Heurística para mejorar la convergencia del método de descomposición de Benders

Heuristic to improve convergence of the Benders decomposition method

Di Novella, Pedro
Dpto. Procesos y Sistemas, Universidad Simón Bolívar, Caracas, Venezuela
dinovell@usb.ve

Recibido: 02-02-2005 Revisado: 07-01-2008

Resumen

El objetivo fundamental de este trabajo es el desarrollo e implementación de una heurística para resolver problemas de Programación Entera Mixta, con tiempos de convergencia mejores que los métodos exactos. La heurística propuesta es aplicada en el método de descomposición de Benders, obteniéndose tiempos de convergencia hasta 49 veces más rápidos que el método de Benders sin la heurística. Las pruebas computacionales fueron realizadas usando una implementación del Método de Benders original y su versión modificada en el lenguaje C.

Palabras claves: Heurística, programación entera mixta, método de descomposición de Benders.

Abstract

The main objective of this paper is the development and implementation of a heuristic for solving Mixed Integer Programming Problems. The heuristic proposed is applied in the Benders Decomposition Method and provides better convergence times than exact methods. Results show convergence times up to 49 times faster than applying Benders' method without the heuristic. The computational tests were made using an implementation of the original Benders' Method and its modified version in C language.

Key words: Heuristic, mixed integer programming, Benders decomposition method.

1 Introducción

Existen dos razones primordiales para estudiar los problemas con variables enteras. En primer lugar, no se permiten, en muchas aplicaciones, los valores fraccionarios para las variables de decisión. Al redondear la solución obtenida de un problema, sin tomar en cuenta el requerimiento de ser entera, esta puede arrojar resultados muy deficientes. En consecuencia, se requieren métodos para obtener una solución óptima entera. Estos problemas aparecen en el modelaje de procesos físicos, procesos económicos, procesos de decisión de múltiples etapas, etc. La industria es una prolija fuente de problemas y situaciones que pueden ser modeladas como problemas enteros o enteros mixtos, como por ejemplo la aplicación de la tecnología de grupo en un sistema de manufactura flexible.

Una segunda razón para estudiar la programación en enteros es que ofrece una mayor flexibilidad en el modelado, a través del uso de variables binarias (0-1). Por ejemplo, al modelar problemas de presupuesto de capital, de diseño de sistemas de distribución (Geoffrion y Graves, 1974) o de ubicación de entes prestadores de servicio (Sweeney, Mairose y Martin, 1979), este tipo de variables permite que se pueden incorporar diversas consideraciones administrativas importantes mediante el uso de restricciones de elección múltiple, restricciones mutuamente excluyentes, restricciones condicionales y de correquisito, etc. Más aún, encontramos situaciones que sin ser de naturaleza discreta, con frecuencia el problema objeto de estudio es discretizado con el fin de simplificar su análisis y modelaje.

La descomposición de Benders (Benders, 1962) y (Lasdon, 1970) tiene una aplicación directa en procesos de decisión de múltiples etapas (planificación a largo plazo) (Alguacil y Conejo, 2000) y (Bai y Shahidehpour, 1996).

Este tipo de procesos aparece naturalmente en problemas como el de expandir el sistema generador de energía eléctrica de un país (Di Novella, Gorenstin, da Costa y Pacciornick, 1995), (Marin y Salmeron, 1998), (Oliveira, Costa y Binato, 1995) y (Romero y Monticelli, 1994). Al mismo tiempo, se puede aplicar a la planificación de mantenimiento de unidades de generación de energía eléctrica (Marwali y Shahidehpour, 1998, 1999) y (Silva, Morozowski, Fonseca, Oliveira, Melo y Mello, 1995), estimación de los costos de producción de energía eléctrica (Hobbs y Ji, 1995), planificación del nivel del embalse (Escudero, de la Fuente, García y Prieto, 1996) y del inventario de combustible para la generación de energía eléctrica bajo incertidumbre (Supatgiat, Takriti y Wu, 1998), etc.

2 Descomposición de Benders.

La descomposición de Benders se utiliza para resolver problemas de Programación Lineal Entera Mixta (PLEM). Este método se basa en la descomposición de un PLEM en sus partes entera y continua, cada una de las cuales puede ser resuelta con algoritmos apropiados. Sin pérdida de generalidad, podemos formular un PLEM como Min $z(x,y) = c\,x + d\,y$ sujeto a:

$$\begin{aligned} &A\,x + D\,y \geq b \\ &x \geq 0, y \geq 0 \\ &x \in Z^n \end{aligned} \tag{1}$$

donde $c \in R^n$, $d \in R^q$, A y D son matrices $p \times n$ y $p \times q$ respectivamente y $b \in R^p$.

