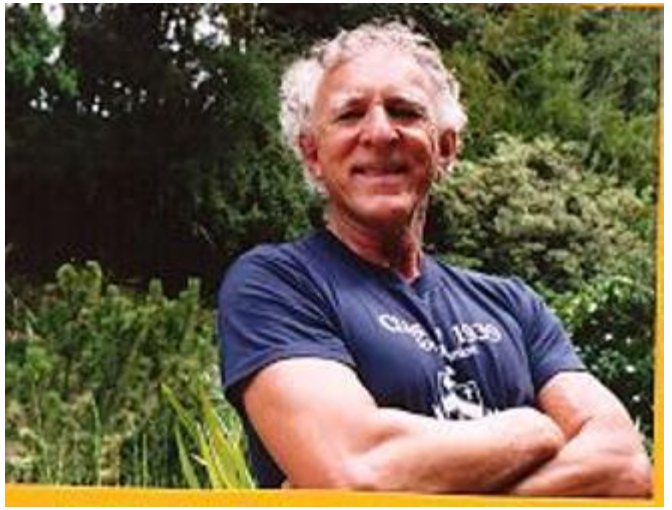
A Million-Dollar Matrix

Cleve Moler
MathWorks
Householder XXII

- Redheffer
- Möbius & Mertens
- Redheffer Matrix
- Code
- Five Ways

Redheffer

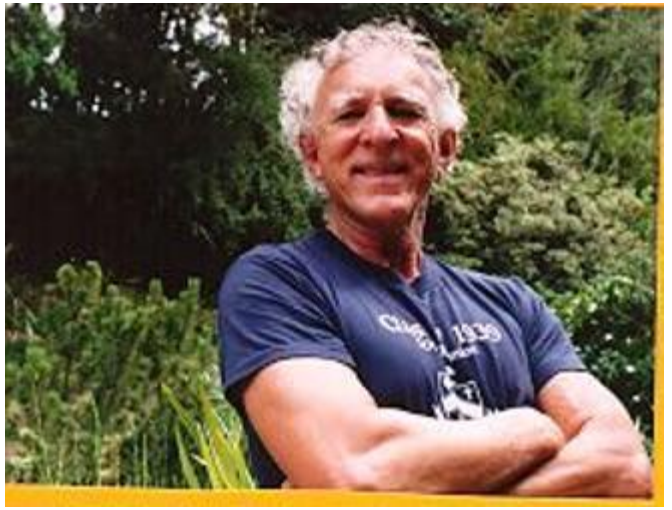


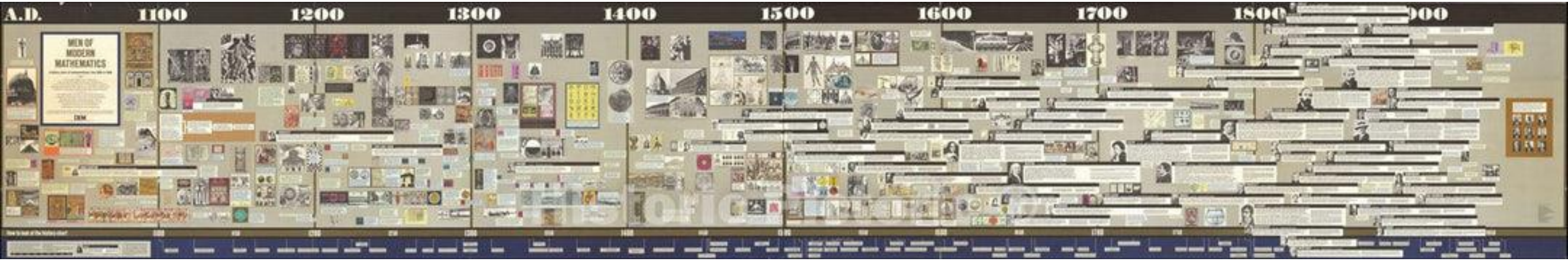
Raymond Redheffer (1921-2005)

UCLA, 1950-2000

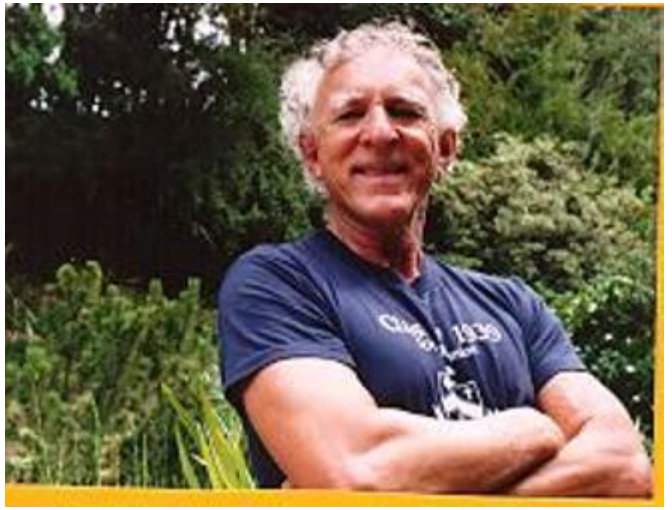
Redheffer

Raymond Redheffer (1921-2005)



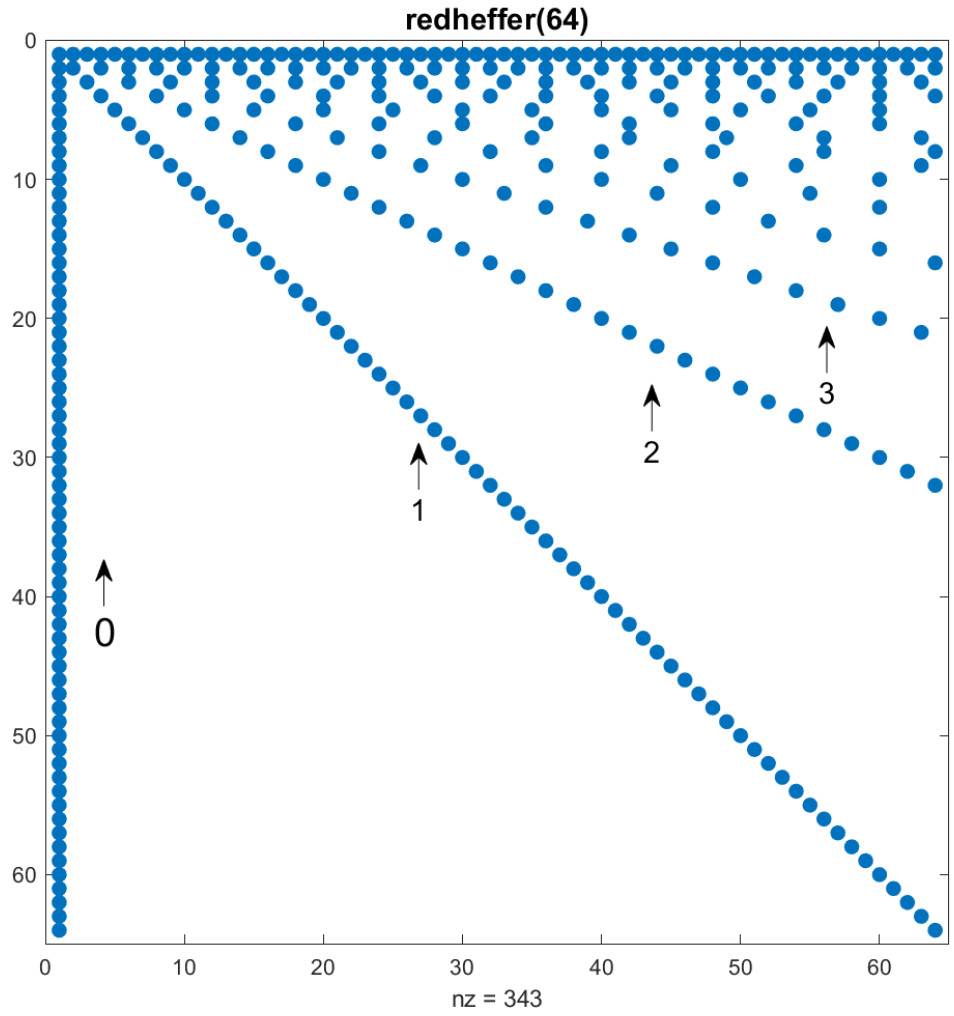


Redheffer Matrix



(1977)

