

REVISIÓN CRÍTICA DE LA UTILIZACIÓN DEL MÉTODO DEL GRADIENTE CONJUGADO Y EXTENSIONES EN LA MINIMIZACIÓN DE FUNCIONES NOLINEALES CON CONDICIONES

L. F. ESCUDERO
CENTRO DE INVESTIGACIÓN UAM-IBM

En este trabajo se efectúa una revisión de la aplicación del método del gradiente conjugado (diseñado para resolver sistemas de ecuaciones lineales) a la optimización de funciones no-lineales (no cuadráticas) en el conjunto factible definido por condiciones lineales. Se describen las desventajas de su utilización; se sugiere su sustitución por algoritmos que, precisando parecida capacidad de memoria, no adolecen de sus inconvenientes. Los resultados computacionales que se acompañan, avalan esta sustitución.

Keywords: REINITIALIZATION, QUASI-NEWTON METHODS WITH LIMITED MEMORY, PRECONDITIONING.

1. INTRODUCCION Y MOTIVACION.

Un problema de programación no lineal sin condiciones (PNSC) consiste en:

$$\min\{F(X) \mid X \in R^n\} \quad (1.1)$$

tal que $F(X)$ es una función no lineal con las siguientes propiedades: $F(X)$ es continua y, al menos, dos veces diferenciable, y los niveles $L(X) = \{X: F(X) \leq F(\bar{X})\}$ están finitamente acotados y cerrados. El objetivo consiste en obtener un punto mínimo local débil, sea \bar{X} ; es decir, un punto para el cual $\exists \delta > 0$ tal que $F(\bar{X}) \leq F(X) \forall X: \|X - \bar{X}\| \leq \delta$. Sea $g(\bar{X}) \equiv \bar{g}$ el gradiente de $F(X)$ evaluado en \bar{X} y $G(\bar{X}) \equiv \bar{G}$ su matriz Hessiana. Una condición suficiente para \bar{X} es que $g(\bar{X}) = 0$ y $d^t \bar{G} d > 0 \forall d = X - \bar{X}$. Es preciso notar que $g(\bar{X}) = 0$ y $d^t \bar{G} d \geq 0 \forall d = X - \bar{X}$ sólo es condición necesaria.

El método más clásico para obtener \bar{X} es el método Newton que, dada una estimación inicial, sea $X^{(0)}$, obtiene una secuencia de direcciones de búsqueda $\{d^{(k)}\}$ tal que para la iteración k resuelve el sistema

$$G^{(k-1)} d^{(k)} = -g^{(k-1)} \quad (1.2)$$

tal que obtiene la estimación $X^{(k)}$,

$$X^{(k)} = X^{(k-1)} + \alpha^{(k)} d^{(k)} \quad (1.3)$$

donde $\alpha^{(k)}$ es la amplitud de paso en la iteración k , tal que minimiza $F(X)$ en la dirección $d^{(k)}$,

$$\alpha^{(k)} = \arg \min\{F(X^{(k-1)} + \alpha d^{(k)}): \alpha > 0\} \quad (1.4)$$

La resolución de (1.4), denominada búsqueda lineal exacta, aunque no es tan compleja como minimizar $F(X)$ todavía tiene una gran dificultad; alternativamente, se utilizan métodos de búsqueda lineal aproximada en los que para una dirección de búsqueda $d^{(k)}$ suficientemente descendente

$$(-g^{(k-1)t} d^{(k)}) / (\|g^{(k-1)}\|_2 \|d^{(k)}\|_2) \geq \sigma \quad (1.5)$$

donde σ es una tolerancia positiva que evita que el gradiente y la dirección sean cuasi-ortogonales, el algoritmo es globalmente convergente si $\alpha^{(k)}$ reduce suficientemente $F(X)$ y, en concreto, si se cumplen /5/ las condiciones GPW:

$$(i) |g(X^{(k-1)} + \alpha^{(k)} d^{(k)})^t d^{(k)}| \leq -ng(X^{(k-1)})^t d^{(k)} \quad (1.6)$$

o, alternativamente /15/, si el gradiente es difícil ó costoso de calcular,

$$\frac{|F(X^{(k-1)} + \alpha^{(k)} d^{(k)}) - F(X^{(k-1)} + \nu d^{(k)})|}{\alpha^{(k)} - \nu} \leq -\eta g(X^{(k-1)})^t d^{(k)} \quad (1.7)$$

donde ν es una scalar tal que $0 < \nu < \alpha^{(k)}$, y $0 < \eta \leq 1$.

Es preciso notar que $\alpha^{(k)}$ satisface (1.4) - para $\eta=0$.

$$(ii) F(X^{(k-1)}) - F(X^{(k-1)} + \alpha^{(k)} d^{(k)}) \geq \mu \alpha^{(k)} g(X^{(k-1)})^t d^{(k)} \quad (1.8)$$

para $0 < \mu \leq 0.5$. Valores típicos, $\eta=0.9$ y $\mu=10^{-4}$ tal que si $\mu \leq \eta$, $\exists \alpha$ que cumple las condiciones GPW. Condiciones (1.6) y (1.7) evitan que $\alpha^{(k)}$ tome un valor excesivamente pequeño; condición (1.8) evita que $\alpha^{(k)}$ tome un valor excesivamente grande.

El método Newton es importante como standard con el que comparar métodos alternativos. -- Sus ventajas e inconvenientes están suficientemente documentados en la literatura. Brevemente, su principal ventaja consiste en que es local y cuadráticamente convergente (ver e.g. en /12/ la definición de los conceptos aquí utilizados). Sus inconvenientes principales son: no es globalmente convergente, no existe solución en el sistema (1.2) si $G^{(k-1)}$ es singular, $G^{(k-1)}$ puede no ser positiva definida (pd) en problemas no convexos (y, por tanto, $d^{(k)}$ puede no ser descendente) y, en cualquier caso, es preciso evaluar la matriz Hessiana y resolver el sistema n-dimensional (1.2) en cada iteración k.

Los principales métodos alternativos al método Newton son los siguientes (ver en la Sec. 5 los resultados computacionales):

(a) Probablemente, los métodos Quasi-Newton sean los métodos más satisfactorios para obtener \bar{x} en problemas de escala reducida (sea $n \leq 300$), ver en /6/ una panorámica muy actualizada. Eliminan los inconvenientes teóricos del método Newton y su convergencia es superlineal. Sin embargo, requieren el almacenamiento de una matriz simétrica nxn (la aproximación $B^{(k-1)}$ de la matriz $G^{(k-1)}$) y la resolución del sistema (1.2), donde G es sustituida por B. Para problemas de gran escala, su utilización es prohibitiva.

