**\_\_\_**

**/ \_ \**

**| (\_) |**

**\\_\_, |**

**/\_/**

**\_\_\_\_ \_ \_\_**

**| \_ \ \_ \_\_ \_\_\_ \_\_ \_ \_ \_\_ \_\_ \_ \_ \_\_ \_\_\_ \_\_ \_ \_\_\_(\_) /\_/ \_ \_\_**

**| |\_) | '\_\_/ \_ \ / \_` | '\_\_/ \_` | '\_ ` \_ \ / \_` |/ \_\_| |/ \_ \| '\_ \**

**| \_\_/| | | (\_) | (\_| | | | (\_| | | | | | | (\_| | (\_\_| | (\_) | | | |**

**|\_| |\_| \\_\_\_/ \\_\_, |\_| \\_\_,\_|\_| |\_| |\_|\\_\_,\_|\\_\_\_|\_|\\_\_\_/|\_| |\_|**

**|\_\_\_/**

**\_\_\_\_\_ \_**

**| \_\_\_\_|\_ \_\_ | |\_ \_\_\_ \_ \_\_ \_\_ \_**

**| \_| | '\_ \| \_\_/ \_ \ '\_\_/ \_` |**

**| |\_\_\_| | | | || \_\_/ | | (\_| |**

**|\_\_\_\_\_|\_| |\_|\\_\_\\_\_\_|\_| \\_\_,\_|**

**\_ \_ \_\_\_\_ \_**

**\_\_\_ \_\_\_ \_ \_\_ ( | ) \_\_ ) \_ \_\_ \_\_ \_ \_ \_\_ \_\_\_| |\_\_**

**/ \_\_/ \_ \| '\_ \ V V| \_ \| '\_\_/ \_` | '\_ \ / \_\_| '\_ \**

**| (\_| (\_) | | | | | |\_) | | | (\_| | | | | (\_\_| | | |**

**\\_\_\_\\_\_\_/|\_| |\_| |\_\_\_\_/|\_| \\_\_,\_|\_| |\_|\\_\_\_|\_| |\_|**

**\_ \_\_\_\_ \_ \_ \_**

**\_\_ \_ \_ \_\_ \_\_| | | \_\_ ) \_\_\_ \_ \_ \_ \_\_ \_\_| ( | )**

**/ \_` | '\_ \ / \_` | | \_ \ / \_ \| | | | '\_ \ / \_` |V V**

**| (\_| | | | | (\_| | | |\_) | (\_) | |\_| | | | | (\_| |**

**\\_\_,\_|\_| |\_|\\_\_,\_| |\_\_\_\_/ \\_\_\_/ \\_\_,\_|\_| |\_|\\_\_,\_|**

**Contenido**

**=========**

**1. Introducción.**

**2. Modelo Programación Lineal Entera.**

**3. Algoritmo de "Branch and Bound".**

**4. Interpretación Gráfica del B&B.**

**5. Un Ejemplo con Acotamiento Al Final.**

**6. Un Ejemplo sin Acotamiento.**

**7. Un Ejemplo con Acotamiento Temprano.**

**8. Un Ejemplo Binario.**

**9. Resumen.**

**10. Ejercicios.**

**1. Introducción.**

**================**

**El método de "branch and bound", conocido como ramificación y acotamiento se puede utilizar tanto para problemas mixtos como para problemas puros. Este método parte el área de las soluciones factibles en partes más pequeñas hasta obtener una solución óptima.**

**Como todos los métodos de ramificación, si el número de variables es demasiado grande, este método puede tardar un gran número de iteraciones en resolver el problema.**

**2. Modelo Programación Lineal Entera.**

**=====================================**

**Un modelo es de programación lineal entera si se puede escribir de la siguiente forma:**

**Maximizar: c1 x1 + c2 x2 + ... + cn xn = z**

**Con las**

**restricciones: a11 x1 + a12 x2 + ... + a1n xn <= b1**

**a21 x1 + a22 x2 + ... + a2n xn <= b2**

**. . .**

**. . .**

**. . .**

**am1 x1 + am2 x2 + ... + amn xn <= bm**

**con x1,x2,...,xn >= 0**

**y x1,x2,...,xp p<=n son valores enteros.**

**Algunas variantes de este modelo son:**

**() El modelo puede ser para maximizar o minimizar.**

**() Cada una de las restricciones puede relacionarse con bij mediante cualquiera de los símbolos = , <= , >=.**

**() Si todas las variables están restringidas a tomar valores enteros se dice que se trata de un problema de programación entera puro.**

**() Al contrario, si algunas variables deben tomar valores enteros y otras pueden tomar valores continuos se denomina un problema de programación entera mixto.**

**() Cuando las variables deben tomar valores de 0 o 1 únicamente se denomina un problema de programación entera 0-1 o binario.**

**3. Algoritmo de "Branch and Bound".**

**===================================**

**Este algoritmo se suele llamar por diferentes nombres, "bifurcación y acotamiento" o bien de "ramificación y acotamiento". Se ha optado por utilizar el nombre en inglés de "Branch and Bound".**

**Considere que la función objetivo es de maximización. La idea principal del algoritmo consiste en encontrar una solución inicial continua. Si la solución da valores enteros se detiene. Si la solución tiene valores continuos entonces se toma cualquier variable, por ejemplo la variable xj. Se toma i1 <= xj <= i2 con la condición que i1,i2 sean valores enteros consecutivos, entonces se crean dos nuevos problemas cada uno de ellos con las restricciones xj <= i1 y xj >= i2. Este procedimiento se denomina bifurcación y tiene el efecto de reducir el área de la región factible y obligar a xj a tomar valores enteros.**

**El proceso de ramificación continua hasta que se obtiene una primera solución entera. El valor de "z" de esta primera solución entera se vuelve una cota inferior para el problema y todos los demás problemas cuyas aproximaciones sean enteras o no y que den valores de "z" menores a la cota inferior.**

**Si la función objetivo es de minimización, el procedimiento es el mismo, excepto que se emplean cotas superiores. De esta manera la primera solución entera se convierte en una cota superior para el problema y se eliminan los programas lineales cuyo valor de "z" sea mayor que la cota superior actual. A continuación se describe el algoritmo.**

**Algoritmo**

**---------**

**(1) Resuelva el problema como continuo.**

**(2) Examine la solución óptima. Si la solución satisface las restricciones enteras termine, de lo contrario pase al siguiente paso.**

