

# Tarea 3: MÉTODO DE REGIÓN DE CONFIANZA

Juan Gerardo Fuentes Almeida

**Abstract**—En esta práctica se implementa el algoritmo de Región de Confianza al problema de registro rígido de imágenes.

## 1 INTRODUCCIÓN

Los Métodos de la Región de Confianza constituyen una metodología reciente en optimización, se basan en un problema de aproximación más que en una dirección.

Se construye una función modelo  $m_k(x)$  cuyo comportamiento cerca de la iteración actual  $x_k$  es similar a la función objetivo  $f$ , por esta razón se le llama método de Región de Confianza.

Como el modelo  $m_k$  puede no ser una buena aproximación de  $f$  cuando  $x$  está lejos de  $x_k$ , se restringe la búsqueda de un óptimo de  $m_k$  en alguna región en torno a  $x_k$ .

Si la solución candidata no produce un decrecimiento suficiente en  $f$  se concluye que la región de confianza es demasiado grande y debe reducirse para después resolver el subproblema asociado.

## 2 TEORÍA

### 2.1 Región De Confianza

El modelo en un método de región de confianza está usualmente definido por una función cuadrática de la forma:

$$m_k(x_k + p) = f(x_k) + p^T \nabla f(x_k) + \frac{1}{2} p^T B_k p$$

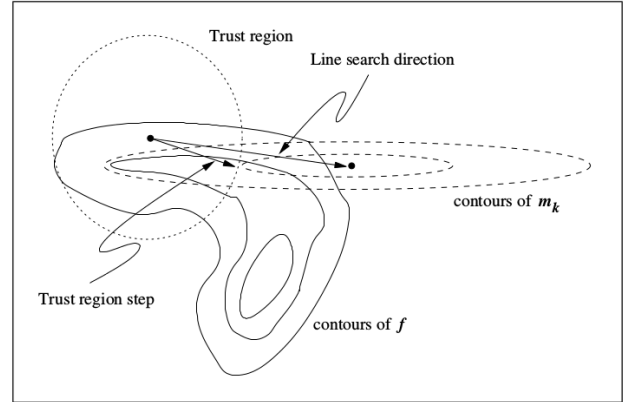
donde la matriz  $B_k$  es, o bien el Hessiano  $\nabla^2 f(x_k)$  o alguna aproximación de éste.

Si  $B_k = 0$  y se define la región de confianza como la una norma Euclídea, el problema de región de confianza se resume en:

$$\min f(x_k) + p^T \nabla f(x_k) \text{ sujeto a } \|p\| \leq \Delta_k$$

Un algoritmo de región de confianza más interesante se obtiene eligiendo  $B_k$  como el Hessiano  $\nabla^2 f(x_k)$  en el modelo cuadrático. Como se tiene la restricción de la región de confianza  $\|p\| \leq \Delta_k$  no es necesario que  $\nabla^2 f(x_k)$  sea definido positivo pues se tiene asegurada la existencia de una solución  $p_k$ .

El tamaño de la región de confianza es crítico para la eficacia de cada paso. Si la región de confianza es muy pequeña, el algoritmo pierde la oportunidad de dar un paso importante para desplazarse cerca del minimizador de la función objetivo  $f$ .



Búsqueda en Línea vs Región de Confianza

Por otro lado, si es muy grande, el minimizador del problema puede estar lejos del minimizador de la función objetivo en la región, dando como resultado un avance no deseado, una solución para esta situación sería reducir el tamaño de la región de confianza, e intentar resolver nuevamente.

En consecuencia a lo anterior se tiene que la elección de  $\Delta_k$  debe basarse en la función  $f$  mediante la función  $m_k$  empleando el siguiente parámetro.

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)}$$

donde  $p_k$  es un paso a partir de la iteración actual  $x_k$ . Al numerador se le llama reducción actual y al denominador reducción prevista.

Debemos notar que como el paso  $p_k$  lo podemos obtener minimizando el modelo  $m_k$  en una región que incluye  $p = 0$ , tomando en cuenta que la reducción prevista será siempre no negativa, entonces si  $\rho_k$  es negativo el nuevo valor objetivo  $f(x_k + p_k)$  es mayor que el valor actual  $f(x_k)$  por lo que el paso debe ser rechazado.

