

BREVE MANUAL DE WINQSB

PROFESOR: DAVID LAHOZ ARNEDO

PROGRAMACIÓN LINEAL

Definición: Un problema se define de programación lineal si se busca calcular el máximo o el mínimo de una función lineal, la relación entre las variables es solamente lineal.

Por ejemplo:

1.- maximizar el beneficio de una empresa que vende dos productos X1 y X2 a 12 y 10 unidades cada uno: $\text{Max } Z = 12X1 + 10X2$

2.- minimizar el coste de una empresa que compra dos productos X1 y X2 a 5 y 7 unidades cada uno: $\text{Min } Z = 5X1 + 7X2$

Las funciones lineales no están acotadas, no tienen máximos ni mínimos numéricos. En un problema de programación lineal las variables no suelen ser libres sino que deben cumplir una serie de restricciones. Estas restricciones acotan los posibles valores de la solución.

Por ejemplo:

1.- se deben de producir más de 20 unidades entre los dos productos: $X1 + X2 \geq 20$.

2.- un producto consume 10 horas de elaboración y otro 15 y se dispone de 100 horas como máximo: $10X1 + 15X2 \leq 100$

Las restricciones son desigualdades de funciones lineales.

Para ver como se introducen los datos de un problema de programación lineal en WINQSB se trabajará con el siguiente ejemplo:

Ejemplo1.

Una empresa distribuye coches y motos entre las fábricas y los concesionarios. La distribución la realiza en tren o en camión. Las fábricas tienen almacenadas 150 coches y 100 motos. En cada vagón de tren podemos colocar 5 coches y 2 motos y en cada camión 2 coches y 3 motos. Si la distribuidora obtiene 12 y 8 unidades de beneficio por cada vagón de tren o camión, respectivamente, ¿Cuántas vagones y camiones debe de utilizar para obtener el máximo beneficio?

Planteamiento del problema:

Se definen X1 como el nº de vagones y X2 como el de camiones.

Función Objetivo, F. O.: Se quiere maximizar el beneficio total. Si se ganan 12 unidades por cada vagón $12X1$ y 8 por cada camión $8X2$, en total $12X1 + 8X2$.

Restricciones: Solo se dispone de 150 coches y 100 motos para distribuirlos en vagones, o en camiones, $2X1 + 3X2 \leq 100$ motos y $5X1 + 2X2 \leq 150$ coches, .

Expresión matemática:

$\text{Max } Z = 12X1 + 8 X2$	Beneficio total
Sujeto a	
$5X1 + 2X2 \leq 150$	Nº de coches
$2X1 + 3X2 \leq 100$	Nº de motos
y por supuesto $X1 \geq 0$ y $X2 \geq 0$.	

Resolución en WINQSB:

Se abre WINQSB en el modulo *Linear and Integer Programming*. Para introducir un nuevo problema se selecciona *File/ New Problem*. Aparece una pantalla (Figura 1) donde se introducen las especificaciones del problema a resolver.

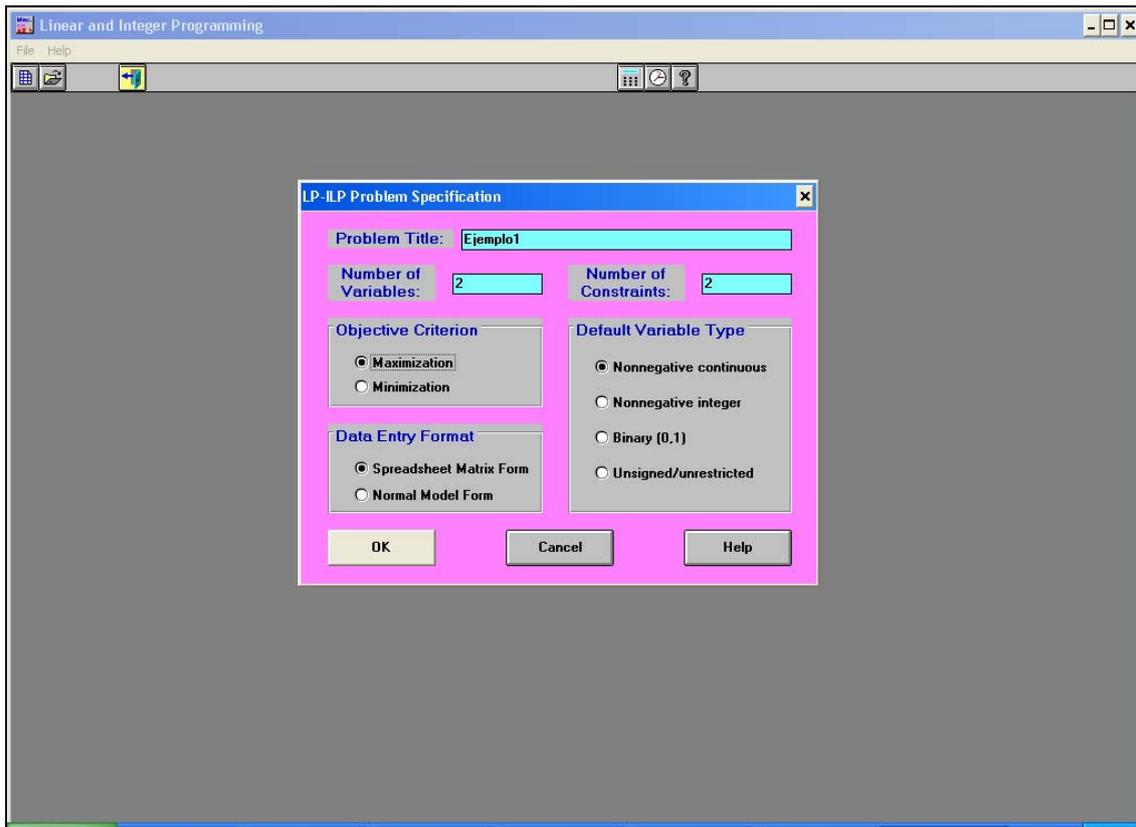


Figura 1: Pantalla inicial de Programación Lineal y Entera.

