Métodos Numéricos para la Ciencia y la Ingeniería

Camila Castillo Pinto. RUT 18.889.762-2

27 de Octubre, 2015.

1 Introducción.

En esta tarea se pedía integrar la ecuación de Poisson para el potencial electrostático dentro de una caja de 10 $[cm]$ x 15 $[cm]$ conectada a tierra. La ecuación a integrar es:

$$\nabla^2 V(x,y) = -\rho(x,y) \tag{1}$$

Dentro de la caja hay una línea que cumple con condiciones de borde derivativas:

$$\frac{dV}{dn} = \pm 1 \tag{2}$$

tomando el signo positivo cuando la derivada perpendicular a la linea es calculada encima de ella y el signo negativo cuando la derivada es calculada bajo ella.

Además dentro de la caja hay una letra 'c' (por la primera letra del nombre del alumno) contenida dentro del rectngulo centrado con lados 5 $[cm]$ x 7 $[cm]$. Dentro de la letra la densidad es constante y la carga total es de 1 $[C]$:

$$Q = \int \rho(x,y) dxdy = 1[C] \tag{3}$$

Pero como la densidad de carga es constante:

$$Q = 1[C] = \rho \int dxdy = \rho A \tag{4}$$

donde A corresponde al área. La ecuación (4) nos entrega una forma de calcular la densidad de carga constante ρ cuando se conoce el área de la letra.

1.1 Procedimiento

El enunciado pedía usar un reticulado con $h = 0.2[cm]$ e implementar el método de sobrerelajación sucesiva con distintos valores de w.

La implementación del código se hizo de forma gradual, desde lo más sencillo hasta lo más complejo. Para ver este proceso, se recomienda revisar el historial de comentarios del código.

La implementación del método consistió en:

- En primer lugar se definieron las dimensiones de la grilla (10 $[cm]$ x 15 $[cm]$), el número de pasos y usando el h que se daba por enunciado. Cabe destacar que como en este caso la grilla no es cuadrada, se ocupan diferente número de paso para x y para y. La grilla se representa por una matriz de arreglos (arreglo de arreglos).

- Se define una función que retorne el valor de ρ para cada par (x, y). Recordar que dentro de la región que define la letra 'C' la densidad de carga es constante y fuera de ella la densidad de carga es nula. De esta manera, usando la ecuación (4), podemos calcular la densidad dentro de la letra, que da como resultado $\rho = 1/15[C/cm^2]$, donde el área que ocupa la letra 'C' es $15[cm^2]$.

- Se define una función que corresponda a una iteración del método de sobrerelajación para el rectángulo. Esta iteración se divide por sectores del rectángulo principal, ya que tenemos dos situaciones en él: por una parte tenemos la letra 'C' y por otra parte tenemos una línea que cumple una condición derivativa.

  En primer lugar dividimos en dos sectores: sector de la línea (condición derivativa) y sector restante. El sector restante se recorre utilizando la fórmula para la sobrerelajacion (vista en clases con más detalles):

$$V_{i,j} = (1-w)V_{i,j} + \frac{w(V_{i+1,j} + V_{i-1,j} + V_{i,j+1} + V_{i,j-1} + h^2\rho(x,y))}{4} \tag{5}$$

  El sector de la línea debe ser dividido a su vez en dos: encima de la línea y debajo de la línea. Para ambos sectores se recorre utilizando la fórmula para sobrerelajación cuando hay condiciones tipo Neumann (visto en clases):

$$V_{i,j} = (1-w) * V_{i,j} + \frac{w(V_{i+1,j} + v_{i-1,j} + v_{i,j-1} + h^2\rho(x,y) + hg_i)}{3} \tag{6}$$

  donde g_i corresponde al valor de la derivada. En el caso encima de la línea $g_i = +1$ y en el caso debajo de la línea $g_i = -1$.

- Se implementa un algoritmo que itere la función del punto anterior, en un cierto número o bajo un criterio de convergencia. En este algoritmo se guardan los valores resultantes de V.

- Finalmente, se grafican los resultados utilizando algoritmos de gráficos vistos en clases. Es importante notar que la matriz V que se obtuvo luego de todas las iteraciones estaba girada, por lo que se tuvo que transponer la matriz V para graficarla.

