

Proyecciones Paramétricas para el Escape de Aristas en Poliedros de Forma $Ax \leq b$

Oscar Y. Buitrago^{(1,2)*}, Andrés L. Ramírez⁽¹⁾ y Rodrigo A. Britto⁽³⁾

(1) Facultad de Ingeniería, Universidad Militar Nueva Granada, Carrera 11 No. 101-80, Bogotá, Colombia.

(2) Facultad de Ingeniería, Universidad de Carabobo, Avenida Universidad, Bárbula, Valencia, Venezuela.

(3) Facultad de Administración, Universidad de Los Andes, Carrera 1 N° 18A- 12 Bogotá, Colombia

(e-mail: oscar.buitrago@unimilitar.edu.co; andres1729@yahoo.com; ro-britt@uniandes.edu.co)

* Autor a quien debe dirigirse la correspondencia

Recibido Mar. 16, 2017; Aceptado May. 22, 2017; Versión final Jul. 14, 2017, Publicado Dic. 2017

Resumen

El objetivo de este trabajo es proponer un procedimiento para la creación de vectores de escape diseñados específicamente para realizar proyecciones ortogonales desde las aristas de poliedros $Ax \leq b$ hacia el interior de los mismos. El procedimiento propuesto puede ser implementado en algoritmos de punto interior para la optimización de problemas de programación lineal, pues en los mismos se emplean diversas estrategias para evitar llegar hasta la frontera de los poliedros. Para esto se desarrolló un procedimiento de proyección hacia el interior del poliedro desde tres situaciones y se probó con un ejemplo en contexto. Los resultados muestran que la proyección realizada en la iteración en que se aplica el escape no solo permite salir de un punto subóptimo sino que también mantuvo el último valor de la función objetivo. Por lo tanto, se puede concluir que no es necesario evitar la frontera del poliedro o ajustar valores de paso como ocurre en otras aplicaciones.

Palabras clave: vector de escape; poliedro; optimización; proyecciones ortogonales; algoritmo de punto interior

Parametric Projections to Escape from Edges in Polyhedrons of the Type $Ax \leq b$

Abstract

The aim of this paper is to propose a procedure to create escape vectors specifically developed to perform an orthogonal projection from an $Ax \leq b$ type polyhedron's edge to its interior. The proposed procedure can be implemented in interior-point methods for solving linear programming problems since in these methods various strategies are used to avoid reaching the borders of the polyhedron. A projection procedure was developed to the polyhedron's interior starting from three scenarios and it was used on a context case demonstrating its applicability. The results show that the projection carried out in the iteration in which the escape is done not only allows abandoning from a suboptimal solution but it also kept the same objective function value. Therefore, it can be concluded that interior-point methods do not need to avoid the frontier or to adjust values, as it occurs in other applications.

Keywords: escape vector; polyhedron; optimization; orthogonal projections; interior-point method

INTRODUCCIÓN

El cómo realizar proyecciones desde la frontera de un poliedro $Ax \leq b$ hacia el interior del mismo pareciera ser un problema con solución trivial. Sin embargo, hacer este tipo de movimientos resulta complicado especialmente cuando se parte de aristas o puntos extremos. Una implicación inmediata se da en métodos de punto interior para solucionar problemas de programación lineal (PL), pues en ellos el acercarse a la frontera del poliedro puede hacer que la siguiente solución de prueba quede acorralada (Winston, 2004) o que se realicen búsquedas lineales unidimensionales (Mehrotra, 1992). Por ello se han propuesto diversos métodos para evitar aproximaciones a los límites del poliedro, tales como los de escalado afín de Dikin (1967) y de Todd (1991), y el de barrera logarítmica (Eiselt y Sandblom, 2007).