(b) En problemas super-escala (sea, $300 < n \leq 600$), el método Newton Truncado /4/ todavía produce resultados satisfactorios; su convergencia es superlineal, elimina los inconvenientes teóricos del método Newton y no requiere obtener y, por tanto, almacenar la matriz B (sólo necesita el producto de esta matriz por un vector, sea Bd tal que este producto se puede aproximar por diferencias finitas) y, finalmente, sólo requiere resolver (1.2) de una forma exacta en el límite.

Los métodos Quasi-Newton se utilizan con resultados satisfactorios en programación no lineal con condiciones (PNCC) si el número de variables superbásicas n_g (ver en /14/ su definición) es tal que $n_g \leq 300$; ver /7/ para el caso de condiciones no lineales y /8/ para el caso de condiciones lineales. Para $300 < n_g \leq 600$, el método Newton Truncado también produce resultados satisfactorios para el caso de condiciones lineales; ver /9/.

El principal inconveniente del método Newton Truncado estriba en que, en cada iteración interna requerida para resolver el sistema (1.2) es preciso obtener una evaluación adicional del gradiente en el caso muy frecuente de tener que aproximar el producto Bd ; este inconveniente no es grave para problemas super-escala, pero para problemas supra-escala ($n > 600$) es preciso utilizar métodos alternativos tal que continuen siendo, al menos, localmente convergentes, aunque la convergencia ya no sea superlineal.

(c) Uno de los métodos hoy en día más utilizados en problemas supra-escala es el método del gradiente conjugado /13/, /16/. Su ventaja principal consiste en que no es necesario almacenar ninguna matriz, ni estimar el producto Bd , ya que la nueva dirección de búsqueda se obtiene en base a combinaciones lineales del gradiente evaluado en la iteración anterior y de las direcciones obtenidas en iteraciones previas. Su principal inconveniente consiste en que las direcciones de búsqueda obtenidas, en su aplicación a la optimización de funciones no lineales generales en PNCC, no son forzosamente descendentes salvo que la búsqueda lineal en la iteración previa sea -

exacta; este inconveniente se agrava en el caso de PNCC, dado que el margen permitido para la amplitud de paso generalmente es limitado (ver sec. 2). Otra importante desventaja en la utilización del gradiente conjugado en PNCC consiste en que dado un cambio de base (i.e., una variable básica alcanza uno de sus límites y es sustituida por una variable normalmente superbásica, ver /14/), la información obtenida en iteraciones anteriores no puede ser utilizada en la iteración correspondiente; ello fuerza a una convergencia más lenta.

Este trabajo está organizado como sigue. En las sec. 2 y 3 se revisan los algoritmos más utilizados basados en el gradiente conjugado, efectuando una revisión crítica de los mismos en su utilización en PNCC. En la sec. 4 se describen algoritmos alternativos al gradiente conjugado para resolver problemas en gran escala. En la sec. 5 se efectúa una evaluación final de los algoritmos del gradiente conjugado basada en experimentación numérica con casos reales tal que, desde un punto de vista computacional, se sugiere su no utilización en programación no lineal con condiciones. Asimismo, en la sec. 5 se recoge la metodología básica seguida para la aplicación del método del gradiente conjugado y extensiones a la resolución de problemas de PNCC.

En los trabajos /10/, /11/ que son continuación de éste se describe detalladamente en el primero, un nuevo algoritmo para minimizar $F(X)$ que, basado en las conclusiones de éste, no adolece de sus inconvenientes en PNCC; y, en el segundo, se presenta el análisis computacional.

2. ALGORITMO DEL GRADIENTE CONJUGADO TRADICIONAL.

Sea $x^{(0)}$ un punto inicial y $g^{(0)}$ su gradiente. Sea $k \geq 1$ la iteración considerada, tal que requiere el gradiente $g^{(k-1)}$ obtenido al finalizar la iteración anterior $k-1$. La dirección de búsqueda $d^{(k)}$ es tal que

$$d^{(1)} = -g^{(0)} \tag{2.1}$$

$$d^{(k)} = -g^{(k-1)} + \beta^{(k-1)} d^{(k-1)} \text{ para } k \geq 2 \tag{2.2}$$

donde /12/

$$\beta^{(k-1)} = (\|g^{(k-1)}\|_2^2) / (\|g^{(k-2)}\|_2^2) \tag{2.3}$$

tal que el nuevo punto $x^{(k)}$ al finalizar la iteración k será (1.3).

Si $F(X)$ es cuadrática, es decir

$$F(X) = c^t X + 1/2 X^t G X \tag{2.4}$$

donde G es una matriz simétrica positiva definida (spd), las direcciones (2.2) son idénticas a las obtenidas con el método del gradiente conjugado introducido en /16/ para resolver el sistema $GX = -c$, tal que si $\alpha^{(k)}$ satisface (1.4),

$$\alpha^{(k)} = -(g^{(k-1)t} d^{(k)}) / (d^{(k)t} G d^{(k)}) \text{ para } k \geq 1 \tag{2.5}$$

Los gradientes son mutuamente ortogonales $(g^{(i)t} g^{(j)}) = 0 \text{ } i \neq j$, las direcciones están conjugadas $(d^{(i)t} G d^{(j)}) = 0 \text{ } i \neq j$, y el algoritmo tiene la propiedad denominada terminación cuadrática (el número teórico de iteraciones para obtener \tilde{x} es $m_G \leq n$, donde m_G es el número de valores propios diferentes en G).

