Libro: Quarteroni, "Scientific Computing with MATLAB and Octave"

EJERCICIO 1.10:

La siguiente secuencia converge a π :

$$Z_2 = 2$$

$$Z_{n+1} = 2^{n-1/2}\sqrt{1-\sqrt{1-(4^{1-n})(Z_n)^2}}\text{ , para n=2,3,...} \qquad (1)$$

Con el MATLAB se obtienen los siguientes resultados:

n	Z_n
2	2
3	2,82842712474619
4	3,06146745892072
5	3,12144515225805
6	3,13654849054593
7	3,14033115695474
8	3,14127725093276
9	3,14151380114415
10	3,14157294036557
11	3,14158772527996
12	3,14159142150463
13	3,14159234561108
14	3,14159257654500
15	3,14159263346325
16	3,14159265480759
17	3,14159264532122
18	3,14159260737572
PI	3,14159265358979

Z_16 tiene 9 cifras significativas exactas, es la mejor aproximación a PI.

Con Z_16, el error relativo es:

|z_16-PI|/PI = |1,21779697437319E-09|/PI = 3,87636816307694E-10

Note que el error relativo (para Z_16) es aproximadamente 10^(-n) con n=9=número de cifras significativas exactas que tiene Z_16.

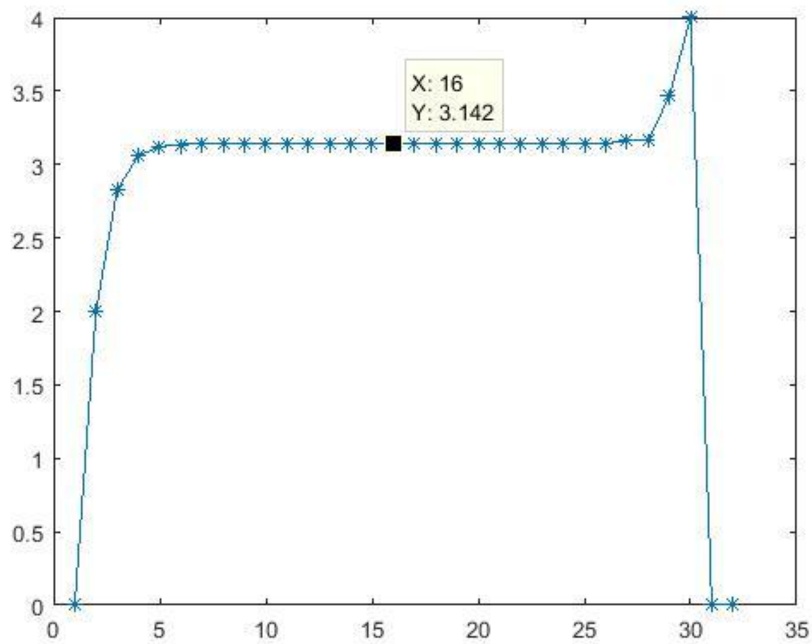

Figura 1. Z_n versus n

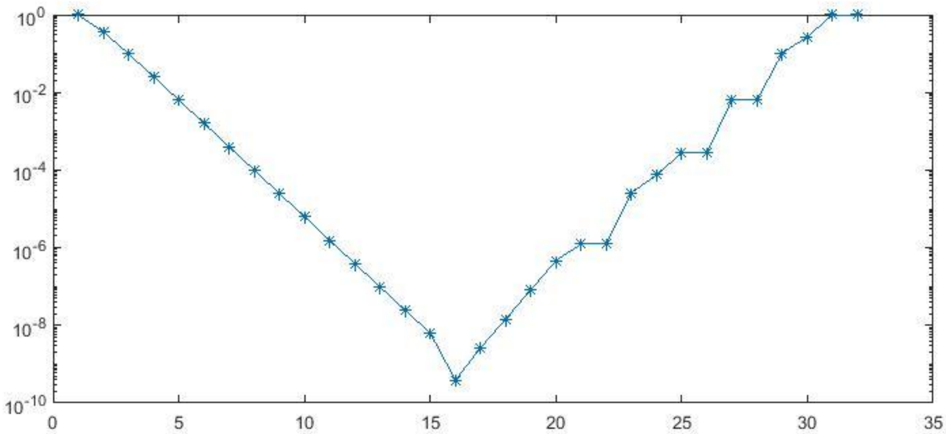

Figura 2. Error relativo versus n

Durante el cálculo, el error de redondeo (roundoff) se va acumulando debido a las operaciones de resta, potencia, raíz cuadrada. Y a partir de Z_17 el error relativo es cada vez mayor.

Notamos además que,

1 - 4^(1-n)*(Z_n)^2

en la ecuación (1) es, según el MATLAB, igual a uno cuando n=30. Vea a continuación:

```
>> format long e

>> 4^(1-30)*Z(30)^2

ans =

   5.551115123125783e-17
```

```
>> r=1-ans

r =

     1
```

El número r=1 - 4^(1-30)*(Z_30)^2 tiene 17 cifras significativas (o más), pero sabemos que el computador (precisión doble) representa de 15 a 16 cifras significativas de manera exacta. Y en este caso r es representado por 1. Luego, Z_(n+1) en la ecuación (1) es cero para n >=30 (Ver también Figura 1).

