Métodos numéricos en LSIP

Valentín Jornet

Seminario en Homenaje al Prof. Marco A. López

20 de febrero de 2010

Outline

Referencias para Métodos en Semi-Infinita

Referencias para Métodos en Semi-Infinita

- Toda la información necesaria para esta presentación se ha obtenido de los capítulos 11 y 12 del texto **Semi-Infinite Linear Programming**, publicado por JOHN WILEY & SONS en 1998 y cuyos autores son Goberna, Miguel A.; **López, Marco A.**

Referencias para Métodos en Semi-Infinita

- Toda la información necesaria para esta presentación se ha obtenido de los capítulos 11 y 12 del texto **Semi-Infinite Linear Programming**, publicado por JOHN WILEY & SONS en 1998 y cuyos autores son Goberna, Miguel A.; **López, Marco A.**
- Aunque los siguientes trabajos también pueden considerarse aportaciones a los métodos numéricos en semi-infinita:

Referencias para Métodos en Semi-Infinita

- Toda la información necesaria para esta presentación se ha obtenido de los capítulos 11 y 12 del texto **Semi-Infinite Linear Programming**, publicado por JOHN WILEY & SONS en 1998 y cuyos autores son Goberna, Miguel A.; **López, Marco A.**
- Aunque los siguientes trabajos también pueden considerarse aportaciones a los métodos numéricos en semi-infinita:
  - Anderson,Edward J.; Goberna, Miguel A.; **López, Marco A.** Simplex-like trajectories on quasi-polyhedral sets. Math. Oper. Res. 26 (2001)

Referencias para Métodos en Semi-Infinita

- Toda la información necesaria para esta presentación se ha obtenido de los capítulos 11 y 12 del texto **Semi-Infinite Linear Programming**, publicado por JOHN WILEY & SONS en 1998 y cuyos autores son Goberna, Miguel A.; **López, Marco A.**
- Aunque los siguientes trabajos también pueden considerarse aportaciones a los métodos numéricos en semi-infinita:

  - Anderson,Edward J.; Goberna, Miguel A.; **López, Marco A.**
    Simplex-like trajectories on quasi-polyhedral sets. Math. Oper. Res. 26 (2001)
  - Auslender, A.; Goberna, M.A. ; **López, M.A.**
    Penalty and smoothing methods for convex semi-infinite programming. Math. Oper. Res. (2009)

Presentación del problema

Aunque a estas alturas el problema de LSIP es un viejo conocido, lo presentamos de nuevo y hacemos mención de sus elementos más importantes:

Presentación del problema

Aunque a estas alturas el problema de LSIP es un viejo conocido, lo presentamos de nuevo y hacemos mención de sus elementos más importantes:

- Problemas primal y dual

$$
(\mathrm{P}) \quad \begin{array}{ll} \min & c'x \\ s.a & a_t'x \geq b_t \quad , \ t \in T \end{array}
\qquad
(\mathrm{D}) \quad \begin{array}{ll} \max & \sum_{t \in T} \lambda_t b_t \\ s.a & \sum_{t \in T} \lambda_t a_t = c \\ & \lambda \in \mathbb{R}_+^{(T)} \end{array}
$$

Presentación del problema

Aunque a estas alturas el problema de LSIP es un viejo conocido, lo presentamos de nuevo y hacemos mención de sus elementos más importantes:

- Problemas primal y dual

$$\text{(P)} \quad \begin{array}{ll} \text{mín} & c'x \\ s.a & a_t'x \geq b_t \quad , \; t \in T \end{array} \qquad \text{(D)} \quad \begin{array}{ll} \text{máx} & \sum_{t\in T} \lambda_t b_t \\ s.a & \sum_{t\in T} \lambda_t a_t = c \\ & \lambda \in \mathbb{R}_+^{(T)} \end{array}$$

- Denotamos por F el conjunto factible del primal y por Λ el conjunto factible del dual.