**(3) Divida el problema en dos partes, con [t] parte entera de t:**

**Problema 1: max z = cx**

**ax <= b**

**xk <= [t] <--**

**x >= 0**

**Problema 2: max z = cx**

**ax <= b**

**xk >= [t]+1 <--**

**x >= 0**

**(4) Resuelva el problema 1 y 2 separadamente.**

**(5) Si para alguno de los problemas se encuentra una solución entera o es infactible no se divide más. El proceso de bifurcación se continua hasta que se obtiene la primera solución entera. El valor de la función objetivo para esta primera solución entera se vuelve una cota inferior para el problema y todos los problemas cuyas aproximaciones, enteras o no, den valores de la función objetivo menores que la cota se descartan.**

**(6) Si quedan nodos por explorar vuelva al punto (3) y repita el proceso.**

**Consideraciones para los cálculos**

**---------------------------------**

**Cuando se realizan las ramificaciones se va construyendo un árbol de soluciones. Para escoger cuál de los nodos se debe resolver se pueden utilizar las técnicas de "profundidad primero" o de "anchura primero". En general los problemas de "branch and bound" constan de tres partes:**

**() Selección del nodo.**

**Para problemas de maximización se realizará la ramificación del programa que tenga el mayor valor de z. De manera recíproca para los problemas de minimización se realizará la ramificación del programa que tenga el menor valor de z.**

**() "Branching" o Ramificación.**

**Suelen utilizarse varias técnicas. En algunos casos se escoge la variable que esté más cercana a 0.50 en su parte decimal. En otros casos se suele escoger la variable con el mayor valor fraccional. En caso de empate se puede seleccionar la nueva variable de manera arbitraria. Para efectos de mantener un orden se escogerá la variable con mayor valor fraccionario.**

**() "Bounding" o Acotamiento.**

**Se termina el árbol cuando se llega a un nodo infactible, a un nodo factible lo que significa que tiene soluciones enteras. Cuando se detecta un nodo con soluciones enteras, su valor de "z" se convierte en un cota inferior para los demás nodos, la cual servirá como mecanismo de acotamiento.**

**4. Interpretación Gráfica del B&B.**

**==================================**

**Se tiene el siguiente problema.**

**Problema 1:**

**-----------**

**(0) max z = 3 x1 + 4 x2**

**(1) 1 x1 + 1 x2 <= 10**

**(2) -1 x1 + 1 x2 <= 7**

**(3) x1,x2 >= 0 y enteros**

**Al resolver con simplex se obtiene la siguiente solución.**

**x1 = 3/2 = 1 1/2 = 1.50**

**x2 = 17/2 = 8 1/2 = 8.50**

**z = 77/2 = 38 1/2 = 38.50**

**Ninguna de las soluciones es entera, se toma cualquier variable, en este caso se tomará x1 y se resuelven dos nuevos problemas al agregar a cada uno las restricciones x1 <= 1 y x1 >= 2.**

**Problema 1.1:**

**-------------**

**(0 ) max z = 3 x1 + 4 x2**

**(1 ) 1 x1 + 1 x2 <= 10**

**(2 ) -1 x1 + 1 x2 <= 7**

**(3b) x1 <= 1**

**Al resolver con simplex se obtiene una solución entera. Por lo tanto no se necesita realizar más ramificaciones.**

**x1 = 1**

**x2 = 8**

**z = 35**

**Problema 1.2:**

**-------------**

**(0 ) max z = 3 x1 + 4 x2**

**(1 ) 1 x1 + 1 x2 <= 10**

**(2 ) -1 x1 + 1 x2 <= 7**

**(3b) x1 >= 2**

**Al resolverlo se obtiene la siguiente solución. No hay necesidad de ramificar más los problemas y esta es la solución óptima del problema pues es mayor que la anterior.**

