

LINEAMIENTOS GENERALES DE SOLUCIÓN
PARA PROGRAMACIÓN LINEAL CON
COEFICIENTES BORROSOS

GENERAL GUIDELINES SOLUTION FOR LINEAR
PROGRAMMING WITH FUZZY COEFFICIENTS

SERGIO G. DE-LOS-COBOS-SILVA*

ANTONIO TERCEÑO-GÓMEZ†

MIGUEL A. GUTIÉRREZ-ANDRADE‡

*Received: 14/Nov/2011; Revised: 20/May/2013;
Accepted: 27/May/2013*

*Universidad Autónoma Metropolitana-Iztapalapa, Departamento de Ingeniería Eléctrica, Av. San Rafael Atlixco 186, Col. Vicentina, Del. Iztapalapa, México D.F., C.P. 09340, México. E-Mail: cobos@xanum.uam.mx

†Universitat Rovira i Virgili, Departamento de Gestión de Empresas y Economía, Avinguda de la Universitat 1, 43204, Reus, España. E-MAil: atg@fcee.urv.es

‡Misma dirección que/Same address as: S. de-los-Cobos-Silva. E-Mail: gamma@xanum.uam.mx

Resumen

En este trabajo se introduce tanto a la Programación Posibilística como a la Programación Borrosa como paradigmas que permiten resolver problemas de optimización cuando los coeficientes del modelo de programación lineal o las restricciones del mismo se presentan como números borrosos, en lugar de números exactos (crisp, en inglés). Se presentan algunos ejemplos basados en [1].

Palabras clave: Programación posibilística, programación borrosa, optimización.

Abstract

This work introduce to the Possibilistic Programming and the Fuzzy Programming as paradigms that allow to resolve problems of linear programming when the coefficients of the model or the restrictions on the same are presented as fuzzy numbers, rather than exact numbers (crisp). This work presents some examples based on [1].

Keywords: Possibilistic programming, fuzzy programming, optimization.

Mathematics Subject Classification: 65J05, 90C05, 90C70, 65K05, 65K99.

1 Introducción

Se presentan dos instrumentos de programación matemática que utilizan instrumentos de la teoría de los subconjuntos borrosos. Como se indica en [1]:

1. Programación Borrosa. En este caso, y al contrario de lo que ocurre con un programa matemático convencional, el decisor no pretende “optimizar” de forma estricta un objetivo, sino que únicamente pretende obtener un valor para este que sea “satisfactorio”. El decisor tampoco expresa las limitaciones a las que se enfrenta en su intento de optimizar el objetivo de forma rígida (es decir, mediante “ $\leq b$ ” ó “ $\geq b$ ” de forma estricta), sino que vienen dadas de forma flexible, de forma que son expresadas como “ \lesssim ” ó “ \gtrsim ”, es decir, las restricciones no vienen dadas por “estrictamente menor/mayor que b ” sino como “razonablemente”, “esencialmente”, “más o menos” menor/mayor que b .

2. Programación Posibilística. Se trata de programación convencional donde los coeficientes del programa (beneficios unitarios, costes unitarios, etc.) son conocidos de forma imprecisa, y, su valor es representado, por números borrosos. Por ejemplo, no afirmaremos que los “beneficios unitarios del producto 1 son de 3 u.m.”, sino “aproximadamente 3 u.m.”. Ello puede representarse mediante un número borroso triangular, es decir, “más o menos 3”.

Al respecto, se pueden encontrar varios trabajos que presentan las bases y aplicaciones, pueden consultarse entre otros: [1], [2], [3], [4], [5], [6], [7], [8], [9] y [10].

2 Aspectos generales

En programación matemática mono-objetivo se desea encontrar una solución $X = (x_1, x_2, \dots, x_n)$ que optimice cierta función $z = f(X)$ sujeta a m restricciones $g_i(x_1, x_2, \dots, x_n) \leq bi, i = 1, 2, \dots, m$.