R(k,j) = 1, if j == 1 or k divides j
= 0, otherwise


redheffer(64)
0
10
20
30
40
50
60
0
1
2
3
nz = 343

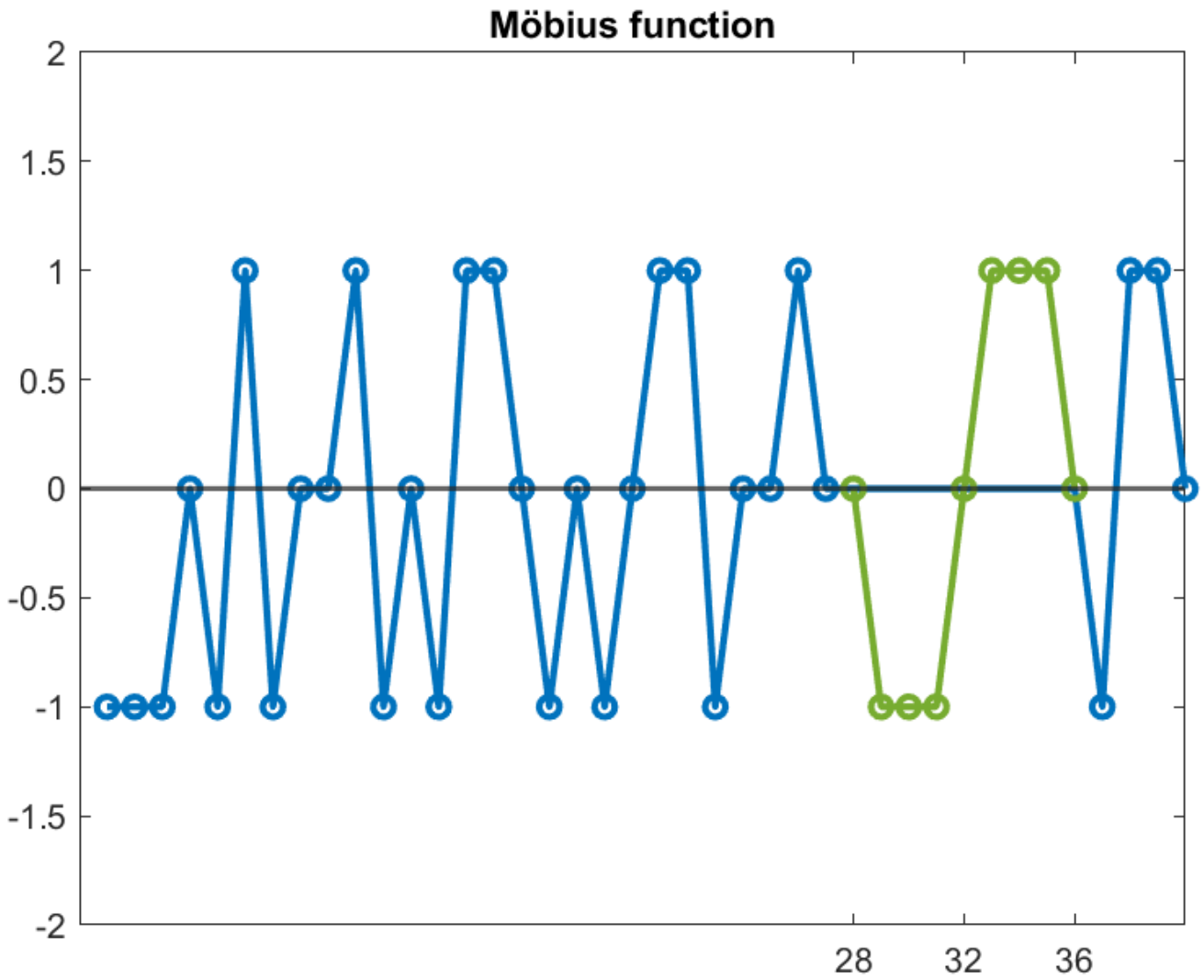
- **Möbius & Mertens**

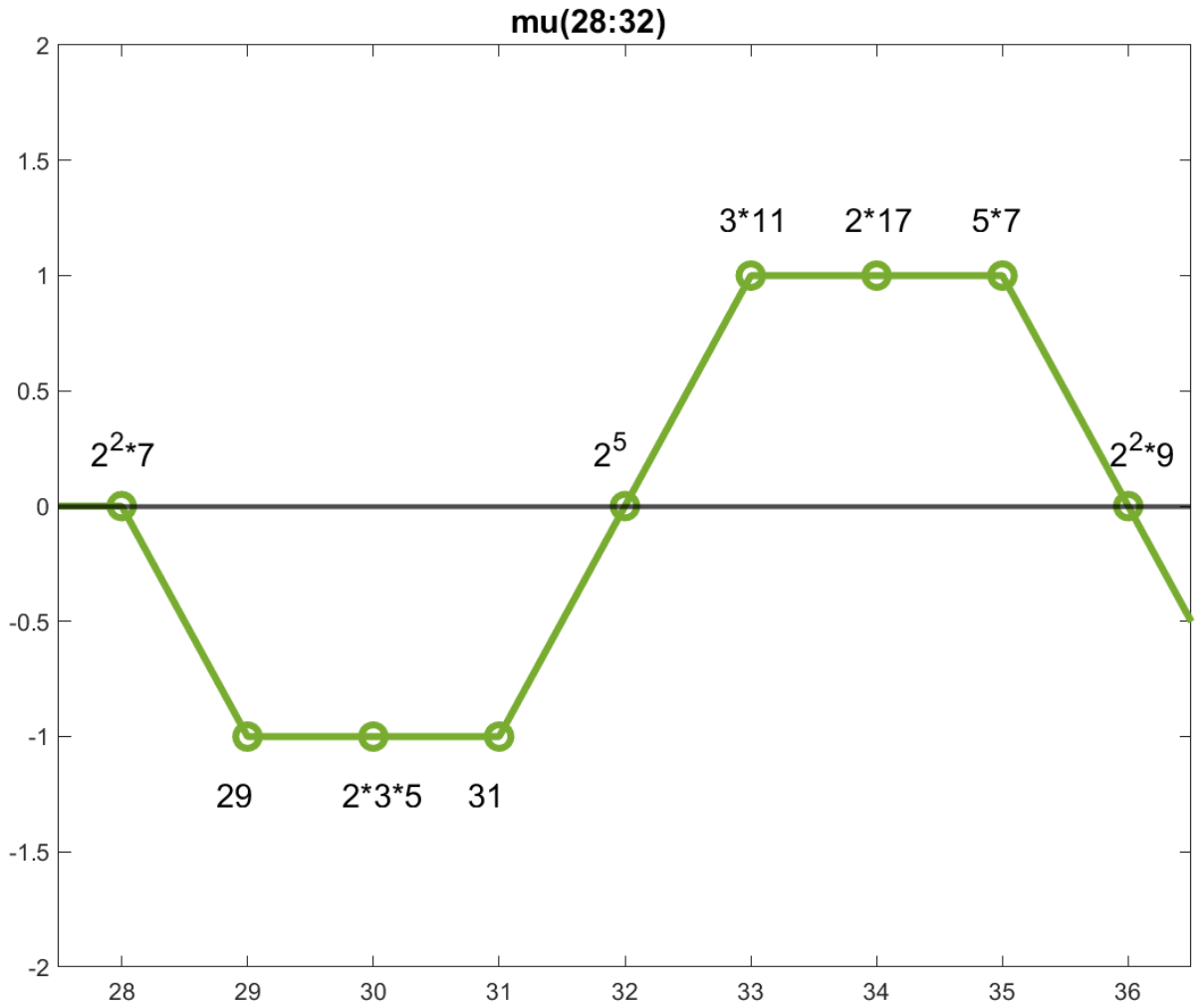
Möbius Function

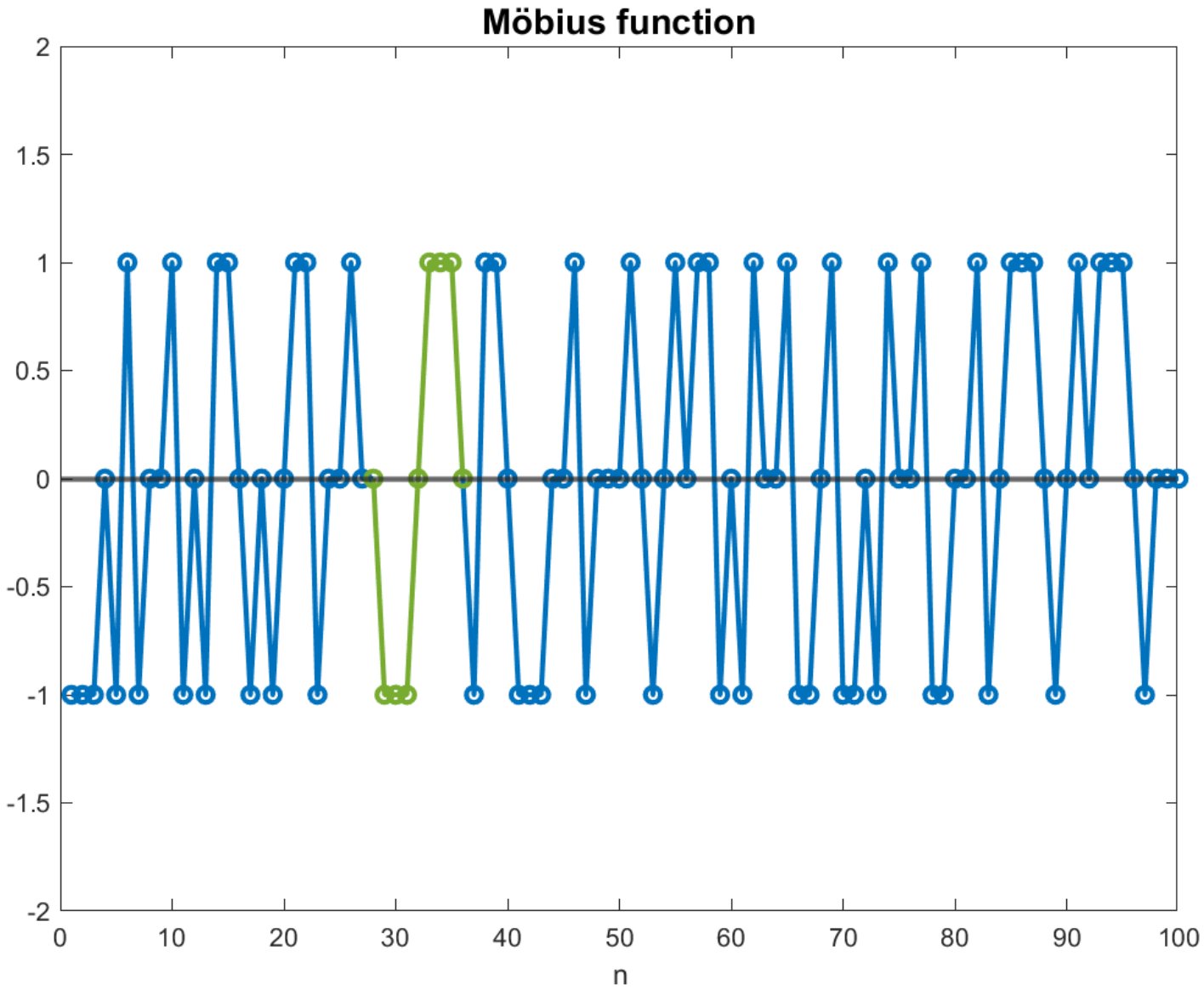
August Möbius (1832)

μ_k = (-1)^(number distinct prime factors)