Recientemente estudios de solución y aplicación de PL o MIP (por las siglas en inglés de mixed-integer programming) incluyen; la expansión de redes de transmisión eléctrica (Tejada et al., 2014), la selección de portafolios dados una rentabilidad y riesgo (Guastaroba et al., 2014), la optimización de sistemas de energía residenciales (Lauinger et al., 2016), la minimización de tiempo y costo de transporte de mercancías desde diferentes orígenes a diferentes destinos con diferentes medios de transporte (Baidya et al., 2016), la solución del problema de reconstrucción de árboles generadores mínimos (Fortz et al., 2017) y la optimización del manejo de especies invasivas (Kıbiş y Büyüktaktakın, 2017). Otro aspecto que refuerza la importancia de proponer escapes de las fronteras de los poliedros definidos por las restricciones de los problemas PL, es que aun cuando el método de solución más famoso es el Simplex (Dantzig, 1951) que se mueve sobre los bordes del poliedro, existen muchos algoritmos que transitan por el interior del mismo y que teóricamente lo aventajan por su complejidad computacional (Powell, 1993). Entre estos se cuentan el elipsoidal (Khachiyan, 1982) y su variante proyectiva (Karmarkar, 1984).

La búsqueda de métodos de solución para algoritmos de punto interior (que son a los que se les puede incorporar la estrategia de escape propuesta) continúa, y se han encontrado propuestas recientes como las de Mehendale (2015) y la de Berti et al (2017). Además, las aplicaciones de este tipo de algoritmos no se limitan a problemas PL. Por ejemplo, Klintberg y Gros (2016) recurren a uno para enfrentar problemas de control óptimo en procesos industriales, Wu et al (2016) utilizan un algoritmo de punto interior para optimización del flujo de energía eléctrica, Touil et al (2017) proponen un algoritmo primal-dual (de punto interior) para solucionar problemas de programación semidefinida. Finalmente, Dattaa et al (2017) incorporan un procedimiento de punto interior para solucionar problemas de optimización multiobjetivo en regiones no convexas.

En este trabajo se propone una forma para volver al interior de poliedros de la forma $Ax \leq b$ desde las proximidades de las fronteras. Aunque pueda tener otras aplicaciones, una muy importante es que abre las puertas al desarrollo de nuevos algoritmos de solución de problemas PL o a modificaciones a los ya existentes, puesto que se elimina la restricción de no acercarse a la frontera. Específicamente, para escapar de la frontera se plantea el diseño de proyecciones paramétricas sobre hiperplanos perpendiculares al vector gradiente del problema PL. De esta forma se mantiene el valor de la función objetivo correspondiente al punto desde el que se aplica el escape. Si la propuesta se incorpora a un algoritmo iterativo se obtiene un nuevo punto interior (que no reduce el valor de la función objetivo alcanzado hasta el momento) para la continuación de las iteraciones en busca de la solución óptima.

Con respecto a la estructura del artículo, en primer lugar se muestra el problema que se busca solucionar. Después se describe la propuesta para escapar de puntos de frontera (correspondiente a puntos de convergencia no óptima en algoritmos de punto interior) y la forma de cálculo de los que se denominan vectores de escape. Posteriormente, se realizan las correspondientes demostraciones sobre el método de escape propuesto y para hacer una ilustración de cómo funciona, se formula un problema PL al que se le aplica el método de proyecciones ortogonales paramétricas de Ramírez et al., (2012) (en realidad se puede elegir cualquier procedimiento que llegue a la frontera), haciendo que converja a una arista (no óptima) para desde allí implementar el vector de escape. Finalmente se presentan las correspondientes conclusiones.

PLANTEAMIENTO DEL PROBLEMA

Inicialmente hay que solucionar el problema de cómo realizar una proyección desde la frontera de un poliedro $Ax \leq b$ hacia el interior del mismo. Sin embargo, encontrar una respuesta a esta cuestión no es suficiente porque al momento de implementar esta estrategia en un algoritmo de solución de problemas PL se requiere que el valor de la función objetivo logrado no disminuya. Es decir, no es deseable que un algoritmo de solución pueda retroceder.