Presentación del problema

Aunque a estas alturas el problema de LSIP es un viejo conocido, lo presentamos de nuevo y hacemos mención de sus elementos más importantes:

- Problemas primal y dual

$$(\mathrm{P})\quad \begin{array}{ll} \min & c'x \\ s.a & a_t'x \geq b_t \quad ,\ t \in T \end{array} \qquad (\mathrm{D})\quad \begin{array}{ll} \max & \sum\limits_{t\in T} \lambda_t b_t \\ s.a & \sum\limits_{t\in T} \lambda_t a_t = c \\ & \lambda \in \mathbb{R}_+^{(T)} \end{array}$$

- Denotamos por F el conjunto factible del primal y por Λ el conjunto factible del dual.
- Los conjuntos $F_\alpha = \{x \in F \,|\, c'x \leq \alpha\}$ son los conjuntos de nivel del problema primal.

Presentación del problema

Aunque a estas alturas el problema de LSIP es un viejo conocido, lo presentamos de nuevo y hacemos mención de sus elementos más importantes:

- Problemas primal y dual

$$(P) \quad \begin{array}{ll} \text{mín} & c'x \\ s.a & a_t'x \geq b_t \quad , \; t \in T \end{array} \qquad (D) \quad \begin{array}{ll} \text{máx} & \sum_{t \in T} \lambda_t b_t \\ s.a & \sum_{t \in T} \lambda_t a_t = c \\ & \lambda \in \mathbb{R}_+^{(T)} \end{array}$$

- Denotamos por F el conjunto factible del primal y por Λ el conjunto factible del dual.
- Los conjuntos $F_\alpha = \{x \in F \,|\, c'x \leq \alpha\}$ son los conjuntos de nivel del problema primal.
- Se representa por F^* el conjunto óptimo del primal y por Λ^* el conjunto óptimo del dual.

Clasificación

Según el texto **Semi-Infinite Linear Programming**, los métodos numéricos para la resolución de un problema de LSIP pueden clasificarse en:

Clasificación

Según el texto **Semi-Infinite Linear Programming**, los métodos numéricos para la resolución de un problema de LSIP pueden clasificarse en:

- Métodos de discretización:

  Mediante parrillas

  Mediante planos de corte

Clasificación

Según el texto **Semi-Infinite Linear Programming**, los métodos numéricos para la resolución de un problema de LSIP pueden clasificarse en:

- Métodos de discretización:
  - Mediante parrillas
  - **Mediante planos de corte**
- Métodos de reducción local

Clasificación

Según el texto **Semi-Infinite Linear Programming**, los métodos numéricos para la resolución de un problema de LSIP pueden clasificarse en:

- Métodos de discretización:

  Mediante parrillas

  Mediante planos de corte

- Métodos de reducción local
- Métodos de intercambio

Clasificación

Según el texto **Semi-Infinite Linear Programming**, los métodos numéricos para la resolución de un problema de LSIP pueden clasificarse en:

- Métodos de discretización:
  - Mediante parrillas
  - **Mediante planos de corte**
- Métodos de reducción local
- Métodos de intercambio
- **Métodos del tipo simplex**

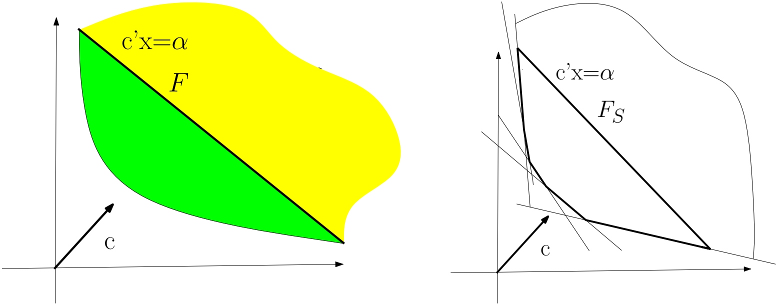
Clasificación

Según el texto **Semi-Infinite Linear Programming**, los métodos numéricos para la resolución de un problema de LSIP pueden clasificarse en:

- Métodos de discretización:
  - Mediante parrillas
  - **Mediante planos de corte**
- Métodos de reducción local
- Métodos de intercambio
- **Métodos del tipo simplex**
- Métodos de descenso

Conjuntos de nivel

Todos los métodos de discretización necesitan de la existencia de un conjunto $S \subset T$ tal que sus conjuntos de nivel sean no vacíos y acotados.
Un tal S existe si, y sólo si, F^* es no vacío y acotado.