= 0 repeated prime factor


Möbius function
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
28
32
36

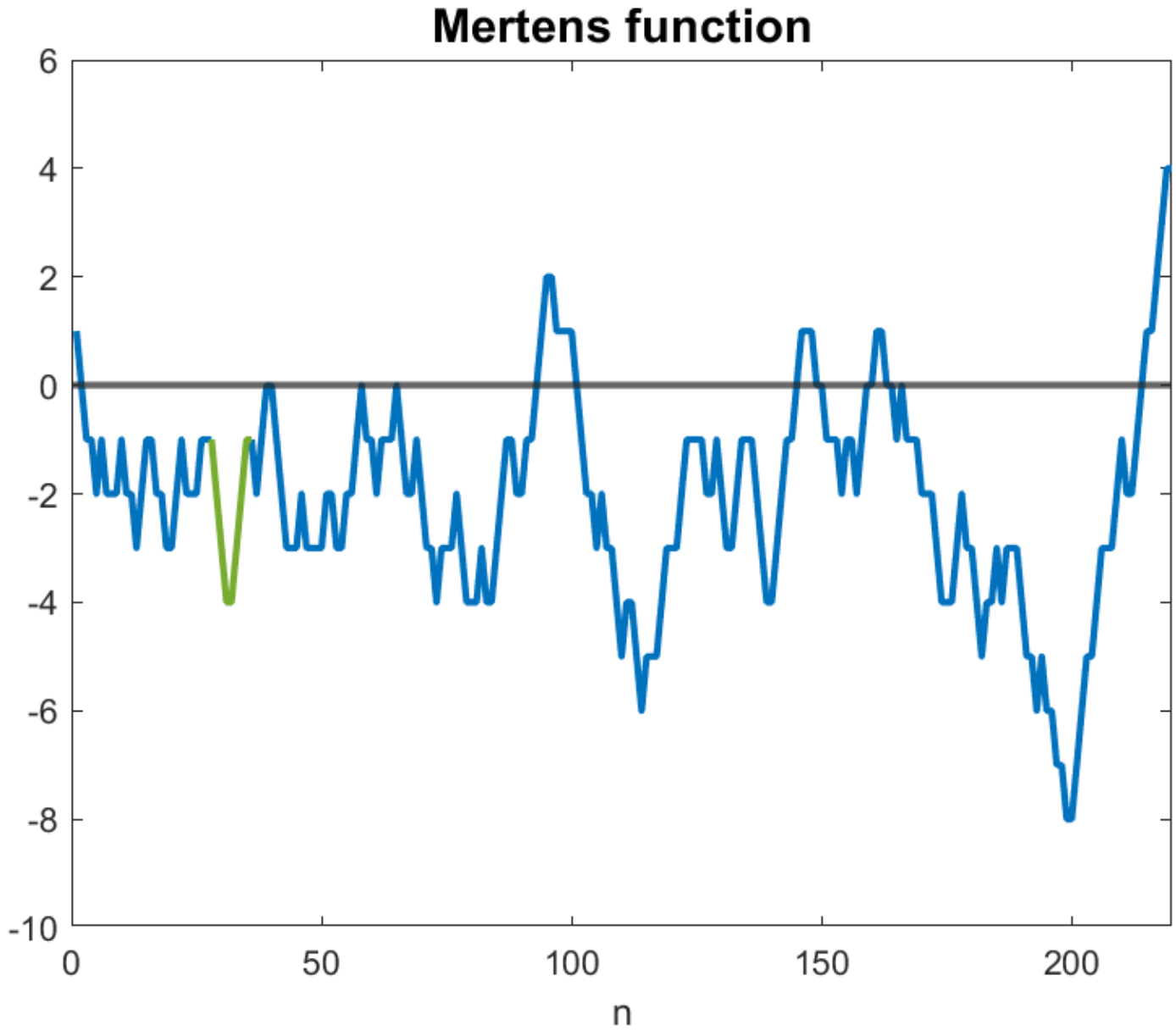

mu(28:32)
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
28
29
30
31
32
33
34
35
36
2^2*7
29
2*3*5
31
2^5
3*11
2*17
5*7
2^2*9


Möbius function
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
10
20
30
40
50
60
70
80
90
100
n

Mertens Function

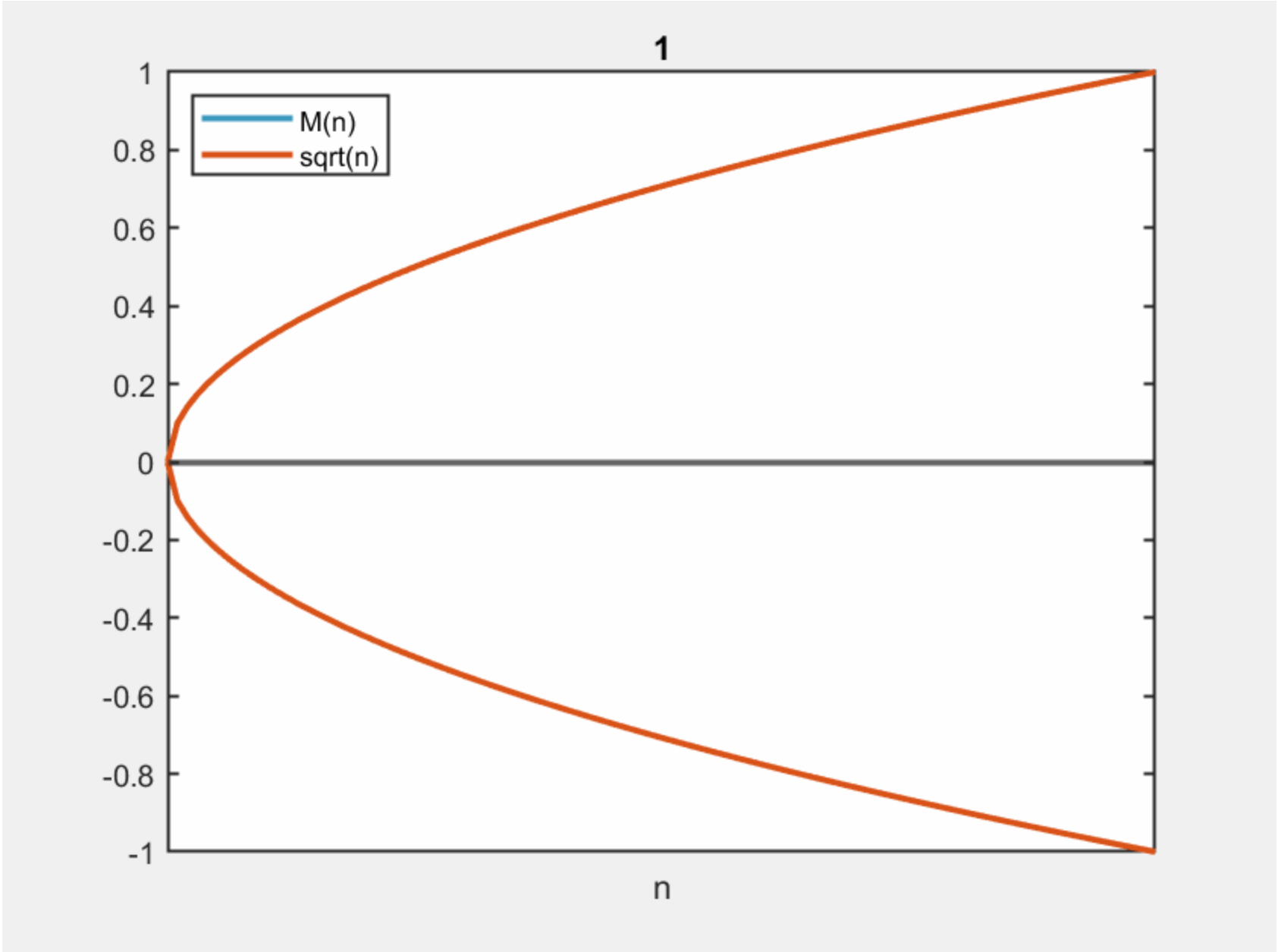
Franz Mertens (1897)

