



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

# Métodos de gradiente. Métodos de Krylov

Damián Ginestar Peiró

Departamento de Matemática Aplicada

Universidad Politécnica de Valencia

Curso 2012-2013

- 1 Método de descenso rápido
- 2 Método del gradiente conjugado
- 3 Métodos de Krylov
  - Método del residuo conjugado
  - método GMRES

# Método de descenso rápido

- Resolver  $Ax = b$ , con  $A$  simétrica y definida positiva (SPD).
- Definimos la función cuadrática  $\phi: \mathbb{R}^n \rightarrow \mathbb{R}$

$$\phi(y) = \frac{1}{2}(y - x)^T A(y - x) = \frac{1}{2}e^T A e .$$

- Se tiene  $\phi(y) \geq 0 \forall y \neq 0$  (definición de matriz SPD).
- Error  $e = y - x$ .

## Teorema

La solución del sistema  $Ax = b$  es el mínimo de la función  $\phi(y)$ .

$$\phi(y) = \frac{1}{2}(y - x)^T A(y - x) = \frac{1}{2}e^T A e$$

- $\phi(y_k) = \text{constant}$  representa un hiperelipsoide en un espacio de dimensión  $n$ .
- El centro geométrico es la solución  $x$  del sistema lineal (mínimo).
- Construir una sucesión  $\{y_k\}$  tal que  $\lim_{k \rightarrow \infty} y_k = x$ .

$$y_{k+1} = y_k + \alpha_k p_k$$

- Hace falta determinar la dirección  $p_k$  y  $\alpha$ .

# Método de descenso rápido

- Este método construye una sucesión que va hacia el centro del hiperelipsoide en la dirección del **gradiente**.
- El gradiente de  $\phi$  en el punto  $y_k$  es

$$\nabla\phi(y_k) = \frac{1}{2}\nabla e_k^T A e_k = \nabla \left( \frac{1}{2}y_k^T A y_k - y_k^T b + \frac{1}{2}x^T x \right) = A y_k - b = -r_k$$

- Como la dirección del vector gradiente es hacia fuera, la dirección buscada coincide con el residuo  $r_k$  en la aproximación actual.
- En consecuencia la nueva aproximación es

$$y_{k+1} = y_k + \alpha_k r_k$$

donde  $\alpha_k$  es una constante a determinar. ¿Cómo? Minimizando  $\phi(y)$  en la dirección buscada  $r_k$ .

- Es decir, nuestro  $\alpha_k$  es la solución

$$\alpha_k = \operatorname{argmin}_{\beta} \phi(y_k + \beta r_k)$$

# Método de descenso rápido

Desarrollando la función  $\phi(y_k + \beta r_k)$  se tiene un polinomio de segundo grado en la variable  $\beta$ .

$$\begin{aligned}\phi(y_k + \beta r_k) &= (y_k + \beta r_k - x)^T A(y_k + \beta r_k - x) \\ &= (y_k + \beta r_k - x)^T (Ay_k + \beta Ar_k - b) \\ &= (y_k + \beta r_k - x)^T (\beta Ar_k - r_k) \\ &= (\beta r_k - e_k)^T (\beta Ar_k - r_k) \\ &= \beta^2 r_k^T Ar_k - \beta (r_k^T r_k + e_k^T Ar_k) + x^T r_k \\ &= \beta^2 r_k^T Ar_k - 2\beta r_k^T r_k + \text{const.}\end{aligned}$$

Como  $r_k^T A r_k > 0$  el mínimo de  $\phi$  se alcanza cuando

$$\alpha_k \equiv \beta = \frac{r_k^T r_k}{r_k^T A r_k}$$

Otra forma:

Resolver  $\frac{\partial \phi}{\partial \beta} = 0$ .

# Método de descenso rápido

- La  $k + 1$  iteración se puede representar como

$$\begin{aligned}r_k &= b - Ax_k \\ \alpha_k &= \frac{r_k^T r_k}{r_k^T Ar_k} \\ y_{k+1} &= y_k + \alpha_k r_k\end{aligned}$$

Notar que el coste computacional es principalmente dos productos matriz-vector.