**x1 = 2**

**x2 = 8**

**z = 38**

**El siguiente esquema muestre al proceso de ramificación y acotamiento que se llevó a cabo:**

**P.1**

**-----------**

**x1 = 1.50**

**x2 = 8.50**

**z = 38.50**

**/ \**

**x1<=1 / \ x1>=2**

**/ \**

**P.1.1 P.1.2**

**------- -------**

**x1 = 1 x1 = 2**

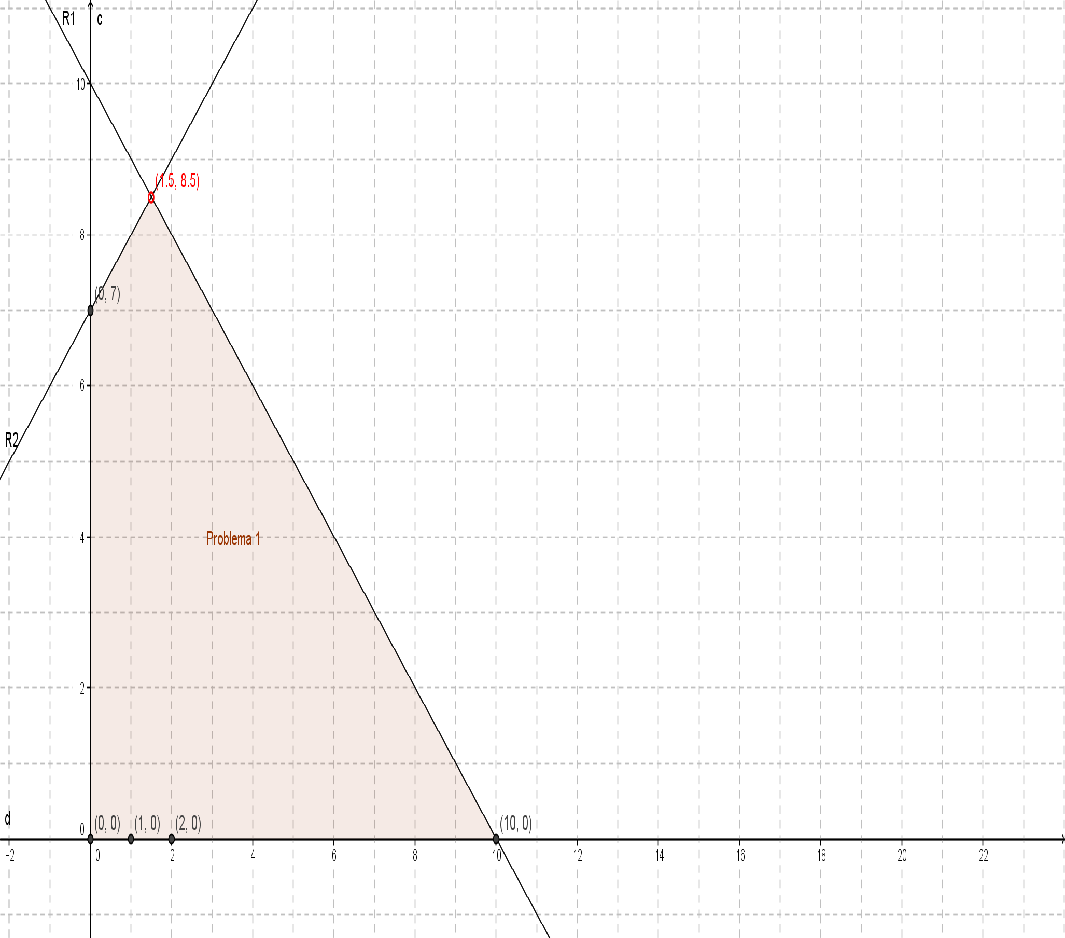
**x2 = 8 x2 = 8**

**z = 35 z = 38**

**De manera gráfica se puede observar como el algoritmo de "branch and bound" parte la región factible de manera que obliga a las variables a tomar valores enteros.**

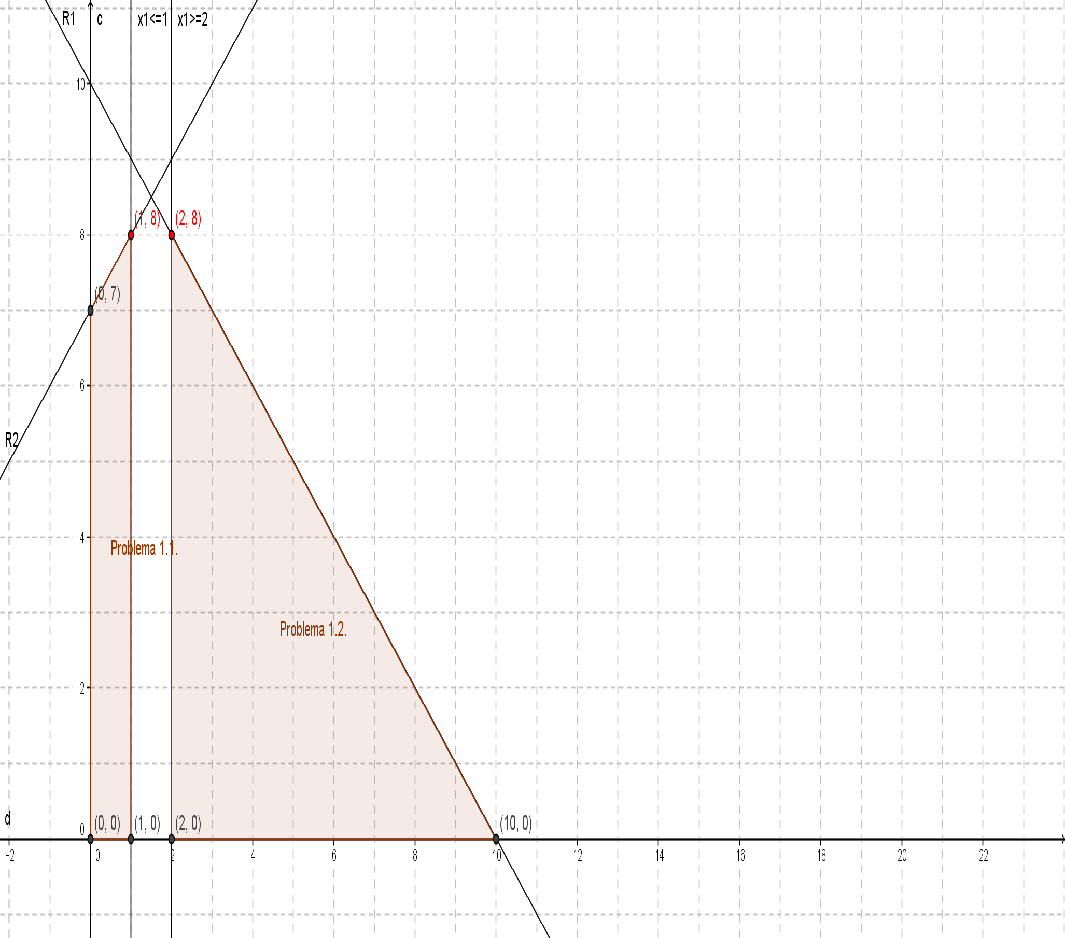
**### Gráfico.**

**### Problema 1.**

****

**### Gráfico.**

**### Problemas 1.1 y 1.2.**

****

**5. Un Ejemplo con Acotamiento Al Final.**

**=======================================**

**Se tiene el siguiente problema.**

**Problema 1**

**----------**

**(0) max z = 3 x1 + 4 x2**

**(1) 2 x1 + 1 x2 <= 6**

**(2) 2 x1 + 3 x2 <= 9**

**(3) x1,x2 >= 0 y enteros.**

**Se tiene la solución:**

**x1 = 9/4 = 2 1/4 = 2.25**

**x2 = 3/2 = 1 1/2 = 1.50**

**z = 51/4 = 12 3/4 = 12.75**

**Por lo tanto se resuelven los problemas 1.1 y 1.2.**

**Problema 1.1**

**------------**

**(0) max z = 3 x1 + 4 x2**

**(1) 2 x1 + 1 x2 <= 6**

**(2) 2 x1 + 3 x2 <= 9**

**(3) x2 <= 1**

**Cuya solución es:**

**x1 = 5/2 = 2 1/2 = 2.50**

**x2 = 1**

**z = 23/2 = 11 1/2 = 11.50**

**Problema 1.2**

**------------**

**(0) max z = 3 x1 + 4 x2**

**(1) 2 x1 + 1 x2 <= 6**

**(2) 2 x1 + 3 x2 <= 9**

**(3) x2 >= 2**

**Cuya solución es:**

**x1 = 3/2 = 1 1/2 = 1.50**

**x2 = 2**

**z = 25/2 = 12 1/2 = 12.50**

**De estos dos problemas se escoge para desarrollar el problema 1.2 por tener el mayor valor de "z". De esta forma se generan dos nuevos problemas los que se denominarán 1.2.1 y 1.2.2.**