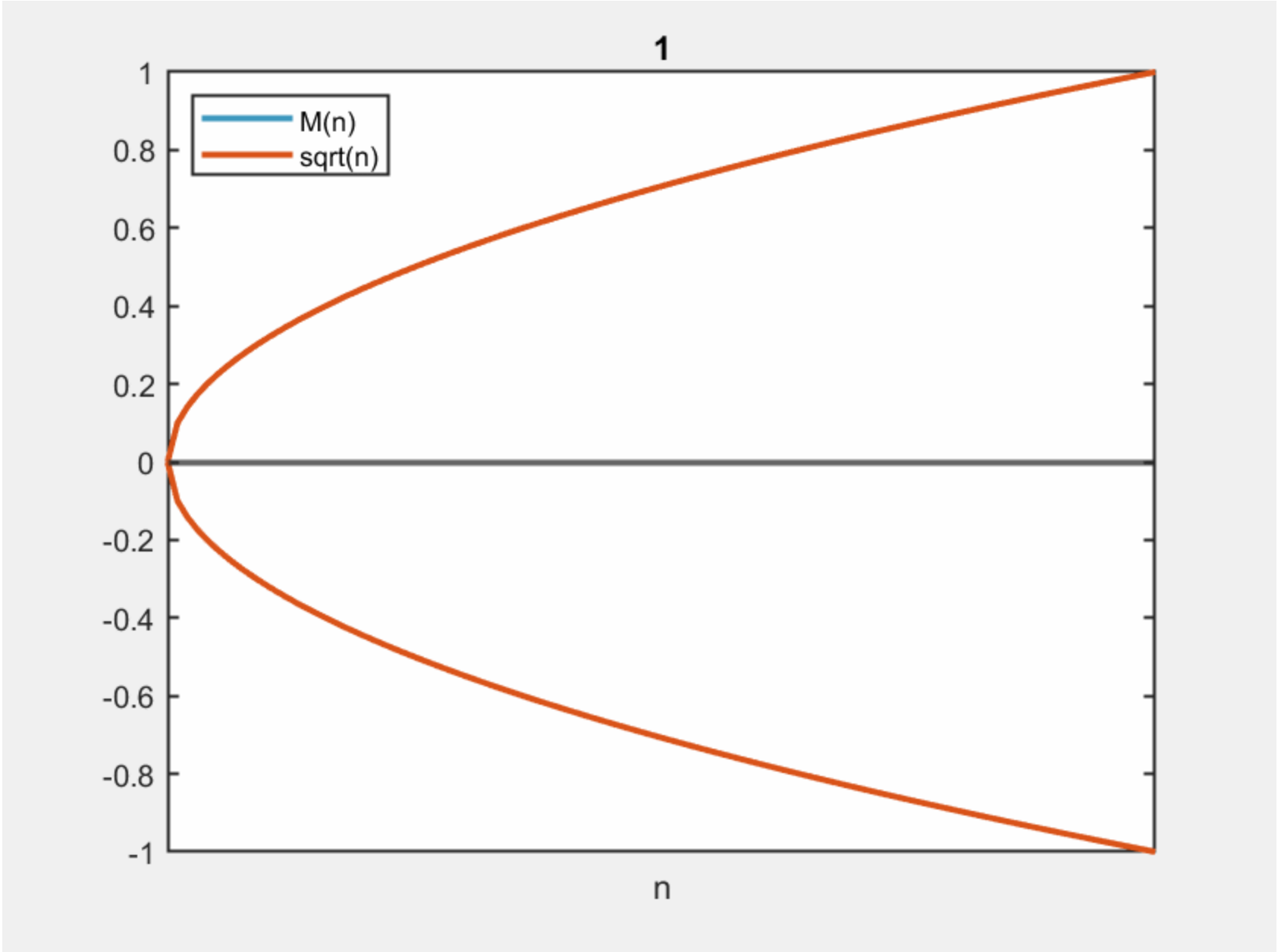
$$M_n = \sum^{n} \mu_k$$

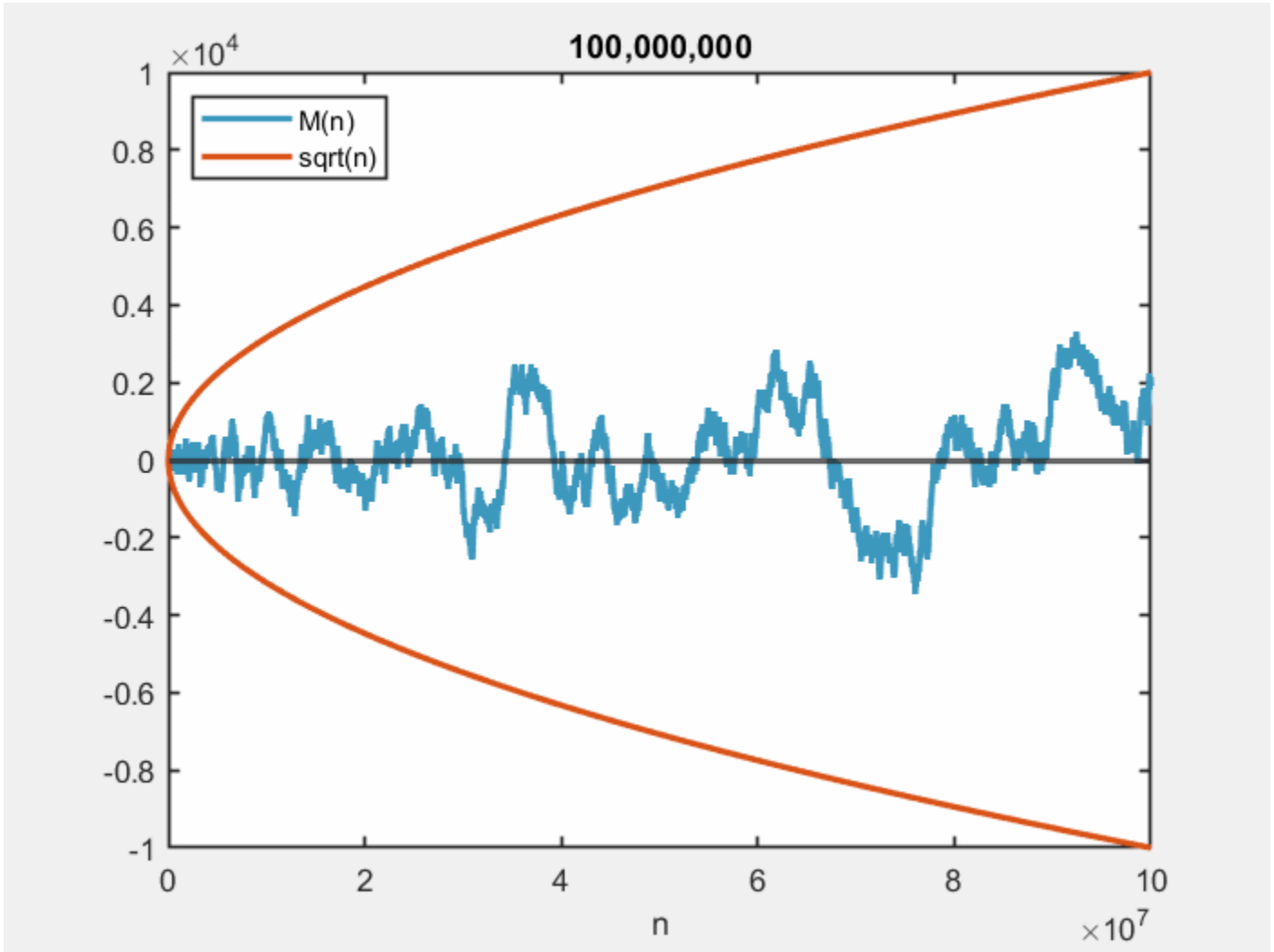

Mertens function
6
4
2
0
-2
-4
-6
-8
-10
0
50
100
150
200
n

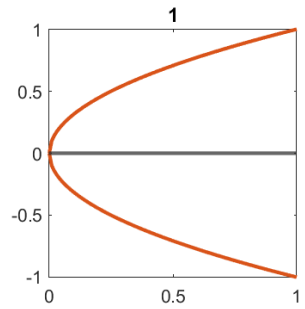
Redheffer = Mertens

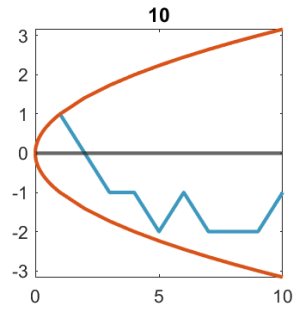
$$\det(R(n)) = M(n)$$

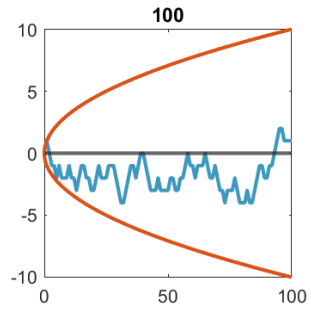

1
M(n)
sqrt(n)
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
n

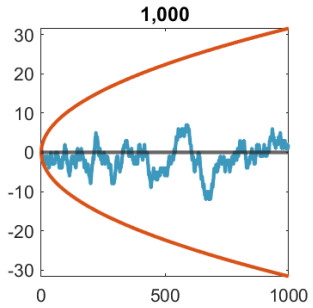

1
M(n)
sqrt(n)
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
n

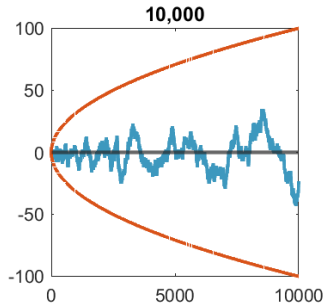

100,000,000
×10^4
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
M(n)
sqrt(n)
0
2
4
6
8
10
n
×10^7

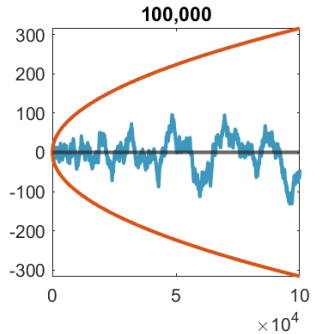

1
1
0.5
0
-0.5
-1
0
0.5
1

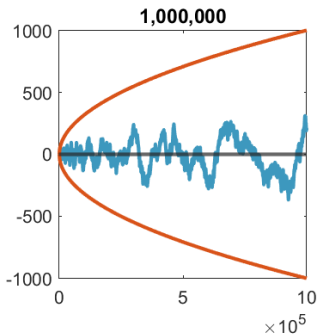

10
3
2
1
0
-1
-2
-3
0
5
10

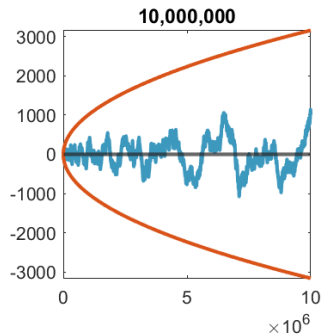

100
10
5
0
-5
-10
0
50
100

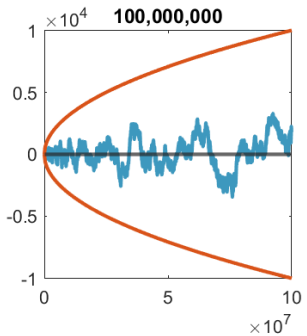

1,000
30
20
10
0
-10
-20
-30
0
500
1000


10,000
100
50
0
-50
-100
0
5000
10000


100,000
300
200
100
0
-100
-200
-300
0
5
10
×10^4


1,000,000
1000
500
0
-500
-1000
0
5
10
×10^5


10,000,000
3000
2000
1000
0
-1000
-2000
-3000
0
5
10
×10^6


×10^4
100,000,000
1
0.5
0
-0.5
-1
0
5
10
×10^7

Mertens Conjecture

$$|M(n)| < \sqrt{n}$$

Stieltjes(1885), Mertens(1897)

Mertens Conjecture

Implies Riemann Hypothesis

Redheffer

$$|\det(R(n))| < \sqrt{n}$$

Implies Riemann Hypothesis

Riemann Hypothesis

G. F. Bernard Riemann (1859)

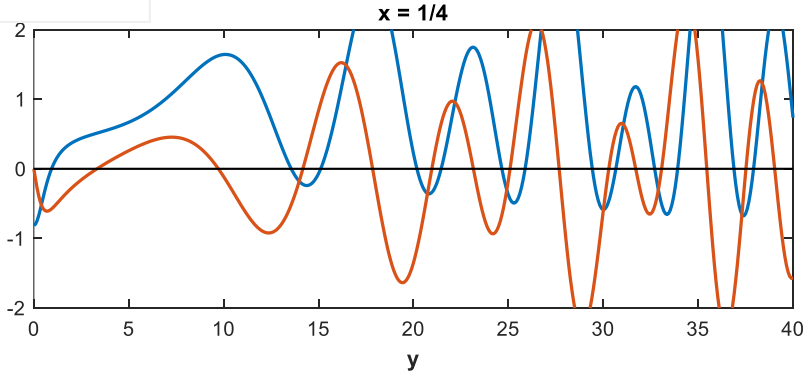
$\zeta(s) = 0$

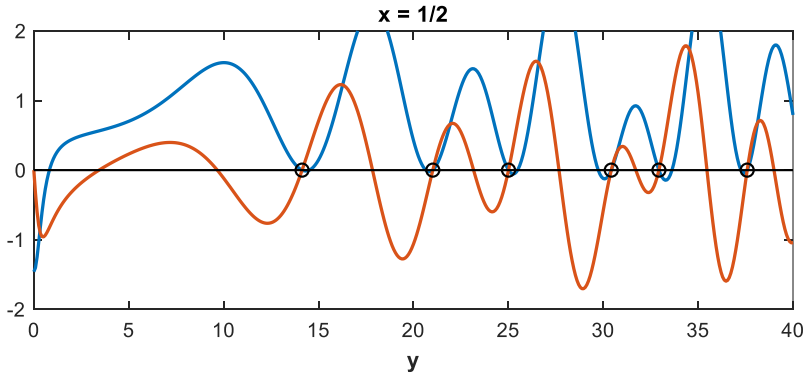
“Most important unsolved problem in math”

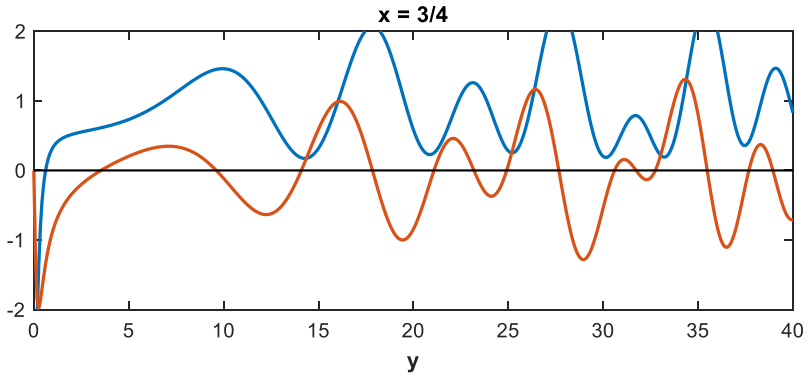
Clay Millenium Prize, $1 million

zeta(x+iy) = 0

x = 1/2







Riemann Computations

Computation of the first N non-trivial zeros of ζ(s).

authors	year	N
Riemann	1854	?
Gram	1903	10
Backlund	1914	79
Hutchinson	1925	138
Titchmarsh	1936	1,041
Turing	1953	1,104
Lehmer	1956	25,000
Meller	1958	35,337
Lehman	1966	250,000
Rosser, Yohe, Schoenfield	1969	3,502,500
Brent	1977	40,000,000
Brent	1979	81,000,001
Brent, Van_de_Lune, Te_Riele, Winter	1982	200,000,001
Van_de_Lune, Te_Riele, Winter	1986	1,500,000,001
Van_de_Lune	2001	100,000,000,000
Wedeniwski	2003	250,000,000,000
Gourdon	2004	10,000,000,000,000

Spoiler . . .