¿Cómo actúan?

- Todos los métodos para resolver un problema LSIP generan una sucesión, $\{x^r\}$, $(\{\lambda^r\})$ que, bajo determinadas condiciones, converge a la solución óptima, x^*, $(\{\lambda^*\})$.

¿Cómo actúan?

- Todos los métodos para resolver un problema LSIP generan una sucesión, $\{x^r\}$, $(\{\lambda^r\})$ que, bajo determinadas condiciones, converge a la solución óptima, x^*, $(\{\lambda^*\})$.
- Las diferencias entre ellos radican en:

¿Cómo actúan?

- Todos los métodos para resolver un problema LSIP generan una sucesión, $\{x^r\}$, $(\{\lambda^r\})$ que, bajo determinadas condiciones, converge a la solución óptima, x^*, $(\{\lambda^*\})$.
- Las diferencias entre ellos radican en:
  - La forma de generar las sucesiones $\{x^r\}$ y $\{\lambda^r\}$.

¿Cómo actúan?

- Todos los métodos para resolver un problema LSIP generan una sucesión, $\{x^r\}$, $(\{\lambda^r\})$ que, bajo determinadas condiciones, converge a la solución óptima, x^*, $(\{\lambda^*\})$.
- Las diferencias entre ellos radican en:
  - La forma de generar las sucesiones $\{x^r\}$ y $\{\lambda^r\}$.
  - La rapidez de convergencia.

¿Cómo actúan?

- Todos los métodos para resolver un problema LSIP generan una sucesión, $\{x^r\}$, $(\{\lambda^r\})$ que, bajo determinadas condiciones, converge a la solución óptima, x^*, $(\{\lambda^*\})$.
- Las diferencias entre ellos radican en:
  - La forma de generar las sucesiones $\{x^r\}$ y $\{\lambda^r\}$.
  - La rapidez de convergencia.
  - La robustez del método.

¿Cómo actúan?

- Todos los métodos para resolver un problema LSIP generan una sucesión, $\{x^r\}$, $(\{\lambda^r\})$ que, bajo determinadas condiciones, converge a la solución óptima, x^*, $(\{\lambda^*\})$.
- Las diferencias entre ellos radican en:
  - La forma de generar las sucesiones $\{x^r\}$ y $\{\lambda^r\}$.
  - La rapidez de convergencia.
  - La robustez del método.
- Se admite que las estrategias más eficientes combinan los métodos de discretización con los de reducción local.

Outline

Método de Kelley

- Supongamos dado un conjunto finito $S = T_0 \subset T$, cuyos conjuntos de nivel son no vacíos y acotados.

Método de Kelley

- Supongamos dado un conjunto finito $S = T_0 \subset T$, cuyos conjuntos de nivel son no vacíos y acotados.
- Consideremos el problema de programación lineal

$$
\begin{array}{lll}
(\mathsf{P}_0) & \text{mín} & c'x \\
 & s.a & a_t'x \geq b_t \quad , \; t \in T_0.
\end{array}
$$

Método de Kelley

- Supongamos dado un conjunto finito $S = T_0 \subset T$, cuyos conjuntos de nivel son no vacíos y acotados.
- Consideremos el problema de programación lineal

$$\begin{array}{lll} (\mathsf{P}_0) & \text{mín} & c'x \\ & s.a & a_t'x \geq b_t \quad , \ t \in T_0. \end{array}$$

- Sea su conjunto factible el conjunto convexo poliedral $F_0 \supset F$.