Puesto que una aplicación que resulta evidente para implementar la estrategia de escape de puntos de frontera en poliedros es incorporarla a algoritmos de punto interior, se toma uno de ellos para definir

términos asociados a problemas PL y proyecciones y a la vez ilustrar el problema de llegar a la frontera del poliedro y no escapar de allí. Para este propósito, se hace una breve descripción del método de proyecciones ortogonales paramétricas (Ramírez et al., 2012). En ella se define un problema de programación lineal como $\max(\min) z = dx$, sujeto a: $Ax \leq b$; $x \geq 0$, donde d es el vector de los n coeficientes de la función objetivo, A $[m \times n]$ la matriz de coeficientes de las m restricciones, x el vector que contiene las n variables del problema, b el vector de los lados derechos de las m restricciones y 0 un vector compuesto por n ceros. En su propuesta de proyecciones paramétricas ortogonales, los autores reescriben el PL como se indica en (1).

$$\max(\min) z = dx, \text{ sujeto a: } \pi_i \leq b_i, i \leq m \quad x \geq 0 \quad (1)$$

Cada vector fila de A se denomina f_i , de esta forma la i -ésima restricción se puede definir como se expresa en (2).

$$f_i x \leq b_i \quad i = 1, 2, \dots, m. \quad (2)$$

El lado izquierdo de la desigualdad mostrada en la expresión (2) es el término π_i de la inecuación presentada en la región factible del problema de optimización definido en (1) y representa el correspondiente lado izquierdo de la i -ésima restricción.

El algoritmo de proyecciones paramétricas tiene dos etapas: inicialización y ciclo iterativo. En la primera se construyen $n-1$ vectores ortogonales al gradiente de la función objetivo, d , y se definen $n-1$ rectas paramétricas a partir de un punto interior inicial. Proyectando para cada recta, tanto en sentido positivo como negativo, se encuentran las $2(n-1)$ intersecciones con las caras del poliedro y se calcula un centroide C para desde allí proyectar d .

El ciclo iterativo consiste en (i) realizar proyecciones ortogonales al vector gradiente desde el punto interior C , (ii) determinar las intersecciones de las proyecciones sobre cada una de las caras del poliedro y (iii) calcular un nuevo centroide C_i , más cercano al óptimo que su predecesor. Desde C_i se vuelve a proyectar d hasta la cara más cercana del poliedro. Desde esta intersección se realiza una proyección ortogonal hasta otra cara del poliedro y se calcula el nuevo centroide con estos puntos. La iteración de este procedimiento debe permitir acercarse tanto como se quiera al punto óptimo.

El problema con este tipo de procedimientos es que se puede converger a un punto no óptimo ubicado en las aristas del poliedro. La propuesta para salir de este tipo de puntos y lograr la convergencia a la solución óptima es calcular para cada restricción un vector de escape ortogonal al gradiente, que como consecuencia mantenga constante el valor de la función objetivo que se ha logrado y, por supuesto, que apunte al interior de la región factible.

PROPUESTA DE VECTORES DE ESCAPE

Cada uno de los hiperplanos $f_i x = b_i$ que limitan la región factible tiene como vector ortogonal a f_i . En general, estos vectores no son perpendiculares al gradiente. Sin embargo, una característica deseable en un "vector de escape" (definiendo así a uno que permita proyectar desde un punto de frontera hacia el interior del poliedro) es que sus proyecciones mantengan el valor de la función objetivo constante e igual al que tenga el punto del que se está escapando.

Se propone definir como vector de escape de la i -ésima restricción la diferencia entre el correspondiente vector ortogonal normalizado y la proyección del mismo sobre el gradiente normalizado (los vectores d y f_i normalizados se simbolizan con el supra índice n). Conceptualmente el vector de escape para la i -ésima cara (i -ésimo hiperplano soporte del poliedro o i -ésima restricción tomada como igualdad) se presenta en (3).

$$f_i^n - \text{proy}_{d^n} f_i^n \quad (3)$$

La dirección del vector de escape conceptualizado en (3) es perpendicular al gradiente y por lo tanto permite proyectar desde la frontera hacia el interior del poliedro con la ventaja de no disminuir el valor de la función objetivo. Así, algebraicamente el vector de escape para el i -ésimo hiperplano está dado por (4).

$$E_i = \begin{cases} -f_i^n - (-f_i^n - d^n) d^n & \text{si } \pi_i \leq b_i \\ f_i^n - (f_i^n - d^n) d^n & \text{si } \pi_i \geq b_i \end{cases} \quad (4)$$

Al proyectar los vectores de escape, el orden de magnitud de la norma no incide en el resultado del punto encontrado sobre la cara más cercana, sino que simplemente cambia el valor del parámetro de proyección. No obstante, para hacer operaciones entre ellos (e.g. sumas) el valor de la norma sí es importante. Por ello, para combinarlos y obtener una dirección que permita volver al interior desde aristas del poliedro, los vectores de escape deben estar normalizados.