**Problema 1.2.1**

**--------------**

**(0) max z = 3 x1 + 4 x2**

**(1) 2 x1 + 1 x2 <= 6**

**(2) 2 x1 + 3 x2 <= 9**

**(3) x2 >= 2**

**(4) x1 <= 1**

**Se obtiene la solución:**

**x1 = 1**

**x2 = 7/3 = 2 1/3 = 2.3333**

**z = 37/3 = 12 1/3 = 12.3333**

**Problema 1.2.2**

**--------------**

**(0) max z = 3 x1 + 4 x2**

**(1) 2 x1 + 1 x2 <= 6**

**(2) 2 x1 + 3 x2 <= 9**

**(3) x2 >= 2**

**(4) x1 >= 2**

**Este problema no tiene solución factible.**

**Se desarrollará ahora el problema 1.2.1 por tener el mayor valor de z. Lo que producirá los problemas 1.2.1.1 y 1.2.1.1.**

**Problema 1.2.1.1**

**----------------**

**(0) max z = 3 x1 + 4 x2**

**(1) 2 x1 + 1 x2 <= 6**

**(2) 2 x1 + 3 x2 <= 9**

**(3) x2 >= 2**

**(4) x1 <= 1**

**(5) x2 <= 2**

**Cuya solución es:**

**x1 = 1**

**x2 = 2**

**z = 11**

**Problema 1.2.1.2**

**----------------**

**(0) max z = 3 x1 + 4 x2**

**(1) 2 x1 + 1 x2 <= 6**

**(2) 2 x1 + 3 x2 <= 9**

**(3) x2 >= 2**

**(4) x1 <= 1**

**(5) x2 >= 3**

**Cuya solución es:**

**x1 = 0**

**x2 = 3**

**z = 12**

**Resulta innecesario desarrollar el problema 1.1 pues se ha encontrado una solución entera con un valor de z mayor. De esta forma el valor de z funciona como una cota inferior al problema.**