- De  $y_{k+1} = y_k + \alpha_k r_k$  se sigue que

$$r_{k+1} = b - Ax_{k+1} = b - Ax_k - A\alpha_k r_k = r_k - \alpha_k Ar_k,$$

- Los residuos consecutivos  $r_{k+1}, r_k$  son ortogonales (demostración: **Ejercicio**).
- El error  $e_{k+1}$  es  $A$  ortogonal a la dirección  $r_k$ . (demostración: **Ejercicio**).



## Algoritmo: Descenso rápido

Input:  $y_0, A, b, k_{\max}, \text{tol}$

- $r_0 = b - Ay_0, k = 0$
- while  $\|r_k\| > \text{tol} \|b\|$  and  $k < k_{\max}$  do
  - 1  $z = Ar_k$
  - 2  $\alpha_k = \frac{r_k^T r_k}{z^T r_k}$
  - 3  $y_{k+1} = y_k + \alpha_k r_k$
  - 4  $r_{k+1} = r_k - \alpha_k z$
  - 5  $k = k + 1$
- end while

## Lema

Sea  $A$  simétrica definida positiva y sean  $0 < \lambda_n \leq \dots \leq \lambda_2 \leq \lambda_1$  sus valores propios. Si  $P(t)$  es un polinomio real, entonces

$$\|P(A)x\|_A \leq \max_{1 \leq j \leq n} |P(\lambda_j)| \cdot \|x\|_A, \quad x \in \mathbb{R}^n$$

donde  $\|x\|_A = \sqrt{x^T A x}$ .

## Teorema

Sean las mismas condiciones que en el lema anterior. La sucesión  $\{y_k\}$  del método de descenso rápido satisface

$$\|y_k - x\|_A \leq \left( \frac{\lambda_1 - \lambda_n}{\lambda_1 + \lambda_n} \right)^k \|y_0 - x\|_A$$

donde  $x$  es la solución exacta del sistema.

## Teorema

$$\sqrt{\phi(y_k)} = \sqrt{e_k^T A e_k} = \|e_k\|_{A,2} \leq \mu^k \|e_0\|_{A,2}, \quad \text{donde} \quad \mu = \frac{\kappa(A) - 1}{\kappa(A) + 1}$$

- Cuando los sistemas vienen de discretizar ecuaciones EDPs,  $\kappa(A)$  puede ser muy grande.

- Se estima el número de iteraciones para ganar  $p$  dígitos en la aproximación de la solución:

$$\frac{\|e_k\|_A}{\|e_0\|_A} \leq 10^{-p} \text{ resolviendo } \left( \frac{\kappa(A) - 1}{\kappa(A) + 1} \right)^k \leq 10^{-p}$$

- Tomando logaritmos y usando la aproximación de primer orden de Taylor  $\log \frac{\kappa(A) - 1}{\kappa(A) + 1} \approx \frac{-2}{\kappa(A) + 1}$ , se obtiene

$$k \approx \frac{\log 10}{2} p (\kappa(A) + 1)$$

# Método del gradiente conjugado

- Es una mejora del Descenso rápido. La sucesión de recurrencia es similar

$$y_{k+1} = y_k + \alpha_k p_k$$

- Las direcciones se construyen como

$$p_0 = r_0$$

$$p_k = r_k + \beta_k p_{k-1}, \quad k > 0$$

- Se exige que las direcciones sean  $A$  conjugadas

$$p_{k-1}^T A p_k = 0,$$

es decir,  $p_k$  y  $p_{k-1}$  son  $A$ -ortogonales.