En programación lineal, tanto la función objetivo como las restricciones son de tipo lineal y se puede representar el modelo en general de la forma:

$$\text{Maximizar } Z = c_1x_1 + c_2x_2 + \dots + c_nx_n$$

sujeto a:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &\leq b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &\leq b_2 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n &\leq b_m. \end{aligned}$$

3 Programación borrosa

Con el fin de proporcionar una idea más clara sobre la utilización de la programación borrosa, llamada también programación flexible, nos basaremos en [1] donde se presenta el siguiente ejemplo:

“Una determinada empresa se dedica a la producción de dos bienes A y B, cuyo margen de beneficio por unidad producida es de 3 u.m. y 10 u.m. respectivamente. Asimismo, se sabe que la producción de 1 unidad de bien A consume 2 unidades de materia prima y la producción de 1 unidad de bien B consume

4 unidades de materia prima. La disponibilidad de materia prima es de 2000 unidades. Asimismo, para la producción de 1 unidad de bien A es necesario el empleo de 2 trabajadores, y para la producción de una unidad de bien B, es necesario el empleo de uno. Por un pacto con los sindicatos, el número de trabajadores contratados debe ser mayor de 800.”

La programación borrosa permitirá formular modelos en los siguientes casos:

1. Cuando la intención del decisor no es tanto la de optimizar, sino la de obtener cierta satisfacción sobre el cumplimiento de objetivos y restricciones.
2. Cuando existe un conocimiento “impreciso” de los recursos o de los requerimientos. Utilizando la programación borrosa, la disponibilidad de recursos del ejemplo anterior se expresaría, por ejemplo, como “no deben consumirse más de 2000 unidades de materia prima, pero se aceptaría, como máximo, un exceso de consumo de 200 unidades”. Por otra parte, el requerimiento de contratación podría expresarse como “seguro que no tendremos problemas con los sindicatos si se contratan 800 trabajadores o más, no obstante, seguramente acabarían aceptando que el número de trabajadores contratados que fuera inferior a 800 en, como máximo, 80 trabajadores.

Un modelo de programación borrosa puede formularse como:

Encontrar (x_1, x_2, \dots, x_n) tal que:

- Al menos se alcanza la mínima aspiración buscada sobre los objetivos.
- Al menos se alcanza la mínima aspiración buscada sobre el cumplimiento de las restricciones.

El concepto clave en programación borrosa es la **medición de la satisfacción** sobre el cumplimiento de las restricciones o los objetivos, la que se realiza mediante funciones de pertenencia. Se debe tener en consideración que:

- El valor que proporciona máxima satisfacción no puede ser mejor al que se obtiene cuando se permite máxima flexibilidad.

- El peor valor admitido no puede ser peor al que se obtiene cuando no se permite ninguna flexibilidad.

Para observar lo anterior, pondremos los modelos y sus soluciones, sin permitir ninguna flexibilidad y permitiendo máxima flexibilidad del problema planteado.

- Sin permitir flexibilidad:

$$\text{Maximizar } z = 3A + 10B$$

sujeto a:

$$2A + 4B \leq 2000$$

$$2A + B \geq 800.$$

- Permitiendo máxima flexibilidad:

$$\text{Maximizar } z = 3A + 10B$$

sujeto a:

$$2A + 4B \leq 2200$$

$$2A + B \geq 720,$$

con solución (entera) : $z = 5272, A = 114, B = 493$.

Ahora bien, como la función objetivo se desea maximizar, el valor deseado será de 4600 o superior al considerar las restricciones flexibles. Ahora se desea encontrar los valores de la máxima satisfacción α considerando una gradación lineal tanto del objetivo como de las restricciones, esto lo realizaremos considerando las funciones de pertenencia μ , como se verá a continuación.

3.1 Funciones de pertenencia

En el caso de que se tenga una flexibilidad que va de un valor b_i a un valor $b_i + d_i$, la función de pertenencia μ en forma de gradación lineal será:

$$\mu_i(x_1, \dots, x_n) = \begin{cases} 1 & \text{si } g_i(x_1, \dots, x_n) \leq b_i \\ \frac{b_i + d_i - g_i(x_1, \dots, x_n)}{d_i} & \text{si } b_i \leq g_i(x_1, \dots, x_n) \leq b_i + d_i \\ 0 & \text{si } g_i(x_1, \dots, x_n) \geq b_i + d_i. \end{cases}$$

Ver la Figura 1, que ilustra esta situación.