Aunque computacionalmente (2.3) es la más estable, hay diversas fórmulas que son teóricamente equivalentes. Para su obtención, es preciso reconocer que la conjugancia de las direcciones es equivalente a la ortogonalidad,

$$y^{(i)t} d^{(j)} = 0 \text{ } i \neq j \tag{2.6}$$

donde $y^{(i)} \equiv g^{(i)} - g^{(i-1)}$, ya que

$$y^{(i)} = g^{(i)} - g^{(i-1)} = G(x^{(i)} - x^{(i-1)}) = \alpha^{(i)} G d^{(i)} \tag{2.7}$$

$$y^{(i)t} d^{(j)} = \alpha^{(i)} d^{(j)t} G d^{(i)} = 0 \text{ } i \neq j \tag{2.8}$$

Consecuentemente, la fórmula más obvia para $\beta^{(k-1)}$ se obtiene premultiplicando (2.2) por $y^{(k-1)}$ y exigiendo que $\beta^{(k-1)}$ satisfaga la ortogonalidad (2.8) de donde

$$\beta^{(k-1)} = (y^{(k-1)t} g^{(k-1)}) / (y^{(k-1)t} d^{(k-1)}) \tag{2.9}$$

Es preciso notar que para una búsqueda lineal exacta, $g^{(j)t} d^{(j)} = 0$ tal que, utilizando la formulación de $d^{(k)}$ (2.2), resulta $y^{(k-1)t} d^{(k-1)} = \|g^{(k-2)}\|_2^2$ y, por tanto, la nueva fórmula /20/ será

$$\beta^{(k-1)} = (y^{(k-1)t} g^{(k-1)}) / (\|g^{(k-2)}\|_2^2) \quad (2.10)$$

Es preciso notar que (2.9) exige solamente la ortogonalidad (2.8), y (2.10) exige dicha ortogonalidad y una búsqueda lineal exacta; no precisan por tanto que F(X) sea cuadrática tal que los gradientes sean ortogonales, y las direcciones sean conjugadas. En cambio (2.3) exige adicionalmente que F(X) sea cuadrática y, por tanto, $y^{(k-1)t} g^{(k-1)} = \|g^{(k-1)}\|_2^2$.

Cuando F(X) es una función no lineal general, las fórmulas (2.2), (2.9) y (2.10) ya no son equivalentes, y la propiedad de terminación cuadrática ya no está garantizada; en este caso, sugerimos la utilización de (2.9). Tradicionalmente, después de cada ciclo de n iteraciones, se sustituye la dirección (2.2) por la dirección localmente "más descendente" tal que

$$d^{(k)} = -g^{(k-1)} \quad \text{para las iteraciones} \\ k=1, n+1, \dots, in+1 \quad (2.11)$$

La estrategia de reinicialización (2.11) se basa en la criticable presunción de que la reducción en F(X) será mayor que si se utiliza la fórmula (2.2).

Las condiciones GPW, aplicadas a $\alpha^{(k-1)}$, no garantizan que $d^{(k)}$ sea descendente para F(X) no lineal general y búsqueda lineal aproximada; por tanto, es preciso añadir una condición adicional si $\eta > 0$, ya que si se premultiplica (2.2) por $g^{(k-1)}$ resulta

$$g^{(k-1)t} d^{(k)} = -g^{(k-1)t} g^{(k-1)} + \beta^{(k-1)} g^{(k-1)t} g^{(k-1)} \quad (2.12)$$

Para $\eta=0$, $g^{(k-1)t} d^{(k-1)} = 0$ y $d^{(k)}$ es descendente; para $\eta > 0$, $X^{(k-1)}$ puede ser tal que (2.12) sea no negativo. Sea σ un escalar positivo suficientemente pequeño, y $\bar{g}^{(k-1)}$, $\bar{d}^{(k)}$ y $\bar{\beta}^{(k-1)}$ los valores relativos al punto $X^{(k-2)} + \bar{\alpha}^{(k-1)} d^{(k-1)}$ tal que $\bar{\alpha}^{(k-1)}$ es un valor posible de $\alpha^{(k-1)}$; se admite $\bar{\alpha}^{(k-1)}$ como la amplitud de paso en la iteración k-1 si, además de cumplir las condiciones GPW, se satisface

$$-\bar{g}^{(k-1)t} \bar{d}^{(k)} \geq \sigma \| \bar{g}^{(k-1)} \|_2 \| \bar{d}^{(k)} \|_2 \quad (2.13)$$

Es preciso notar que no es necesario obtener

explícitamente $\bar{d}^{(k)}$, ya que $\bar{g}^{(k-1)t} \bar{d}^{(k)}$ y $\| \bar{d}^{(k)} \|_2$ se obtienen en base a $\| \bar{g}^{(k-1)} \|_2 \bar{\beta}^{(k-1)}$ (2.9), $\bar{g}^{(k-1)t} d^{(k-1)}$ y $\| d^{(k-1)} \|_2$; ver (2.13). Valor típico $\sigma=10^{-4}$.

Ahora bien, en PNCC se exige que $\alpha^{(k)} \leq \alpha_M^{(k)}$, donde $\alpha_M^{(k)} > 0$ es el límite máximo permisible para mantener la factibilidad de la nueva solución /8/. Entonces, es posible que no exista ningún valor de α que, satisfaciendo las condiciones GPW, cumpla la condición (2.13), y además sea factible. En este caso, una alternativa consiste en reinicializar el algoritmo con $d^{(k)} = -g^{(k-1)}$ lo que provocaría una convergencia más lenta.

Denominemos algoritmo del gradiente conjugado tradicional (TCG) al algoritmo del gradiente conjugado con (2.2), (2.9), condiciones GPW y (2.13), y la reinicialización (2.11) después de un ciclo de n iteraciones o cuando $\alpha^{(k)}$ no satisfaga dichas condiciones. Dicho algoritmo tiene una convergencia cuasi-lineal tal que requiere entre n y 5n iteraciones para obtener \bar{x} , siendo típico - 2n.

3. ALGORITMO DEL GRADIENTE CONJUGADO CON DIRECCION DE REINICIALIZACION.

Aunque F(X) es monotonamente decreciente en el algoritmo TGC, la reducción en F(X) es frecuentemente pobre cuando se utiliza (2.11) en comparación con el caso en el que la reinicialización no hubiese tenido lugar. Parece más interesante reinicializar $d^{(k)}$ con $d^{(k-1)}$. Ahora bien, una dirección inicial arbitraria no garantiza la terminación cuadrática para F(X) cuadrática ya que las direcciones no están conjugadas. Se demuestra /1/ que dada una dirección inicial arbitraria $d^{(1)}$, los vectores $d^{(1)}$ y

$$d^{(k)} = -g^{(k-1)} + \beta^{(k-1)} d^{(k-1)} + \gamma^{(k-1)} d^{(1)} \quad \text{para } k \geq 2 \quad (3.1)$$

con $\beta^{(k-1)}$ (2.9) y, análogamente, $\gamma^{(k-1)} = y^{(1)t} g^{(k-1)} / y^{(1)t} d^{(1)}$, están mutuamente conjugados. Extendiendo esta fórmula al caso general no lineal, resulta

