

Implementación computacional en GNU Octave del método BFGS.

Integrantes:

- Jessica Espinoza Quesada - Carnet 2018135811
- Jose David Sánchez Schnitzler - Carnet 2018142388
- Tomás Felipe Segura Monge - Carnet 2018099729

En la presente sección de la tarea, se realizó la implementación del algoritmo BFGS, cuyas siglas corresponden a Broyden, Fletcher, Goldfarb y Shanno. El mismo es un algoritmo de optimización de búsqueda local.

Es un tipo de algoritmo de optimización de segundo orden, que pertenece a una clase de algoritmos denominados métodos Cuasi-Newton que se aproximan a la segunda derivada para problemas de optimización donde esta no se puede calcular.

El algoritmo BFGS es quizás uno de los algoritmos de segundo orden más utilizados para la optimización numérica. La optimización implica encontrar valores para los parámetros de entrada que maximizan o minimizan una función objetivo. A continuación, veremos cómo obtener la máxima transferencia de potencia de un circuito eléctrico simple, empleando el método BFGS.

1. Función de Dolan.

Se evaluó el algoritmo con dicha función, como se observa en la imagen 2 y los resultados obtenidos se resumen en la tabla 1. Además, se muestra la gráfica de iteraciones versus error en la figura 1.

Aproximación del punto	Iteraciones	Error final
(4.7353, 56.8136, 38.0958, 6.4772, -2.0616	50	3.6075

Cuadro 1: Resultados obtenidos tras evaluar el algoritmo BFGS con la función de Dolan.

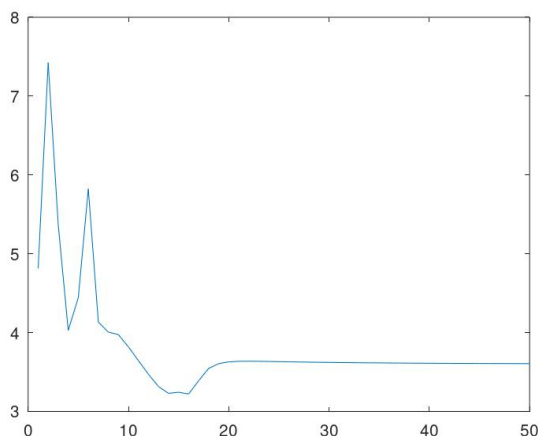


Figura 1: Gráfica de error para la función de Dolan.

The screenshot shows the GNU Octave environment. The main editor window displays a MATLAB script named `p3_solucion_aplicacion.m`. The script defines a function `p3_solucion_aplicacion(tol, iterMax)` that implements the BFGS algorithm. It includes comments in Spanish explaining the function's purpose, parameters, and the Dolan function used for testing. The script calls `p3_bfgs` with specific parameters and returns the results `[xk, k, error]`.

Below the editor, the 'Ventana de comandos' (Command Window) shows the output of the script. It displays the values of `xk`, `k`, and `error`, along with the results of the Dolan function evaluation.

```

function [xk, k, error] = p3_solucion_aplicacion(tol, iterMax)
%Esta función se encarga de insertar los datos del problema de ingeniería elegido en la función BFGS
%
%Sintaxis: p3_solucion_aplicacion(tol, iterMax)
%
%Parámetros Iniciales:
%     tol = un número positivo que representa a la tolerancia para el criterio: ||gradiente(f(x_k))|| <= tol
%     iterMax = cantidad de iteraciones máximas
%
%Parámetros de Salida:
%     xk = aproximación de los valores de convergencia de la función f
%     k = número de iteraciones realizados
%     error = ||gradiente(f(x_k))||
%
%Se inserta la ecuación del problema de ingeniería seleccionado en el metodo bfgs realizado en el archivo p3_bfgs.m
[xk, k, error] = p3_bfgs('(x + 1.7*y)*sin(x) - 1.5*z - 0.1*m*cos(m+n-x) + 0.2*(n**2) - y - 1', ['x' 'y' 'z' 'm' 'n'], 10**-5, 50) #Dolan Function
end

```

Output in the Command Window:

```

56.8136
38.0958
6.4772
-2.0616

k = 50
error = 3.6075
ans =

4.7353
56.8136
38.0958
6.4772
-2.0616

```

Figura 2: Evaluación del algoritmo BFGS en GNU Octave con la función de Dolan.