**A continuación se muestra el árbol de soluciones de este problema.**

**P.1**

**---------**

**x1= 2.25**

**x2= 1.50**

**z =12.75**

**/ \**

**x2<=1 / \ x2>=2**

**/ \**

**/ \**

**P.1.1 P.1.2**

**--------- ---------**

**x1= 2.50 x1= 1.50**

**x2= 1 x2= 2**

**z =11.50 z =12.50**

**\*Acotado\* / \**

**x1<=1 / \ x1>=2**

**/ \**

**/ \**

**P.1.2.1 P.1.2.2**

**--------- ---------**

**x1= 1 \*Infactible\***

**x2= 2.333**

**z =12.333**

**/ \**

**x2<=2 / \ x2>=3**

**/ \**

**/ \**

**P.1.2.1.1 P.1.2.1.2**

**--------- ---------**

**x1= 1 x1= 0**

**x2= 2 x2= 3**

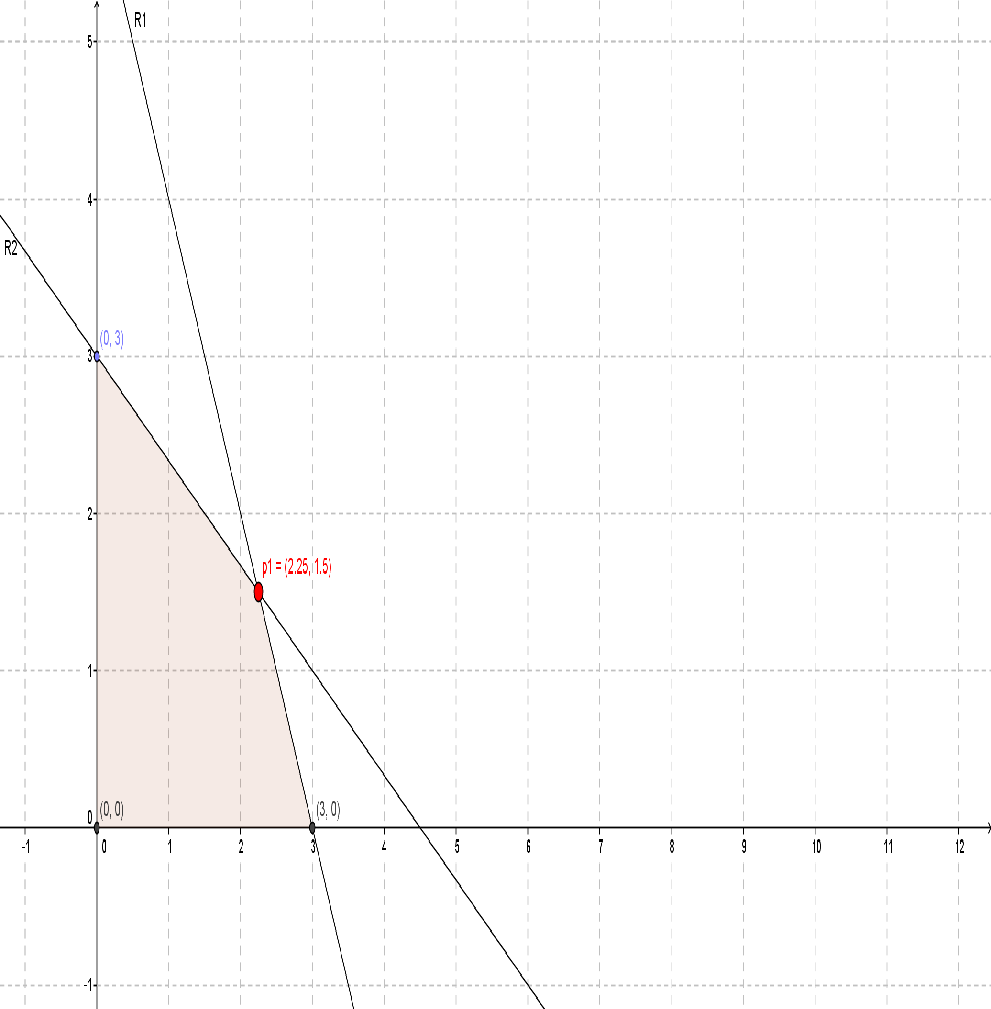
**z =11 z =12**

**La solución estará dada por x1=0, x2=3 y z=12.**

**Se puede explorar el proceso que se ha realizado de manera gráfica.**

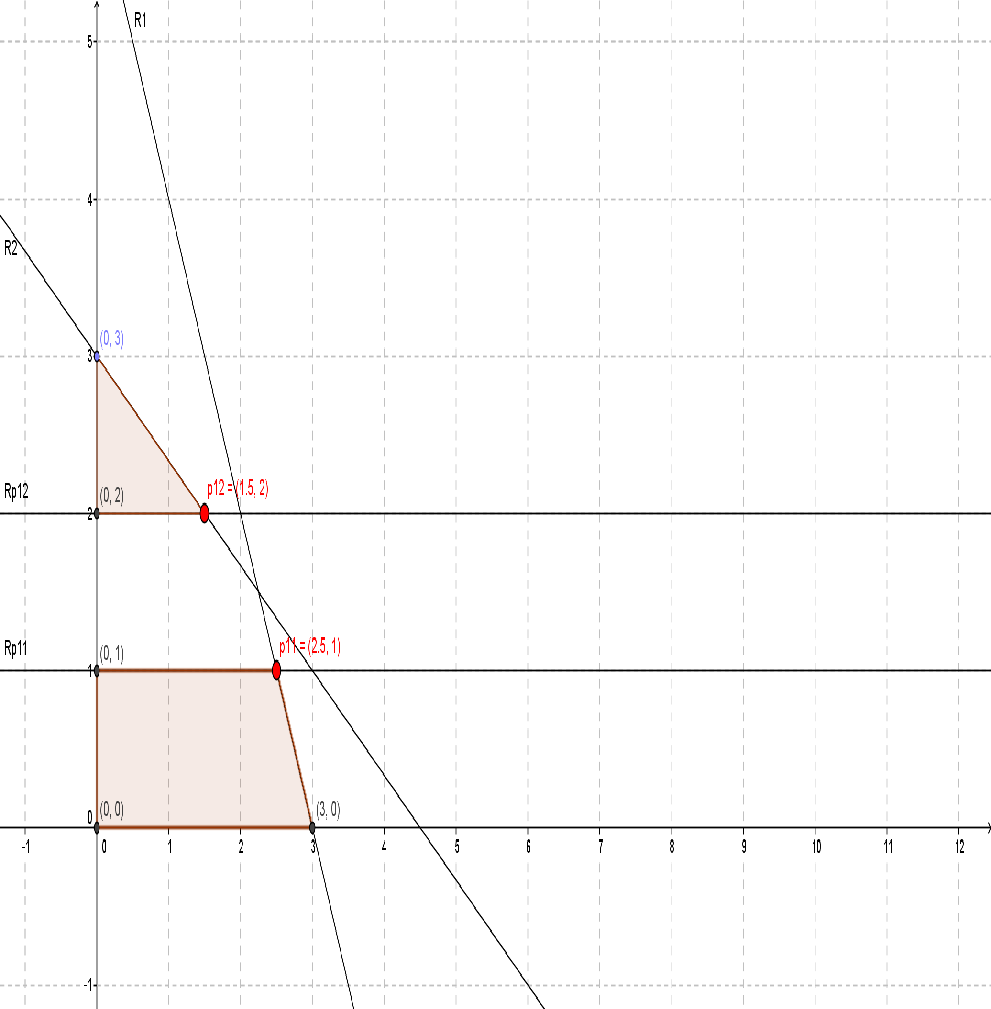
**### Gráfico.**

**### Problema 1**

****

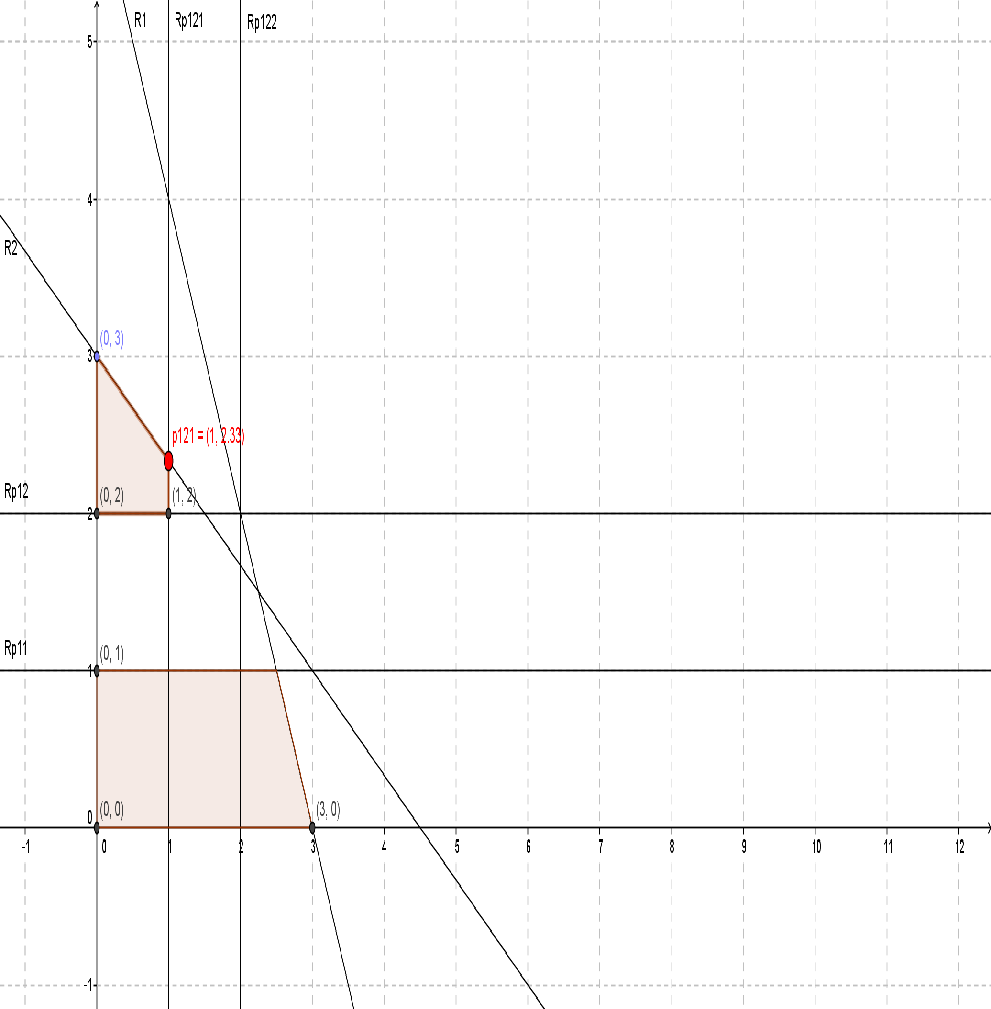
**### Gráfico.**

**### Problemas 1.1 y 1.2**

****

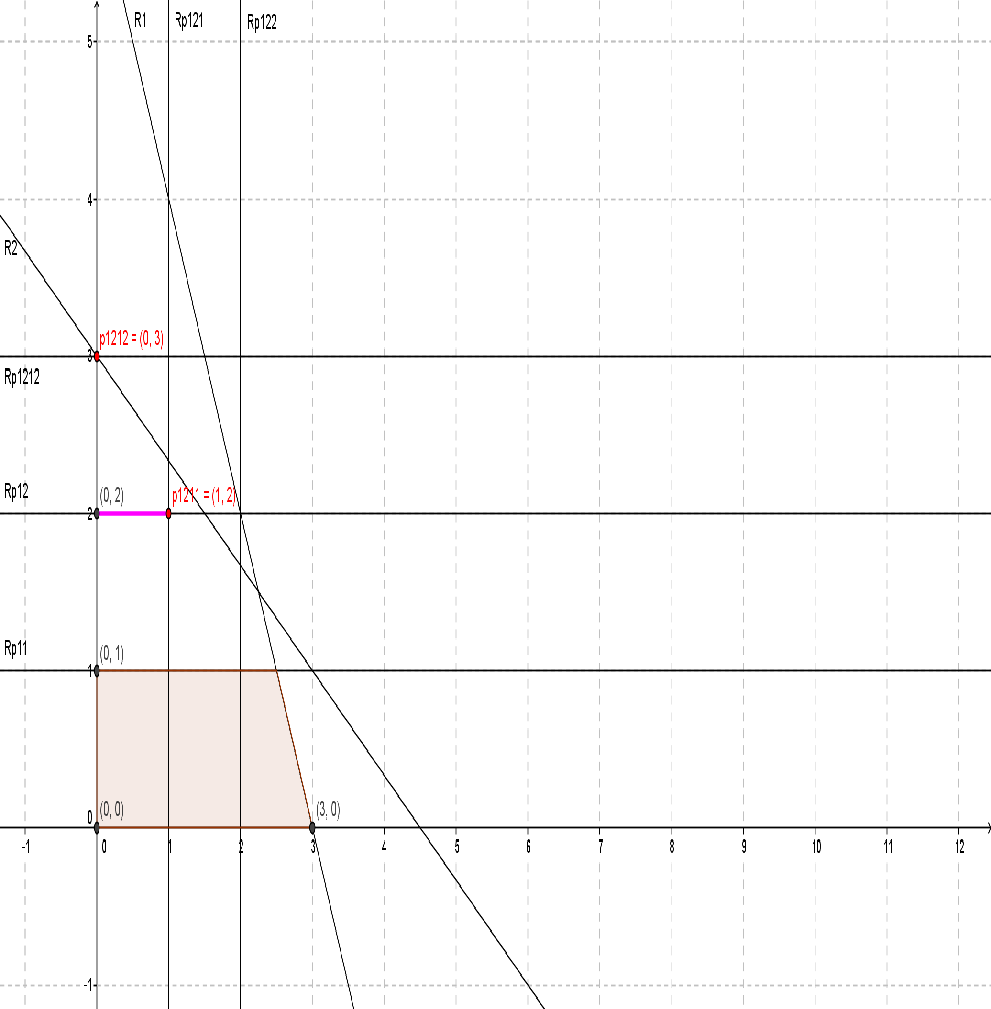
**### Gráfico.**

**### Problema 1.2.1 y 1.2.2**

****

**### Gráfico.**

**### Problemas 1.2.1.1. y 1.2.1.2**

****

**6. Un Ejemplo sin Acotamiento.**

**==============================**

**Se tiene el siguiente problema:**

**(0) max z = 1 x1 + 1.1 x2**

**(1) 3 x1 + 2 x2 <= 12**

**(2) 1 x2 <= 2**

**(3) x1,x2 >= 0 y enteros.