- Por tanto, se debe cumplir

$$\beta_k = -\frac{r_k^T A p_{k-1}}{p_{k-1}^T A p_{k-1}}$$

# Método del gradiente conjugado

- Como en el método de descenso más rápido, la elección de  $\alpha_k$  se obtiene minimizando  $\phi(y_{k+1}) = \phi(y_k + \alpha_k p_k)$  dando la expresión

$$\alpha_k = \frac{r_k^T p_k}{p_k^T A p_k}$$

- Residuos consecutivos como en el método de descenso más rápido satisfacen la relación de recurrencia

$$r_{k+1} = r_k + \alpha_k A p_k$$

# Método del gradiente conjugado

## Teorema

Las sucesiones de vectores  $\{r_i\}$  y  $\{p_i\}$  satisfacen las siguientes relaciones

$$(i) \quad p_i^T r_j = 0, \quad 0 \leq i < j \leq k,$$

$$(ii) \quad r_i^T r_j = 0, \quad i \neq j, \quad 0 \leq i, j \leq k,$$

$$(iii) \quad p_i^T A p_j = 0, \quad i \neq j, \quad 0 \leq i, j \leq k,$$

$$(iv) \quad \text{env}\{r_0, r_1, \dots, r_k\} = \text{env}\{p_0, p_1, \dots, p_k\} = \mathcal{K}(A, r_0, k + 1),$$

$$\text{donde } \mathcal{K}(A, r_0, k + 1) = \text{env}\{r_0, A r_0, \dots, A^k r_0\}.$$

## Corolario

El método del gradiente conjugado obtiene la solución del sistema de  $n$  ecuaciones en como máximo  $n$  iteraciones del GC.

## Otras relaciones útiles

- ①  $p_k^T r_k = r_k^T r_k$ . Ya que de  $e_k^T A p_j = 0$  se sigue  $r_k^T p_j = 0$  y, por tanto,

$$p_k^T r_k = (r_k + \beta_{k-1} p_{k-1})^T r_k = r_k^T r_k$$

- ②  $r_k^T A p_k = p_k^T A p_k$ .

- ③ Combinando 1 y 2, se obtiene una definición alternativa de  $\alpha_k$ :

$$\alpha_k = \frac{r_k^T p_k}{p_k^T A p_k} = \frac{r_k^T r_k}{r_k^T A p_k}$$

- ④ Formulación alternativa de  $\beta_k$ . Como  $p_k^T A p_k = p_k^T \frac{1}{\alpha_k} (r_k - r_{k+1}) = \frac{1}{\alpha_k} r_k^T r_k$

$$r_{k+1}^T A p_k = r_{k+1}^T \frac{1}{\alpha_k} (r_k - r_{k+1}) = -\frac{1}{\alpha_k} r_{k+1}^T r_{k+1}$$

Por tanto

$$\beta_k = -\frac{r_{k+1}^T p_k}{p_k^T A p_k} = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$$



## Algoritmo: Gradiente conjugado

Input:  $y_0, A, b, k_{\max}, \text{tol}$

- $r_0 = p_0 = b - Ax_0, k = 0$
- while  $\|r_k\| > \text{tol} \|b\|$  and  $k < k_{\max}$  do
  - 1  $z = Ap_k$
  - 2  $\alpha_k = \frac{p_k^T r_k}{z^T p_k}$
  - 3  $y_{k+1} = y_k + \alpha_k p_k$
  - 4  $r_{k+1} = r_k - \alpha_k z$
  - 5  $\beta_k = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$
  - 6  $p_{k+1} = r_{k+1} + \beta_k p_k$
  - 7  $k = k + 1$
- end while

## Ejercicio

Aplicar el algoritmo del gradiente conjugado para el problema

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

usando como aproximación inicial  $x_0 = (0, 0)^T$ .