$$d^{(1)} = -g^{(0)} \quad (3.2)$$

$$d^{(t+1)} = -g^{(t)} + \beta^{(t)} d^{(t)} \quad \text{para } t=1, n, 2n, \dots, in \quad (3.3)$$

$$d^{(k)} = -g^{(k-1)} + \beta^{(k-1)} d^{(k-1)} + \gamma^{(k-1)} d^{(t)} \quad (3.4)$$

para $k=3, \dots, n, n+2, \dots, 2n, \dots, in+2, \dots, (i+1)n$, donde $\beta^{(t)}$ y $\beta^{(k-1)}$ se expresarán según (2.9) y $\gamma^{(k-1)} = \frac{y^{(t)T} g^{(k-1)}}{y^{(t)T} d^{(t)}}$; $d^{(t)}$ es la dirección de reinicialización del nuevo ciclo $i+1$, y es la última dirección obtenida en el ciclo i .

Se obtienen mejores resultados si $\beta^{(t)}$ se sigue obteniendo con (2.9), pero $\beta^{(k-1)}$ y $\gamma^{(k-1)}$ son la solución del sistema lineal -

$$\begin{pmatrix} y^{(k-1)T} d^{(k-1)} & y^{(k-1)T} d^{(t)} \\ y^{(t)T} d^{(k-1)} & y^{(t)T} d^{(t)} \end{pmatrix} \begin{pmatrix} \beta^{(k-1)} \\ \gamma^{(k-1)} \end{pmatrix} = \begin{pmatrix} y^{(k-1)T} g^{(k-1)} \\ y^{(t)T} g^{(k-1)} \end{pmatrix} \quad (3.5)$$

De esta forma se asegura que $d^{(k)}$ es ortogonal a $y^{(k-1)}$ e $y^{(t)}$, aunque $F(X)$ no sea cuadrática. El sistema (3.5) se resuelve por sustitución simple ya que $y^{(t)T} d^{(k-1)} = 0$ debido a la forma en que se han obtenido $\beta^{(k-2)}$ para $k=3, n+2, \dots, in+2$, y $\beta^{(k-2)}$ y $\gamma^{(k-2)}$ para $k=4, \dots, n, n+3, \dots, 2n, \dots, in+3, \dots, (i+1)n$. Por tanto, $\gamma^{(k-1)}$ no resulta modificado por utilizar (3.5) y en $\beta^{(k-1)}$ se introduce un término corrector.

Se sugiere inicializar un nuevo ciclo $i+1$ con $d^{(k)}$ (ver más abajo) si se satisface alguna de las siguientes condiciones:

(i) Se han obtenido n direcciones de búsqueda en el ciclo anterior.

$$(ii) \quad |g^{(k)T} g^{(k-1)}| > \rho_g \|g^{(k)}\|_2^2 \quad (3.6)$$

donde ρ_g debe ser positiva; valor típico, $\rho_g = 0.2$. Su motivación [21] consiste en la ortogonalidad de los gradientes para $F(X)$ cuadrática y búsqueda lineal exacta.

$$(iii) \quad -g^{(k-1)T} d^{(k)} < \rho_d \|g^{(k-1)}\|_2^2 \quad (3.7)$$

donde ρ_d debe ser positivo; valor típico, $\rho_d = 0.8$. En este caso se supone que $d^{(k)}$ no es suficiente descendente

$$(iv) \quad F(X^{(k-1)} + \alpha^{(k)} d^{(k)}) < F(X^{(k-1)} + \alpha^{(k)} d^{(k-1)}) \quad (3.8)$$

donde $d^{(k)}$ se ha obtenido utilizando (2.2) - (2.3), y $\alpha^{(k)}$ es la correspondiente amplitud de paso que debe satisfacer las condiciones GPW y (2.13) para $\alpha^{(k)} \leq \alpha_M^{(k)}$.

Es preciso notar que $d^{(k)} = -g^{(k-1)} + \beta^{(k-1)} d^{(k-1)}$ debe ser una dirección descendente; por tanto si no se satisface la condición (3.2) para $d^{(k)}$, se reinicializa $d^{(k)} := -g^{(k-1)}$.

Tal como se ha indicado, el número de iteraciones para obtener \bar{X} si $F(X)$ es cuadrática, G es spd y las direcciones están conjugadas es $m_G \leq n$, donde m_G es el número de valores propios diferentes en G . Por tanto la convergencia será superior si se sustituye el problema original por otro equivalente en el que para la nueva matriz, sea R resulta que $m_R < m_G$. El concepto de preconditionamiento [2] consiste en obtener dicha transformación. La matriz preconditionante W que mejores resultados nos ha dado al optimizar una función no lineal general (equilibrado precisión, tiempo de cálculo y memoria requerida) es la aproximación diagonal $\bar{B}^{(k-1)}$ de la actualización BFGS Quasi-Newton de la matriz Hessiana $G^{(k-1)}$, tal que $\frac{1}{\bar{B}^{(0)T}} I = 1$ y

$$\bar{B}_\ell^{(k)} = \bar{B}_\ell^{(k-1)} + \frac{1}{y^{(k)T} p^{(k)}} y_\ell^{(k)2} + \frac{1}{g^{(k-1)T} d^{(k)}} g_\ell^{(k-1)2} \quad (3.9)$$

para $\ell=1, \dots, n$ y $k \geq 1$.

Las operaciones a efectuar en la iteración $k \geq 1$ de los algoritmos del gradiente conjugado tradicional preconditionado PTCG y del gradiente conjugado con reinicialización PRCG son las siguientes:

Algoritmo PTCG

$$\text{Obtener } z^{(k-1)} : Wz^{(k-1)} = -g^{(k-1)} \quad (3.10)$$

$$\text{Asignar } d^{(k)} = -z^{(k-1)} + \beta^{(k-1)} d^{(k-1)}$$

$$\text{donde } \beta^{(k-1)} = 0 \text{ para } k=1, \text{ y} \quad (3.11)$$

$$\beta^{(k)} = \frac{y^{(k-1)T} z^{(k-1)}}{y^{(k-1)T} d^{(k-1)}} \text{ para } k > 1$$

$$\text{Asignar } X^{(k)} = X^{(k-1)} + \alpha^{(k)} d^{(k)}$$

Algoritmo PRCG

Asignar $d^{(k)} = -z^{(k-1)} + \beta^{(k-1)} d^{(k-1)} + \gamma^{(k-1)} d^{(k)}$

Asignar $x^{(k)} = x^{(k-1)} + \alpha^{(k)} d^{(k)}$

donde $z^{(k-1)}$ se obtiene en (3.10), $\beta^{(t)}$ se obtiene como en (3.11) y $\beta^{(k-1)}$ y $\gamma^{(k-1)}$ se obtienen en (3.5) donde g es sustituido por z .