Estrategia de escape

Para estructurar el procedimiento de escape se consideraron tres situaciones de acuerdo con la ubicación de puntos en la frontera del poliedro $Ax \leq b$ (n variables y m restricciones): (i) sobre un único hiperplano, (ii) sobre la intersección de dos hiperplanos y (iii) sobre la intersección de más de dos hiperplanos.

Desde puntos ubicados sobre un hiperplano

Retornar al interior del poliedro desde un punto P' que pertenece solamente a uno de los hiperplanos que definen la región factible es sencillo. Basta con proyectar el vector de escape del hiperplano en cuestión, de acuerdo con la ecuación (4), hasta que se encuentre con un hiperplano diferente para tener un punto R . El punto medio o cualquier combinación lineal convexa entre P' y R proporciona un punto interior.

Desde puntos ubicados sobre la intersección de dos hiperplanos

Para proyectarse al interior del poliedro desde un punto P' que se localiza en la intersección de dos hiperplanos se deben identificar los mismos. Luego, con la ecuación (4) calcular los dos respectivos vectores de escape, normalizarlos y hacer la suma vectorial. Finalmente, el vector resultante debe ser normalizado para obtener el vector de escape W .

Desde puntos ubicados sobre la intersección de más de dos hiperplanos

Cuando el punto de frontera P' del que es necesario escapar se encuentra localizado sobre la intersección de más de dos hiperplanos, es necesario:

1. Definir el conjunto H cuyos elementos son los subíndices de los hiperplanos $\pi_h = b_h$ sobre cuya intersección se encuentra ubicado el punto P' . De esta forma se elabora una lista L conformada por un subconjunto, cuyo cardinal es menor o igual a m , del conjunto formado por el total de los hiperplanos asociados a las restricciones del poliedro.
2. Obtener los vectores de escape normalizados de los hiperplanos pertenecientes a L según la ecuación (4).
3. Regresar un cambio porcentual sobre la proyección que llevó a P' para de esta forma obtener un punto interior P_0 ubicado a menos de δ unidades de los hiperplanos pertenecientes a L .
4. Proyectar $P_0 + t^*E_1$ hasta la cara f_k usando el primer vector de escape de la lista L . Si el punto Q_1 resultante incide sobre una sola cara, hacer $Q_1 + t^*E_k$. Si incide sobre dos caras, proyectar desde Q_1 la combinación de los dos vectores de escape correspondientes. Si incide sobre más de dos caras, anidar el procedimiento descrito.
5. Continuar el procedimiento hasta incidir en un punto R sobre un hiperplano (cara del poliedro) que no pertenezca a L .
6. Calcular el vector que va de P_0 a R y normalizarlo para obtener el vector de escape W .

Demostración: Sea H el conjunto de subíndices h de los hiperplanos $\pi_h = b_h$ cuya intersección contiene el punto de frontera P' y sea P_0 un punto interior a menos de δ unidades (medidas perpendicularmente al gradiente) de los hiperplanos $\pi_h = b_h$, $h \in H$. Se mostrará que el algoritmo no puede converger a un punto Q que pertenezca a algún π_h .

Suponga que el algoritmo converge a Q . Esto significa que dado un $\varepsilon > 0$, existe $N \in \mathbb{N}$, tal que si $n > N$

entonces $d(P_0, Q) < \varepsilon$. Considere las esferas S_Q y S_{P_n} centradas en P' y con radios $\|P'Q\|$ y $\|P'P_n\|$ respectivamente. Entonces todos los puntos P_m con $m > n$ deben estar afuera de S_{P_n} y dentro de S_Q , pues la sucesión de distancias $\|P'P_n\|$ es estrictamente creciente.