Mertens Conjecture is False

Andrew Odlyzko and Herman te Riele (1985)

$$\limsup_{n \to \infty} M(n)/\sqrt{n} > 1.06$$

Estimate $n > 10^{30}$

Andrew Odlyzko (“old lisco”)

Bell Labs, 1971-2001

Univ. Minnesota, 2001+

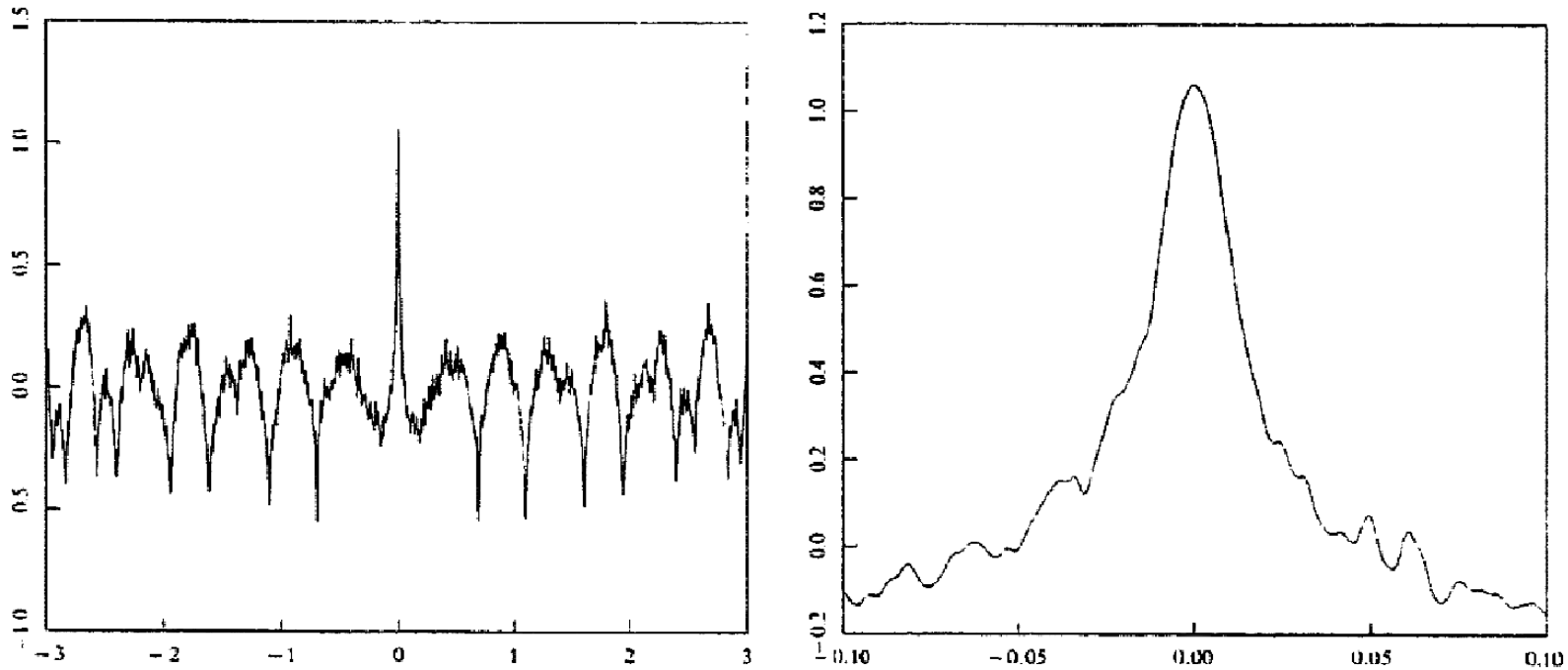
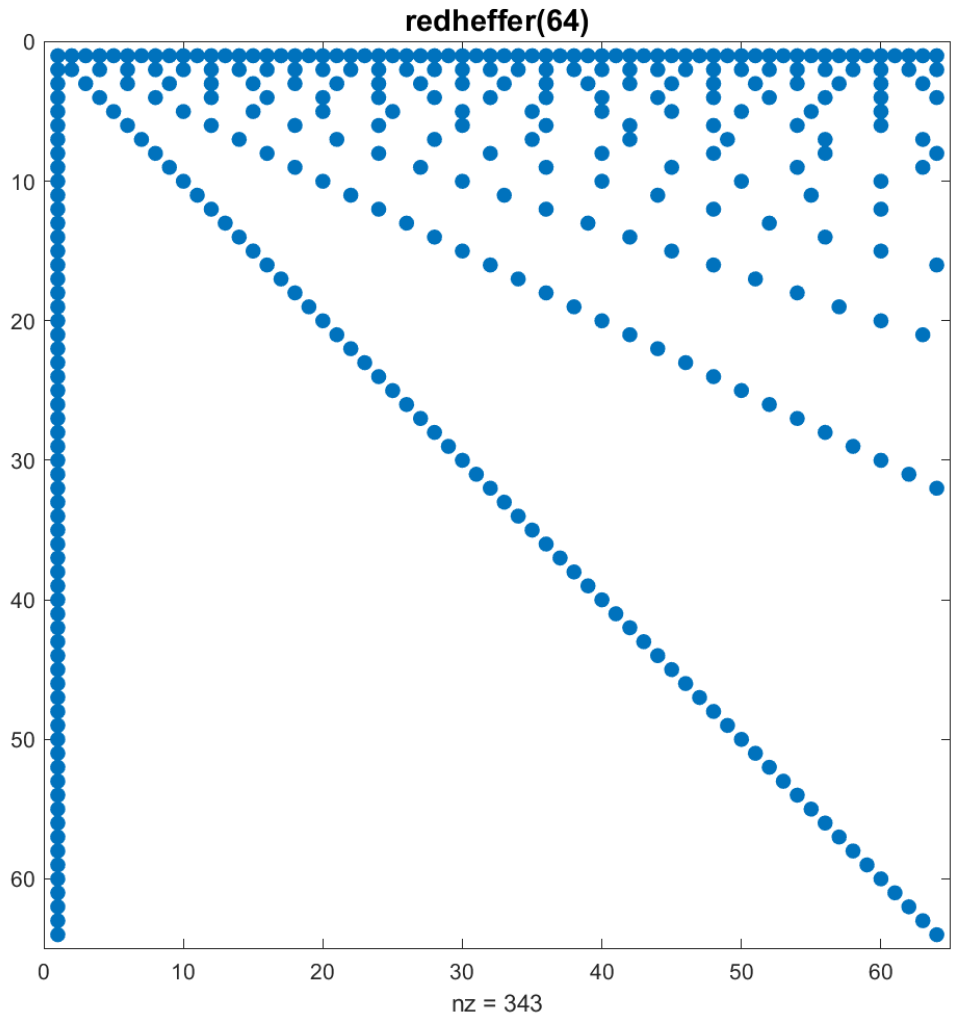


Figure 3. Graph of the function $h(y_0 + t)$ for $t \in [-3, +3]$ (left), with enlargement of its central part (right).

$$y_0 = -14045\ 2896805929\ 9804679036\ 1630399781\ 1274005919\ 9978973803\ 9965960762.521505.$$

- **Redheffer Matrix**


redheffer(64)
0
10
20
30
40
50
60
0
10
20
30
40
50
60
nz = 343

Redheffer Matrix

```
help gallery/redheff
```

```
A has N-FLOOR(LOG2(N))-1 eigenvalues equal to 1,
a real eigenvalue approximately SQRT(N),
a negative eigenvalue approximately -SQRT(N),
and the remaining eigenvalues are provably "small".
```

Redheffer Matrix

```
help gallery/redheff
```

```
N = 64
A has        57          eigenvalues equal to 1,
a real eigenvalue approximately    8.0
a negative eigenvalue approximately   -8.0   ,
and the remaining eigenvalues are provably "small".
```

Redheffer Matrix

```
eig(redheffer(64))

10.0445 + 0.0000i
-5.5442 + 0.0000i
 0.0726 + 0.0000i
 0.3213 + 0.4487i
 0.3213 - 0.4487i
 0.8923 + 0.1262i
 0.8923 - 0.1262i
 1.0000 + 0.0000i
    . . . .
 1.0000 + 0.0000i
```

57 times

Redheffer Matrix

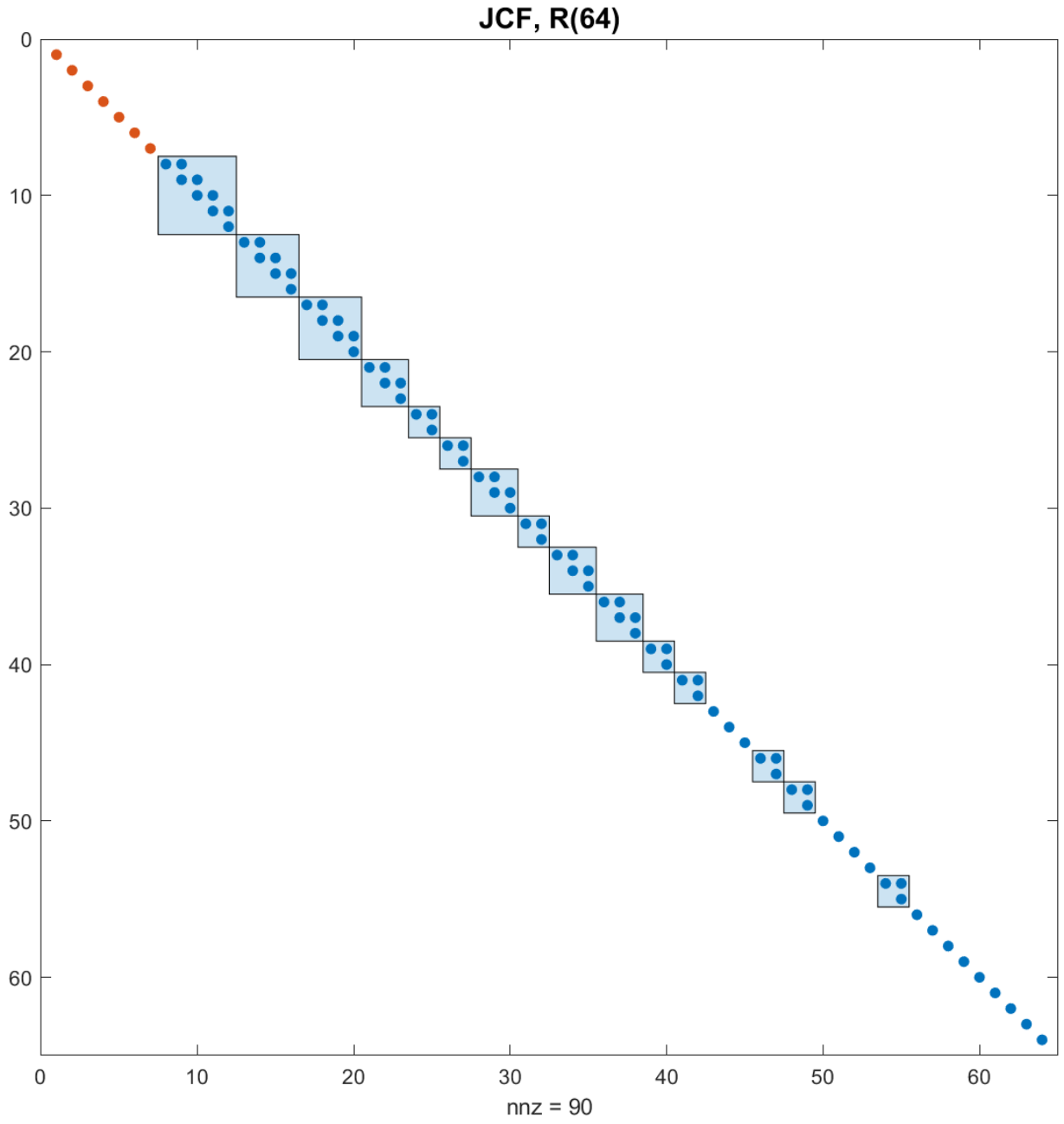
The characteristic polynomial of R(64) is