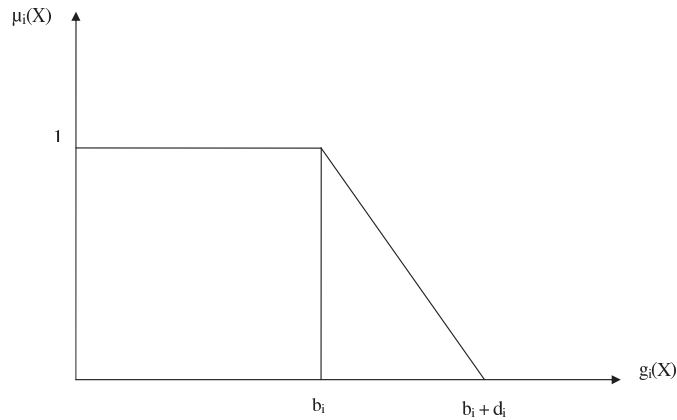


Figura 1: Gráfica de la función pertenencia cuando se tiene una flexibilidad para un aumento de recursos.

En el caso de que se tenga una flexibilidad que va de un valor b_i a un valor $b_i - d_i$, la función de pertenencia μ en forma de gradación lineal será (ver Figura 2):

$$\mu_i(x_1, \dots, x_n) = \begin{cases} 1 & \text{si } g_i(x_1, \dots, x_n) \geq b_i \\ \frac{g_i(x_1, \dots, x_n) - (b_i - d_i)}{d_i} & \text{si } b_i \geq g_i(x_1, \dots, x_n) \geq b_i - d_i \\ 0 & \text{si } g_i(x_1, \dots, x_n) \leq b_i - d_i. \end{cases}$$

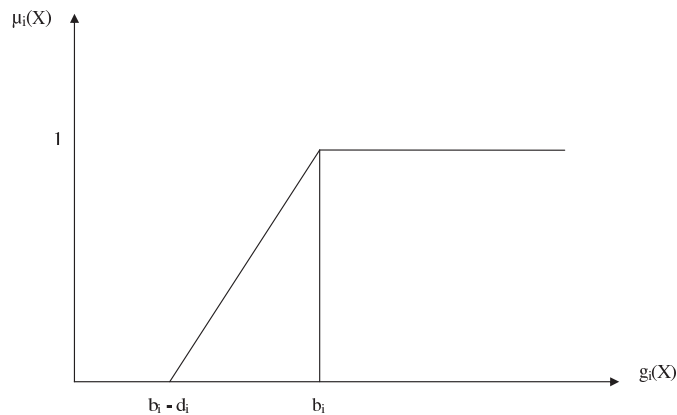


Figura 2: Gráfica de la función pertenencia cuando se tiene una flexibilidad para una disminución de recursos.

De manera análoga a las restricciones, se encuentra la función de pertenencia para el valor de la función objetivo.

Por lo que, formulando el programa del ejemplo donde se desea encontrar el máximo nivel α , para el cual, la satisfacción de cada uno de los objetivos y restricciones sea igual o mayor, el modelo a resolver queda como:

Maximizar α

sujeto a:

$$\begin{aligned} \frac{3A + 10B - 4600}{672} &\geq \alpha \\ 3A + 10B &\leq 5272 \\ 3A + 10B &\geq 4600 \\ \frac{2200 - (2A - 4B)}{200} &\geq \alpha \\ 2A - 4B &\leq 2200 \\ 2A - 4B &\geq 2000 \\ \frac{2A + B - 720}{80} &\geq \alpha \\ 2A + B &\leq 800 \\ 2A + B &\geq 720 \\ 0 \leq \alpha &\leq 1. \end{aligned}$$

Este modelo tiene la solución (entera) : $\alpha = .497024$, $A = 154$, $B = 446$, lo que no da los valores para la materia prima de 2100 unidades y para el número de trabajadores igual a 762.

4 Programación posibilística

Un programa lineal posibilístico es un programa lineal en el cual, los coeficientes del mismo son conocidos de forma "imprecisa" mediante números borrosos. Este se puede expresar como:

$$\text{Maximizar } Z = \tilde{c}_1 x_1 + \tilde{c}_2 x_2 + \dots + \tilde{c}_n x_n$$

sujeto a:

$$\begin{aligned} \widetilde{a}_{11}x_1 + \widetilde{a}_{12}x_2 + \dots + \widetilde{a}_{1n}x_n &\leq \widetilde{b}_1 \\ \widetilde{a}_{21}x_1 + \widetilde{a}_{22}x_2 + \dots + \widetilde{a}_{2n}x_n &\leq \widetilde{b}_2 \\ &\vdots \\ \widetilde{a}_{m1}x_1 + \widetilde{a}_{m2}x_2 + \dots + \widetilde{a}_{mn}x_n &\leq \widetilde{b}_m, \end{aligned}$$

donde el símbolo $\widetilde{}$ sobre los coeficientes significa que se trata de un número borroso. En este trabajo nos enfocaremos sólo al caso de números borrosos triangulares (ver Figura 3).