Sea π_n el hiperplano tangente a S_{P_n} en el punto P_n . El vector que va de P_n a P_{n+1} está contenido en π_n y $\pi_n \cap S_Q$ no debe ser encontrada este vector antes que el hiperplano en que se encuentra P_{n+1} . Pero a medida que $\varepsilon \rightarrow 0$, $S_Q \cap \pi_n \rightarrow Q$ y por lo tanto P_{n+1} está fuera de S_Q . Entonces por contradicción, el algoritmo de escape no puede converger a un punto Q que pertenezca a algún(os) $\pi_h = b_h$, $h \in H$. Por lo tanto el algoritmo de escape encontrará finalmente otro hiperplano y el vector que une a P' con el punto hallado sobre el nuevo hiperplano es un vector de escape. La figura 1 permite interpretar geoméricamente la demostración.

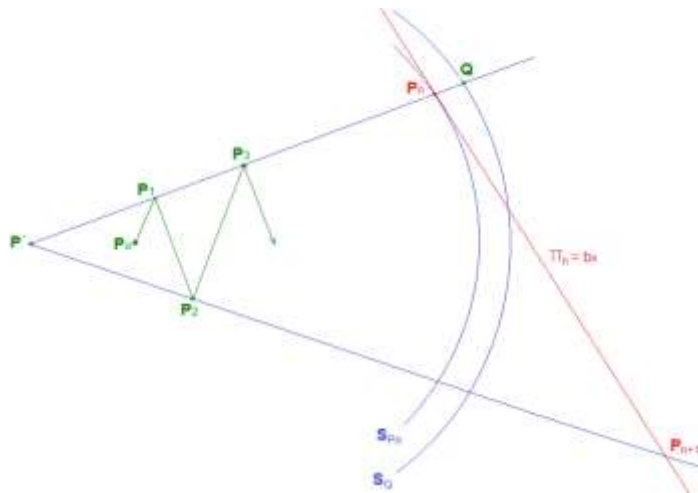


Fig. 1: Estrategia general de escape.

Ejemplo de aplicación

Se diseñó el problema de programación lineal (5) con el propósito específico de que el algoritmo de proyecciones ortogonales converja a una arista. En el caso del problema de PL (5), el poliedro definido por la región factible tiene cinco aristas, cada una conformada por la intersección de dos hiperplanos, tal y como se puede apreciar en la Figura 2.

$$\text{Maximizar: } z = x_1 + x_2 + 10x_3; \text{ sujeto a: } 7x_1 + x_2 + x_3 \leq 20; x \geq 0 \quad (5)$$

La solución óptima de (5) se encuentra en $[0 \ 0 \ 20]$ y cuya función objetivo tiene un valor de 200. Partiendo desde el punto interior $P_0 = [1 \ 5 \ 1]$ con $z = 16$, el algoritmo converge al punto de frontera $P_f = [0 \ 6.19 \ 13.81]$ con $z = 144.29$ y que se encuentra ubicado sobre la arista conformada por las restricciones $7x_1 + x_2 + x_3 = 20$ y $x_1 = 0$. En la figura 2 se muestra la secuencia de puntos con que el algoritmo de proyecciones ortogonales lleva del punto inicial P_0 al punto no óptimo de convergencia P_f . Alcanzar el punto óptimo del problema (5) implica salir de P_f y continuar iterando, escapando tantas veces como sea necesario de aristas o de puntos de frontera. Aplicando (4) se calculan los vectores de escape, E_i , para las caras que forman la arista en la que se encuentra P_f . En (6) se muestra el resultado obtenido para el vector de escape normalizado, E_1 , correspondiente al hiperplano $7x_1 + x_2 + x_3 = 20$.

$$E_1 = [-0.9867 \ -0.1191 \ 0.1106] \quad (6)$$

El vector de escape normalizado, E_2 , para el hiperplano $x_1 = 0$ se presenta en la expresión (7).

$$E_2 = [0.9951 \ -0.0098 \ -0.0985] \quad (7)$$

Los vectores de escape E_1 e E_2 , dados por (6) y (7) respectivamente, se muestran en la figura 2. Cada uno de ellos se apunta al interior del poliedro desde puntos en la respectiva cara para la que se generaron y que no estén sobre una arista, cara impropia o punto extremo. En contrapartida, las proyecciones de E_1 e E_2 salen del poliedro en aristas y, por lo tanto, estos dos vectores (normalizados) se deben combinar para obtener una dirección dirigida al interior. La combinación de estos vectores es el vector combinado de la figura 2 y cuyas coordenadas son $[0.0646 \ -0.9936 \ 0.0929]$.