Método de Kelley

- Supongamos dado un conjunto finito $S = T_0 \subset T$, cuyos conjuntos de nivel son no vacíos y acotados.
- Consideremos el problema de programación lineal

$$\begin{array}{lll} (\mathrm{P}_0) & \text{mín} & c'x \\ & s.a & a_t'x \geq b_t \quad , \ t \in T_0. \end{array}$$

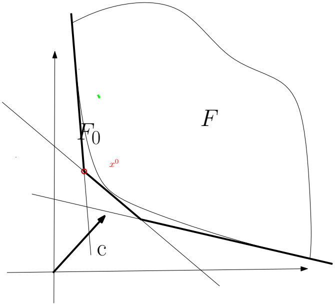
- Sea su conjunto factible el conjunto convexo poliedral $F_0 \supset F$.
- Sea x^0 una solución óptima de (P_0).

Método de Kelley

- Supongamos dado un conjunto finito $S = T_0 \subset T$, cuyos conjuntos de nivel son no vacíos y acotados.
- Consideremos el problema de programación lineal

$$\begin{array}{lll} (\mathsf{P}_0) & \text{mín} & c'x \\ & s.a & a_t'x \geq b_t \quad , \ t \in T_0. \end{array}$$

- Sea su conjunto factible el conjunto convexo poliedral $F_0 \supset F$.
- Sea x^0 una solución óptima de (P_0).



-

Método de Kelley

¿Cómo se construye el conjunto T_{r+1} a partir de T_r?

Método de Kelley

¿Cómo se construye el conjunto T_{r+1} a partir de T_r?

- Mediante hiperplanos que dejan en uno de sus semiespacios todo el conjunto factible y que dan lugar a ***cortes factibles***.

Método de Kelley

¿Cómo se construye el conjunto T_{r+1} a partir de T_r?

- Mediante hiperplanos que dejan en uno de sus semiespacios todo el conjunto factible y que dan lugar a ***cortes factibles***.
- Sea $g(t, x^r) = a_t' x^r - b_t$.

Método de Kelley

¿Cómo se construye el conjunto T_{r+1} a partir de T_r?

- Mediante hiperplanos que dejan en uno de sus semiespacios todo el conjunto factible y que dan lugar a ***cortes factibles***.
- Sea $g(t, x^r) = a_t' x^r - b_t$.
- Como $x^r \notin F$, existe algún $t \in T$ tal que $g(t, x^r) < 0$. Se trata de una restricción violada por x^r.

Método de Kelley

¿Cómo se construye el conjunto T_{r+1} a partir de T_r?

- Mediante hiperplanos que dejan en uno de sus semiespacios todo el conjunto factible y que dan lugar a ***cortes factibles***.
- Sea $g(t, x^r) = a_t' x^r - b_t$.
- Como $x^r \notin F$, existe algún $t \in T$ tal que $g(t, x^r) < 0$. Se trata de una restricción violada por x^r.
- Cualquier restricción violada sirve para construir el nuevo conjunto T_{r+1}.

Método de Kelley

¿Cómo se construye el conjunto T_{r+1} a partir de T_r?

- Mediante hiperplanos que dejan en uno de sus semiespacios todo el conjunto factible y que dan lugar a ***cortes factibles***.
- Sea $g(t, x^r) = a_t' x^r - b_t$.
- Como $x^r \notin F$, existe algún $t \in T$ tal que $g(t, x^r) < 0$. Se trata de una restricción violada por x^r.
- Cualquier restricción violada sirve para construir el nuevo conjunto T_{r+1}.
- En la práctica se elige una de las más violadas, es decir una de las que hacen $g(t, x^r) = a_t' x^r - b_t$ mínimo o próximo a ser mínimo.

Método de Kelley

¿Cómo se construye el conjunto T_{r+1} a partir de T_r?