4. METODOS QUASI-NEWTON CON MEMORIA LIMITADA.

La fórmula BFGS para la actualización de la matrix Hessiana $G^{(k-1)}$ está considerada como una de las formulaciones más estables de la familia Quasi-Newton /6/.

$$H^{(k)} = H^{(k-1)} - \frac{p^{(k)} y^{(k)t} H^{(k-1)} + H^{(k-1)} y^{(k)} p^{(k)t}}{p^{(k)t} y^{(k)}} + \left(1 + \frac{y^{(k)t} H^{(k-1)} y^{(k)}}{p^{(k)t} y^{(k)}} \right) \frac{p^{(k)} p^{(k)t}}{p^{(k)t} y^{(k)}} \quad (4.1)$$

La matrix $H^{(k)}$ es spd si $H^{(k-1)}$ también lo es y si $p^{(k)t} y^{(k)} > 0$; facilmente se puede observar que las condiciones GPW satisfacen la última condición. Por tanto, si (a) $H^{(0)}$ es spd (se utiliza normalmente una matrix diagonal y, la mayoría de las veces, la matrix identidad), (b) se utiliza la formulación (4.1), y (c) la amplitud de paso $\alpha^{(k)}$ se obtiene mediante una búsqueda lineal aproximada tal que se satisfagan las condiciones GPW, el algoritmo produce direcciones de búsqueda monotónicamente descendentes, al menos teóricamente, -- con una convergencia superlineal (ver e.g. -- /12/, /14/). Ahora bien, en problemas de gran escala ($n > 300$), su utilización es prohibitiva ya que requiere almacenar $n(n+1)/2$ elementos y resolver un sistema de n ecuaciones.

Se demuestra /17/ que el algoritmo basado en la actualización BFGS (4.1) y en una búsqueda lineal exacta produce las mismas direcciones de búsqueda que el algoritmo PTCG, si en éste, en vez de utilizar una matrix preconditionante fija W , se actualiza esta matrix con cualquier formula de la subfamilia de métodos Quasi-Newton, denominada familia Broyden (ver e.g. /6/). Esta interpretación motiva la utilización de algoritmos que mejoren el ritmo de convergencia de los algoritmos basados en el gradiente conjugado y, aunque exijan mayor

capacidad de memoria en un ordenador, no adolezcan del inconveniente de los métodos Quasi-Newton: memoria para almacenar una matrix simétrica n -dimensional y resolución del correspondiente sistema de n -ecuaciones; estos algoritmos (ver /2/, /3/10/, /1 /, /18/, /19/, -- entre otros) se denominan métodos Quasi-Newton con Memoria Limitada (LSQN) tal que sólo se utilizan las últimas, sea m iteraciones -- para la obtención (no explícita) de la aproximación BFGS (4.1).

La formulación $H^{(k)}$ (4.1) es una función en $(H^{(k-1)}, p^{(k)}, y^{(k)})$ y por tanto, $H^{(k)} = f(H^{(0)}, \{p^{(j)}\}, \{y^{(j)}\})$ tal que la dirección Quasi-Newton de búsqueda $d^{(k)} = -H^{(k-1)} g^{(k-1)}$ puede expresarse como una combinación lineal de vectores obtenidos en las iteraciones anteriores: $x^{(j)}$ y $g^{(j)}$ para $j=0,1,2,\dots,k-1$, -- sin necesitar almacenar ninguna matrix. Los métodos LSQN utilizan sólo las m últimas iteraciones, tal que

$$d^{(k)} = -H^{(k-1)} g^{(k-1)}$$

$$H^{(k-1)} g^{(k-1)} = c1\{H^{(k-2)} g^{(k-1)}, H^{(k-2)} y^{(k-1)}, p^{(k-1)}, \dots\}$$

$$H^{(k-m+1)} g^{(k-1)} = c1\{H^{(k-m)} g^{(k-1)}, H^{(k-m)} y^{(k-m+1)}, p^{(k-m+1)}\} \quad (4.2)$$

$$H^{(k-m)} g^{(k-1)} = c1\{H^{(k-m-1)} g^{(k-1)}, H^{(k-m-1)} y^{(k-m)}, p^{(k-m)}\}$$

donde $c1$ es la abreviatura de "combinación lineal" y $H^{(k-m-1)} = W^{-1}$. Normalmente, $W=I$; en la Sec. 5 se recogen los resultados obtenidos utilizando la matrix diagonal $\bar{B}^{(k-1)}$ -- (3.9). Es preciso notar que $H^{(j-1)} y^{(j)}$, para $j=k-1,\dots,k-m-1$ se obtiene de forma análoga al procedimiento (4.2) utilizado para obtener $H^{(j-1)} g^{(k-1)}$.

Se puede observar que los métodos LSQN no requieren una gran capacidad de memoria para valores razonables de m ; en la Sec. 5 se recogen los resultados obtenidos para $m=1,2,3,4$. Este tipo de método tiene alguna de las propiedades de los métodos Quasi-Newton tal que la matrix implícita $H^{(k-1)}$ es spd (y, por tanto, la dirección de búsqueda $d^{(k)}$ es descendente) si la matrix W es spd y e.g. se satisfacen las condiciones GPW. La dirección LSQN

para $m=1$ es una direcci3n PTCG, si se utiliza la f3rmula BFGS (4.1) para obtener $d^{(k)}$ en (4.2) y se efectúa una búsqueda lineal exacta para obtener $\alpha^{(k)}$. La convergencia de los métodos LSQN se incrementa con el parámetro m , pero también se incrementan las necesidades de almacenamiento y el tiempo de computaci3n. Para $m=2$; sólo se precisan 7 vectores con dimensi3n n además del espacio requerido por la matriz W (normalmente, diagonal) /10/, /11/; el balance entre necesidades de almacenamiento y tiempo de computaci3n parece adecuado (ver Sec. 5).