La primera casilla es para introducir un nombre para el problema (*Problem Title*), en nuestro caso Ejemplo 1. En la segunda casilla se introducen el nº de variables (*Number of variables*) que tiene el problema, en Ejemplo1 son 2, el nº de vagones y el nº de camiones. La siguiente casilla es para el número de restricciones (*Number of Constraints*) del problema, en este caso 2, motos y coches, sin contar las restricciones de que sean positivas. Se selecciona maximizar o minimizar en la casilla de *Objective Criterion*. En el Ejemplo1 *Maximization*. La última opción es para decidir con que tipos de variables se va a trabajar (*Default Variable Type*): reales positivas (*Nonnegative continuous*), enteras no negativas (*Nonnegatives integer*), binarias o dicotómicas (*Binary*) o sin ninguna restricción de valor (*Unsigned/Unrestricted*).

Hay dos formas de introducir los datos (*Data Entry Format*), como tabla (*Spreadsheet Matrix Form*) o de forma libre (*Normal Model Form*). Aquí solo se va a desarrollar la forma de tabla, más intuitiva y sencilla.

En la Figura 1 se puede ver como queda la pantalla después de introducir las especificaciones del problema Ejemplo1.

Se pulsa *OK* y aparece la tabla donde se van a introducir los coeficientes, (Figura 2). El nombre de la tabla es el mismo que se ha dado al problema. Como nombre de las columnas las variables y como nombres de las filas la Función Objetivo y las restricciones numeradas, C1, C2,... En la parte inferior se da información sobre el tipo de las variables y las cotas de estas. En el Ejemplo1 la cota inferior es 0 y la cota superior es infinito (*M* en la celda) y las variables son de tipo continuo (*Continuous*).

Para estudiar mejor los resultados del problema antes de resolverlo se vas a cambiar el nombre de las variables y de las restricciones. Para cambiar el nombre de las variables se selecciona *Edit/ Variable Names*. En nuestro caso podemos llamar TREN y CAMIÓN a las variables. Para las restricciones se elige *Edit/ Constraint Names* y se

modifican los nombres a MOTOS y COCHES. También se puede modificar el nombre del problema (*Edit/ Problem Name*) y el criterio de la F. O. (*Edit/ Objective Function Criterion*).

Ahora ya se pueden introducir los coeficientes del problema. En la primera fila se introducen los coeficientes que acompañan a las variables en la Función Objetivo, F. O., debajo de la variable tren se coloca el 12 y debajo de camión el 8, ya que la F. O. de Ejemplo1 es MAXIMIZAR $12 * VAGÓN + 8 * CAMIÓN$.

Los coeficientes de las variables en las restricciones se introducen de manera análoga, colocando según el nombre de la restricción y el nombre de la variable el coeficiente que corresponda. Por ejemplo el coeficiente que acompaña a CAMIÓN en la restricción de COCHES es 2, en la celda con nombre de fila COCHE y nombre de COLUMNA CAMIÓN se escribe 2. Además para las restricciones hay que especificar la dirección de la desigualdad y el valor tope de esta. Si se desea modificar la dirección de alguna desigualdad, pulsando dos veces sobre ella con el ratón va mostrando diferentes posibilidades. Si se quiere alguna desigualdad diferente a las que ofrece el programa se puede introducir por teclado. La cota de la desigualdad se introduce donde pone *R.H.S.* (algo así como el valor a la derecha de la desigualdad en inglés).

Si las variables tuvieran alguna restricción de valor, tanto superior como inferior, está se introduciría en la fila *UpperBound* o *LowerBound*, respectivamente. En el Ejemplo1 la cota mínima de las variables es cero, no tiene sentido un número negativo, y la máxima infinito, M en el programa.

Y si las variables no pudieran ser reales o continuas y tuvieran que ser binarias o enteras,... bastaría con pulsar dos veces sobre la fila *VariableType* con la columna de la variable que corresponda.

En la Figura 2 se muestra como queda la tabla una vez introducidos los coeficientes del problema Ejemplo1.

The screenshot shows the LPILP software window titled 'Linear and Integer Programming'. The main window contains a table for 'Ejemplo1' with the following data:

Variable -->	TREN	CAMION	Direction	R. H. S.
Maximize	12	8		
COCHES	5	2	<=	150
MOTOS	2	3	<=	100
LowerBound	0	0		
UpperBound	M	M		
VariableType	Continuous	Continuous		

At the top of the table, there is a status bar for 'MOTOS : R.H.S.' with a value of 100. The software interface includes a menu bar (File, Edit, Format, Solve and Analyze, Results, Utilities, Window, WinQSB, Help) and a toolbar with various icons. A large watermark 'LPILP' is visible in the background of the main window.