Además se probaron los valores de $w = 1$, $w = 0.6$, $w = 1.2$ y $w = 1.4$, ya que la solución se vuelve estables para valores de w entre 0 y 2, y los valores recomendados (en clase) son $w = 1.2$ y $w = 1.4$. //

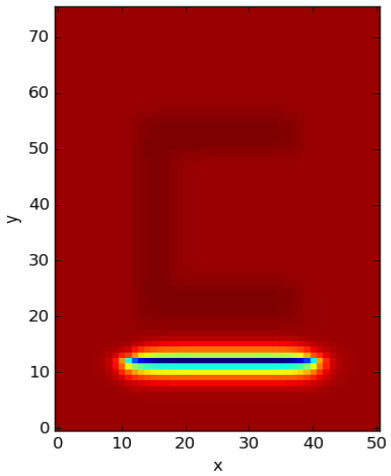
El criterio de convergencia utilizado consistió en tomar los puntos de la grilla que no fuesen nulos, es decir tomar los puntos de la letra y los puntos de la línea, y se calcula el valor absoluto de la diferencia entre los valores de cada punto en la grilla anterior y en la

grilla actual. Se toma aquella diferencia entre puntos que sea máxima y se compara con un valor determinado de tolerancia. Si de esta comparación resulta que aquel máximo es mayor que la tolerancia, entonces aún no ha convergido (retorna True); en cambio, si el máximo es menor o igual que la tolerancia, entonces la solución ha convergido (retorna False) y no se vuelve a iterar sobre ella.

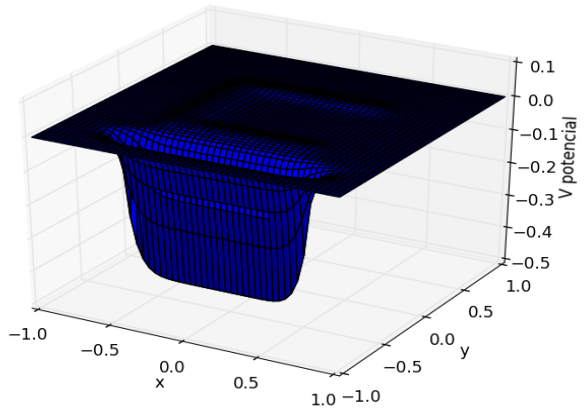
Los valores para la tolerancia que se usaron fueron: 10^{-1}, 10^{-3} y 10^{-5}. Los últimos dos fueron probados sólo para $w = 1.2$ y $w = 1.4$.

1.2 Resultados y Análisis

En primer lugar se probó el algoritmo para 20 iteraciones, para cada valor de w. Las figuras 1, 2, 3 y 4 muestran lo obtenido.



(a) Gráfico de profundidad para la solución, w=0.6



(b) Gráfico en 3D. w=0.6

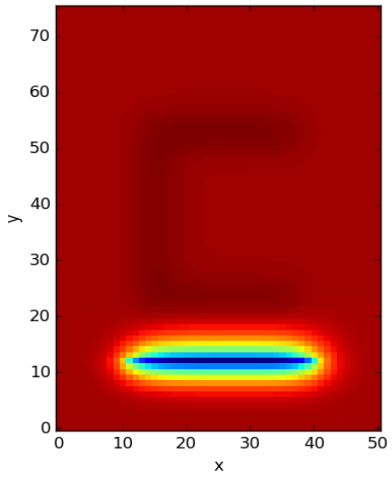
Figure 1: Solución obtenida luego de 20 iteraciones, para w=0.6.

En las figuras 1, 2, 3 y 4 se logra observar claramente la letra 'C' y la línea donde está la condición de borde. Además se puede notar que para el valor de $w = 0.6$ la línea es delgada y se va engrosando a medida que usamos un w más alto, $w = 1.4$ por ejemplo.