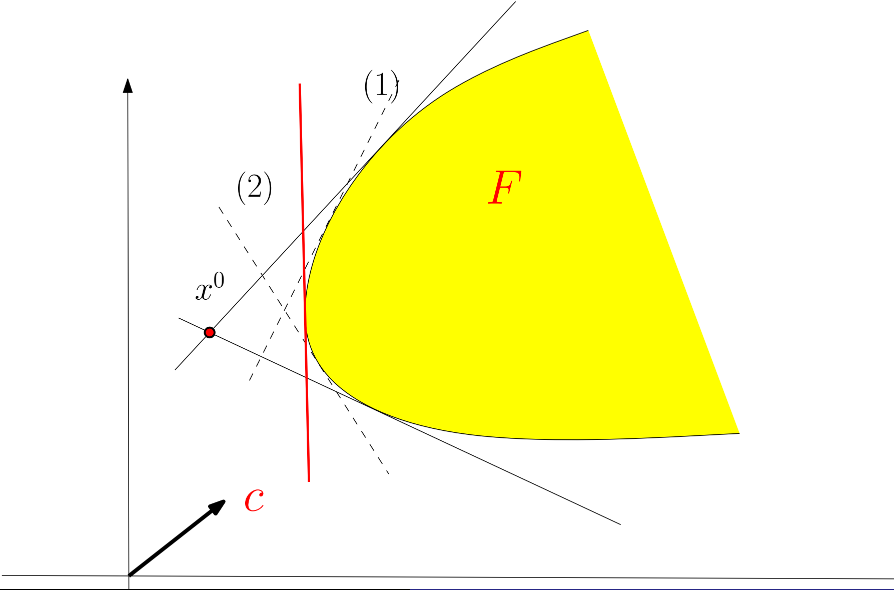
- Mediante hiperplanos que dejan en uno de sus semiespacios todo el conjunto factible y que dan lugar a ***cortes factibles***.
- Sea $g(t, x^r) = a_t' x^r - b_t$.
- Como $x^r \notin F$, existe algún $t \in T$ tal que $g(t, x^r) < 0$. Se trata de una restricción violada por x^r.
- Cualquier restricción violada sirve para construir el nuevo conjunto T_{r+1}.
- En la práctica se elige una de las más violadas, es decir una de las que hacen $g(t, x^r) = a_t' x^r - b_t$ mínimo o próximo a ser mínimo.
- Cuanto menor sea $g(t, x^r) = a_t' x^r - b_t$, más profundo es el corte, es decir, más nos aproximamos al conjunto factible del problema inicial.

Método de Kelley

¿Cómo se construye el conjunto T_{r+1} a partir de T_r?

- Mediante hiperplanos que dejan en uno de sus semiespacios todo el conjunto factible y que dan lugar a ***cortes factibles***.
- Sea $g(t, x^r) = a_t' x^r - b_t$.
- Como $x^r \notin F$, existe algún $t \in T$ tal que $g(t, x^r) < 0$. Se trata de una restricción violada por x^r.
- Cualquier restricción violada sirve para construir el nuevo conjunto T_{r+1}.
- En la práctica se elige una de las más violadas, es decir una de las que hacen $g(t, x^r) = a_t' x^r - b_t$ mínimo o próximo a ser mínimo.
- Cuanto menor sea $g(t, x^r) = a_t' x^r - b_t$, más profundo es el corte, es decir, más nos aproximamos al conjunto factible del problema inicial.
- Paramos cuando $\inf g(t, x^r) = a_t' x^r - b_t > -\varepsilon$.

Método de Kelley



Outline

Método de Elzinga-Moore: las condiciones

- Se parte de un polítopo F_r que contiene algún conjunto de nivel no vacío del problema inicial.

Método de Elzinga-Moore: las condiciones

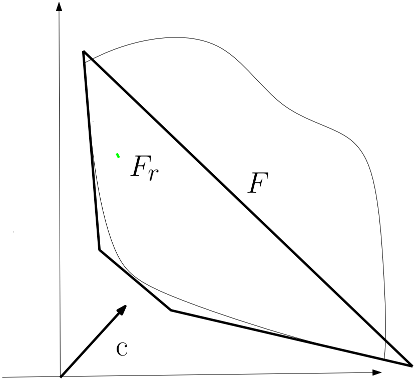
- Se parte de un polítopo F_r que contiene algún conjunto de nivel no vacío del problema inicial.
- Una condición suficiente para que tal polítopo exista es que F^* sea un conjunto no vacío y acotado.