EJERCICIO 1.12:

La siguiente serie converge a π :

$$\lim_{N\to\infty} S_N = \lim_{N\to\infty} \sum_{n=0}^{N-1} 16^{-n}\left(\frac{4}{8n+1} - \frac{2}{8n+4} - \frac{1}{8n+5} - \frac{1}{8n+6}\right) \qquad (2)$$

El símbolo S_N representa a la suma de los N primeros sumandos de la serie.

Con el MATLAB se obtienen los siguientes resultados:

N	S_N
1	3,13333333333333
2	3,14142246642247
3	3,14158739034658
4	3,14159245756744
5	3,14159264546034
6	3,14159265322809
7	3,14159265357288
8	3,14159265358897
9	3,14159265358975
10	3,141592653589791
11	3,141592653589793
12	3,141592653589793
13	3,141592653589793
14	3,141592653589793
15	3,141592653589793
PI	3,141592653589793

S_11 (la suma de los 11 primeros sumandos) tiene las 16 cifras significativas de la variable “pi” almacenada en el MATLAB. Así, el error relativo para S_11 es cero.

S_10 tiene 15 cifras significativas exactas, y su error relativo (ver Figura 4) es aproximadamente 10^(-n) con n=15=número de cifras significativas exactas que tiene S_10.

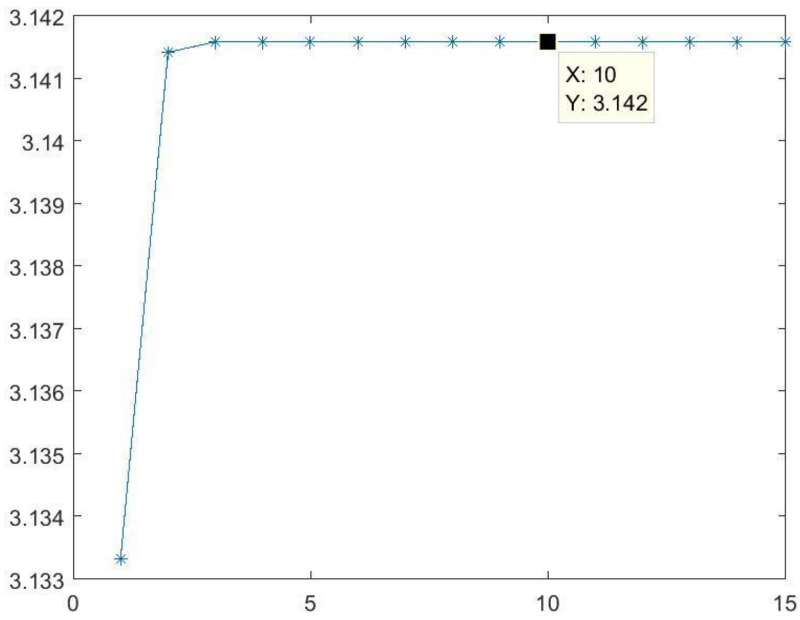

Figura 3. S_N versus N

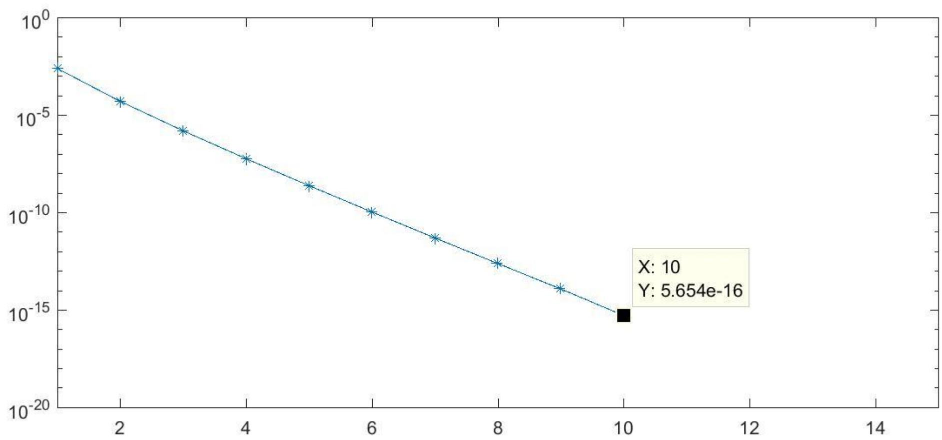

Figura 4. Error relativo versus N

Parece que el error de redondeo (debido a las operaciones de punto flotante) está controlado. El error predominante es la diferencia entre S_N y su límite.