2. Máxima transferencia de potencia en un circuito (Ingeniería Electrónica).

El circuito de resistencias simple que se presenta en la figura 3 contiene tres resistores fijos y uno ajustable. Los resistores ajustables se llaman potenciómetros. Los valores de los parámetros son $V = 60 \text{ V}$, $R_1 = 8 \Omega$, $R_2 = 12 \Omega$ y $R_3 = 10 \Omega$. El objetivo es encontrar la máxima transferencia de potencia a través de las terminales 1 y 2 y el valor de la resistencia R_a que lo permite.[1]

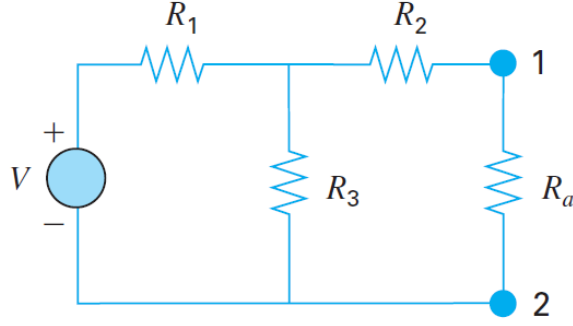


Figura 3: Un circuito de resistencias con un resistor ajustable, o potenciómetro.

A partir de las leyes de Kirchhoff se puede obtener la siguiente expresión para la potencia del circuito:

$$P(R_a) = \frac{\left(\frac{V R_3 R_a}{R_1(R_a + R_2 + R_3) + R_3 R_a + R_3 R_2} \right)^2}{R_a} \quad (1)$$

Se debe maximizar $P()$ en función de R_a . Para ello, se hará uso del algoritmo BFGS como se observa en la figura 4. Para ello, se tomó como valor inicial el número 15. Los resultados se resumen en la tabla de 2 y la gráfica de error versus iteraciones se aprecia en la figura 5.

Resistor R_a	Potencia $P(R_a)$
14.836 Ω	18mW

Cuadro 2: Resultados obtenidos tras evaluar el algoritmo BFGS con la función de máxima transferencia de potencia.

```

Octave
Archivo Editar Depurar Ventana Ayuda Noticias
Directorio actual: C:\Users\jessi\Downloads\Tarea 1\Parte 3
Editor
Archivo Editar Ver Depurar Ejecutar Ayuda
p3_bfgs.m p3_solucion_aplicacion.m
1 function [xk, k, error] = p3_solucion_aplicacion(tol, iterMax)
2
3 %Esta función se encarga de insertar los datos del problema de ingeniería elegido en la función BFGS
4
5 %Sintaxis: p3_solucion_aplicacion(tol, iterMax)
6
7 %Parámetros Iniciales:
8 %   tol = un número positivo que representa a la tolerancia para el criterio: ||gradiente(f(x_k))|| <= tol
9 %   iterMax = cantidad de iteraciones máximas
10
11 %Parámetros de Salida:
12 %   xk = aproximación de los valores de convergencia de la función f
13 %   k = número de iteraciones realizados
14 %   error = ||gradiente(f(x_k))||
15
16 %Se inserta la ecuación del problema de ingeniería seleccionado en el metodo bfgs realizado en el archivo p3_bfgs.m
17
18 [xk, k, error] = p3_bfgs('(x + 1.7*y)*sin(x) - 1.5*z - 0.1*m*cos(m+n-x) + 0.2*(n**2) - y - 1', ['x' 'y' 'z' 'm' 'n'], 10**-5, 50) #Dolan Function
19
20
21 [xk, k, error] = p3_bfgs('((60*10*x/(8*(x+22)+10*x+120))**2)/x - y', ['x' 'y'], 10**-5, 15)
22
23 end
24
Línea: 21 Columna: 94 Codificación: SYSTEM (CP1252) Fin de línea: CRLF
Documentación Editor de variables Editor
Ventana de comandos
p3_bfgs at line 85 column 14
p3_solucion_aplicacion at line 21 column 19
xk =
    14.836
    18.000
k = 15
error = 1.0017
ans =
    14.836
    18.000
>> |

```

Figura 4: Evaluación del algoritmo BFGS para la máxima transferencia de potencia.

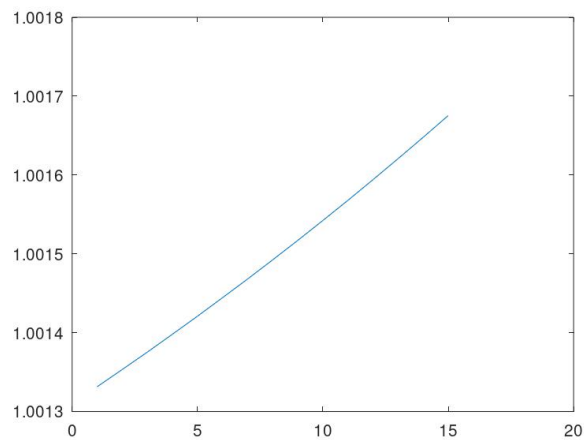


Figura 5: Gráfica de error para la función de máxima transferencia de potencia.

Referencias

- [1] Steven C. Chapra, Raymond P. Canale (2007) *Métodos numéricos para ingenieros*, "Mc Graw Hill, 5ta ed.