El algoritmo de descomposición de Benders consiste en los siguientes pasos:

1. Inicializar: el número de iteraciones $K = 0$ y $\bar{z} = +\infty$, el limite superior de la función objetivo del problema original.
2. Resolver el problema maestro $z = \text{Min } c\,x + \alpha$ sujeto a:

$$\begin{aligned} &\alpha \geq \pi k\,(b - A\,x) && k = 1, ..., K_1 \\ &\mu k\,(b - Ax) \leq 0 && k = K_1+1, ..., K \\ &x \geq 0,\ x \in Zn,\ \alpha \in R \end{aligned} \tag{2}$$

3. Sea $\{x^*_{k+1}, \alpha^*_{k+1}\}$ la solución óptima de las Ecs. (2). Calcular el límite inferior para el valor óptimo de la función objetivo del problema original $\underline{z} = c\,x^*_{k+1} + \alpha^*_{k+1}$
4. Hacer $K = K + 1$ y resolver el problema esclavo $w(x^*_k) = \text{Min } dy$ sujeto a

$$\begin{aligned} &D\,y \geq b - A\,x^*_k \\ &y \geq 0 \end{aligned} \tag{3}$$

Si el problema tiene solución óptima finita ir al paso 6, caso contrario ir al paso 5.

5. Sea μ^k la solución extrema homogénea de (2.3). Agregar al problema (2.2) una nueva restricción $\mu k\,(b - Ax) \leq 0$. Ir al paso 2.
6. Sea y^*_k la solución de (2.3). Calcular el límite superior para el valor óptimo de la función objetivo del problema original
7. $\bar{z} = \min\{\bar{z}, c\,x^*_k + d\,y^*_k\}$.
8. Si $\bar{z} - \underline{z}$ es menor que una tolerancia dada, la solución asociada a $\bar{z}$ es la óptima. Fin. En caso contrario, ir al paso 8.
9. Agregar al problema (2.2) una nueva restricción $\alpha \geq \pi^K (b - A\,x)$ donde π^K es el vector de las variables duales del problema (2.3). Ir al paso 2.

3 Heurística para la descomposición de Benders.

Un gran problema que aparece al aplicarse el algoritmo de descomposición de Benders para resolver el problema (2.1), es que el problema maestro debe ser resuelto hasta hallarse una solución óptima. En la resolución de este problema se utiliza el algoritmo de Branch and Bound. Dependiendo del problema a ser resuelto, se puede necesitar un enorme período de tiempo para resolver el problema maestro, lo cual hace que no sea posible aplicar la descomposición de Benders para resolver el problema.

Geoffrion y Graves (1974), mostraron que no era necesario el cálculo de la solución óptima a cada iteración. Basado en este resultado fue desarrollada una heurística, la cual dispensa el cálculo de la solución óptima a cada iteración. La heurística propuesta en este trabajo consiste en resolver el problema maestro hasta que el algoritmo Branch and Bound obtenga la primera solución factible.

La sucesión de puntos $\underline{z}$ generada por el método de Benders en el paso 3 es monótona creciente y cada punto es una cota inferior del valor óptimo de la función objetivo del problema (2.1). Al aplicarse la heurística planteada en este trabajo, la sucesión generada deja de ser monótona creciente, además no se puede asegurar que sus puntos sean una cota inferior del problema original. Entonces, para poder garantizar la convergencia del algoritmo, debemos agregar al problema maestro, partir de la segunda iteración, una restricción adicional que es basada en la condición de convergencia del algoritmo de Benders. Esta condición, dada en el paso 7, puede ser expresada como $\bar{z}\ \underline{z} < \varepsilon$, donde ε la tolerancia dada para la convergencia. En cuanto el algoritmo no converja, la desigualdad

$$\underline{z} \leq \bar{z} - \varepsilon \tag{4}$$

es válida para el problema maestro. Por la definición de $\underline{z}$ en el paso 3 del algoritmo, tenemos que es igual al valor de la función objetivo del problema maestro. Por lo tanto podemos rescribir la restricción (3.1) como:

$$cx + \alpha \leq \bar{z} - \varepsilon \quad (5)$$

La desigualdad (5) deja de ser válida cuando se cumple la condición de convergencia del algoritmo de descomposición de Benders, por lo tanto la nueva condición de convergencia pasa a ser que el problema maestro con la restricción (5) sea infactible. La desigualdad (5) es de muy fácil implantación y en cada iteración sólo se debe modificar su lado derecho, lo cual es un trabajo insignificante desde el punto de vista computacional.

4 Resultados computacionales.

A continuación se presentan la descripción de los problemas de prueba utilizados y los resultados obtenidos de la aplicación del algoritmo de Benders original y de la heurística propuesta. Las pruebas fueron realizadas utilizando un microcomputador con procesador Pentium III 866 MHz, con 128 Mb de memoria RAM.