Figura 2: Pantalla de introducción de los datos del problema Ejemplo1.

Una vez introducido el problema lo mejor es guardarlo en un archivo para poder recuperar los datos en cualquier instante. Seleccionando *File/ Save Problem* o *File/ Save Problem as* aparecerá la pantalla típica de guardar de Windows. Se puede cambiar el nombre del archivo y la carpeta donde se quiera guardar. Notar que la extensión que le da el programa al archivo es *.lpp. La pantalla de salvado se muestra en Figura 3.

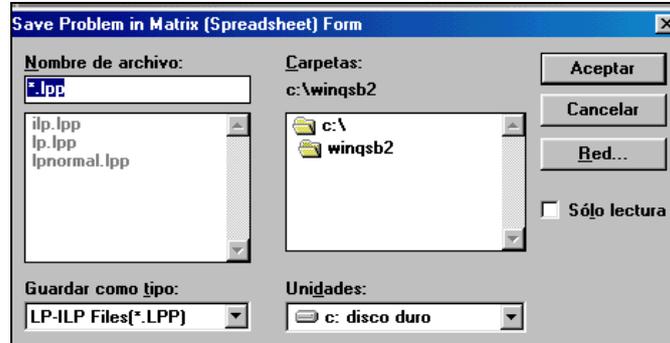


Figura 3: Pantalla de WINQSB para guardar los datos de un problema.

Para resolver el problema se puede elegir entre dos opciones, ver solo el resultado o ver los pasos del método de resolución. WINQSB utiliza tres métodos de resolución el "simplex", "ramificación y acotación" (Bound And Branch) y "método gráfico". Para variables continuas y sin restricciones utiliza el primero o el último mientras que el segundo es para variables binarias y enteras. Los dos primeros, además, se pueden resolver viendo todas las iteraciones. Para resolver el problema por el método numérico sin ver los pasos (ya sea por simplex o ramificación según las variables) se selecciona la opción *Solve and Analyze/ Solve the Problem* o pulsando sobre el botón . Aparece una pequeña ventana indicando que el problema ha sido resuelto y pulsando sobre *Aceptar* mostrará la tabla con la solución, si ha podido calcularla. En la Figura 4 se muestra el resultado que obtiene WINQSB para el problema Ejemplo1.

Combined Report for Ejemplo1							
	19:20:58	8/05/2003 19:20:57 p.m.	8/05/2003 19:20:57 p.m.	8/05/2003 19:20:57 p.m.	8/05/2003 19:20:57 p.m.		
	Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status	Allow Min.
1	TREN	22.7273	12.0000	272.7273	0	basic	5.33
2	CAMION	18.1818	8.0000	145.4545	0	basic	4.80
	Objective	Function	(Max.) =	418.1818			
	Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price	Allow Min. I
1	COCHES	150.0000	<=	150.0000	0	1.8182	66.6
2	MOTOS	100.0000	<=	100.0000	0	1.4545	60.0

Figura 4: Tabla de solución del problema Ejemplo1.

Los valores que muestra son los de la solución óptima del problema. En el Ejemplo1 para maximizar los beneficios se deben distribuir 22.72 vagones y 18.18 camiones (Figura 4). Contabiliza las ganancias con cada elemento y muestra el valor de la F. O.: 418.18 unidades. Esto es lo máximo que se puede obtener con las restricciones del problema. Además de la solución del problema muestra cuanto pueden variar los coeficientes de las variables en la F.O. sin que se modifique la solución. Por ejemplo el beneficio por vagón es de 12 unidades, aunque se modifica este valor entre 5.33 (*Allowable Min. c[j]*) y 20 (*Allowable Max. c[j]*) el número de vagones que se deben de distribuir para maximizar los beneficios sigue siendo 22.72.

En la parte inferior de la tabla de soluciones (Figura 4) se da información sobre como quedan las restricciones con la solución dada. Para la restricción de coches existía un tope de 150 (*Right Hand Side o R. H. S.*) y la solución llega a utilizarlos todos, los 150 (*Left Hand Side*), no hay excedente ni holgura (*Slack or Surplus*). Por último muestra cuanto pueden variar las cotas de las restricciones sin que la solución varíe, similar al caso de los coeficientes de las variables en la función objetivo. Variando los 150 coches entre 66.66 y 250 la solución no se modifica.

Si se quiere resolver el problema por el método gráfico hay que seleccionar *Solve And Analyze/ Graphic Method* o pulsar sobre el botón . Al hacerlo aparece una ventana para seleccionar en que ejes se muestran cada variable. Por ejemplo se puede utilizar el eje X para la variable TREN y el eje Y para CAMIÓN. Pulsando sobre *Aceptar* muestra la pantalla con las variables en los ejes, la F. O., las restricciones y la solución (Figura 5).