**

**Se obtiene la solución:**

**x1 = 8/3 = 2 2/3 = 2.6666**

**x2 = 2**

**z = 73/15 = 4 13/15 = 4.8666**

**Por lo que se debe ramificar sobre x1.**

**Con x1 <= 2 y x1 >= 3.**

**Problema 1.1**

**------------**

**(0 ) max z = 1 x1 + 1.1 x2**

**(1 ) 3 x1 + 2 x2 <= 12**

**(2 ) x2 <= 2**

**(3b) x1 <= 2**

**Solución:**

**x1 = 2**

**x2 = 2**

**z = 21/5 = 4 1/5 = 4.20**

**Problema 1.2**

**------------**

**(0 ) max z = 1 x1 + 1.1 x2**

**(1 ) 3 x1 + 2 x2 <= 12**

**(2 ) x2 <= 2**

**(3b) x1 >= 3**

**Solución:**

**x1 = 3**

**x2 = 3/2 = 1 1/2 = 1.50**

**z = 93/20 = 4 13/20 = 4.65**

**Se debe dividir este problema en dos partes.**

**Con x2 <= 1 y x2 >= 2.**

**Problema 1.2.1**

**--------------**

**(0 ) max z = 1 x1 + 1.1 x2**

**(1 ) 3 x1 + 2 x2 <= 12**

**(2 ) x2 <= 2**

**(3b) x1 >= 3**

**(4b) x2 <= 1**

**Solución:**

**x1 = 10/3 = 3 1/3 = 3.3333**

**x2 = 1**

**z = 133/30 = 4 13/30 = 4.65**

**Se debe dividir este problema en dos partes.**

**Con x1 <= 3 y x1 >= 4.**

**Problema 1.2.2**

**--------------**

**(0 ) max z = 1 x1 + 1.1 x2**

**(1 ) 3 x1 + 2 x2 <= 12**

**(2 ) x2 <= 2**

**(3b) x1 >= 3**

**(4b) x2 >= 2**

**La solución de este problema es no factible, por lo que no se tomará en cuenta en la solución, ni es necesario ramificarlo más.**