## Solución:

- $x_0 = (0, 0)^T$ .
- $p_0 = r_0 = b = (1, 0)^T$ .
- $\alpha_0 = \frac{r_0^T r_0}{p_0^T A p_0} = \frac{1}{2}$ ,  $x_1 = x_0 + \alpha_0 p_0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ 0 \end{pmatrix}$
- $r_1 = r_0 - \alpha_0 A p_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} - \frac{1}{2} \begin{pmatrix} 2 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ \frac{1}{2} \end{pmatrix}$ ,  $r_1^T r_0 = 0$
- $\beta_0 = \frac{r_1^T r_1}{r_0^T r_0} = \frac{1}{4}$ ,  $p_1 = r_1 + \beta_0 p_0 = \begin{pmatrix} 0 \\ \frac{1}{2} \end{pmatrix} + \frac{1}{4} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{4} \\ \frac{1}{2} \end{pmatrix}$
- $\alpha_1 = \frac{r_1^T r_1}{p_1^T A p_1} = \frac{2}{3}$   
 $x_2 = x_1 + \alpha_1 p_1 = \begin{pmatrix} \frac{1}{2} \\ 0 \end{pmatrix} + \frac{2}{3} \begin{pmatrix} \frac{1}{4} \\ \frac{1}{2} \end{pmatrix} = \begin{pmatrix} \frac{2}{3} \\ \frac{1}{3} \end{pmatrix}$
- $r_2 = 0$  solución exacta

## Teorema

Sea  $A \in \mathbb{R}^{n \times n}$  simétrica y definida positiva. Sea  $x$  la solución exacta del sistema  $Ax = b$ . Entonces la sucesión de vectores del Gradiente Conjugado  $\{y_k\}$  cumple

$$\|x - y_k\|_A \leq 2 \left( \frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \right)^k \|x - y_0\|_A$$

donde  $\kappa_2(A) = \|A\|_2 \|A^{-1}\|_2$ .

Vimos que el método de Richardson construye la sucesión

$$x_{k+1} = x_0 + \sum_{i=0}^k (I - A)^i r_0 .$$

así  $x_{k+1} \in \text{span}\{x_0, r_0, Ar_0, A^2r_0, \dots, A^k r_0\}$ .

Para los residuos se tiene

$$r_{k+1} = b - Ax_0 - A \sum_{i=0}^k (I - A)^i r_0 = r_0 - A \sum_{i=0}^k (I - A)^i r_0 .$$

y así  $r_{k+1} \in \text{span}\{r_0, Ar_0, A^2r_0, \dots, A^{k+1}r_0\}$ .

El espacio  $\text{span}\{r_0, Ar_0, A^2r_0, \dots, A^{k-1}r_0\}$  se llama **subespacio de Krylov** de dimensión  $k$ , correspondiente a la matriz  $A$  y residuo inicial  $r_0$

$$K^k(A; r_0) = \text{span}\{r_0, Ar_0, A^2r_0, \dots, A^{k-1}r_0\}$$

¿Se puede usar la información contenida en  $K^k(A; r_0)$  de forma más eficiente que con el método de Richardson?

Por ejemplo, como ya hemos visto, el método de descenso más rápido se basa en iteraciones de la forma

$$\begin{aligned}x_{k+1} &= x_k + \alpha_k r_k \\r_{k+1} &= b - Ax_{k+1} = r_k - \alpha_k Ar_k .\end{aligned}$$

Para este método  $x_{k+1} \in x_0 \cup K^{k+1}(A; r_0)$

Los métodos de proyección de un paso se basan en una combinación óptima de los dos últimos vectores base del subespacio de Krylov. ¿Es posible contruir una combinación lineal óptima de todos los vectores base del subespacio de Krylov?

La respuesta en dos pasos:

- Primero vemos cómo construir una base para  $K^k(A; r_0)$ ;
- Después veremos cómo construir una aproximación óptima como una combinación lineal de los vectores base. Primero para matrices simétricas.



La base más simple para el subespacio de Krylov  $K^k(A; r_0)$  es la base:  $r_0, Ar_0, A^2r_0, \dots, A^{k-1}r_0$ .

Pero esta base está mal condicionada ya que  $A^{k-1}r_0$  apuntará cada vez más en la dirección del autovector dominante de  $A$ .

Una base estable y ortogonal se puede construir usando el método de Arnoldi.