En la Tabla 1 se presentan algunas estadísticas de los problemas de prueba. Para cada problema se presenta el número de restricciones, el número de variables continuas, el número de variables enteras y el número de elementos diferentes de cero de la matriz de los coeficientes de las restricciones.

Tabla 1. Problemas de prueba

Problema	Número de restricciones	Número de variables continuas	Número de variables enteras	Elementos no nulos de la matriz de coeficientes
1	16	649	144	12343
2	64	649	462	64453
3	22	649	194	17791
4	88	649	628	97008
5	109	2491	836	285697
6	436	3141	2360	1159052
7	109	2210	1117	285697
8	436	2347	3154	1159052

En la Tabla 2 se presentan los resultados obtenidos al aplicarse el método de Benders sin heurística. Para cada problema se presenta el número de iteraciones realizadas, los límites inferior y superior obtenidos del valor óptimo de la función objetivo del problema, la diferencia entre ambos límites y el tiempo de procesamiento en segundos.

Se puede observar que para los problemas 2 y 4 no se pudo obtener la solución del problema maestro de la quinta iteración, a pesar de haberse colocado un límite de 200000 nodos en el algoritmo de Branch and Bound. Para el problema 7 no se pudo obtener la solución del problema maestro de la decimoctava iteración, mientras que para el problema 8 no se pudo obtener la solución en la séptima iteración, a pesar de haberse colocado un límite de 100000 nodos en el algoritmo de Branch and Bound para ambos problemas. Debido a la no convergencia del método en estos cuatro problemas, se observa que la diferencia entre los límites inferior y superior es muy grande.

Tabla 2. Resultados obtenidos sin utilizar la heurística.

Sin heurística					
Problema	Número de iteraciones	Limite inferior	Limite superior	Diferencia entre límites	Tiempo de procesamiento (s)
1	48	815,54	815,54	0	53,06
2	4	1,11	614,55	613,44	22499,67
3	47	815,21	815,21	0	80,83
4	4	1,11	614,38	613,27	36158,57
5	384	70007,95	70007,95	0	541,39
6	110	282,63	297,38	14,75	101193,89
7	17	29755,91	31162,52	1406,61	185773,39
8	6	0,00	26830,86	26830,86	23998,45

Tabla 3. Resultados obtenidos utilizando la heurística.

Con heurística					
Problema	Número de iteraciones	Limite inferior	Limite superior	Diferencia entre límites	Tiempo de procesamiento (s)
1	53	809,38	809,38	0	18,16
2	22	4,27	4,60	0,33	5526,88
3	52	809,04	809,04	0	48,05
4	19	8,24	10,51	2,27	7127,82
5	346	70000,50	70000,51	0,01	729,96
6	125	364,24	367,06	2,82	6747,62
7	76	30871,18	30888,71	17,53	16837,59
8	100	345,74	347,79	2,05	51102,14

La Tabla 3 presenta los resultados obtenidos al aplicarse el método de Benders con la heurística propuesta.

Se puede observar que para todos los problemas, la diferencia entre los límites inferior y superior son menores que los correspondientes a los obtenidos sin utilizar la heurística. Salvo para el problema 6, los límites obtenidos son mejores a los conseguidos sin la aplicación de la heurística. Para el problema 4 no se pudo obtener la solución del problema maestro de la vigésima iteración con un límite de 50000 nodos en el algoritmo de Branch and Bound. En el problema 6 el programa paró al alcanzarse el número máximo de iteraciones permitidas, el cual fue para este problema de 125. Para el problema 7 no se pudo obtener la solución del problema maestro de la iteración 77, para el cual fue colocado un límite de 60000 nodos en el algoritmo de Branch and Bound. En el problema 8 el programa paró al alcanzarse el número máximo de 100 iteraciones permitidas para este problema.

La Tabla 4 muestra los tiempos promedio de procesamiento por iteración, en segundos, para las dos implementaciones del algoritmo, así como la relación (tiempo promedio de procesamiento sin heurística) / (tiempo promedio de procesamiento con heurística).

Para comparar los tiempos de procesamiento se utilizaron los tiempos promedio de procesamiento por iteración, debido a la diferencia en el número de iteraciones. Se puede notar que el método de Benders con la heurística presenta un mejor desempeño en términos de tiempo de procesamiento, en media 1480,45% menor que los tiempos obtenidos por el método de Benders sin la heurística, siendo la mejor relación obtenida la del problema 7, con 4832,51%.

Tabla 4. Relación de los tiempos promedio de procesamiento por iteración.