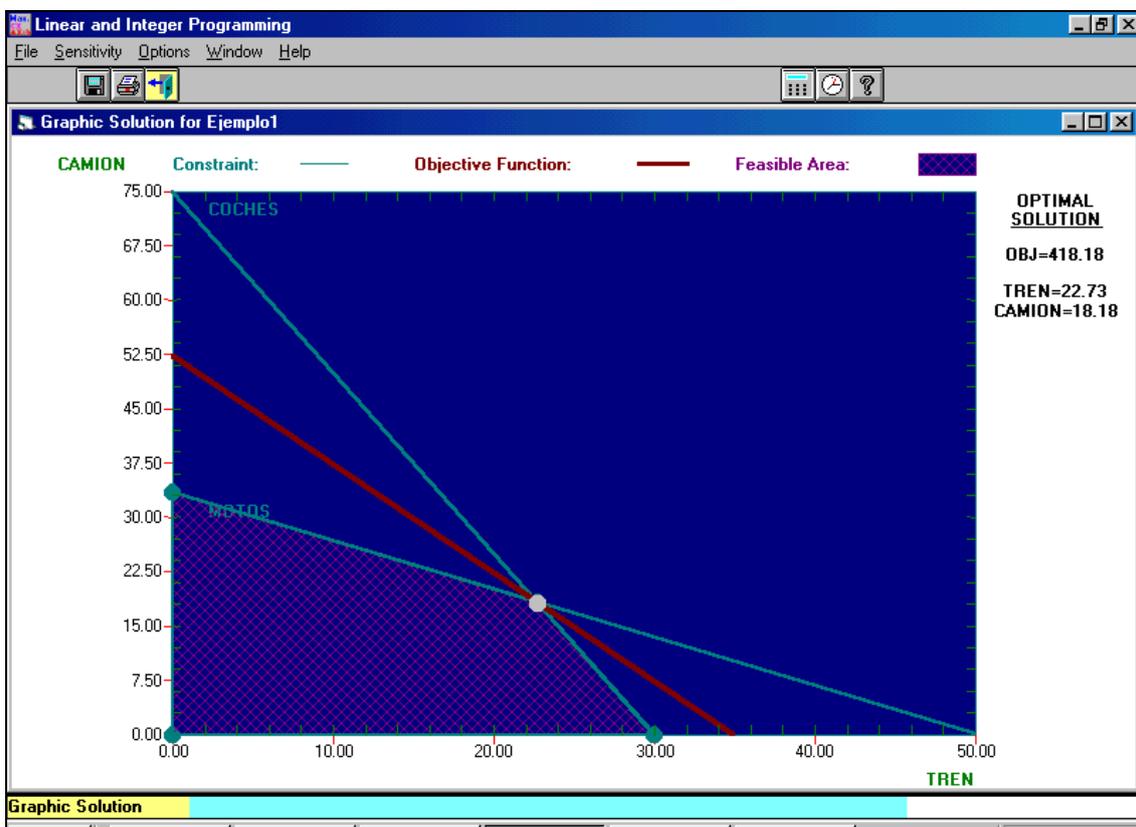


Figura 5: Solución del problema Ejemplo1 por el método gráfico.

Tal como se había seleccionado en el eje X se representa la variable TREN y en el eje Y la variable CAMIÓN. En color azul muestra las rectas correspondientes a las restricciones (*Constraint*). Se puede observar el área factible (*Feasible Area*), con fondo a cuadros rojos, que recoge todos los puntos que cumplen las restricciones. Como puntos más gruesos marca los vértices de la zona. Normalmente, estos son los candidatos a la solución óptima, ya que son factibles, cumplen la restricciones y es fácil que allí se dé el máximo o el mínimo de la F. O. La recta que representa la F. O. se va moviendo por el área factible hasta encontrar el máximo dentro de ella. En el gráfico se representa en color marrón (*Objective Function*). El máximo lo alcanza en el punto que marca en blanco, y sus coordenadas se muestran a la derecha del gráfico: TREN=22.73 y CAMION=18.18. Estos valores dan un valor máximo a la Función Objetivo de 418.18.

Si los colores del gráfico no son lo suficientemente claros se pueden modificar en *Options/ Changes XY Ranges and Colors*. Permite modificar el rango de visión de las variables en el gráfico.

El método gráfico para dos variables es muy claro, para más pierde claridad. solo se pueden representar pares de variables, con lo que se muestra el área factible solo para estas y no para todas.

Otra forma de resolver el problema es mostrando todos los pasos del método. No es una forma nueva de resolución es el método numérico pero mostrando todas las iteraciones. En Ejemplo1 como las variables son continuas va mostrando las diferentes tablas que el método del simplex va obteniendo hasta encontrar la solución óptima. Para seleccionar este forma de resolución hay pulsar sobre *Solve and Analyze/ Solve and Display Steps* o sobre el botón . Va mostrando lo que se conoce como la tabla del simplex de cada iteración. Para pasar a la siguiente iteración se vuelve a pulsar sobre la misma opción, y así hasta que encuentre la solución óptima. En ese momento aparece una ventana con el mensaje *"The simplex method is completed"*, esto es, el método del simplex ya ha encontrado la solución óptima del problema. No muestra la solución. Para poder verla se selecciona *Results/ Solution Summary*. Y para estudiar como han quedado las restricciones del problema se elige *Results/ Constrains Summary*. (Recordar que resolviendo el problema directamente esta información aparecía en la misma tabla resumen).

Modificación del problema:

Ejemplo1.b:

Supongamos que se dispone de una nueva forma de distribución. Por ejemplo que se permita la distribución de coches y motos por barco. En cada barco entran 4 de las primeras y 3 de las segundas. El beneficio que obtenemos por cada barco es de 4 unidades.