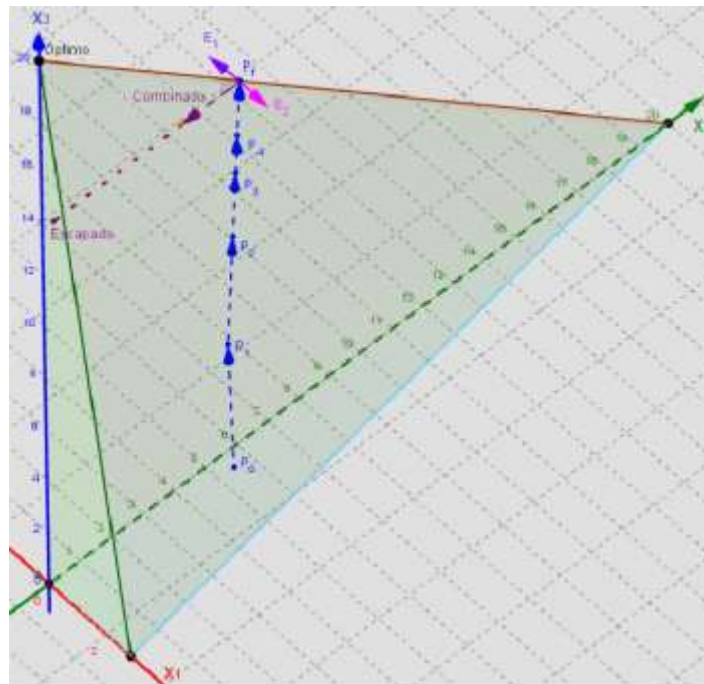


Fig. 2: Escape de un punto sobre una arista en la frontera del poliedro.

Proyectando el vector combinado desde P_f se llega al punto escapado de la figura 2, que se ubica sobre otra cara del poliedro. La combinación lineal convexa entre el punto escapado $[0,4034 \ 0 \ 14,381]$ y P_f también está en el interior del poliedro. La figura 3 presenta una ampliación para mostrar que el regreso al interior desde un punto de frontera ubicado en un único hiperplano, se hace con el respectivo vector de escape. El punto escapado se promedia con P_f para tener el centroide $C_m [0,2017 \ 3,0952 \ 14,0950]$, que es un punto interior del poliedro con un valor z igual al de P_f y al del punto escapado.

Desde C_m se proyecta d para llegar a $P_{2f} [0,2793 \ 3,1728 \ 14,8716]$, cuyo valor objetivo es 152.16 y está sobre el hiperplano $7x_1 + x_2 + x_3 = 20$. Para retornar al interior, se proyecta el vector de escape E_1 , dado por (6), llegando a $P_{2Escapado} [0 \ 3,1391 \ 14,9029]$. Se calcula el centroide C_{2m} (punto medio entre P_{f2} y P_{e2}) y proyecta nuevamente d . El resultado de esta proyección es $P_{3f} [0,1946 \ 3,2109 \ 15,4363]$ con $z = 158$.