Problema	Tiempo promedio de procesamiento por iteración (s)		Sin heurística / Con heurística
	Sin heurística	Con heurística	
1	1,11	0,34	3,23
2	5624,92	251,22	22,39
3	1,72	0,92	1,86
4	9039,64	375,15	24,10
5	1,41	2,11	0,67
6	919,94	53,98	17,04
7	10927,85	221,55	49,33
8	3999,74	511,02	7,83

5 Conclusiones.

En este trabajo fue presentada una heurística, que al ser empleada en el método de descomposición Benders permite resolver problemas de Programación Lineal Entera Mixta, donde el número de variables enteras varía de 144 a 3154.

Como se pudo observar en los resultados computacionales mostrados en la sección anterior, en cinco de los ocho problemas de prueba, se presentaron problemas de convergencia cuando se utilizó del algoritmo de Benders original. El método de Benders presentó también un elevado tiempo de convergencia, debido a que en cada iteración se debe resolver el problema maestro hasta obtenerse la solución óptima. Estos problemas presentados determinan que no sea recomendable utilizar el método de Benders en su forma original, en la resolución de problemas con un elevado número de variables enteras.

Al aplicarse el método de Benders con la heurística propuesta, se garantiza su correcta convergencia debido al resultado expuesto en Geoffrion y Graves (1974). Esta combinación permite obtener la precisión del método de Benders con una velocidad de convergencia hasta 49 veces más rápidos que el método original.

6 Agradecimiento.

El autor agradece al Decanato de Investigación y Desarrollo de la Universidad Simón Bolívar el apoyo otorgado a la realización de la presente investigación a través del proyecto S1-PN: Desarrollo de Heurísticas para Programación Entera.

Referencias

Alguacil N. y Conejo AJ, 2000, Multiperiod optimal power flow using Benders decomposition, IEEE Transactions on Power Systems, Vol. 15, No 1.

Bai X, y Shahidehpour SM, 1996, Hydro-thermal scheduling by tabu search and decomposition method, IEEE Transactions on Power Systems, Vol. 11, No 2, pp.968-974.

Benders JF, 1962, Partitioning procedures for solving mixed-variable programming problems, Numerische Mathematik, Vol, 4, pp. 238-252.

Di Novella P, Gorenstin B, da Costa J y Pacciornick N, 1995, Planejamento sob incertezas, XIII Seminário Nacional de Produção e Transmissão de Energia Elétrica, Florianópolis, Brasil.

Escudero LF, de la Fuente JL, García C y Prieto FJ, 1996, Hydropower generation management under uncertainty via scenario analysis and parallel computation, IEEE Transactions on Power Systems, Vol. 11, No 2, pp. 683-689,

Geoffrion AM y Graves GW, 1974, Multicommodity distribution system design by benders decomposition, Management Science, Vol. 20, No. 5.

Hobbs BF y Ji Y, 1995, Bounding approach to multiarea probabilistic production costing, IEEE Transactions on Power Systems, Vol. 10, No. 2, pp. 853-859.

Lasdon LS, 1970, Optimization theory for large systems, Macmillan Series in Operations Research.

Marín A y Salmeron J, 1998, Electric capacity expansion under uncertain demand: Decomposition approaches, IEEE Transactions on Power Systems, Vol. 13, No. 2, pp. 333-339.

Marwali MKC y Shahidehpour SM, 1998, Integrated generation and transmission maintenance scheduling with network constraints, IEEE Transactions on Power Systems, Vol. 13, No. 3, pp. 1063-1068.

Marwali MKC y Shahidehpour SM, 1999, Long-term transmission and generation maintenance scheduling with network, fuel and emission constraints, IEEE Transactions on Power Systems, Vol. 14, No. 3, pp. 1160-1165.

Oliveira GC, Costa APC y Binato S, 1995, Large scale transmission network planning using optimization and heuristic techniques, IEEE Transactions on Power Systems, Vol. 10, n 4, pp. 1828-1834.

Romero R y Monticelli A, 1994, Zero-one implicit enumeration method for optimizing investments in transmission expansion planning, IEEE Transactions on Power Systems, Vol.9, No.3, pp.1385-1391.

Silva EL, Morozowski M, Fonseca LGS, Oliveira GC, Melo ACG y Mello JCO, 1995, Transmission constrained maintenance scheduling of generating units: a stochastic programming approach, IEEE Transactions on Power Systems, Vol.10, No.2, pp.95-701.

Supatgiat C, Takriti S y Wu L, 1998, Coordinating fuel inventory and electric power generation under uncertainty, Proceedings of the American Power Conference v 1, Illinois Inst of Technology, Chicago, IL, USA, pp.417-421.

Sweeney DJ, Mairose L y Martin R, 1979, Strategic Planning in Bank Location, AIDS Proceedings.