Método de Elzinga-Moore: las condiciones

- Se parte de un polítopo F_r que contiene algún conjunto de nivel no vacío del problema inicial.
- Una condición suficiente para que tal polítopo exista es que F^* sea un conjunto no vacío y acotado.
- No se requiere que $F \subset F_r$.

Método de Elzinga-Moore: las condiciones

- Se parte de un polítopo F_r que contiene algún conjunto de nivel no vacío del problema inicial.
- Una condición suficiente para que tal polítopo exista es que F^* sea un conjunto no vacío y acotado.
- No se requiere que $F \subset F_r$.
-



Método de Elzinga-Moore

Dado F_r, se calcula el centro, x^r, de la mayor bola contenida en F_r.

Método de Elzinga-Moore

Dado F_r, se calcula el centro, x^r, de la mayor bola contenida en F_r.

- **¿Cómo se obtiene F_{r+1} a partir de F_r?**

Método de Elzinga-Moore

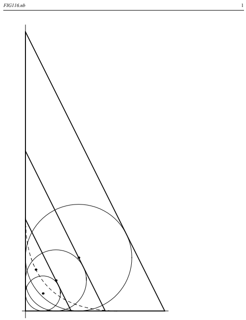
Dado F_r, se calcula el centro, x^r, de la mayor bola contenida en F_r.

- **¿Cómo se obtiene F_{r+1} a partir de F_r?**
- Dos casos pueden darse:

Método de Elzinga-Moore

Dado F_r, se calcula el centro, x^r, de la mayor bola contenida en F_r.

- **¿Cómo se obtiene F_{r+1} a partir de F_r?**
- Dos casos pueden darse:
    - Si $x^r \in F$, tomamos $F_{r+1} = \{x \in F_r \mid c'x \leq c'x^r\}$. Es lo que se conoce como un ***corte objetivo*** sobre F_r.

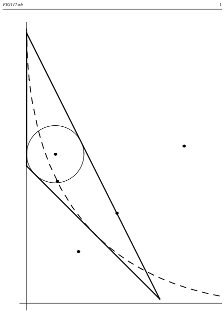


Método de Elzinga-Moore

- Si $x^r \notin F$, tomamos $F_{r+1} = \{x \in F_r \mid a'x \geq b\}$, donde $a'x \geq b$ es un corte factible.

Método de Elzinga-Moore

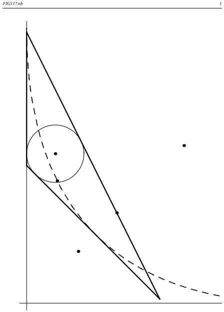
- Si $x^r \notin F$, tomamos $F_{r+1} = \{x \in F_r \mid a'x \geq b\}$, donde $a'x \geq b$ es un corte factible.



-

Método de Elzinga-Moore

- Si $x^r \notin F$, tomamos $F_{r+1} = \{x \in F_r \,|\, a'x \geq b\}$, donde $a'x \geq b$ es un corte factible.



-
- El algoritmo termina cuando el radio de la última bola obtenida es suficientemente pequeño.

Método simplex: nomenclatura y condiciones iniciales

- Sea $\Lambda = \left\{ \lambda \in \mathbb{R}_+^{(T)} \mid \sum_{t \in T} \lambda_t a_t = c \right\} \neq \emptyset$, con $c \neq 0_n$.

Método simplex: nomenclatura y condiciones iniciales

- Sea $\Lambda = \left\{ \lambda \in \mathbb{R}_+^{(T)} \mid \sum_{t \in T} \lambda_t a_t = c \right\} \neq \emptyset$, con $c \neq 0_n$.
- Definimos soporte de λ, $\operatorname{supp} \lambda = \{t \in T \mid \lambda_t \neq 0\}$.