En este caso se esta añadiendo una nueva variable, X3 o BARCO, al problema Ejemplo1. Se selecciona *Edit/ Insert a Variable* y aparece una ventana par introducir la posición de la nueva variable en la tabla. Se puede elegir entre usar el nombre por defecto, X3, o darle un nombre diferente. En el ejemplo la nueva variable se puede situar colocarla al final de la tabla (*The end*) y darle el nombre (*Inserted Name*) BARCO. Aparece de nuevo la tabla con los coeficientes del problema original más una columna con el nombre de la nueva variable. Se deben de introducir los coeficientes de la nueva variable, tanto en la F. O. como en las restricciones. Después de introducir de los coeficientes de BARCO en la tabla Ejemplo1 queda como se muestra en la Figura 6.

Variable -->	TREN	CAMION	BARCO	Direction	R. H. S.
Maximize	12	8	4		
COCHES	5	2	4	<=	150
MOTOS	2	3	3	<=	100
LowerBound	0	0	0		
UpperBound	M	M	M		
VariableType	Continuous	Continuous	Continuous		

Figura 6: Tabla de coeficientes con la incorporación de la nueva variable.

Si se resuelve el problema con este nuevo planteamiento se puede comprobar que la solución no varía. El beneficio de la distribución por barco es demasiado pequeño para que sea seleccionado (Figura 7). En la solución óptima la variable Barco toma valor 0.

Report for Ejemplo1								
12:40:43		Friday	May	09	2003			
Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status	Allowable Min. c(j)	Allowable Max. c(j)	
1	TREN	22.7273	12.0000	272.7273	0	basic	5.3333	20.0000
2	CAMION	18.1818	8.0000	145.4545	0	basic	4.8000	18.0000
3	BARCO	0	4.0000	0	-7.6364	at bound	-M	11.6364
Objective	Function	(Max.) =	418.1818					
Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price	Allowable Min. RHS	Allowable Max. RHS	
1	COCHES	150.0000	<=	150.0000	0	1.8182	66.6667	250.0000
2	MOTOS	100.0000	<=	100.0000	0	1.4545	60.0000	225.0000

Figura 7: Solución del problema con la incorporación de la nueva variable.

Puesto que esta nueva distribución no nos aporta nada la podemos eliminar del problema con *Edit/ Delete a Variable*. Se selecciona la variable BARCO y se pulsa *Aceptar*. Después pregunta si se esta seguro de querer borrarla. Vuelve a mostrar la tabla del problema inicial habiendo eliminado la variable BARCO con todos sus coeficientes, como la Figura2 con el problema Ejemplo1 original.

Ejemplo1.c:

Ahora acompañando a los productos deben de ir trabajadores de la empresa. Cada vagón de tren necesita de 4 trabajadores y cada camión de 2. La empresa solo dispone de 80 empleados. En este caso se esta añadiendo una nueva restricción al problema: $4 \cdot \text{TREN} + 2 \cdot \text{CAMION} \leq 80 \text{ EMPLEADOS}$. Para hacer esto en WINQSB se selecciona *Edit/ Insert a Constraint*. Se abre la misma pantalla que a la hora de insertar una nueva variable. Se introduce la nueva restricción al final de la tabla con el nombre EMPLEADOS. En la tabla de coeficientes se introducen los correspondientes a la nueva restricción, así como la desigualdad y la cota (Figura 8).

Variable -->	TREN	CAMION	Direction	R. H. S.
Maximize	12	8		
COCHES	5	2	<=	150
MOTOS	2	3	<=	100
EMPLEADOS	4	2	<=	80
LowerBound	0	0		
UpperBound	M	M		
VariableType	Continuous	Continuous		

Figura 8: Tabla del problema con la incorporación de la nueva restricción

Si se resuelve el problema con la nueva restricción, esta vez si se modifica la solución. La solución que se había obtenido en el problema original sobrepasa el número de empleados de la empresa. El área factible en este caso se reduce y la nueva solución es distribuir 5 vagones y 30 camiones, reduciéndose el beneficio a 300 unidades (Figuras 9 y 10). Notar que con esta nueva solución no se distribuyen 65 coches, *Slack or Surplus* en restricción COCHES.

Combined Report for Ejemplo1							
13:03:04		9/05/2003 13:03:04 p.m.	9/05/2003 13:03:04 p.m.	9/05/2003 13:03:04 p.m.	9/05/2003 13:03:04 p.m.		
Decision Variable	Solution Value	Unit Cost or Profit c(j)	Total Contribution	Reduced Cost	Basis Status	Allowable Min. c(j)	Allowable Max. c(j)
1	TREN	5.0000	12.0000	60.0000	0	basic	5.0000
2	CAMION	30.0000	8.0000	240.0000	0	basic	30.0000
Objective	Function	(Max.) =	300.0000				
Constraint	Left Hand Side	Direction	Right Hand Side	Slack or Surplus	Shadow Price	Allowable Min. RHS	Allowable Max. RHS
1	COCHES	85.0000	<=	150.0000	65.0000	0	85.0000
2	MOTOS	100.0000	<=	100.0000	0	1.0000	100.0000
3	EMPLEADOS	80.0000	<=	80.0000	0	2.5000	80.0000