# Métodos de Krylov

Se elige un vector inicial  $q_1$  con  $\|q_1\|_2 = 1$ .

```
FOR  $k = 1, \dots$  DO (iteración)
   $v = Aq_k$ 
  FOR  $i = 1, \dots, k$  (ortogonalización)
     $h_{i,k} = v^T q_i$ 
     $v = v - h_{i,k} q_i$ 
  END FOR
   $h_{k+1,k} = \|v\|_2$ 
  IF  $h_{k+1,k} = 0$  STOP (subespacio invariante)
   $q_{k+1} = v/h_{k+1,k}$  (nuevo vector)
END FOR
```

El método de Arnoldi se puede resumir en:

$$H_k = \begin{bmatrix} h_{1,1} & \dots & \dots & h_{1,k} \\ h_{2,1} & \ddots & & \vdots \\ & \ddots & \ddots & \vdots \\ 0 & & h_{k,k-1} & h_{k,k} \end{bmatrix}$$

y  $Q_k = [q_1 \ q_2 \ \dots \ q_k]$  entonces

$$AQ_k = Q_k H_k + h_{k+1,k} q_{k+1} e_k^T$$

donde  $e_k$  es el  $k$  – esimo vector de la base canónica de  $\mathbb{R}^k$ .

Si  $A$  es simétrica de acuerdo a la relación de Arnoldi

$$Q_k^T A Q_k = H_k .$$

Como  $A$  es simétrica

$$H_k^T = Q_k^T A^T Q_k = Q_k^T A Q_k = H_k .$$

Así  $H_k$  es simétrica y Hessenberg superior  
luego  $H_k$  es tridiagonal.

Así

$$H_k = \begin{bmatrix} h_{1,1} & h_{2,1} & & & 0 \\ h_{2,1} & \ddots & \ddots & & \\ & \ddots & \ddots & & h_{k,k-1} \\ 0 & & & h_{k,k-1} & h_{k,k} \end{bmatrix} .$$

Con  $\alpha_k = h_{k,k}$  y  $\beta_k = h_{k-1,k}$  el método de Arnoldi se simplifica en el método de Lanczos. Con el método de Lanczos es posible calcular un nuevo vector base ortogonal utilizando sólo los dos vectores base previos.

El método de Arnoldi y el método de Lanczos se propusieron originalmente como métodos iterativos para calcular autovalores de la matriz  $A$ :

$$Q_k^T A Q_k = H_k$$

es 'casi' una transformación de similitud. Los autovalores de  $H_k$  se llaman los valores de Ritz de  $A$ .

El método de Lanczos nos proporciona un método económico para calcular vectores base ortogonales para el subespacio de Krylov  $K^k(A; r_0)$ .

Nuestra aproximación se escribe

$$x_k = x_0 + Q_k y_k$$

donde  $y_k$  se determina de forma que o bien se minimiza

$$f(x_k) = \|x_k - x\|_A^2 = (x_k - x)^T A (x_k - x)$$

respecto de la norma inducida por  $A$  (simétrica y definida positiva) o bien

$$g(x_k) = \|A(x_k - x)\|_2^2 = r_k^T r_k,$$

se minimiza la norma del residuo.

Se considera la minimización del error en la  $A$ -norma:

$$x_k = x_0 + Q_k y_k \Rightarrow f(x_k) = (x_0 + Q_k y_k - x)^T A (x_0 + Q_k y_k - x) .$$

Se calcula la derivada respecto de  $y_k$  y  $\frac{\partial f(x_k)}{\partial y_k} = 0$ . Con lo que

$$Q_k^T A Q_k y_k = Q_k^T r_0$$

y con  $Q_k^T A Q_k = T_k$ ,  $r_0 = \|r_0\| q_1$  se tiene

$$T_k y_k = \|r_0\| e_1$$

con  $e_1$  el primer vector de la base canónica.



Es fácil ver que los residuos son ortogonales a los vectores base

$$r_k = r_0 - AQ_k y_k \Rightarrow Q_k^T r_k = Q_k^T r_0 - Q_k^T A Q_k y_k = 0$$

Esta condición es equivalente a minimizar  $f(x_k)$  cuando  $A$  es SPD. En este caso se obtiene el método del gradiente conjugado.