$$p(z) * (z - 1)^{57}$$

where p(z) =

$$z^7 - 7^z^6 - 42*z^5 + 127*z^4 - 130*z^3 + 67*z^2 - 18*z + 1$$

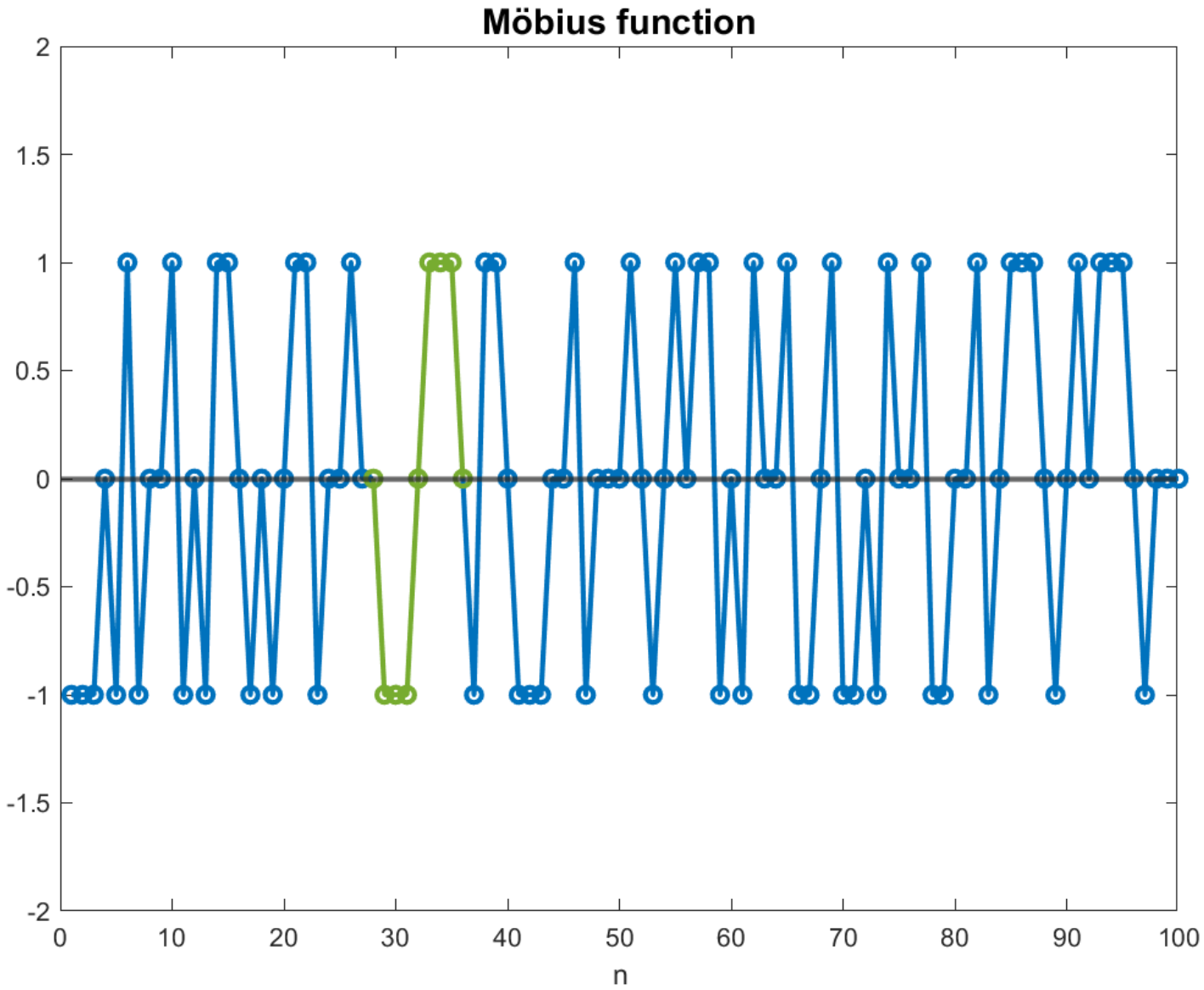
M(64)


JCF, R(64)
0
10
20
30
40
50
60
0
10
20
30
40
50
60
nnz = 90

- **Code**

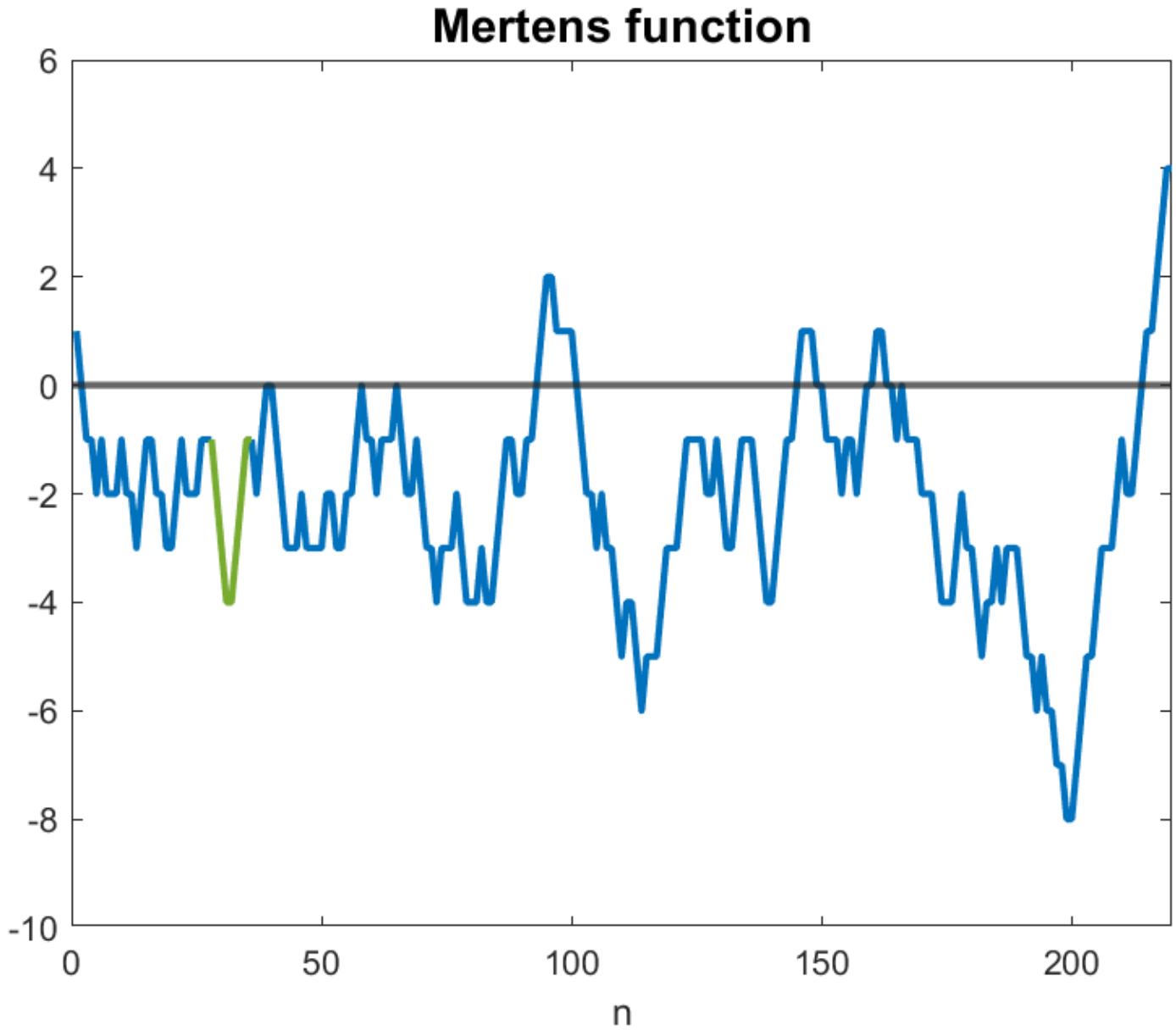
Möbius Function

```
function mu = mobius(n)
    mu = ones(1,n);
    mu(1) = -1;
    for p = primes(n)
        mu(p^2:p^2:n) = 0;
        mu(p:p:n) = -mu(p:p:n);
    end
end
```


Möbius function
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
10
20
30
40
50
60
70
80
90
100
n

Mertens Function

```
function M = mertens(n)
    mu = mobius(n);
    M = cumsum(mu);
end
```


Mertens function
6
4
2
0
-2
-4
-6
-8
-10
0
50
100
150
200
n

Redheffer Matrix

```
function R = redheffer(n)
    k = 1:n;
    R = (mod(k,k') == 0);
    R(:,1) = 1;
end
```

Redheffer Matrix

```
>> R = redheffer(12)
R =
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     0     1     0     1     0     1     0     1     0     1
     1     0     1     0     0     1     0     0     1     0     0     1
     1     0     0     1     0     0     0     1     0     0     0     1
     1     0     0     0     1     0     0     0     0     1     0     0
     1     0     0     0     0     1     0     0     0     0     0     1
     1     0     0     0     0     0     1     0     0     0     0     0
     1     0     0     0     0     0     0     1     0     0     0     0
     1     0     0     0     0     0     0     0     1     0     0     0
     1     0     0     0     0     0     0     0     0     1     0     0
     1     0     0     0     0     0     0     0     0     0     1     0
     1     0     0     0     0     0     0     0     0     0     0     1
```

Sparse Redheffer

```
function R = sparse_redheffer(n)
    j(1:n) = (1:n)';
    k(1:n) = 1;
    m = n;
    for i = 2:n
        t = [1 i:i:n]';
        p = length(t);
        j(m+(1:p)) = t;
        k(m+(1:p)) = i;
        m = m+p;
    end
    R = sparse(k,j,1,n,n);
end
```

Sparse Redheffer

```
>> R = sparse_redheffer(12);

   (1,1)        1
   (2,1)        1
   (3,1)        1
   (4,1)        1
   (5,1)        1
   (6,1)        1
   (7,1)        1
   (8,1)        1
   (9,1)        1
  (10,1)        1
  (11,1)        1
  (12,1)        1
       . . . .
   (1,12)       1
   (2,12)       1
   (3,12)       1
   (4,12)       1
   (6,12)       1
  (12,12)       1
```