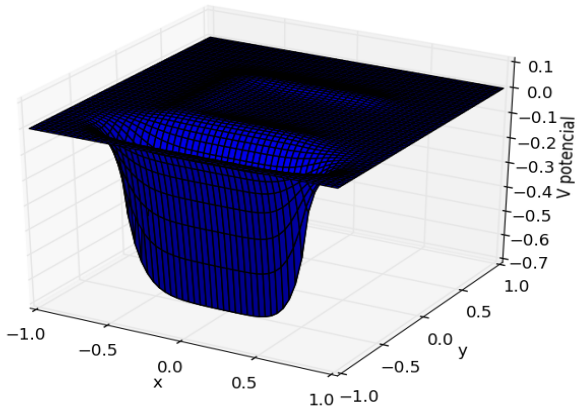
Luego se probó con el criterio de convergencia y se evaluó en cuántas iteraciones convergía la solucion para cada valor de w. El valor de tolerancia usado en esta etapa fue de $1 \cdot 10^{-1}$.

Para $w = 0.6$ la solución convergió luego de 2075 iteraciones, bajo la tolerancia dada.
Para $w = 1.0$ la solución convergió luego de 1726 iteraciones, bajo la tolerancia dada
Para $w = 1.2$ la solución convergió luego de 1141 iteraciones, bajo la tolerancia dada.
Para $w = 1.4$ la solución convergió luego de 897 iteraciones, bajo la tolerancia dada.

Se puede observar que con los valores de w recomendados en clase, la solución converge más rápido que los valores de w más bajos, para la tolerancia que se definió.

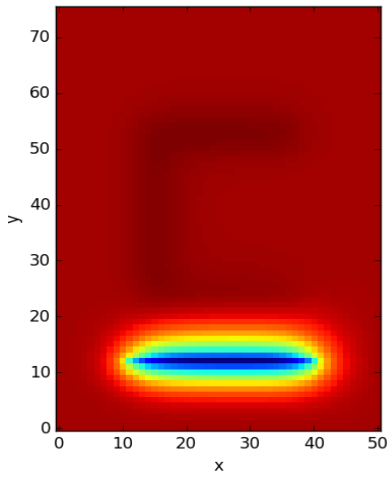


(a) Gráfico de profundidad para la solución, w=1.0

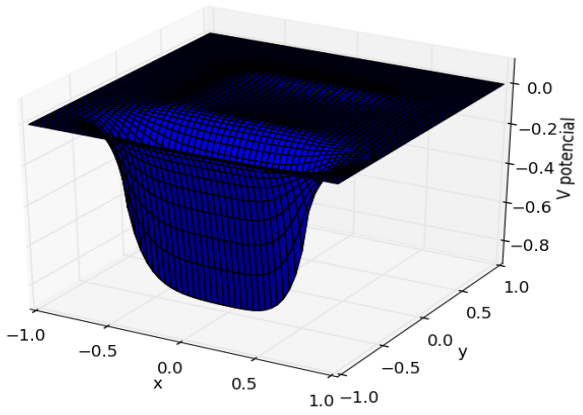


(b) Gráfico en 3D. w=1.0

Figure 2: Solución obtenida luego de 20 iteraciones, para w=1.0.



(a) Gráfico de profundidad para la solución, w=1.2



(b) Gráfico en 3D. w=1.2

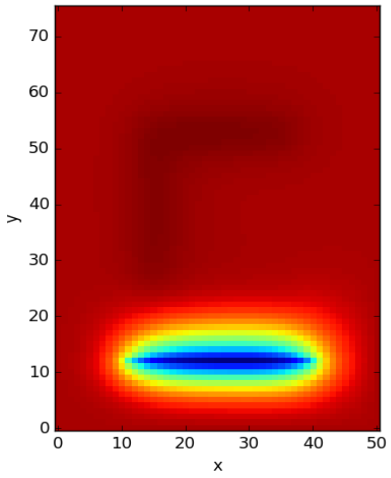
Figure 3: Solución obtenida luego de 20 iteraciones, para w=1.2.

La figura 5 muestra la solución una vez que convergió bajo este criterio, para $w = 1.4$.

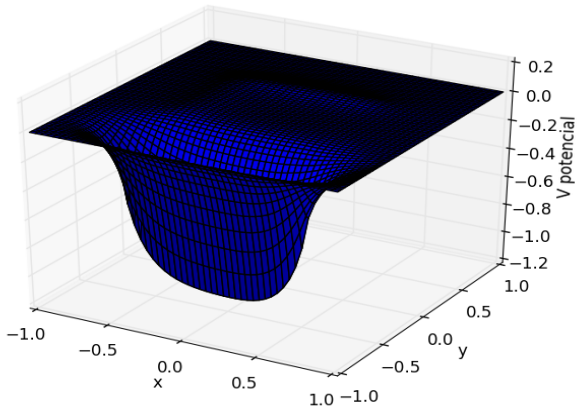
De la figura 5, podemos observar que ya no se nota la letra 'C' y la línea se muestra con más definición.