Figura 9: Solución del problema con la incorporación de la nueva restricción

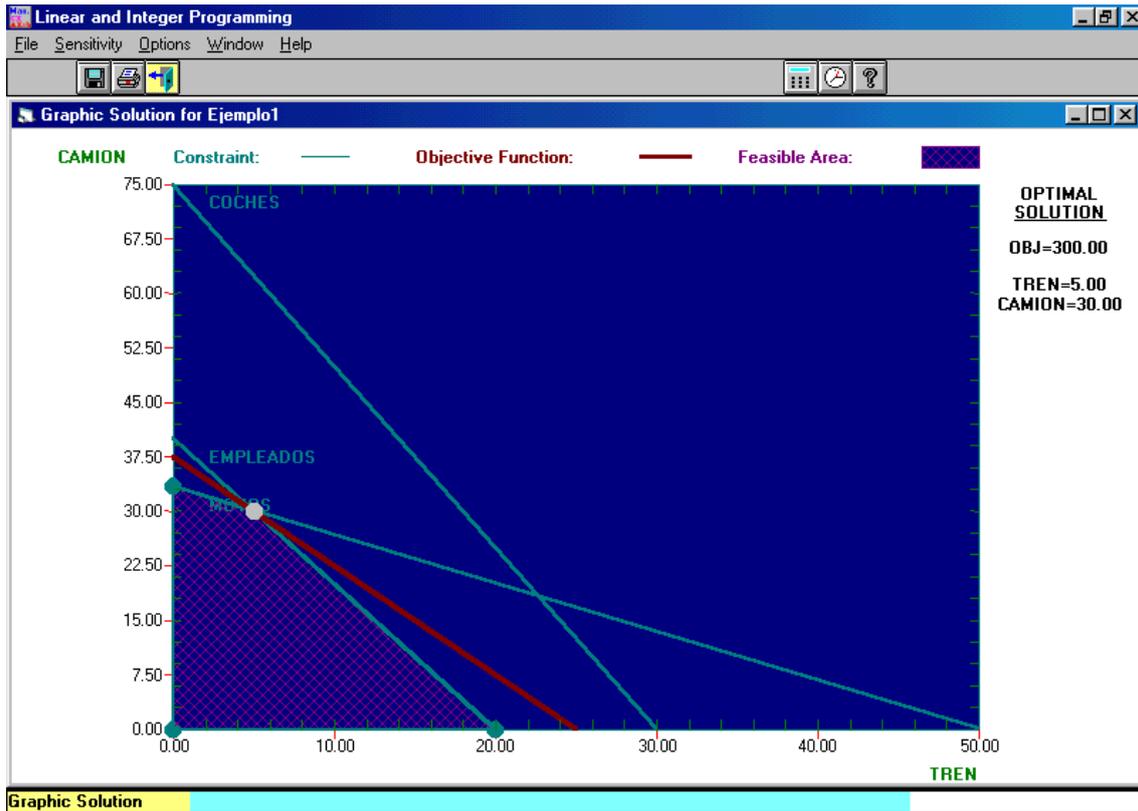


Figura 10: Gráfica del problema con la incorporación de la nueva restricción (notar la modificación del área factible del problema inicial).

Análisis paramétrico del problema:

WINQSB permite el estudio paramétrico del problema, ver que ocurre con la solución si se modifica algún coeficiente o si se añade alguna nueva relación entre las variables. Por ejemplo si se desea estudiar como varía la solución del problema si se modifica el beneficio por camión.

Se selecciona *Solve and Analyze/ Perform Parametric Analysis* o se pulsa sobre el botón

. Aparece, entonces, una pantalla en la que se debe de elegir en que expresión se desea realizar el estudio paramétrico; la F. O. (*Objective Function*) o las restricciones (*R. H. S.*). Si la elección es la Función Objetivo se puede realizar el estudio en cualquiera de la variables o en una combinación de estas. Si es en las restricciones hay que elegir entre una de estas y una combinación lineal de ellas. En el caso de que desee hacer una modificación en una combinación lineal se debe pulsar sobre *Perturbation Vector* y en la nueva pantalla colocar los coeficientes de la combinación lineal.

Por ejemplo para ver como se modifica la solución al variar el beneficio que se obtiene por cada camión. Se selecciona la F. O. y después la variable CAMION. Se pulsa *OK* y aparece una tabla donde se resume el estudio paramétrico (Figura 11).

Analysis for Ejemplo1 -- Objective Function							
Range	From Coeff. of CAMION	To Coeff. of CAMION	From OBJ Value	To OBJ Value	Slope	Leaving Variable	Entering Variable
1	8.0000	18.0000	300.0000	600.0000	30.0000	TREN	Slack_EMPLEADOS
2	18.0000	M	600.0000	M	33.3333		
3	8.0000	6.0000	300.0000	240.0000	30.0000	CAMION	Slack_MOTOS
4	6.0000	-M	240.0000	240.0000	0		

Figura 11: Tabla que resume el estudio paramétrico de la variable CAMION en la F. O.

La tabla muestra los intervalos entre los que se mueve el coeficiente sin modificar las variables de la solución. Lógicamente la solución si varía. Por ejemplo si el coeficiente está entre 8 (*From Coeff. of CAMION*) y 18 (*To Coeff. of CAMION*) entonces la F. O. varía entre 300 (*From OBJ Value*) y 600 (*To OBJ Value*) con las mismas variables en la solución. En el momento que el coeficiente supere el valor 18 sale la variable TREN (*Leaving Variable*) y entra la variable Holgura de Empleados (*Entering Variable*). Se pueden ver los resultados gráficamente seleccionando *Results/ Graphic Parametric Analysis*. En el eje X de la gráfica se representan los diferentes valores que va tomando el coeficiente de la variable CAMION en la F. O. y en el eje Y los valores de la Función Objetivo (Figura 12).