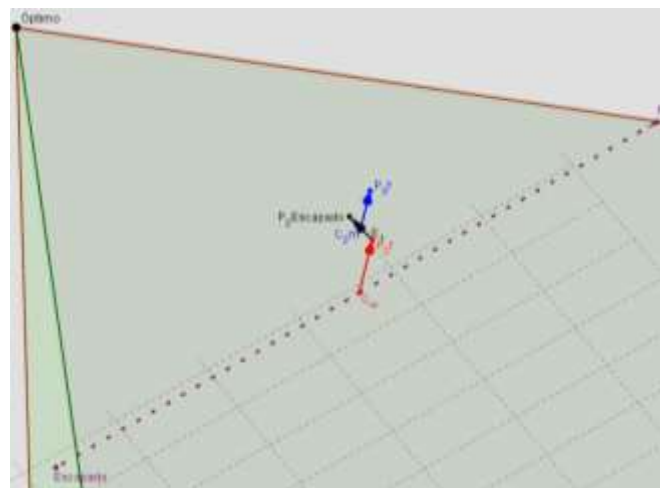


Fig. 3: Escape de un punto de frontera ubicado sobre un hiperplano

DISCUSION

La revisión de la literatura muestra que realizar proyecciones hasta la frontera del poliedro $Ax \leq b$ es una acción que se evita en los algoritmos de punto interior porque limita el avance del proceso o precipita convergencias a puntos no óptimos. La propuesta presentada permite obtener vectores de escape desde puntos de frontera, como se demostró, y por ello se abre la posibilidad de diseñar algoritmos en los que no sea necesario tomar previsiones para evitar acercarse a la frontera. La otra alternativa es incorporar la estrategia de escape a algoritmos ya existentes.

Para ilustrar con un ejemplo el funcionamiento de los vectores de escape, es suficiente con definir un poliedro y localizarse en un punto sobre su frontera. Sin embargo, para contextualizar el cómo puede darse una situación en que se llegue a un punto de frontera en un poliedro $Ax \leq b$, se tomó el algoritmo de proyecciones ortogonales paramétricas (Ramírez, et al, 2012) e intencionalmente se le hizo converger a un punto no óptimo, sin embargo pudo haberse elegido cualquier método de punto interior pues el propósito no es mostrar el funcionamiento del mismo sino presentar un ejemplo con contexto.

Una característica importante de los vectores de escape diseñados y que se pone de manifiesto en el ejemplo, es que estos mantienen el valor de la función objetivo logrado hasta el momento. Este hecho es importante si se piensa en una aplicación en algoritmos de punto interior porque si se hace la combinación lineal convexa (por simplicidad puede ser el punto medio, pero dependerá del diseño del algoritmo) entre el punto desde el que se escapa y el punto al que se escapó, se dispone de un nuevo punto interior para continuar con el proceso iterativo sin reducir el valor de la función objetivo.

CONCLUSIONES

Se puede concluir que se diseñó un procedimiento general para proyectarse hacia el interior de un poliedro $Ax \leq b$ desde puntos ubicados en su frontera. La estrategia de escape se subdividió en tres situaciones: (i) puntos ubicados sobre un único hiperplano, (ii) sobre la intersección de dos hiperplanos y (iii) sobre la intersección de más de dos hiperplanos. Puesto que se deben hacer sumas entre vectores y a que las magnitudes de los mismos pueden ser bastante disímiles, se concluye que estas operaciones se deben realizar con vectores normalizados. Finalmente, se presentaron las correspondientes demostraciones y un ejemplo para mostrar que la incorporación de la estrategia en algoritmos de tipo proyectivo permite que estos escapen de falsas convergencias, proporcionándoles un nuevo punto interior a partir del cual continuar iterando.

Finalmente, teniendo a disposición vectores y una estrategia de escape, no es necesario evitar la frontera del poliedro o ajustar valores de paso como ocurre en otras aplicaciones, por ejemplo en el algoritmo primal dual de punto interior (Mehrotra, 1992).

AGRADECIMIENTOS

Este estudio es producto derivado del proyecto de investigación INV-ING-2105 financiado por la Vicerrectoría de Investigaciones de la Universidad Militar Nueva Granada (Colombia), vigencia 2016.