Independientemente del tipo de convergencia, la principal ventaja de los métodos LSQN sobre los métodos PTCG en programaci3n no lineal con condiciones (PNCC) consiste en que sólo exigen que W sea una matriz spd y e.g. las condiciones GPW se satisfagan para $\alpha^{(k)} \leq \alpha^{(k-1)}$; la matriz W , obtenida con la formulaci3n (3.9), es spd y, para valores apropiados de η en (1.6), la amplitud de paso factible satisface las condiciones GPW; por tanto, la direcci3n de búsqueda es siempre descendente.

5. ANALISIS COMPUTACIONAL.

En esta secci3n se recogen los principales resultados del análisis numérico efectuado con los métodos del gradiente conjugado en base a los tres problemas reales descritos en /10/, /11/; ver también la tabla 1. Los problemas I y II consisten en la optimizaci3n de la explotaci3n de un sistema de generaci3n eléctrica por medios hidráulicos en el que las condiciones son de dos tipos: ecuaciones de demanda eléctrica, y de balance multiperiodo multireserva de flujo de agua. El Problema III es una aproximaci3n Lagrangiana de un sistema de flujo óptimo eléctrico en el que la funci3n de pérdidas se incorpora a la funci3n objetivo. El Problema I es muy estable; los otros dos problemas son altamente inestables. El Problema I es de escala reducida, el Problema II es de super-escala, y el Problema III es de supra-escala; ver Sec. 1. En todos los problemas el gradiente es analíticamente evaluado, pero no así la matriz Hessiana.

Se ha efectuado el análisis computacional

con la metodología de un algoritmo de PNCC /8/, tal que el problema consiste en

$$\min \{F(X) \mid \bar{b} \geq AX \geq \underline{b}, U \geq X \geq L\} \tag{5.1}$$

donde A es la matriz $m \times n$ de condiciones, \bar{b} y \underline{b} son, respectivamente, los vectores independientes superior e inferior, y U y L son los vectores que acotan el vector soluci3n X . La direcci3n de búsqueda d para la iteraci3n k se ha obtenido en base a la metodología de Murtagh-Saunders (ver e.g./14/), tal que

$$d = Z d_S \tag{5.2}$$

donde d_S es la direcci3n reducida de búsqueda (también denominada direcci3n superbásica) y Z es una matriz $m \times n_S$ tal que para cualquier vector d_S garantiza que $Ad=0$ y, por tanto, el nuevo punto $X=X^{(k-1)} + \alpha d$ es factible; n_S es la dimensi3n de d_S .

En la metodología utilizada, se obtiene d_S tal que sea una direcci3n de búsqueda suficientemente descendente en el problema

$$\begin{aligned} F(X) &= F(X^{(k-1)}) + g^{(k-1)t} d + 1/2 d^t G^{(k-1)} d + o(\|d\|_2) \\ &= F(X^{(k-1)}) + \underline{h}^t d_S + 1/2 d_S^t \underline{H} d_S + o(\|d_S\|_2) \end{aligned} \tag{5.3}$$

donde \underline{h} (gradiente reducido) y \underline{H} (hessiana reducida) se obtienen utilizando (5.2) tal que

$$\underline{h} = g^{(k-1)t} Z \tag{5.4}$$

$$\underline{H} = Z^t G^{(k-1)} Z \tag{5.5}$$

Los métodos utilizados para obtener d_S en cada iteraci3n son los siguientes:

(i) Método RQN. Se basa en la actualizaci3n del factor R de Cholesky de la aproximaci3n BFGS Quasi-Newton de la matriz \underline{H} (ver e.g. /6/, /14/), tal que d_S se obtiene resolviendo el sistema $R^t R d_S = -\underline{h}$. Está diseñado para problemas de escala reducida. Ver los detalles en /8/.

(ii) Método PRTN. Está basado en el preconditionamiento diagonal de la matriz \underline{H} , tal que d_S se obtiene resolviendo el sistema

TABLA 1
Dimensiones del problema

Problema	m (1)	e (2)	n (3)	Tipos de variables (%) / β /			Densidad (%)		* t (4)	* n _s (5)	* v (5)
				Pura Lineal	No-pura Lineal	No-lineal	A	G			
I	32	19	88	20	58	22	10	18	26	47	15
II	583	241	1479	24	36	40	1	12	502	459	518
III	702	642	1762	11	25	64	8	15	673	810	279

- (1) Número de condiciones
- (2) Número de condiciones de igualdad
- (3) Número de variables
- (4) Número de condiciones activas en el óptimo local \bar{x}^*
- (5) Número de variables activas en el óptimo local \bar{x}^*

TABLA 2
Resultados en el Problema I

Método	Mitn (6)	mitn (7)	FG (8)	Tiempo CPU (secs)
RQN	103	-	274	4.02
PRTN ⁽⁹⁾	90	204	363	4.75
RCG	410	-	698	13.79
PRCG	382	-	615	12.33
TCR	503	-	795	16.83
PTCR	482	-	683	14.82
PR1S	264	-	487	10.62
PR2S	247	-	475	11.56
PR3S	282	-	460	13.56
PR4S	302	-	452	13.84
PR2SA	210	-	396	9.74

- (6) Número de iteraciones externas. Una iteración externa consiste en la obtención de d_s , d , α y por tanto, el nuevo punto X .
- (7) Número de iteraciones internas. Una iteración interna, sea i en una iteración externa dada, sea k consiste en las operaciones a efectuar para la obtención de la estimación, sea $d_s^{(i)}$ de la dirección d_s . La estimación $d_s^{(i)}$ para la última iteración interna i en una iteración externa consiste precisamente en la dirección d_s de la iteración externa considerada. Ver los detalles en /9/.
- (8) Número de evaluaciones de la función objetivo δ del gradiente en iteraciones externas.
- (9) El número total de evaluaciones FG será (7) + (8).

TABLA 3

Resultados en el Problema II

Método	Mitn (6)	mitn (7)	FG (8)	Tiempo CPU (minutos)
PRTN ⁽⁹⁾	141	7334	384	33.42
RCG	6210	-	12813	84.47
PRCG	5816	-	12132	79.87
TCR	8803	-	16132	129.58
PTCR	8161	-	15252	103.52
PR1S	5278	-	10869	66.25
PR2S	5304	-	10464	68.43
PR3S	5402	-	10213	71.12
PR4S	5443	-	9846	73.41
PR2SA	4312	-	8431	42.81

TABLA 4

Resultados en el Problema III

Método	Mitn (6)	FG (8)	Tiempo CPU (minutos)
RCG	9231	16212	131.22
PRCG	8764	15760	120.67
TCR	12832	18628	167.22
PTCR	12012	17293	150.56
PR1S	8056	12680	102.58
PR2S	7982	12174	100.61
PR3S	8043	11910	103.32
PR4S	8098	11615	106.81
PR2SA	7682	11137	83.23

$Hd_S = -h$ con el método Newton-Truncado. Se utiliza la matriz (3.9) como la matriz -precondicionante W . Está diseñado para -problemas de super-escala. Ver los detalles en /9/.