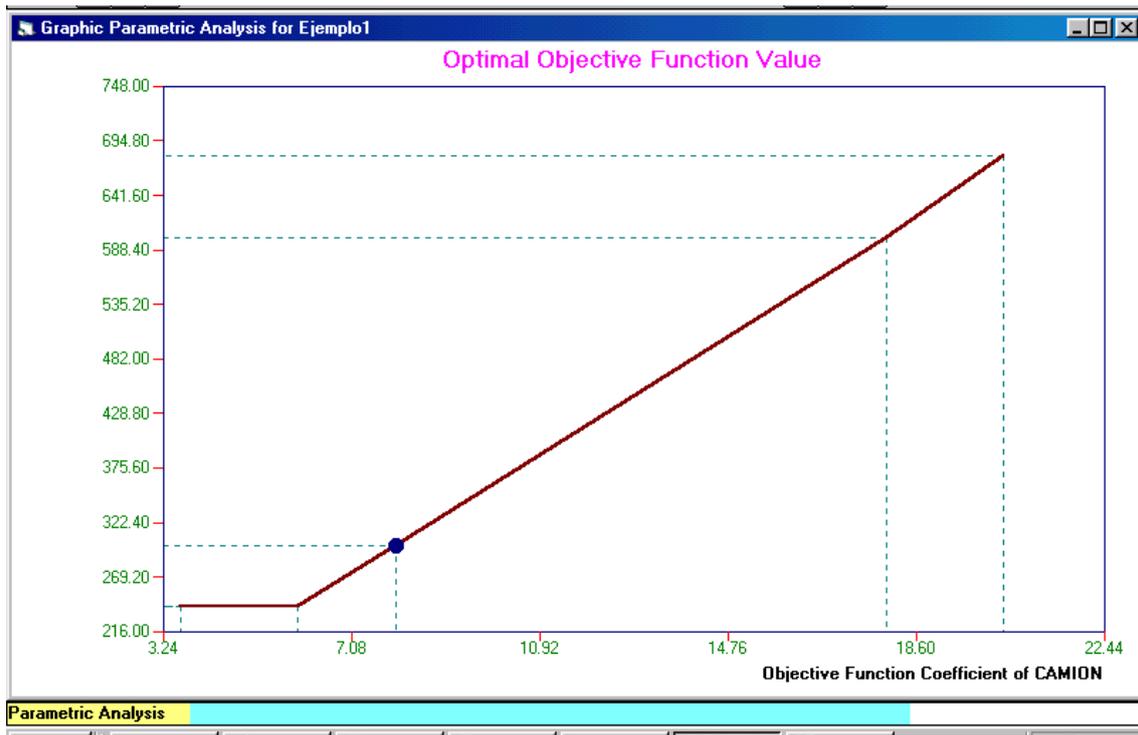


Figura 12: Gráfica que muestra el estudio paramétrico de la variable CAMION en la F. O.

Programación Lineal Entera:

Volvamos al problema original eliminando la restricción de EMPLEADOS. Recordar que entonces se tenía como solución distribuir 22.72 vagones y 18.188 camiones. Lógicamente esta es una solución que no sirve ya que lo normal sería distribuir un número completo de vagones o camiones. No se permiten valores continuos para las variables sino entero. Hasta ahora la única restricción de valor que se ha puesto a las variables es ser mayor que cero. Ahora se eliminan también los valores con decimales. En la ventana inicial de entrada del problema (Figura 1), se selecciona *Nonnegative integer* en el tipo de variables. Si ya se tienen los coeficientes introducidos en la tabla se puede modificar el tipo de variable pulsando dos veces sobre *Variable Type* hasta que muestre *Integer* en ambas (se pueden tener variables de diferentes tipos, enteras y binarias con continuas, etc).

Si se resuelve el problema entero la solución que se obtiene es distribuir 27 vagones y 13 camiones. El beneficio disminuye de 418.18 a 412 unidades. Si las variables son binarias o enteras no se puede resolver gráficamente.

El método de resolución que utiliza WINQSB para calcular la solución con variables enteras o binarias es Ramificación-y-Acotación (Branch-And-Bound). Primero calcula la solución por el método del simplex, si la solución es buena (las variables son enteras

o binarias según el caso), se para. Si no lo son va ramificando a valores enteros (o binarios) y acotando todas las soluciones peores que las ya calculadas. Las diferentes ramificaciones que va haciendo el método se pueden ir viendo resolviendo el problema por pasos. La primera iteración es la solución que teníamos del problema con variables continuas, a partir de estas va estudiando que solución es la mejor tomando valores enteros cercanos a esa solución. En nueve iteraciones encuentra la solución óptima del problema Ejemplo1.

El método de Ramificación-y-Acotación tiene asociados una serie de parámetros para fijar la búsqueda, las acotaciones,... En WINQSB se pueden modificar algunos de estos dentro de *Solve and Analyze: Change Integer Tolerance, Specify Solution Quality y Specify Variable Branching Priorities*.

Todos los estudios paramétricos que hemos realizado para variables continuas se pueden hacer para variables enteras o binarias.