Es más, el error relativo llega a ser cero pues, como se ve en la tabla anterior, S_N se mantiene constante e igual a la variable "pi" del MATLAB (para N>=11). Más explícitamente tenemos que, de la ecuación (2):

S_(N+1) = S_N + 16^(-N)*(4/(8N +1) -2/(8N +4)- 1/(8N +5) - 1/(8N +6))

y, según el MATLAB, para N=11:

```
>> S(11)

ans =

   3.141592653589793e+00
```

```
>> 16^(-11)*( 4/(8*11 +1) -2/( 8*11 +4)- 1/( 8*11 +5) - 1/( 8*11 +6) )

ans =

   1.030971216978887e-16

>> S(12)=S(11)+ans

ans =

   3.141592653589793e+00
```

O sea S(12) es representado por S(11) a pesar de que son diferentes. La razón es la siguiente: S(12) tiene 17 cifras significativas (o más) pero sabemos que el computador (precisión doble) representa de 15 a 16 cifras significativas de manera exacta.

Libro: Süli, “An Introduction to Numerical Analysis”

EJERCICIO 1.3:

Con x_1=100 el método de Newton converge a la raíz positiva, que es aproximadamente x_105. La diferencia entre x_104 y x_105 es menor a 10^(-6).

k	x_k
100	1,69752585652667
101	1,30229522179734
102	1,16208944237534
103	1,14637611343467
104	1,14619324513471
105	1,14619322062058

El error converge linealmente en las primeras 90 iteraciones (ver Figura 5). Aparentemente a partir de allí la convergencia es cuadrática porque entra en la vecindad [ξ-h, ξ+h].

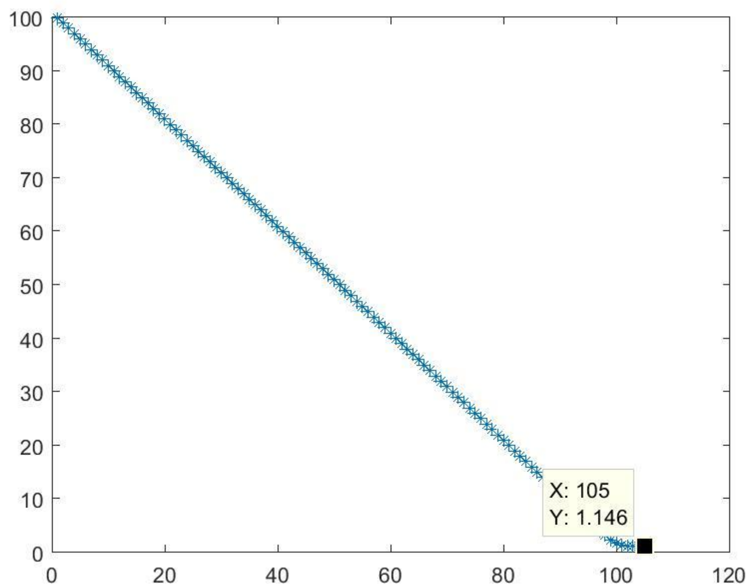

Figura 5. x_k versus k

Con $x_1=0.1$ el método de Newton también converge a la raíz positiva, que es aproximadamente x_{16}. La diferencia entre x_{15} y x_{16} es menor a 10^{-6}.

k	x_k
10	2,031484763
11	1,489039752
12	1,214107636
13	1,149400575
14	1,146200742
15	1,146193221
16	1,146193221

Notamos que x_1 es menor a la raíz y x_2 es mayor a la raíz, como era esperado. El error converge linealmente en las primeras 10 iteraciones (ver Figura 6). Aparentemente a partir de allí la convergencia es cuadrática porque entra en la vecindad $[\xi-h, \xi+h]$.

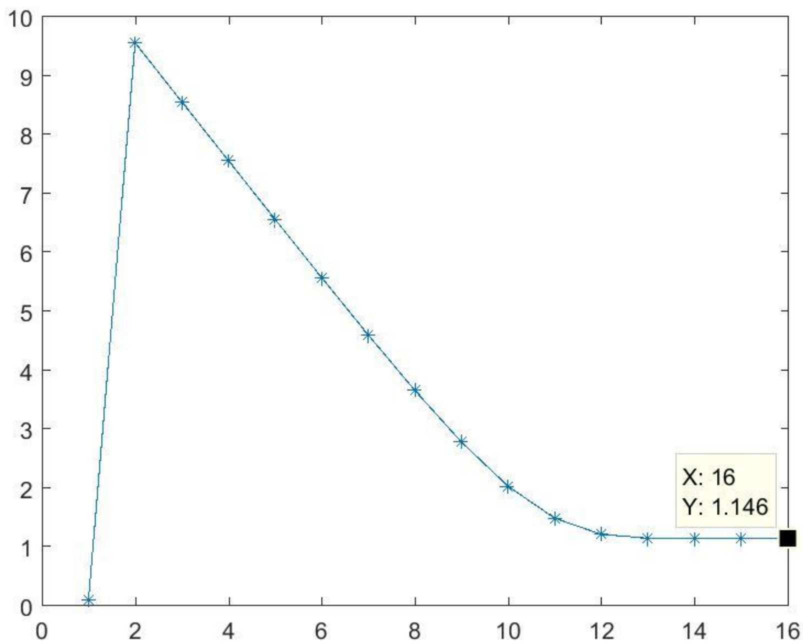

Figura 6. x_k versus k