- (iii) Método RCG (gradiente conjugado con reinicialización) y PRCG (RCG precondicionado) tal que W se obtiene con la formulación (3.9)). Ver Sec. 3.
- (iv) Métodos TCG (gradiente conjugado tradicional) y PTCG (TCG precondicionado) tal que W se obtiene con la formulación - - (3.9)). Ver Sec. 3. Es preciso notar que los métodos indicados en (iii) y (iv) no precisan resolver ningún sistema de ecuaciones (programa cuadrático); en su lugar utilizan las fórmulas descritas en la sec. 3 para PNSC, pero sustituyendo g (gradiente) por h (gradiente reducido) y d (dirección de búsqueda) por d_S (dirección superbásica).
- (v) Métodos PR1S, PR2S, PR3S y PR4S. Están - basados en la metodología Quasi-Newton - con Memoria Limitada tal que $m=1,2,3,4$ - en PRmS. La matriz diagonal precondicionante se obtiene con la formulación (3.9) Ver Sec. 4.
- (vi) Método PR2SA. Basicamente es el mismo método PR2S en el que se sustituye el punto de reinicialización $x_S^{(k-3)}$ por el punto $x_S^{(r)}$ tal que r es la iteración de reinicialización. Ver los detalles en /10/, /11/.

Todos los métodos se han probado en el Problema I. Dado que el método RQN está diseñado para problemas de escala reducida, sólo - se ha podido probar en el Problema I. Por razones análogas, el método PRTN sólo se ha -- probado en los problemas I y II. Las tablas 2,3 y 4 recogen los resultados computacionales obtenidos, respectivamente, en los Problemas I, II y III.

El experimento se ha efectuado en un sistema IBM 370/158 con 8MB de memoria real, 3MB de memoria virtual, utilizando VM/CMS, las rutinas auxiliares del sistema IBM MPSX/370 y el compilador PL/I OPT(2).

La Tabla 2 muestra que cuando la capacidad de memoria requerida para almacenar el factor R de la matriz H no es un gran inconveniente, la metodología basada en la aproximación Quasi-Newton todavía es la que ofrece mejores resultados, principalmente cuando la comparación se efectúa en base al número de evaluaciones de la función objetivo y el gradiente. Es preciso destacar que cada iteración interna para obtener la dirección d_S en el método PRTN requiere una evaluación adicional del gradiente ya que la matriz Hessiana reducida H no se obtiene explícitamente. Es preciso destacar que el tiempo requerido en el método RQN para actualizar - el factor R y resolver el sistema $R^t R d_S = -h$ - domina el tiempo invertido en la evaluación de la función objetivo y gradiente.

Los Problemas II y III son muy inestables y, dadas sus dimensiones, no se puede utilizar el método RQN en su resolución. El Problema II no tiene una densidad muy alta y todavía permite la utilización del método PRTN. Sus resultados son muy satisfactorios. Ver en - /9/ el procedimiento en el que ventajosamente se explota la baja densidad en las matrices A y G .

El Problema III es de gran dimensión y, por tanto, no se puede utilizar los métodos RQN y PRTN. Analizando los resultados de los métodos basados en el gradiente conjugado, se puede observar que el método TCG ofrece resultados muy pobres; su precondicionamiento con la matriz diagonal (3.9) mejora su convergencia, pero en cualquier caso sus resultados todavía no son aceptables. Los métodos RCG y PRCG mejoran la convergencia, pero todavía el tiempo de cálculo es prohibitivo en problemas de grandes dimensiones.

Del análisis de los resultados de los métodos PRmS aplicados a la resolución de los -- tres casos, se puede obtener las siguientes conclusiones. Los métodos Quasi-Newton con memoria limitada tienen una convergencia más rápida que los métodos basados en el gradiente conjugado. No es evidente que para mayores valores de m , la convergencia sea más rápida ya que si el número de evaluaciones de la función objetivo y gradiente decrece a - medida que se aumenta el valor de m , también se incrementa (aunque sea ligeramente) el número

mero de iteraciones y, por otra parte, se incrementan las necesidades de memoria y el número de operaciones a efectuar con el procedimiento (4.2) para obtener la dirección d_S en cada iteración. Parece que $m=2$ equilibra ambos tipos de condiciones.

El método PR2SA es el método que ofrece mejores resultados en el Problema III; no incrementa considerablemente la capacidad de memoria requerida por el método PR2S. Para la obtención de la dirección d_S sólo precisa los vectores $x_S^{(j)}$ y $h^{(j)}$ para $j=k-1, k-2$ y r (donde r es la iteración de reinicialización), y la matriz diagonal $B_S^{(k-1)}$ a obtener con la formulación (3.9) donde $y^{(k)} = h^{(k)} - h^{(k-1)}$, $p^{(k)} = \alpha^{(k)} d_S^{(k)}$ tal que $g^{(k)}$ se sustituye por $h^{(k)}$ y $d^{(k)}$ se sustituye por $d_S^{(k)}$. Se puede observar que la única diferencia entre los métodos PR2S y PR2SA radica en que en el procedimiento (4.2) utilizado para obtener $d_S^{(k)}$ el vector $p^{(k-2)}$ tiene la expresión $x_S^{(k-2)} - x_S^{(k-3)}$ en PR2S y la expresión $x_S^{(k-2)} - x_S^{(r)}$ en PR2SA, donde $x_S^{(j)} = x_S^{(j-1)} + \alpha^{(j)} d_S^{(j)}$ para la iteración j .

Ver en /10/ los criterios para determinar la iteración r (denominada iteración de reinicialización); el objetivo en la sustitución de $x_S^{(k-3)}$ por $x_S^{(r)}$ consiste en evitar el caso muy frecuente de pasos pequeños sucesivos $\{\alpha^{(j)} d^{(j)}\}$ que se producen cuando la dirección de búsqueda se obtiene con los métodos basados en el gradiente conjugado y extensiones (método Quasi-Newton con Memoria Limitada).