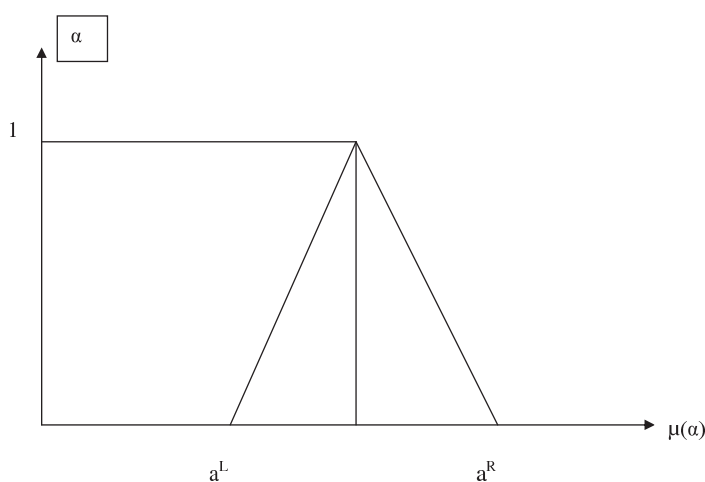


Figura 3: Representación gráfica de un número borroso triangular respecto de su nivel de membresía α .

Definición 1 *Dados dos números borrosos \tilde{a} y \tilde{b} con α -corte de nivel α $a_\alpha = [a^L(\alpha), a^R(\alpha)]$ y $b_\alpha = [b^L(\alpha), b^R(\alpha)]$. Se dice que $\tilde{a} \leq_\alpha \tilde{b}$ para un grado de pertenencia α si:*

$$a^L(\alpha) \leq b^L(\alpha) \text{ y } a^R(\alpha) \leq b^R(\alpha).$$

Definición 2 *Dados dos números borrosos $\tilde{a} = (a^1, a^2, a^3)$ y $\tilde{b} = (b^1, b^2, b^3)$. Se dice que $\tilde{a} \leq \tilde{b}$ si:*

$$a^1 \leq b^1, a^2 \leq b^2, \text{ y } a^3 \leq b^3.$$

De la definiciones anteriores podemos observar que cada restricción de la forma:

$$\widetilde{a}_{i1}x_1 + \widetilde{a}_{i2}x_2 + \dots + \widetilde{a}_{in}x_n \leq \widetilde{b}_i,$$

puede escribirse como:

1. Si se utiliza la definición 1, para un nivel α :

$$\begin{aligned} a_{i1}^L(\alpha) + a_{i2}^L(\alpha) + \dots + a_{in}^L(\alpha) &\leq b_i^L(\alpha) \\ a_{i1}^R(\alpha) + a_{i2}^R(\alpha) + \dots + a_{in}^R(\alpha) &\leq b_i^R(\alpha). \end{aligned}$$

2. Si se utiliza la definición 2:

$$\begin{aligned} a_{i1}^1(\alpha) + a_{i2}^1(\alpha) + \dots + a_{in}^1(\alpha) &\leq b_i^1(\alpha) \\ a_{i1}^2(\alpha) + a_{i2}^2(\alpha) + \dots + a_{in}^2(\alpha) &\leq b_i^2(\alpha) \\ a_{i1}^3(\alpha) + a_{i2}^3(\alpha) + \dots + a_{in}^3(\alpha) &\leq b_i^3(\alpha). \end{aligned}$$

Para el caso de la función objetivo, ésta tendrá un tratamiento multiobjetivo de la forma:

- Si se utiliza la definición 1, para un nivel α :

$$\begin{aligned} c_1^L(\alpha)x_1 + c_2^L(\alpha)x_2 + \dots + c_n^L(\alpha)x_n \\ c_1^R(\alpha)x_1 + c_2^R(\alpha)x_2 + \dots + c_n^R(\alpha)x_n. \end{aligned}$$