REFERENCIAS

- Baidya, A., U. Bera y M. Maiti. The grey linear programming approach and its application to multi-objective multi-stage solid transportation problem. *Opsearch*, 53 (3), 500-522 (2016)
- Berti, L.F., Oliveira, A.R.L. y Ghidini, C.T.L.S. A variation on the interior point method for linear programming using the continued iteration, doi:10.1007/s00186-016-0558-9, *Math Meth Oper Res*, 85 (1), 61-75 (2017)
- Dantzig, G. B. *Activity Analysis of Production and Allocation* (Ed. Koopmans, T. C.), Cowles Commission for Research in Economics, 339-347 (1951)
- Dattaa, S., A. Ghoshb., K. Sanyal y S. Das. A Radial Boundary Intersection aided interior point method for multi-objective optimization, doi:10.1016/j.ins.2016.09.062. *Information Sciences*, 377, 1-16 (2017)
- Dikin, I.I., *Iterative Solution of Problems of Linear and Quadratic Programming*, Soviet Mathematics Doklady, 8, 674-675 (1967)
- Eiselt, H.A., y C.L. Sandblom. *Linear Programming and its Applications*. Berlin: Springer Verlag Berlin Heidelberg, Alemania (2007)
- Fortz, B., O. Oliveira y C. Requejo. Compact mixed integer linear programming models to the minimum weighted tree reconstruction problem, doi:10.1016/j.ejor.2016.06.014. *European Journal of Operational Research*, 256 (1), 242-251 (2017)
- Guastaroba, G., R. Mansini, W. Ogryczak y M. Speranza. Linear Programming Models based on Omega Ratio for the Enhanced Index Tracking Problem, Warsaw University of Technology, Institute of Control & Computation Engineering, (33), 1-29 (2014)
- Karmarkar, N. A new polynomial-time algorithm for linear programming, *Combinatorica* 4, 373-395 (1984)

- Khachiyan, L. G. On the exact solution of systems of linear inequalities and linear programming problems, *USSR Comput, Mathematics and Mathematical Physics*, 22 (4), 239-242 (1982)
- Kibiş, E.Y. y I.E. Büyüktaktın. Optimizing invasive species management: A mixed-integer linear programming approach, doi:10.1016/j.ejor.2016.09.049, *European Journal of Operational Research*, 259, 308-321 (2017)
- Klintberg, E. y S. Gros. An inexact interior point method for optimization of differential algebraic systems, doi:10.1016/j.compchemeng.2016.04.013. *Computers and Chemical Engineering*, 92, 163-171 (2016)
- Lauinger, D., P. Caliandro, J. Van Herle y D. Kuhn, A linear programming approach to the optimization of residential energy systems, *J. Energy Storage*, 7, 24-37 (2016)
- Mehendale, D. P., A New Algorithm for Linear Programming, 1-53, [math.GM] (2015)
- Mehrotra, S., On the Implementation of a Primal-Dual Interior Point Method. *SIAM Journal on Optimization*, 2(4), 575-601 (1992)
- Powell, M.J. On the number of iterations of Karmarkar's algorithm for linear programming. *Mathematical Programming*, 62 (1), 153-197 (1993)
- Ramírez, A., O. Buitrago, R. Britto y A. Fedossova. A new algorithm for solving linear programming problems, *Ingeniería e Investigación*, 32 (2), 68-73 (2012)
- Tejada, D., J. Lopez y M. Rider. Planeamiento de la expansión de la red de transmisión en sistemas eléctricos de potencia considerando reconfiguraciones de red. *Información Tecnológica*, 25(1), 85-96 (2014)
- Todd, M. The Aff ine-Scaling Direction for Linear Programming is a Limit of Projective-Scaling Directions. *Linear Algebra and Its Applications*, 152, 93-105 (1991)
- Touil, I., D. Benterki y A. Yassine. A feasible primal-dual interior point method for linear semidefinite programming, doi:10.1016/j.cam.2016.05.008. *Journal of Computational and Applied Mathematics*, 312, 216-230 (2017)
- Winston, W. *Operations Research: Applications and Algorithms*, 4ª Ed. Cengage Learning, Boston, USA (2003)
- Wu, W., Z. Hu, Y., Song. A New Method for OPF Combining Interior Point Method and Filled Function Method. doi: 10.1109/PESGM.2016.7741321, 2016 IEEE Power and Energy Society General Meeting (PESGM), Boston, USA, 1 al 5 Julio (2016)