Método simplex: nomenclatura y condiciones iniciales

- Sea $\Lambda = \left\{ \lambda \in \mathbb{R}_+^{(T)} \mid \sum_{t \in T} \lambda_t a_t = c \right\} \neq \emptyset$, con $c \neq 0_n$.
- Definimos soporte de λ, $\operatorname{supp} \lambda = \{t \in T \mid \lambda_t \neq 0\}$.
- Sea $\lambda \in \Lambda$ un punto extremo. Un conjunto $S \subset T$ se dice básico para el punto extremo λ si $\operatorname{supp} \lambda \subset S$ y $\{a_t,\ t \in S\}$ es una base de $\mathbb{R}^n$. Si $|\operatorname{supp} \lambda| > n$, podemos encontrar más de un conjunto básico asociado a λ y entonces se dice que λ es un punto extremo degenerado.

Método simplex: nomenclatura y condiciones iniciales

- Sea $\Lambda = \left\{ \lambda \in \mathbb{R}_+^{(T)} \mid \sum_{t \in T} \lambda_t a_t = c \right\} \neq \emptyset$, con $c \neq 0_n$.
- Definimos soporte de λ, $\operatorname{supp} \lambda = \{t \in T \mid \lambda_t \neq 0\}$.
- Sea $\lambda \in \Lambda$ un punto extremo. Un conjunto $S \subset T$ se dice básico para el punto extremo λ si $\operatorname{supp} \lambda \subset S$ y $\{a_t,\ t \in S\}$ es una base de $\mathbb{R}^n$. Si $|\operatorname{supp} \lambda| > n$, podemos encontrar más de un conjunto básico asociado a λ y entonces se dice que λ es un punto extremo degenerado.
- El algoritmo precisa para su arranque de un punto extremo de Λ o del correspondiente algoritmo de purificación para determinarlo.

Método simplex: nomenclatura y condiciones iniciales

- Sea $\Lambda = \left\{ \lambda \in \mathbb{R}_{+}^{(T)} \mid \sum_{t \in T} \lambda_t a_t = c \right\} \neq \emptyset$, con $c \neq 0_n$.
- Definimos soporte de λ, $\operatorname{supp} \lambda = \{t \in T \mid \lambda_t \neq 0\}$.
- Sea $\lambda \in \Lambda$ un punto extremo. Un conjunto $S \subset T$ se dice básico para el punto extremo λ si $\operatorname{supp} \lambda \subset S$ y $\{a_t,\ t \in S\}$ es una base de $\mathbb{R}^n$. Si $|\operatorname{supp} \lambda| > n$, podemos encontrar más de un conjunto básico asociado a λ y entonces se dice que λ es un punto extremo degenerado.
- El algoritmo precisa para su arranque de un punto extremo de Λ o del correspondiente algoritmo de purificación para determinarlo.
- Construye dos sucesiones, $\{x^r\} \subset \overline{F}$ y $\{\lambda^r\}$, puntos extremos de Λ.

Método simplex: nomenclatura y condiciones iniciales

- Sea $\Lambda = \left\{ \lambda \in \mathbb{R}_+^{(T)} \mid \sum_{t \in T} \lambda_t a_t = c \right\} \neq \emptyset$, con $c \neq 0_n$.
- Definimos soporte de λ, $\operatorname{supp} \lambda = \{t \in T \mid \lambda_t \neq 0\}$.
- Sea $\lambda \in \Lambda$ un punto extremo. Un conjunto $S \subset T$ se dice básico para el punto extremo λ si $\operatorname{supp} \lambda \subset S$ y $\{a_t,\ t \in S\}$ es una base de $\mathbb{R}^n$. Si $|\operatorname{supp} \lambda| > n$, podemos encontrar más de un conjunto básico asociado a λ y entonces se dice que λ es un punto extremo degenerado.
- El algoritmo precisa para su arranque de un punto extremo de Λ o del correspondiente algoritmo de purificación para determinarlo.
- Construye dos sucesiones, $\{x^r\} \subset \overline{F}$ y $\{\lambda^r\}$, puntos extremos de Λ.
- En ausencia de degeneración en la sucesión λ^r el algoritmo converge o nos informa de la no acotación de (D) y, consecuentemente, de la inconsistencia de (P).