**Algorithm 2** Trust region

---

**Require:** A maximum step size  $\hat{\Delta}$ , initial step  $\Delta_0 \in (0, \hat{\Delta})$  and update threshold  $\eta$

**while** Stopping criterion not met **do**

Minimize  $m_k(p)$  using an optimization algorithm

$\rho_k = \frac{f(x_k) - f(x_k + p)}{m_k(0) - m_k(p_k)}$

**if**  $\rho_k < \frac{1}{4}$  **then**

$\Delta_{k+1} = \frac{1}{4}\Delta_k$

**else**

**if**  $\rho_k > \frac{3}{4}$  and  $\|p_k\| = \Delta_k$  **then**

$\Delta_{k+1} = \min(2\Delta_k, \hat{\Delta})$

**else**

$\Delta_{k+1} = \Delta_k$

**if**  $\rho_k > \eta$  **then**

$x_{k+1} = x_k + p_k$

**else**

$\Delta_{k+1} = \eta\Delta_k$

$x_{k+1} = x_k$

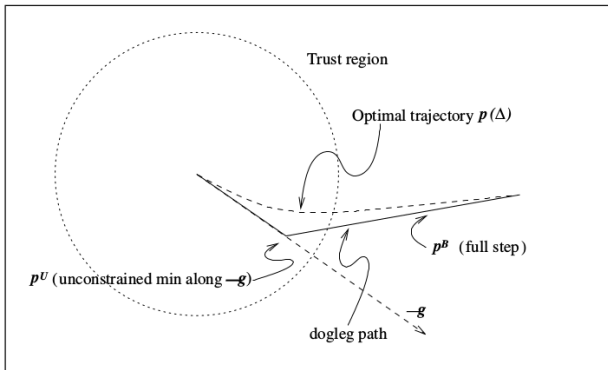
**return**  $x_k$

---

**2.2 Algoritmo de Dog-Leg**

El algoritmo de Dog-Leg se basa en el cálculo del Punto de Cauchy, es el reductor de  $m_k$  a lo largo de la dirección de máximo descenso  $g_k$  sujeto a la región de confianza. No es necesariamente un minizador del modelo, pero si aporta un suficiente descenso dentro de la region de confianza

Este método utiliza el cálculo tanto de la dirección de Newton de descenso, así como la dirección de descenso de gradiente y el radio de la region de confianza para encontrar una  $p_k$  óptima que produzca un suficiente descenso.



Método de DogLeg

**3 RESULTADOS**

Como parte de la práctica, se tomó un conjunto de imágenes de resonancia magnética para implementar un registro rígido y minimizar la función de error determinada por la siguiente expresión:

$I_2$  es una imagen de referencia,  $I_1$  es la imagen observada cuyo dominio está definido en  $x = x_1, x_2$ , de manera que  $\omega$  representa el conjunto de todos los píxeles

**Algorithm 1** Dogleg

---

**Require:** Step size  $\Delta$ , Gradient  $g$  and Hessian  $B$

$p^U = -\frac{g^T g}{g^T B g} g$  (Ecs: 4.11)

$p^B = -(B + \lambda I)^{-1} g$  (Ecs: 4.6)

**if**  $\|p^U\| \geq \Delta_k$  **then**

$p^C = \Delta \frac{p^U}{\|p^U\|}$

**else**

**if**  $\|p^B\| \leq \Delta_k$  **then**

$p^C = p^B$

**else**

(See Nocedal page 75)

Solve  $\|p^U + (\tau - 1)(p^B - p^U)\|^2 = \Delta^2$ .

$p^C = (1 - k)p^U + kp^B$

**return**  $p^C$

---

$$\min_{\alpha} f(\alpha) = \frac{1}{2} \sum_{x \in \Omega} \left( I_2(x) - I_1(x + X\alpha) - \nabla I_1(x + X\alpha)^T X\alpha \right)^2$$

de la imagen,  $X$  y  $\alpha$  son variables definidas de la siguiente manera:

$$X = \begin{bmatrix} x_1 & x_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & x_2 & 1 \end{bmatrix} \text{ y } \alpha = \begin{bmatrix} \cos(\theta) - 1 \\ \sin(\theta) \\ v_1 \\ -\sin(\theta) \\ \cos(\theta) - 1 \\ v_2 \end{bmatrix}$$

Esto permite que la transformación de una imagen a otra este dada por el producto  $x + X\alpha$

El gradiente y el Hessiano de esta función están definidos de la siguiente manera:

$$\nabla f(\alpha) = \sum [I_2(x) - I_1(x)] [-X^T \nabla I_1(x + X\alpha)]$$

$$\nabla^2 f(\alpha) = \sum [X^T \nabla I_1(x + X\alpha)] [X^T \nabla I_1(x + X\alpha)]^T$$

En la práctica, estas funciones fueron las que nos dieron mejores resultados.

El algoritmo lee los archivos de las dos imágenes y realiza la minimización de la función objetivo con el método de la Región de Confianza, en conjunto con el método de Dogleg para el cálculo del punto de Cauchy; se inicia con un desplazamiento inicial de 10 píxeles hacia arriba y hacia la izquierda para acercarnos un poco a la posición de la imagen de referencia.