- Si se utiliza la definición 2:

$$\begin{aligned} c_1^1x_1 + c_2^1x_2 + \dots + c_n^1x_n \\ c_1^2x_1 + c_2^2x_2 + \dots + c_n^2x_n \\ c_1^3x_1 + c_2^3x_2 + \dots + c_n^3x_n. \end{aligned}$$

A manera de ejemplo considere el ejemplo inicial junto con las siguientes tablas:

| | Producto 1 | Producto 2 | Restricción |
|--------------------------|-------------|-------------|--------------------|
| Beneficio unitario | (2, 3, 5) | (7, 10, 12) | |
| Consumo de materia prima | (1.5, 2, 3) | (3, 4, 6) | (1500, 2000, 2200) |
| Mano de obra | (1,2,3) | (0.5, 1, 2) | (790, 800, 850) |

| Datos para un α-corte de 0.5 | Producto 1 | Producto 2 | Restricción |
|---|-------------|-------------|--------------|
| Beneficio unitario | (2.5, 4) | (8.5, 11) | |
| Consumo de materia prima | (1.75, 2.5) | (3.5, 5) | (1750, 2100) |
| Mano de obra | (1.5, 2.5) | (0.75, 1.5) | (795, 825) |

Para el caso del α -corte de 0.5, el modelo multiobjetivo a optimizar será:

$$\text{Maximizar } 2.5A + 8.5B$$

$$\text{Maximizar } 4A + 11B$$

sujeto a:

$$1.75A + 3.5B \leq 1750$$

$$2.5A + 5B \leq 2100$$

$$1.5A + 0.75B \geq 795$$

$$2.5A + 1.5B \geq 825.$$

El modelo antes planteado tiene una solución de valor objetivo de 6799 con $A=428$ y $B=206$.

Para el caso de la primer tabla se tiene el siguiente sistema multiobjetivo:

$$\text{Maximizar } 2A + 7B$$

$$\text{Maximizar } 3A + 10B$$

$$\text{Maximizar } 5A + 12B$$

sujeto a:

$$1.5A + 3B \leq 1500$$

$$2A + 4B \leq 2000$$

$$3A + 6B \leq 2200$$

$$1A + 0.5B \geq 790$$

$$2A + 1B \geq 800$$

$$3A + 2B \geq 850.$$

5 Conclusiones

Existen muchos problemas prácticos en donde es necesario un planteamiento del modelo mediante programación borrosa o mediante programación posibilista. Como se ha podido observar, los planteamientos de los modelos borrosos o posibilísticos se tienen que llevar a un nuevo modelo tipo “exacto” (crisp en inglés) para poder encontrar su solución.

Referencias

- [1] De Andrés Sánchez J.; Terceño Gómez A.,(2002) “Programación matemática y regresión lineal con instrumentos de la teoría de los subconjuntos borrosos”, Departamento de Gestión de Empresas, Facultad de Ciencias Económicas y Empresariales. Universidad Rovira i Virgili.
- [2] Fuller, R. (1986) “On a special type of fuzzy linear programming”, *Colloquia Mathematica Societatis Janos Bolyai* **49**.
- [3] Kaufmann, A.; Gil Aluja, J.; Terceño, A. (1994), “Matemática para la economía y la gestión de empresas”. Foro Científico, Barcelona.
- [4] Lai, Y.J.; Hwang, C.L. (1992) “A new approach to some possibilistic linear programming problem”, *Fuzzy sets and systems* **49**.

- [5] Lai, Y.L.; Hwang, C.L. (1996) “Fuzzy Multiple Objective Decision Making”. Springer-Verlag, Berlin.
- [6] Lai, Y.L.; Hwang, C.L. (1992) “Fuzzy Mathematical Programming”. Springer-Verlag, Berlin.
- [7] Ramik, J.; Rimanek, J. (1985) “Active decision making in fuzzy linear optimization problems”, *European Journal of Operational Research* **41**.
- [8] Rommelfanger, H.; Hanuscheck, R.; Wolf, J. (1989) “Linear programming with fuzzy objectives”, *Fuzzy Sets and Systems* **29**.
- [9] Tanaka, H.; Asai, K. (1984) “Fuzzy linear programming with fuzzy numbers”, *Fuzzy Sets and Systems* **13**.
- [10] Zimmermann, H.-J. (1991) “Fuzzy Set Theory and Its Applications”. Kluwer Academic, Dordrecht.