Cabe destacar que los gráficos de la solución para los otros valores de w, una vez que convergieron, es el mismo que el que se muestra en la Figura 5.

Como ya se mencionó, los valores de w que fueron recomendados en clase convergieron en menor número de iteraciones, por lo que se procedieron a utilizar estos dos de aquí en

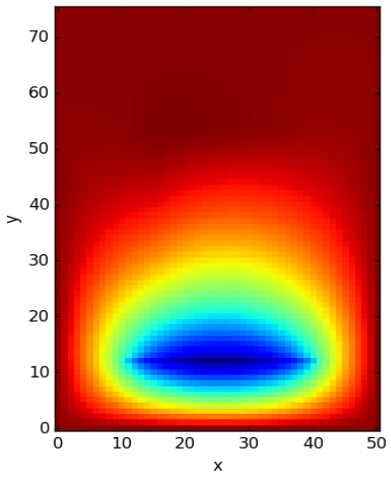


(a) Gráfico de profundidad para la solución, w=1.4

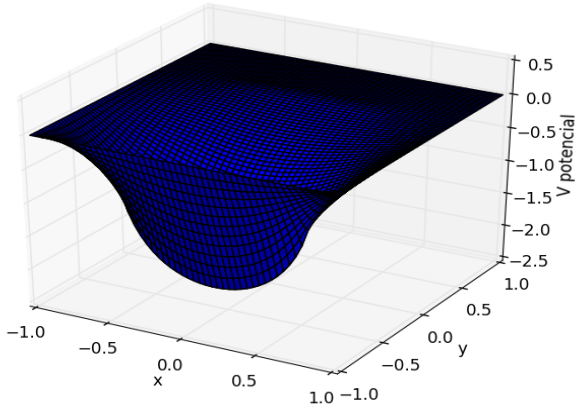


(b) Gráfico en 3D. w=1.4

Figure 4: Solución obtenida luego de 20 iteraciones, para w=1.4.



(a) Gráfico de profundidad para la solución, w=1.4



(b) Gráfico en 3D. w=1.4

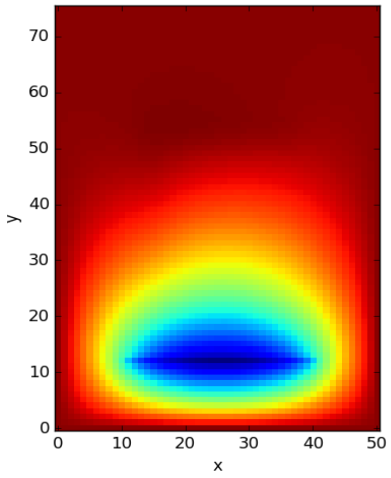
Figure 5: Solución obtenida luego de 897 iteraciones, para w=1.4, con tolerancia $1 \cdot 10^{-1}$.

adelante para evaluarlos bajo un criterio de convergencia más estricto.

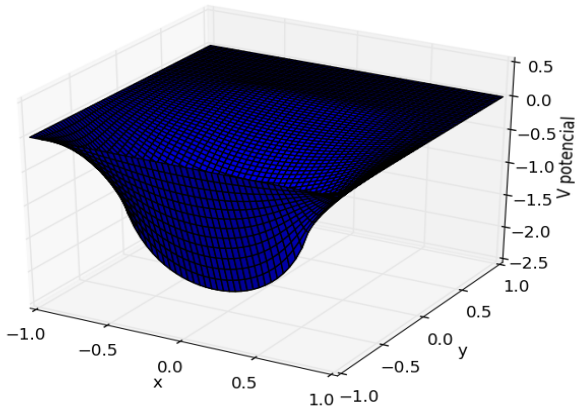
A continuación se utilizó un valor para la tolerancia de $1 \cdot 10^{-3}$.