# Método del residuo conjugado

El gradiente conjugado minimiza la  $A$ -norma del error. Otra forma de construir una aproximación óptima  $x_k$  es minimizar el residuo

$$g(x_k) = \|A(x_k - x)\|_2^2 = r_k^T r_k$$

sobre todos los  $x_k \in \{x_0 \cup K^k(A; b)\}$ .

Definiendo

$$\underline{T}_k = \begin{bmatrix} \alpha_1 & \beta_2 & & & 0 \\ \beta_2 & \alpha_2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & 0 & \ddots & \ddots & \beta_k \\ & & & & \beta_k & \alpha_k \\ & & & & & \beta_{k+1} \end{bmatrix}$$

El método de Lanczos se escribe

$$AQ_k = Q_{k+1}\underline{T}_k$$

# Método del residuo conjugado

El problema es encontrar  $x_k = x_0 + Q_k y_k$  tal que  $\|r_k\|$  es mínimo.

$$r_k = b - Ax_k = r_0 - AQ_k y_k = \|r_0\| q_1 - AQ_k y_k$$

así hay que minimizar

$$\begin{aligned}\|r_k\| &= \|\|r_0\| q_1 - AQ_k y_k\| \\ &= \|\|r_0\| Q_{k+1} e_1 - Q_{k+1} \underline{T}_k y_k\| \\ &= \|\|r_0\| e_1 - \underline{T}_k y_k\|\end{aligned}$$

# Método del residuo conjugado

Resolviendo el sistema sobredeterminado  $\underline{T}_k y_k = \|r_0\| e_1$  se tienen las iteraciones

$$x_k = x_0 + Q_k y_k$$

que minimizan el residuo. El algoritmo resultante se llama MINRES (o residuo conjugado)

# Método del residuo conjugado

```
 $r_0 = b - Ax_0; \quad p_0 = r_0$   
FOR  $k = 0, 1, \dots$ , DO  
     $\alpha_k = \frac{r_k^T Ar_k}{(Ap_k)^T Ap_k}$   
     $x_{k+1} = x_k + \alpha_k p_k$   
     $r_{k+1} = r_k - \alpha_k Ap_k$   
     $\beta_k = \frac{r_{k+1}^T Ar_{k+1}}{r_k^T Ar_k}$   
     $p_{k+1} = r_{k+1} + \beta_k p_k$   
     $Ap_{k+1} = Ar_{k+1} + \beta_k Ap_k$   
END FOR
```

inicializacion

actualizacion residuo

actualizacion direccion

- En el caso de  $A$  simétrica el método de Arnoldi es muy eficiente, los vectores base nuevos se pueden calcular con una recurrencia de tres términos.
- Esto permite construir métodos muy eficientes que combinan recurrencias cortas y una condición de optimalidad para el error.
- Veremos ahora cómo se pueden construir métodos para el caso no simétrico. Estos métodos usan el método de Arnoldi.

# Método GMRES

Método de Arnoldi.

Se elige un vector inicial  $q_1$  con  $\|q_1\|_2 = 1$ .

```
FOR  $k = 1, \dots$  DO           (iteracion)
   $v = Aq_k$ 
  FOR  $i = 1, \dots, k$          ortogonalizacion
     $h_{i,k} = v^T q_i$ 
     $v = v - h_{i,k} q_i$ 
  END FOR
   $h_{k+1,k} = \|v\|_2$ 
  IF  $h_{k+1,k} = 0$  STOP      subespacio invariante
   $q_{k+1} = v/h_{k+1,k}$       nuevo vector base
END FOR
```