Método simplex: el algoritmo

Suponemos que λ^0 es un punto extremo no degenerado de Λ y que $S_0 \subset T$ es el único conjunto básico asociado a λ^0.

Método simplex: el algoritmo

Suponemos que λ^0 es un punto extremo no degenerado de Λ y que $S_0 \subset T$ es el único conjunto básico asociado a λ^0.

- **Paso 1**
  Sea x^r la única solución del sistema $\{a_t' x = b_t,\ t \in S_r\}$. Sea $s_r = \text{ínf}_{t \in T}\, g(t, x^r)$.
  Si $s_r \geq 0$, parar. En otro caso continuar con el **paso 2**.

Método simplex: el algoritmo

Suponemos que λ^0 es un punto extremo no degenerado de Λ y que $S_0 \subset T$ es el único conjunto básico asociado a λ^0.

- **Paso 1**
  Sea x^r la única solución del sistema $\{a_t' x = b_t,\ t \in S_r\}$. Sea $s_r = \text{ínf}_{t \in T}\, g(t, x^r)$.
  Si $s_r \geq 0$, parar. En otro caso continuar con el **paso 2**.
- **Paso 2**
  Sea t_r tal que $g(t_r, x^r) < 0$. Resolvemos el sistema $\{a_t' \alpha^r = a_{t_r},\ t \in S_r\}$.

Método simplex: el algoritmo

Suponemos que λ^0 es un punto extremo no degenerado de Λ y que $S_0 \subset T$ es el único conjunto básico asociado a λ^0.

- **Paso 1**
  Sea x^r la única solución del sistema $\{a_t' x = b_t,\ t \in S_r\}$. Sea $s_r = \text{ínf}_{t \in T}\, g(t, x^r)$.
  Si $s_r \geq 0$, parar. En otro caso continuar con el **paso 2**.
- **Paso 2**
  Sea t_r tal que $g(t_r, x^r) < 0$. Resolvemos el sistema $\{a_t' \alpha^r = a_{t_r},\ t \in S_r\}$.
  - Si $\alpha^r \in \mathbb{R}_-^{(T)}$, parar. $\rho^r = \delta^r - \alpha^r$, siendo δ^r la función característica de $\{t_r\}$, es una dirección de recesión ascendente. $v(D)$ es no acotado.

Método simplex: el algoritmo

Suponemos que λ^0 es un punto extremo no degenerado de Λ y que $S_0 \subset T$ es el único conjunto básico asociado a λ^0.

- **Paso 1**
  Sea x^r la única solución del sistema $\{a_t' x = b_t,\ t \in S_r\}$. Sea $s_r = \inf_{t \in T} g(t, x^r)$.
  Si $s_r \geq 0$, parar. En otro caso continuar con el **paso 2**.
- **Paso 2**
  Sea t_r tal que $g(t_r, x^r) < 0$. Resolvemos el sistema $\{a_t' \alpha^r = a_{t_r},\ t \in S_r\}$.
    - Si $\alpha^r \in \mathbb{R}_-^{(T)}$, parar. $\rho^r = \delta^r - \alpha^r$, siendo δ^r la función característica de $\{t_r\}$, es una dirección de recesión ascendente. $v(D)$ es no acotado.
    - Calcular $\mu_r = \dfrac{\lambda_s^r}{\alpha_s^r} = \min \left\{ \dfrac{\lambda_t^r}{\alpha_t^r} \mid \alpha_t^r > 0 \right\}$.
      Tome $\lambda^{r+1} = \lambda^r + \mu_r \rho^r$.
      Calcule el conjunto básico S_{r+1} asociado a λ^{r+1} y vuelva al **paso 1**