Para $w = 1.2$ la solución convergió luego de 2460 iteraciones, bajo la tolerancia dada. Para $w = 1.4$ la solución convergió luego de 1617 iteraciones, bajo la tolerancia dada.

La Figura 6 muestra la solución obtenida una vez que ésta convergiera, para $w = 1.2$ (el gráfico de la solución obtenida con $w = 1.4$ fue el mismo).



(a) Gráfico de profundidad para la solución, w=1.2



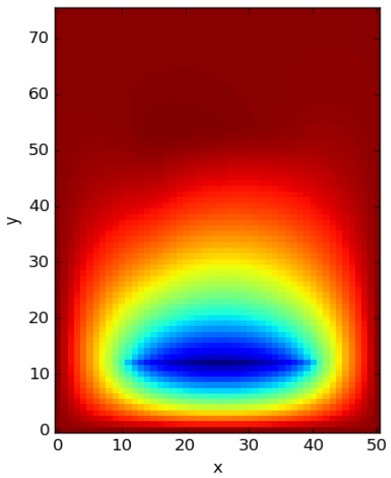
(b) Gráfico en 3D. w=1.2

Figure 6: Solución obtenida luego de 2460 iteraciones, para w=1.2, con tolerancia de $1 \cdot 10^{-3}$.

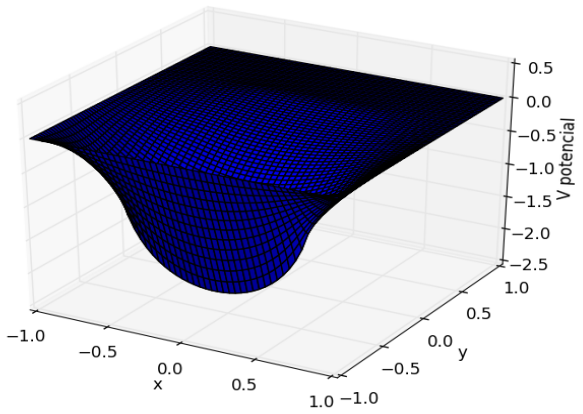
Finalmente se probó con una tolerancia de $1 \cdot 10^{-5}$, para los valores de w recomendados.

Para $w = 1.2$ la solución convergió luego de 3371 iteraciones, bajo la tolerancia dada. Para $w = 1.4$ la solución convergió luego de 2209 iteraciones, bajo la tolerancia dada.

La figura 7 muestra la solución que se obtuvo, luego de que convergiera, para $w = 1.4$.



(a) Gráfico de profundidad para la solución, w=1.4



(b) Gráfico en 3D. w=1.4

Figure 7: Solución obtenida luego de 2209 iteraciones, para w=1.4, con tolerancia de $1 \cdot 10^{-5}$.

1.3 Conclusiones

Se concluye que el método de sobrerelajación sucesiva fue implementado correctamente y se mantuvo estable para los valores de w utilizados, lo cual se esperaba ya que fue información que se entregó en clases.

También se pudo corroborar que la solución convergía, luego de cierto número de iteraciones, por lo que se pudo implementar un criterio de convergencia que, finalmente, sí funcionó para evaluar bajo cuántas iteraciones convergía la solución para distintos valores de w.

Además corroboramos que $w = 1.2$ y $w = 1.4$ son los valores que se recomienda usar, ya que convergen mucho más rápido. En particular $w = 1.4$, ya que, para la tolerancia más baja que se utilizó ($1 \cdot 10^{-5}$), la solución convergió aproximadamente 1000 iteraciones antes que para $w = 1.2$.

Finalmente es importante recordar que un código bien implementado es fundamental, en especial en este tipo de problemas, ya que lo escencial es encontrar una solución bajo un criterio de convergencia, es decir, luego de una gran cantidad de iteraciones. El tiempo que tarde el programa en ejecutarse para una gran cantidad de iteraciones pasará principalmente por cómo esté implementado éste, por lo que no hay que descuidar este factor. Es en este sentido donde cabe destacar que se cree que el programa implementado pudo haber sido mejorado, de manera que fuese más eficiente, en puntos como: el rango en donde se iteraba con la fórmula de sobrerelajación (ya que en ciertas zonas siempre iba a valer nula la solución), la forma para definir dónde hay carga y dónde no.