6. CONCLUSION

En este trabajo se ha efectuado una revisión crítica de la utilización de los métodos del gradiente conjugado en la optimización de una función no lineal (no necesariamente cuadrática) con condiciones lineales.

Del análisis computacional recogido en la Sec. 5 se puede deducir que el preconditionamiento del gradiente (incluso con una matriz diagonal) aunque no produce cambios drásticos en la convergencia de los diversos algoritmos, sus resultados son siempre más satisfactorios que en el caso de no-preconditionamiento.

El método del gradiente conjugado tradicional ofrece peores resultados que el método del gradiente conjugado con reinicialización. Existen indicios de que esta diferencia se puede incrementar en el caso de la optimización de una función no lineal sin restricciones, dado que la reinicialización basada exclusivamente en el gradiente reducido se produciría en menor número de iteraciones al no exigir que la amplitud de paso produzca un punto que pertenezca a un subconjunto dado del espacio R^n (conjunto factible).

En base al desarrollo efectuado en las secciones anteriores (avalado por la experimentación computacional cuyos resultados se recogen en la Sec. 5), los algoritmos del gradiente conjugado tienen los tres siguientes grandes inconvenientes para su utilización en programación no lineal con condiciones. Basados en ellos, desaconsejamos su sustitución por el algoritmo PR2SA.

- (1) Cuando la evaluación del gradiente es muy costosa, o su aproximación por diferencias finitas es también costosa, la condición (2.13) para obtener una dirección descendente supone un esfuerzo excesivo en cada iteración.
- (2) El límite superior impuesto en la amplitud de paso tal que el nuevo punto sea factible puede provocar la imposibilidad de cumplir las condiciones que garantizan que la dirección de búsqueda es descendente y la amplitud de paso reduce suficientemente el valor de la función objetivo (e.g., condiciones GPW y (2.13)). La correspondiente reinicialización basada exclusivamente en el gradiente reducido produce una convergencia lenta.
- (3) No se puede utilizar, en un cambio de base, la información obtenida en iteraciones anteriores. La reinicialización, por tanto, debe efectuarse únicamente con el gradiente reducido.

En /10/, /11/ se describe el algoritmo PR2SA tal que sin requerir la capacidad de almacenamiento de los métodos Quasi-Newton, no exige la condición (2.13) en la amplitud de paso para obtener una dirección descendente y, además, utiliza la información obtenida en

iteraciones anteriores aunque haya un cambio de base.

7. AGRADECIMIENTO.

El autor agradece al examinador anónimo las numerosas sugerencias efectuadas para la mejor presentación y clarificación de muchos de los conceptos y opiniones vertidos en este trabajo.

8. BIBLIOGRAFIA.

- /1/ E.M.L. BEALE, : "A direction of conjugate gradients" en: F.A. Lootsma (ed.), "Numerical methods for nonlinear optimization" Academic Press, Londres, (1972).
- /2/ A.G. BUCKLEY, : "A combined conjugate - - gradient Quasi-Newton minimization algorithm", Mathematical Programming 15, - - 200-210 (1978)
- /3/ A. BUCKLEY y A. LENIR, : "QN-like variable storage conjugate gradient", Mathematical Programming 27, 155-175 (1983)
- /4/ R.S. DEMBO, y T. STEIHAUG, : "Truncated Newton algorithm for large-scale unconstrained optimization", Mathematical Programming, 23, 190-212. (1983).
- /5/ L.C.W. DIXON, : "Introduction to numerical optimization", en: L.W.C. Dixon et al. (eds.), "Nonlinear optimization" - - - Birkhauser, Londres, 1-29 (1980).
- /6/ L.F. ESCUDERO, : "On Hessian matrices in unconstrained optimization", IBM Scientific Center report G320-3416, Palo Alto (California), (1980).
- /7/ L.F. ESCUDERO, : "A projected Lagrangian method for nonlinear programming", IBM - Scientific Center report G320-3407, Palo Alto (California), (1980).
- /8/ L.F. ESCUDERO, : "An algorithm for large scale quadratic programming and its extensions to the linearly constrained - - case", IBM Scientific Center report - - - SCR-01.81, Madrid (1981).
- /9/ L.F. ESCUDERO, : "On diagonally preconditioning the Truncated Newton method for super-scale problems in linearly constrained nonlinear programming, QUESTIO 6, 261-281 (1982).
- /10/ L.F. ESCUDERO, : "On diagonally preconditioning the 2-step BFGS method with accumulated step for linearly constrained nonlinear programming. Method PR2SA", - - European J. of Operational Research 18, 259-274 (1984).
- /11/ L.F. ESCUDERO, : "Computational analysis of the method PR2SA for linearly constrained nonlinear programming", European J. of Operational Research, aparecerá.
- /12/ R. FLETCHER, : "Unconstrained optimization" Wiley, Londres, (1980).
- /13/ R. FLETCHER y C.M. REEVES, : "Function - - minimization by conjugate gradients", Computer J. 7, 149-154 (1964).
- /14/ P.E. GILL, W. MURRAY y M.H. WRIGHT, : - - - "Practical Optimization", Academic Press, Londres, (1981).
- /15/ P.E. GILL, W. MURRAY y M.H. WRIGHT, : "A note on a sufficient decrease for a non-derivate step-length procedure", Mathematical Programming 23, 249-352 (1982).
- /16/ M.R. HESTENES y E. STIEFEL, : "Methods of conjugate gradients for solving linear - - systems", J. Res. Nat. Bur. Standards - - Sec. B 48, 409-436 (1952).
- /17/ L. NAZARETH, : "A relationship between the BFGS and conjugate gradient algorithms - - and its implications for new algorithms, - SIAM J. of Numerical Analysis 16, 794-800, (1979).
- /18/ L. NAZARETH y J. NOCEDAL, : "Conjugate direction methods with variable storage, - - Mathematical Programming 23, 326-340 (1982).
- /19/ J. NOCEDAL, : "Updating Quasi-Newton matrices with limited storage", Mathematics of Computations 35, 773-782. (1980).

- /20/ E. POLAK and G. RIBIERE, : "Note sur la convergence de methodes de directions conjuguees", Rev. Fran. Infor. Rech. Oper. 16, 35-43 (1969).
- /21/ M.J.D. POWELL, : "Restart procedures for the conjugate gradient method", Mathematical Programming 12, 241-254 (1977).