**Problema 1.2.1.1**

**----------------**

**(0 ) max z = 1 x1 + 1.1 x2**

**(1 ) 3 x1 + 2 x2 <= 12**

**(2 ) x2 <= 2**

**(3b) x1 >= 3**

**(4b) x2 <= 1**

**(5b) x1 <= 3**

**Solución:**

**x1 = 3**

**x2 = 1**

**z = 41/10 = 4 1/10 = 4.10**

**Problema 1.2.1.2**

**----------------**

**(0 ) max z = 1 x1 + 1.1 x2**

**(1 ) 3 x1 + 2 x2 <= 12**

**(2 ) x2 <= 2**

**(3b) x1 >= 3**

**(4b) x2 <= 1**

**(5b) x1 >= 4**

**Con la solución**

**x1 = 4**

**x2 = 0**

**z = 4**

**La estructura del proceso de ramificación y acotamiento se muestra en el siguiente diagrama.**

**P.1**

**--------**

**x1=2.6666**

**x2=2**

**z =4.8666**

**/ \**

**x1<=2 / \ x1>=3**

**/ \**

**/ \**

**P.1.1 P.1.2**

**----- --------**

**x1=2 x1=3**

**x2=2 x2=1.50**

**z =4.20 z =4.65**

**/ \**

**x2<=1 / \ x2>=2**

**/ \**

**/ \**

**P.1.2.1 P.1.2.2**

**------- -------**

**x1=3.3333 Infactible**

**x2=1**

**z =4.65**

**/ \**

**x1<=3 / \ x1>=4**

**/ \**

**/ \**

**P.1.2.1.1 P.1.2.1.2**

**--------- ---------**

**x1=3 x1=4**

**x2=1 x2=0**

**z =4.10 z =4**

**Por lo tanto la solución estará dada por:**

**x1 = 2, x2 = 2 y z = 4.20**

**7. Un Ejemplo con Acotamiento Temprano.**

**=======================================**

**Considere el siguiente problema:**

**(0) max z = 7 x1 + 7 x2 + 6 x3 + 9 x4**

**(1) 4 x1 + 5 x2 + 3 x3 + 5 x4 <= 30000**

**(2) 2 x1 + 1.5 x2 + 3 x3 + 3 x4 <= 20000**

**(3) x1,x2,x3,x4 >= 0 y enteros.**

**Cuya solución es:**

**x3 = 5000/3 = 1666 2/3 = 1666.66**

**x4 = 5000**

**z = 55000**

**A continuación se muestran los dos problemas que se generan.**

**Problema 1.1.**

**-------------**

**(0 ) max z = 7 x1 + 7 x2 + 6 x3 + 9 x4**

**(1 ) 4 x1 + 5 x2 + 3 x3 + 5 x4 <= 30000**

**(2 ) 2 x1 + 1.5 x2 + 3 x3 + 3 x4 <= 20000**

**(3b) x3 <= 1666**

**Cuya solución es:**

**x3 = 1666**

**x4 = 25002/5 = 5000 2/5 = 5000.40**

**z = 274998/5 = 54999 3/5 = 54999.60**

**Problema 1.2**

**------------**

**(0 ) max z = 7 x1 + 7 x2 + 6 x3 + 9 x4**

**(1 ) 4 x1 + 5 x2 + 3 x3 + 5 x4 <= 30000**

**(2 ) 2 x1 + 1.5 x2 + 3 x3 + 3 x4 <= 20000**

**(3b) x3 >= 1667**

**Cuya solución es:**

**x1 = 1.00**

**x3 = 1667.00**

**x4 = 4999.00**

**z = 55000.00**

**Al obtenerse una solución entera, su valor de z se utiliza como cota inferior y se considerarán para ramificar únicamente aquellas soluciones con valores mayores a z. Observe el árbol de soluciones que se produce.**

**P.1**

**------------**

**x3= 1666.66**

**x4= 5000.00**

**z = 55000.00**

**/ \**

**x3<=1,666 / \ x3>=1,667**

**/ \**

**/ \**

**P.1.1 P.1.2**

**----------- -----------**

**x3= 1666.00 x1= 1.00**

**x4= 5000.40 x3= 1667.00**

**z =54999.60 x4= 4999.00**

**\*\* z =55000.00**

**No se ramifica. Solución óptima.**

**8. Un Problema Binario.**

**=======================**

**Considere el siguiente problema:**

**(0) max z = 10 x1 + 15 x2 + 50 x3**

**(1) 10 x1 + 15 x2 + 50 x3 <= 70**

**(2) x1 <= 1**

**(3) x2 <= 1**

**(4) x3 <= 1**

**(5) x1,x2,x3 >= 0 y enteras.**

**La solución encontrada es:**

**x1 = 1/2 = 0.50**

**x2 = 1**

**x3 = 1**

**z = 70**

**Como la solución no es entera se pasa a bifurcar el área con respecto a x1. Se utilizan la bifurcación:**

**x1 <= 0 que debido a la no negatividad se convierte en x1=0**

**x1 >= 1 que debido a la restricción 2 se convierte en x1=1**

**Problema 1.1**

**------------**

**(0 ) max z = 10 x1 + 15 x2 + 50 x3**

**(1 ) 10 x1 + 15 x2 + 50 x3 <= 70**

**(2 ) x1 <= 1**

**(3 ) x2 <= 1**

**(4 ) x3 <= 1**

**(5b) x1 = 0**

**Cuya solución es:**

**x1 = 0**

**x2 = 1**

**x3 = 1**

**z = 65**

**Problema 1.2**

**------------**

**(0 ) max z = 10 x1 + 15 x2 + 50 x3**

**(1 ) 10 x1 + 15 x2 + 50 x3 <= 70**

**(2 ) x1 <= 1**

**(3 ) x2 <= 1**

**(4 ) x3 <= 1**

**(5b) x1 = 1**

**Tiene una solución que no es entera y por lo tanto se debe volver a ramificar.**

**x1 = 1**

**x2 = 2/3 = 0.6666**

**x3 = 1**

**z = 70**

**A continuación se muestra el diagrama de ramificaciones. Que se construye al resolver los diferentes subproblemas de ramificación y acotamiento.**

**P.1**

**------**

**x1= 0.50**

**x2= 1**

**x3= 1**

**\_z = 70\_**

**/ \**

**x1=0 / \ x1=1**

**/ \**

**/ \**

**P.1.1 P.1.2**

**------ ------**

**x1= 0 x1= 1**

**x2= 1 x2= 0.6666**

**x3= 1 x3= 1**

**z = 65 z = 70**

**/ \**

**x2=0 / \ x2=1**

**/ \**

**/ \**

**P.1.2.1 P.1.2.2**

**------- -------**

**x1= 1 x1= 1**

**x2= 0 x2= 1**

**x3= 1 x3= 0.90**

**z = 60 z = 70.00**

**/ \**

**x3=0 / \ x3=1**

**/ \**

**/ \**

**P.1.2.2.1 P.1.2.2.2**

**--------- ---------**

**x1= 1 No**

**x2= 1 Factible**

**x3= 0**

**z = 25**

**Al analizar la ramificación anterior se obtiene que la solución óptima del problema está dada por:**

**x1= 0**

**x2= 1**

**x3= 1**

**z = 65**

**9. Resumen.**

**===========**

**Se ha analizado el algoritmo de "branch and bound", conocido como ramificación y acotamiento de acuerdo a la descripción de Land y Doig.**