Otros ejemplos de problemas de programación lineal:

En un problema de programación lineal lo más dificultoso no es la resolución (que como hemos visto es tan sencillo como introducir los datos en WINQSB), sino el planteamiento, saber escribir la función objetivo y las restricciones.

- **Ejemplo2:** Una empresa que fábrica colchones posee dos almacenes a los que envía sus productos. Los colchones los carga en camiones en la fábrica y los descarga en los almacenes. Un almacén está cerca de la empresa por lo que los camiones que van a ella solo necesitan un conductor y un cargador. Los camiones que van al segundo almacén, más alejado y deben pasar la frontera y deben de llevar dos conductores, un cargador y un intérprete. La empresa cuenta con 150 conductores, 120 cargadores y 50 intérpretes. Si el beneficio que obtiene por llevar un camión al primer almacén es 15 y 10 por llevar al segundo, ¿cuantos camiones debe de llevar a cada almacén para maximizar beneficios?. **Solución:** 120 al primer almacén y 6 al segundo con un beneficio de 1860.
- **Ejemplo3:** Una compañía química elabora tres tipos de ingredientes a partir de tres sustancias. Por cada kilo de sustancia produce un kilo de ingrediente. Debe de producir por lo menos 10000 kilos entre los tres ingredientes. Tiene una serie de restricciones técnico- sanitarias: producir menos de 3000 kilos de ingrediente 1, más de 1500 kg. de ingrediente 2 y más de 2000 de ingrediente 3. El costo de cada sustancia es de 8, 10 y 11 unidades respectivamente. ¿cuantos kilos de cada ingrediente debe de elaborar para minimizar los costos?. **Solución:** Elaborar de cada ingrediente 3000, 5000 y 2000 respectivamente.
- **Ejemplo4:** Se quiere maximizar el perímetro de un rectángulo donde la diferencia entre los lados sea mayor que uno y la mitad del perímetro sea mayor que uno. **Solución:** En este caso el problema tiene una solución no acotada (*Unbounded Solution* en solución gráfica y *The problem is unbounded* en solución numérica), la F. O. puede ser infinito. Las variables pueden tomar los valores tan grandes como se quiera con lo que la función objetivo se puede hacer crecer lo que se desee. Notar que en la solución gráfica el área factible no está acotada, no existe ninguna restricción que cierre el área.
- **Ejemplo5:** El mismo problema que antes pero ahora las restricciones son que el doble del primer lado no sobrepase en tres unidades al segundo y que el primero sea por lo menos dos unidades mayor que el segundo. **Solución:** Este problema no tiene solución, no es factible (*Infeasible Solution*), no existe ningún punto que cumpla las dos restricciones a la vez. Si se resuelve gráficamente se puede

comprobar que no existe área factible, ninguna zona cumple las dos restricciones juntas.

- **Ejemplo6:** Resolver el siguiente ejemplo en WINQSB: $\text{Min } Z = -X_1 + X_2$ sujeto a $X_1 + X_2 \geq 1$ y $2X_1 - 2X_2 \leq 1$ y $X_1 \leq 3/2$. **Solución:** Este ejemplo tiene solución múltiple, en este caso infinitas soluciones igual de buenas. Una solución es $X_1=0.75$ y $X_2=0.25$ con F. O.=-0.5, pero igual de buena es $X_1=1.5$ y $X_2=1$ con F. O.=-0.5. Si se resuelve numéricamente aparece un mensaje al lado del valor de la F. O. notificando que existen otras soluciones alternativas igual de buenas (*Alternate Solution Exists!!*). Si se resuelve gráficamente se puede comprobar que la F. O. coincide con una restricción, por lo que cualquier punto que este sobre esa restricción toma el mismo valor en la F. O.
- **Ejemplo7:** Un empresario puede invertir en 8 actividades produciendo unos beneficios de 4, 2, 6, 3, 5, 7, 6, 4 unidades. Dispone de 15 millones para invertir en todas las actividades requiriendo 3, 4, 8, 4, 6, 9, 5, 4 millones respectivamente. Además las actividades 3 y 6 son incompatibles. Si se invierte en la actividad 4 y 7 se debe de invertir en la 2. Si se invierte en la actividad 5 se debe de hacerlo también en la 1. Por otro lado se debe de invertir en al menos tres proyectos. Para la ejecución de los proyectos se necesita una materia prima de la cual se disponen de diez unidades y los consumos respectivos de cada proyecto son 3, 2, 4, 2, 5, 1, 3, 2 unidades. ¿Qué proyectos deberían seleccionarse para maximizar el beneficio?. **Nota:** Ahora las variables son binarias X_i toma el valor 1 si se invierte en la actividad i y 0 en caso contrario. Para plantear las restricciones hay que tener en cuenta que las variables son binarias. Por ejemplo para la restricción de que la actividad 3 y 6 son incompatibles, se tiene en cuenta que entre las dos no valgan más de uno, así las dos no se pueden dar a la vez: $X_3 + X_6 \leq 1$. Las únicas posibilidades son que valgan cero las dos, o que solo una de ellas valga uno, nunca las dos pueden valer uno. **Solución:** Invertir en las actividades 1, 3 y 8. Recordar que hay que modificar el tipo de variables a binarias, en la ventana de introducción al problema (Figura 1) o en la tabla de introducción de los coeficientes (*Variable Type* de la Figura 2).