frac{3}{5}$), mientras que la ordenada al origen de la recta —la *intersección* con el eje x_2—

■ **FIGURA 3.2**
El área sombreada muestra los valores permitidos de (x_1, x_2), llamada la región factible.





■ **FIGURA 3.3**
El valor de (x_1, x_2) que maximiza $3x_1 + 5x_2$ es (2, 6).

es $\frac{1}{5}Z$ (puesto que $x_2 = \frac{1}{5}Z$ cuando $x_1 = 0$). El hecho de que la pendiente esté fija en $-\frac{3}{5}$ significa que *todas* las rectas construidas de esta manera son paralelas.

De nuevo, si se comparan las rectas $10 = 3x_1 + 5x_2$ y $20 = 3x_1 + 5x_2$ en la figura 3.3, es posible observar que la recta que da el valor mayor de Z ($Z = 20$) se encuentra más lejos del origen hacia arriba que la otra recta ($Z = 10$). Este hecho también está implícito en la forma de pendiente-ordenada al origen de la función objetivo, lo que indica que la intersección con el eje x_1 ($\frac{1}{5}Z$) aumenta cuando crece el valor seleccionado de Z.

Estas observaciones implican que el procedimiento de prueba y error para construir las rectas de la figura 3.3 involucra sólo dibujar una familia de rectas paralelas que contengan al menos un punto en la región factible y elegir la que corresponda al mayor valor de Z. La figura 3.3 muestra que esta recta pasa por el punto (2, 6), lo cual indica que la **solución óptima** es $x_1 = 2$ y $x_2 = 6$. La ecuación de esta recta es $3x_1 + 5x_2 = 3(2) + 5(6) = 36 = Z$, lo cual indica que el valor óptimo de Z es $Z = 36$. El punto (2, 6) se encuentra en la intersección de las dos rectas $2x_2 = 12$ y $3x_1 + 2x_2 = 18$, que se muestra en la figura 3.2, por lo que el punto se puede calcular de manera algebraica como la solución simultánea de estas dos ecuaciones.

Una vez estudiado el procedimiento de prueba y error para encontrar el punto óptimo (2, 6) es posible seguir los pasos de este método en otros problemas. En lugar de dibujar varias rectas paralelas, es suficiente marcar una de ellas con una regla para establecer la pendiente y después mover la regla con pendiente fija sobre la región factible en la dirección en que Z mejora. (Cuando el objetivo sea *minimizar* Z la regla deberá moverse en la dirección en que *Z decrece*.) La regla se deja de mover en el momento en que todavía pasa por un punto de esta región. Este punto es la *solución óptima* deseada.

Con frecuencia se hace referencia a este procedimiento como el **método gráfico** de programación lineal. Se puede usar para resolver cualquier problema de programación lineal con dos variables de decisión. Con alguna dificultad es posible extender el método a tres variables de decisión, pero no más de tres. (En el siguiente capítulo se estudia el *método símplex* para resolver problemas más grandes.)

Conclusiones

El equipo de IO utilizó este procedimiento para encontrar que la solución óptima deseada es $x_1 = 2$, $x_2 = 6$, con $Z = 36$. Esta solución indica que la Wyndor Glass Co. debe fabricar los productos 1 y 2 a una tasa de 2 y 6 lotes por semana, respectivamente, con una ganancia total resultante de 36 000 dólares semanales. No existe otra mezcla de los dos productos que sea tan redituable, *de acuerdo con el modelo*.

No obstante, en el capítulo 2 se puso de manifiesto que un buen estudio de investigación de operaciones no sólo encuentra *una* solución para el modelo *inicial* formulado. Cada una de las seis etapas que se describieron es importante, incluso las pruebas exhaustivas del modelo (vea la sección 2.4) y el análisis posóptimo (sección 2.3).

Si reconoce la totalidad de estas realidades prácticas, el equipo de IO está listo para evaluar la validez del modelo de una manera más crítica (esta explicación continuará en la sección 3.3), y para llevar a cabo un análisis de sensibilidad sobre el efecto que tendría el hecho de que las estimaciones dadas en la tabla 3.1 fueran diferentes debido a inexactitudes, cambios en las circunstancias, etc. (Este tema continuará en la sección 6.7.)