En forma compacta

$$H_k = \begin{bmatrix} h_{1,1} & \dots & \dots & h_{1,k} \\ h_{2,1} & \ddots & & \vdots \\ & \ddots & \ddots & \vdots \\ 0 & & h_{k,k-1} & h_{k,k} \end{bmatrix}$$

y  $Q_k = [q_1 \ q_2 \ \dots \ q_k]$  entonces

$$AQ_k = Q_k H_k + h_{k+1,k} q_{k+1} e_k^T$$



Definiendo

$$\underline{H}_k = \begin{bmatrix} h_{1,1} & \dots & \dots & h_{1,k} \\ h_{2,1} & \ddots & & \vdots \\ & \ddots & \ddots & \vdots \\ & & h_{k,k-1} & h_{k,k} \\ O & & & h_{k+1,k} \end{bmatrix}$$

el método de Arnoldi se escribe

$$AQ_k = Q_{k+1}\underline{H}_k$$

El método de Anoldi nos da una base ortogonal para el subespacio de Krylov  $K^k(A; r_0)$ .

Se tiene la aproximación

$$x_k = x_0 + Q_k y_k$$

$y_k$  es tal que minimiza el error

$$f(x_k) = \|x_k - x\|_A^2 = (x_k - x)^T A(x_k - x)$$

respecto de la  $A$ -norma, o bien minimiza el

$$g(x_k) = \|A(x_k - x)\|_2^2 = r_k^T r_k,$$

residuo.

Si  $A$  no es SPD, la  $A$ -norma no está bien definida.

Imponiendo ahora que los residuos sean ortogonales a  $Q_k$  se tiene

$$Q_k^T (r_0 - AQ_k y_k) = 0 \Rightarrow \|r_0\| e_1 - H_k y_k = 0$$

Resolviendo el sistema pequeño

$$y_k = \|r_0\| H_k^{-1} e_1 \quad x_k = x_0 + Q_k y_k$$

Este método se llama FOM.

El FOM es equivalente al GC si  $A$  es SPD. Pero tiene inconvenientes:

- FOM no trata de forma eficiente la memoria:  $Q_k$  se tiene que guardar completa. En cada iteración, un nuevo vector base se tiene que calcular y guardar. La ortogonalización va siendo cada vez más cara con  $k$ .
- FOM no tiene una propiedad de optimalidad.
- El método es finito, pero esto sólo tiene interés teórico ya que se tendrían que calcular  $n$  vectores base y guardarlos.
- FOM no es robusto ya que  $H_k$  podría ser singular.

El método FOM para obtener  $x_k$  es parte de una familia de técnicas para extraer una solución aproximada a partir de un **espacio de búsqueda**  $Q_k$  haciendo que el residuo se ortogonal a un **espacio test**  $W_k$ .

Esto se formula como: Sea  $x_k = x_0 + Q_k y_k$ . Encontrar  $y_k$  tal que

$$W_k^T (r_0 - A Q_k y_k) = 0$$

Estas son las condiciones de Petrov-Galerkin.

Si  $W_k = Q_k$  se llaman condiciones de Galerkin.

Veamos un segundo método para obtener un mínimo del residuo.

Minimizar

$$g(x_k) = \|A(x_k - x)\|_2^2 = r_k^T r_k$$

es un problema bien definido incluso si  $A$  es no simétrica.

El problema es encontrar  $x_k = x_0 + Q_k y_k$  tal que  $\|r_k\|$  is mínimo.

$$r_k = b - Ax_k = r_0 - AQ_k y_k = \|r_0\| q_1 - AQ_k y_k$$

así

$$\begin{aligned}\|r_k\| &= \|\|r_0\| q_1 - AQ_k y_k\| \\ &= \|\|r_0\| Q_{k+1} e_1 - Q_{k+1} \underline{H}_k y_k\| \\ &= \|\|r_0\| e_1 - \underline{H}_k y_k\|\end{aligned}$$

Resolviendo el problema sobredeterminado

$$\underline{H}_k y_k = \|r_0\| e_1$$

se tienen las iteraciones

$$x_k = x_0 + Q_k y_k$$

que minimizan el residuo. El algoritmo resultante se llama GMRES.

GMRES, es uno de los métodos más populares para resolver sistemas no simétricos