Sparse Redheffer

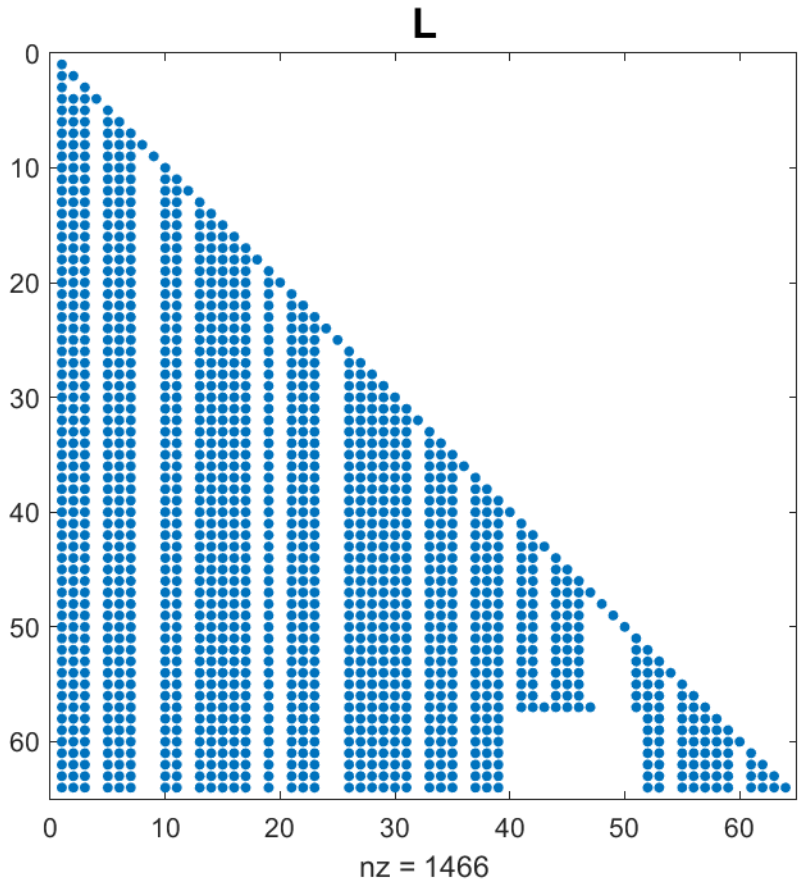
n	tictoc	bytes	nnz	sparsity	det	$\lvert\det\rvert/\sqrt{n}$
10^1	0.000	664	36	0.3600000	-1	0.316
10^2	0.000	10,104	581	0.0581000	1	0.100
10^3	0.002	137,096	8,068	0.0080680	2	0.063
10^4	0.023	1,738,680	103,667	0.0010367	-23	0.230
10^5	0.255	21,067,992	1,266,749	0.0001267	-48	0.152
10^6	2.921	247,520,536	14,970,033	0.0000150	212	0.212
10^7	33.365	2,843,605,816	172,725,363	0.0000017	1037	0.328

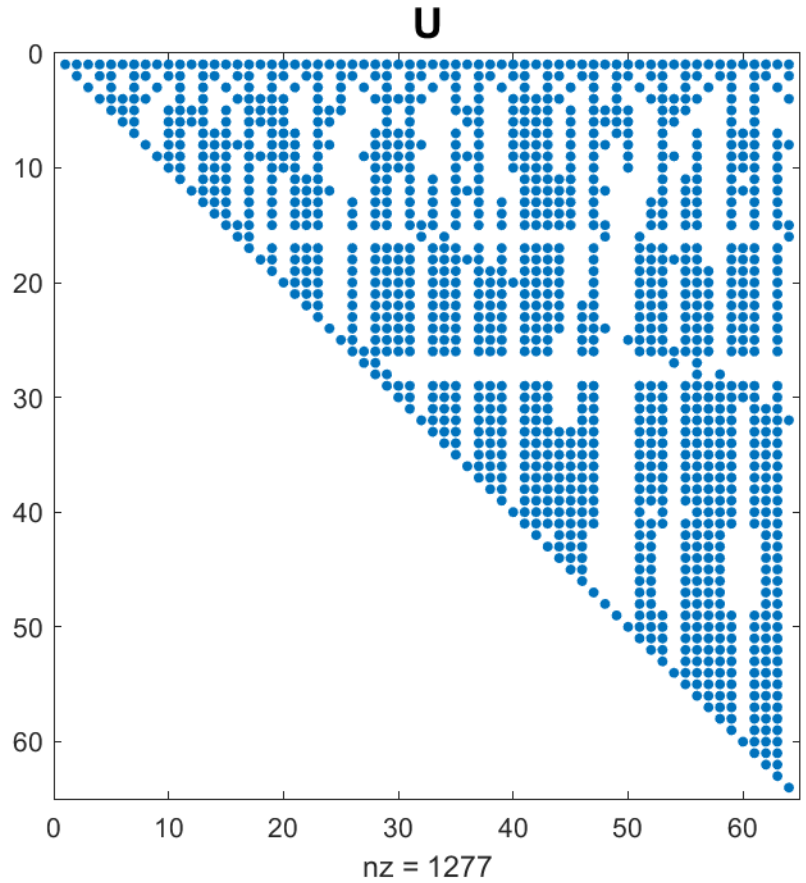
- **Five Ways to Compute the Mertens/Redheffer Function**

#1

```
R = redheffer(n)
M = det(R)
```

#1, R(64)

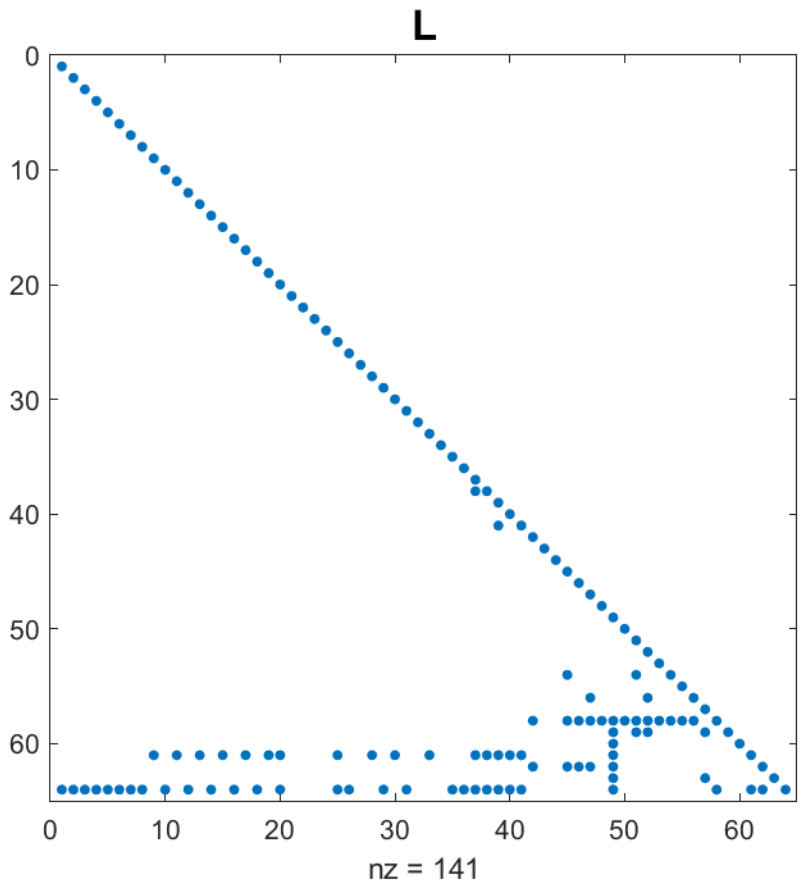

L
0
10
20
30
40
50
60
0
10
20
30
40
50
60
nz = 1466

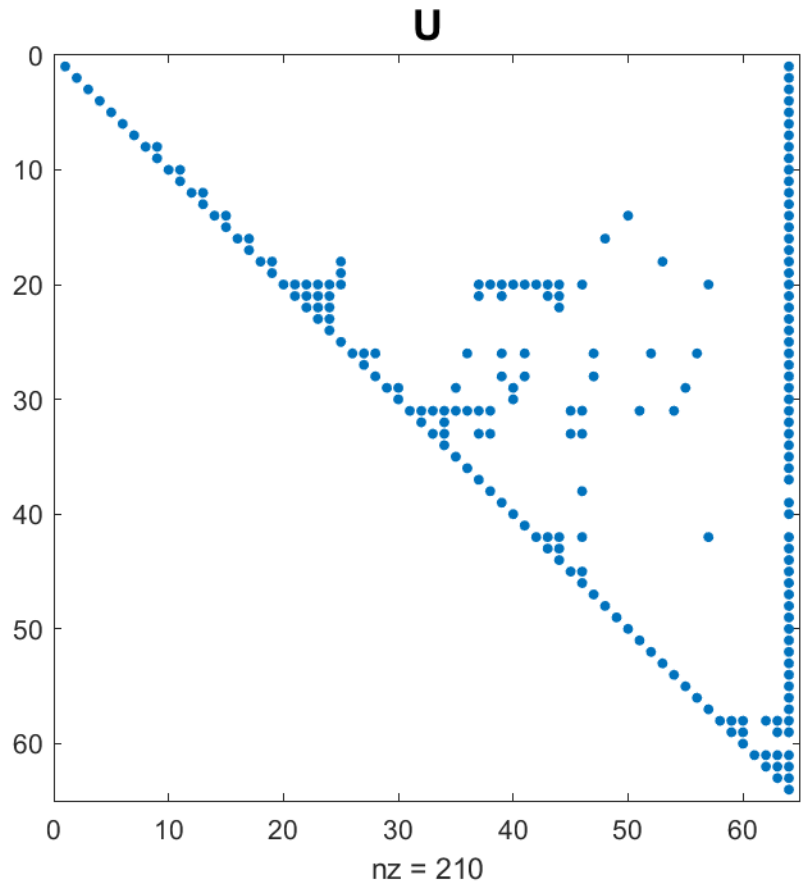

U
0
10
20
30
40
50
60
0
10
20
30
40
50
60
nz = 1277

#2

```
R = sparse_redheffer(n)
[L,U,P,Q] = lu(R)
M = det(L)*det(U)*det(P)*det(Q)
```

#2, R(64)