Continuación del proceso de aprendizaje con OR Courseware

Éste es el primero de muchos puntos en los cuales será útil emplear el *OR Courseware* que se encuentra en el CD que acompaña al libro. Un programa clave en este CD es el llamado OR Tutor que contiene un ejemplo de demostración completo del *método gráfico* que se estudia en esta sección. Esta demostración comienza por la introducción de un problema y la formulación de un modelo de programación lineal, antes de aplicar el método gráfico para resolverlo, con la intención de proporcionar un **ejemplo adicional** de formulación de un modelo. Al igual que muchos otros ejemplos de demostración en otras secciones, éste resalta los conceptos que son difíciles de explicar en una página impresa. En el apéndice 1 se puede consultar la documentación sobre el software.

Si el lector desea ver **más ejemplos** puede consultar la sección de problemas resueltos —**Worked Examples**— en el sitio web del libro. Esta sección incluye unos cuantos ejemplos con soluciones completas para casi todos los capítulos, así como un complemento de los ejemplos del libro y del OR Tutor. Los ejemplos de este capítulo comienzan con un problema relativamente directo que implica la formulación de un pequeño modelo de programación lineal y la aplicación del método gráfico. Los ejemplos siguientes implicarán de manera progresiva un reto mayor.

Otra parte clave del OR Courseware es un programa llamado **IOR Tutorial**. Éste realiza muchos procedimientos interactivos para ejecutar los diferentes métodos de solución que se presentan en el libro, lo que permite que el lector se enfoque en el aprendizaje y la ejecución de la lógica del método en forma eficiente, mientras que la computadora realiza los cálculos numéricos. Se incluye un procedimiento interactivo para aplicar el método gráfico en la programación lineal. Una vez que se haya captado este primer procedimiento, un segundo enfoque permite aplicar con rapidez el método gráfico para desarrollar análisis de sensibilidad sobre el efecto de cambios en los datos del problema. Después, es posible imprimir los trabajos y resultados como una tarea. Como los otros procedimientos del IOR Tutorial, éstos están específicamente diseñados para proporcionar al lector una experiencia de aprendizaje eficiente, amena y enriquecedora mientras realiza sus tareas.

Cuando se formule un modelo de programación lineal con más de dos variables de decisión, por lo que no puede usarse el método gráfico, el *método símplex* descrito en el capítulo 4 permitirá encontrar una solución óptima de inmediato. Obtenerla también es útil para la *validación del modelo* puesto que encontrar una *solución sin sentido* indica que se cometieron errores en la formulación del modelo.

En la sección 1.4 se mencionó que el OR Courseware es una introducción a los tres paquetes de software comerciales que más se usan —Excel Solver, LINGO/LINDO y MPL/CPLEX— para resolver una variedad de modelos de IO. Los tres paquetes incluyen el método símplex para resolver problemas de programación lineal. En la sección 3.5 se describe cómo usar Excel para formular y resolver modelos de programación lineal en el formato de una hoja de cálculo. Las descripciones de los otros paquetes se proporcionan en la sección 3.6 —MPL y LINGO—, los suplementos 1 y 2 de este capítulo en el sitio web del libro —LINGO—, la sección 4.8 —CPLEX y LINDO— y el apéndice 4.1 —LINDO—. Además, el OR Courseware incluye un archivo para cada uno de los tres paquetes que muestra cómo se puede usar para resolver los ejemplos en este capítulo.

3.2 MODELO DE PROGRAMACIÓN LINEAL

El problema de la Wyndor Glass Co. se diseñó para ilustrar un problema común de programación lineal, en versión simplificada. Sin embargo, esta técnica es muy versátil como para describirla mediante un solo ejemplo. En esta sección se presentarán las características generales de los problemas de programación lineal y las distintas formas legítimas del modelo matemático.