**Además de mostrar su funcionamiento para problemas enteros generales se demostró como se debe aplicar para problemas donde las variables toman únicamente los valores de 0 y 1, conocidos como problemas binarios.**

**+++**

**10. Ejercicios.**

**===============**

**() Ejercicio 1.**

**Indique las diferencias fundamentales entre un problema de programación lineal entera y entero binario.**

**() Ejercicio 2.**

**Indique los pasos que lleva el algoritmo de "branch and bound".**

**Investigue sobre el algoritmo problemas binarios de Balas.**

**() Ejercicio 3.**

**Resuelva el siguiente problema de programación entera.**

**max z = 7 x1 + 9 x2**

**-1 x1 + 3 x2 <= 6**

**7 x1 + 1 x2 <= 35**

**x1,x2 >= 0 y enteras.**

**() Ejercicio 4.**

**Resuelva el siguiente problema de programación entera.**

**max z = 1 x1 + 2 x2**

**2 x2 <= 7**

**1 x1 + 1 x2 <= 7**

**2 x1 <= 11**

**x1,x2 >= 0 y enteras.**

**() Ejercicio 5.**

**Resuelva el siguiente problema de programación entera.**

**max z = 2 x1 + 6 x2**

**3 x1 + 1 x2 <= 5**

**4 x1 + 4 x2 <= 9**

**x1,x2 >= 0 y enteros**

**() Ejercicio 6. \***

**Resuelva el siguiente problema de programación entera.**

**max z = 15 x1 + 11 x2**

**1 x1 + 5 x2 <= 20**

**8 x1 + 3 x2 <= 27**

**x1,x2 >= 0 y enteras**

**() Ejercicio 7.**

**Resuelva el siguiente problema de programación entera.**

**max z = 10 x1 + 1 x2**

**2 x1 + 5 x2 <= 11**

**x1,x2 >= 0 y enteros.**

**() Ejercicio 8.**

**Resuelva el siguiente problema de programación entera.**

**max z = 3 x1 + 4 x2**

**2 x1 + x2 <= 6**

**2 x1 + 3 x2 <= 9**

**x1,x2 >= 0 y enteros**

**() Ejercicio 9.**

**Resuelva el siguiente problema de programación entera.**

**min z = x1 + x2**

**2 x1 + 2 x2 >= 5**

**12 x1 + 5 x2 <= 30**

**x1,x2 >= 0 y enteros.**

**() Ejercicio 10.**

**Resuelva el siguiente problema binario.**

**max z = 100 x1 + 60 x2 + 70 x3 + 15 x4 + 15 x5**

**52 x1 + 23 x2 + 35 x3 + 15 x4 + 7 x5 <= 60**

**x1 <= 1**

**x2 <= 1**

**x3 <= 1**

**x4 <= 1**

**x5 <= 1**

**x1,x2,x3,x4,x5 >=0 y enteras**

**() Ejercicio 11. \*\*\***

**Se tiene el siguiente grafo no dirigido. Se desea encontrar la mejor ruta desde el nodo A hasta el nodo T. Para ello se desea construir un modelo en programación lineal binaria que resuelva el problema propuesto.**

**Sugerencia: utilice las siguientes variables:**

**xij: vale 1 si se va del nodo "i" al nodo "j".**

**vale 0 si no se va del nodo "i" al nodo "j".**

**(B)...........(E)**

**. . 7 . . .**

**.´ . .´ . `.**

**2 .´ .2 .´ . `. 5**

**.´ . .´ . `.**

**.´ . .´ 4 . `.**

**.´ . .´ . `.**

**(A)...........(C) .1 (T)**

**. 5 . . . .**

**`. . `. . .´**

**`. . `.3 . .´**

**`. .1 `. . .´ 7**

**4 `. . `. . .´**

**`. . `. . .´**

**(D)...........(F)**

**4**

**R/**

**Se utilizarán las siguientes variables.**

**xij: vale 1 si se va del nodo "i" al nodo "j".**

**vale 0 si no se va del nodo "i" al nodo "j".**

**Como se trata de un grafo no dirigido se utilizarán dos variables para cada arco. Por ejemplo:**

**xcf: representa el arco del nodo "c" al nodo "f".**

**xfc: representa el arco del nodo "f" al nodo "c".**

**Las variables de salida del nodo A y las de llegada al nodo T, serán arcos dirigidos. Es decir tienen únicamente una dirección, pues se debe salir de A y no volver a este nodo. De igual manera una vez que se llega al nodo T no se debe salir de ahí.**

**Es decir existen las variables:**

**Salidas del nodo A: xab, xac, xad**

**Llegadas al nodo T: xet, xft**

**A continuación se muestra el programa lineal de este problema.**

**! Un problema de flujo de redes**

**! resuelto con PL binario**

**!**

**min z = 2 xab + 5 xac + 4 xad**

**+ 2 xbc + 2 xcb**

**+ 1 xcd + 1 xdc**

**+ 7 xbe + 7 xeb**

**+ 4 xce + 4 xec**

**+ 3 xcf + 3 xfc**

**+ 4 xdf + 4 xfd**

**+ 1 xef + 1 xfe**

**+ 5 xet + 7 xft**

**Con las restricciónes:**

**! Nodo a**

**xab + xac + xad = 1**

**! Nodo b**

**xab + xcb + xeb**

**- xbc - xbe = 0**

**! Nodo c**

**xac + xbc + xdc + xec + xfc**

**- xcb - xcd - xce - xcf = 0**

**! Nodo d**

**xad + xcd + xfd**

**- xdc - xdf = 0**

**! Nodo e**

**xbe + xce + xfe**

**- xeb - xec - xef - xet = 0**

**! Nodo f**

**xcf + xdf + xef**

**- xfc - xfd - xfe - xft = 0**

**! Nodo t**

**xet + xft = 1**

**Al resolver este problema se obtiene la siguiente ruta:**

**xab = 1**

**xbc = 1**

**xce = 1**

**xet = 1**

**z = 13**

**Que es equivalente a la ruta:**

**A -> B -> C -> E -> T.**

**Existe una respuesta alternativa dada por:**

**xab = 1**

**xbc = 1**

**xcf = 1**

**xfe = 1**

**xet = 1**

**z = 13**

**Que es equivalente a la ruta:**

**A -> B -> C -> F -> E -> T.**