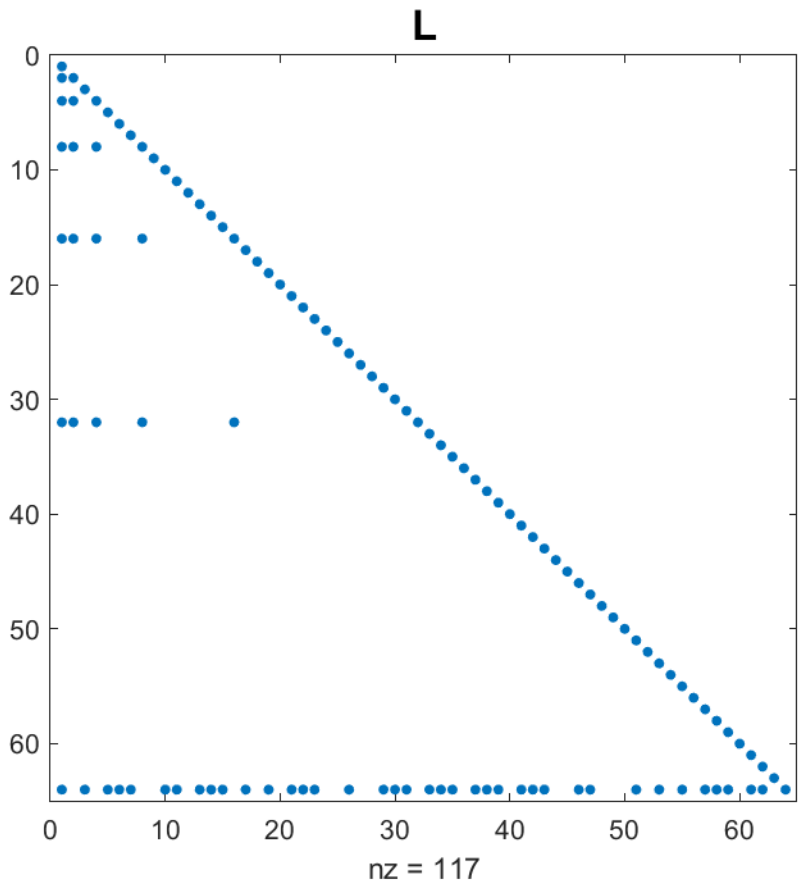

L
0
10
20
30
40
50
60
0
10
20
30
40
50
60
nz = 141

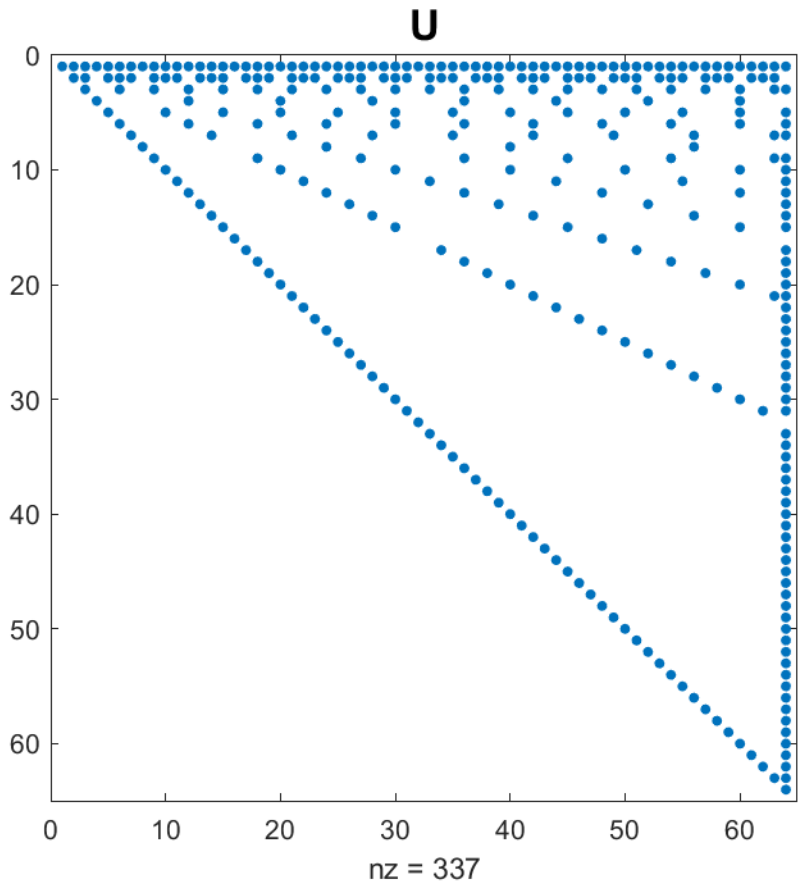

U
0
10
20
30
40
50
60
0
10
20
30
40
50
60
nz = 210

#3

```
R = sparse_redheffer(n)
R(:,[1 n]) = R(:,[n 1])
M = -det(R)
```

#3, R(64)


L
0
10
20
30
40
50
60
0
10
20
30
40
50
60
nz = 117


U
0
10
20
30
40
50
60
0
10
20
30
40
50
60
nz = 337

#4

```
R = sparse_redheffer(n)
T = R(2:n,2:n)
e = ones(n-1,1)
M = 1 - e*(T\e')
```

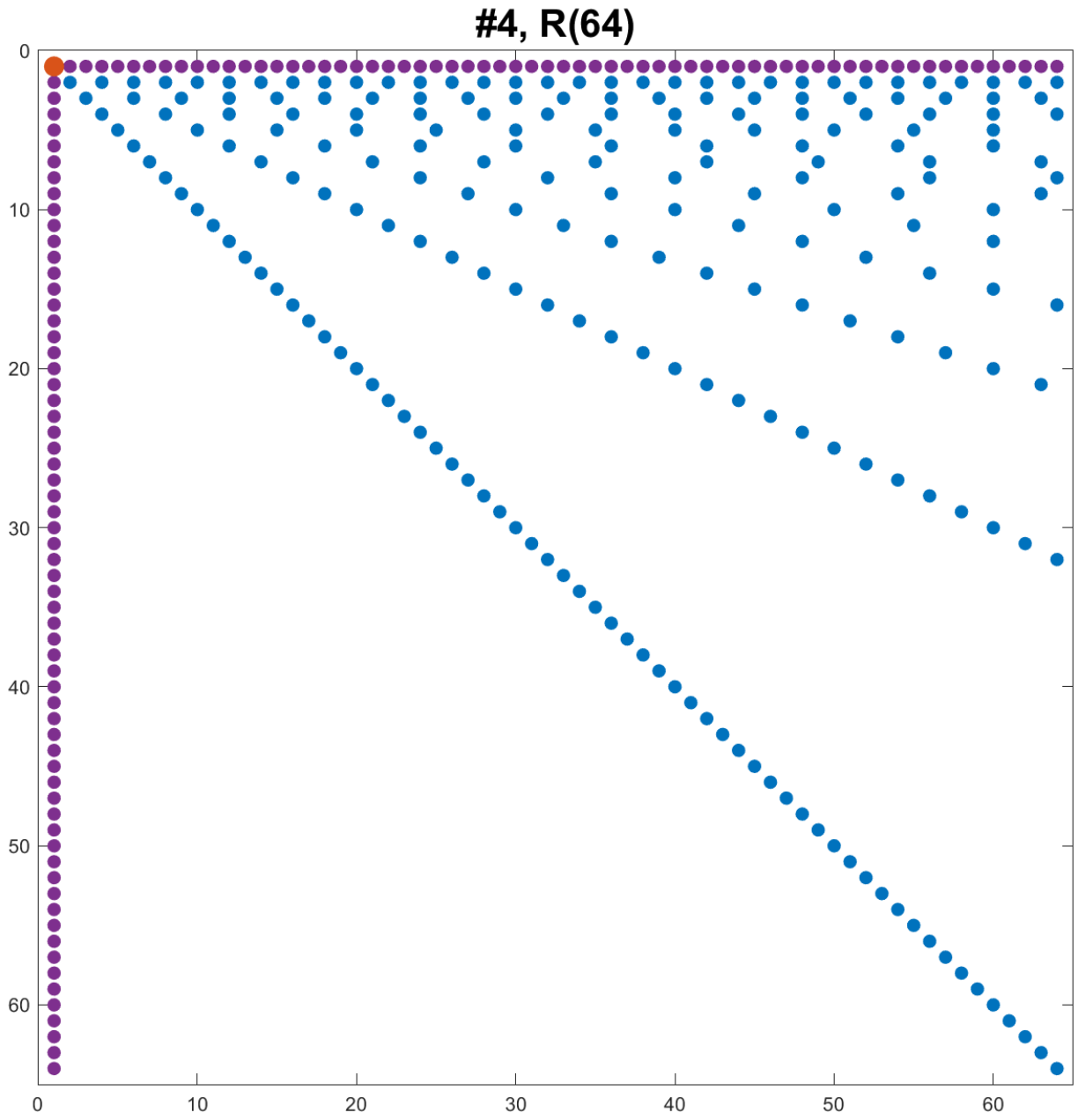
Schur complement

$$R = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

$$\det(R) = \det(D)*\det(A - B*(D\backslash C))$$

Schur complement

```
e = ones(1,n-1)
T = R(2:n,2:n)
R = [1  e
     e' T]
det(R) = 1*det(1 - e*(T\e'))
```


#4, R(64)
0
10
20
30
40
50
60
0
10
20
30
40
50
60

#5

```
mu = mobius(n)
M = cumsum(mu)
```

Five Ways

```
function M = fiveways(n)
    %1
        R = redheffer(n);
        M(1) = det(R);
    %2
        R =  sparse_redheffer(n);
        [L,U,P,Q] = lu(R);
        M(2) = det(L)*det(U)*det(P)*det(Q);
    %3
        R =  sparse_redheffer(n);
        R(:,[1 n]) = R(:,[n 1]);
        M(3) = -det(R);
    %4
        R =  sparse_redheffer(n);
        T = R(2:n,2:n);
        e = ones(1,n-1);
        M(4) = 1 - e*(T\e');
    %5
        mu = mobius(n);
        cmu = cumsum([1 mu(2:end)]);
        M(5) = cmu(n);
end
```

Complexity

	redheffer	function	dets	complexity	M
#1	full	gallery	1	n^3	1
#2	sparse	lu	4	n*log(n)^2	1
#3	sparse	swap	1	n*log(n)^2	1
#4	sparse	\	0	n*log(n)	1
#5	none	primes	0	n*log(log(n))	many

Timing

	2e4	2e5	2e6	2e7
#1	26.33	-	-	-
#2	0.36	21.53	-	-
#3	0.08	1.29	16.71	-
#4	0.05	0.57	6.32	70.85
#5	0.01	0.03	0.27	3.18

Time (seconds)

References

- Möbius (1832), Journal für die reine und angewandte Mathematik. 9: 105–123.
- Riemann (1859), "Ueber die Anzahl der Primzahlen unter einer gegebenen Grösse", Monatsberichte der Berliner Akademie (1892).
- Mertens (1897), Akademie Wissenschaftlicher Wien Mathematik-Naturlich, IIA. 106: 761–830.
- Redheffer (1977), Numerische Methoden bei Optimierungsaufgaben, Band 3: 213–216.
- Odlyzko & te Riele (1985), Journal für die reine und angewandte Mathematik 357: 138–160.
- Barrett & Jarvis (1992), Linear Algebra and Its Applications: 162–164.
- Borwein (2009), https://www.cecm.sfu.ca/~pborwein/course/math08/lecture.pdf

Thanks

- Tim Davis
- John Gilbert
- Pat Quillen
- Steve Lord
- Jan van Lent
- Frank Stenger
- Claude