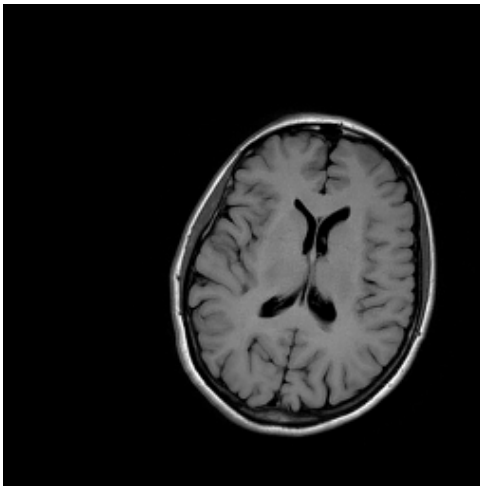


Imagen Inicial u Observada

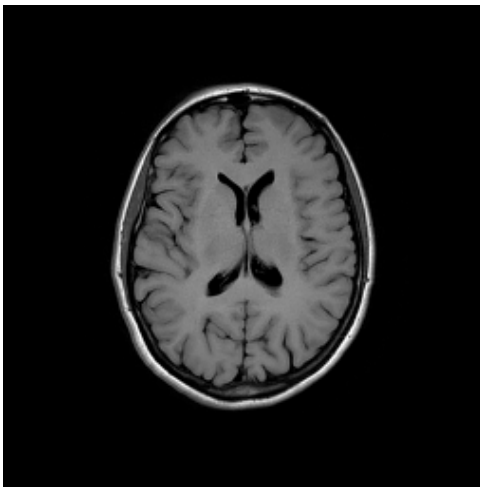


Imagen de referencia

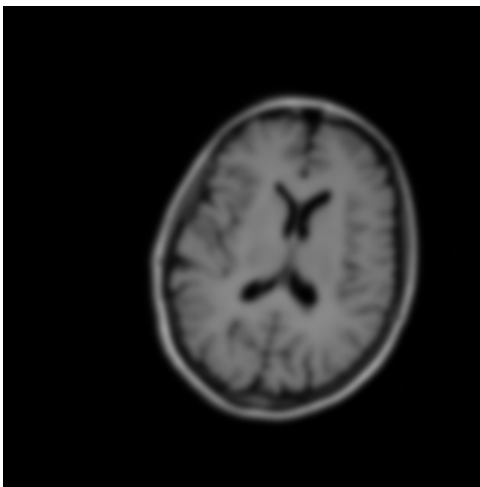


Imagen inicial, se tuvo que hacer un emborronado de la imagen para obtener mejores resultados

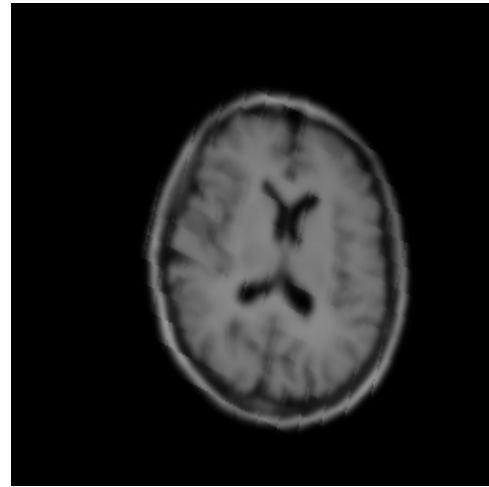


Imagen final despues de 20 mejoras, se aprecia la rotacion que se produce para empalmarse con la imagen de referencia

Aunque la función objetivo siempre muestra valores demasiado grandes, se pudo observar una mejoría en la posición y orientación de la imagen con respecto a la de referencia. Otro comportamiento observado fue que a partir de cierto número de iteraciones, la función objetivo deja de disminuir quedándose en un valor muy grande y reduciendo prácticamente a cero la región de confianza.

#### 4 CONCLUSIONES

En general no se obtuvieron los resultados que se esperaban, se tuvieron muchos problemas, principalmente con respecto a la correcta definición de la función objetivo y sus derivadas, también se tuvieron problemas con los métodos de optimización vistos en clase, los cuales directamente aplicados a este problema no producían los resultados esperados. Se espera que en futuras prácticas se afinen más detalles sobre este tipo de implementaciones.

#### REFERENCES

- [1] Jorge Nocedal and Stephen J. Wright. *Numerical optimization*. Springer series in operations research and financial engineering. Springer, New York, NY, 2. ed. edition, 2006.

#### APPENDIX

El programa está implementado tomando en cuenta todas estandarizaciones indicadas en el curso.

Un *makefile* ha sido generado, el cual soporta los comandos *make*, *run* and *clean*. El programa recibe como primer argumento el nombre del archivo de la imagen inicial, y como segundo argumento el nombre del archivo de la imagen de referencia.