Computation of matrix gamma function

João R. Cardoso[1], Amir Sadeghi[2]

[1]Coimbra Polytechnic – ISEC, Coimbra, Portugal
Institute of Systems and Robotics – Coimbra
jocar@isec.pt

[2]Islamic Azad University, Tehran, Iran

Advances in NLA, Univ. Manchester, May 2019

Motivation

- Matrix gamma function is an important matrix function (connections with differential equations and with other special functions, such as beta function and Bessel function);
- We haven't found any research involving its numerical computation.

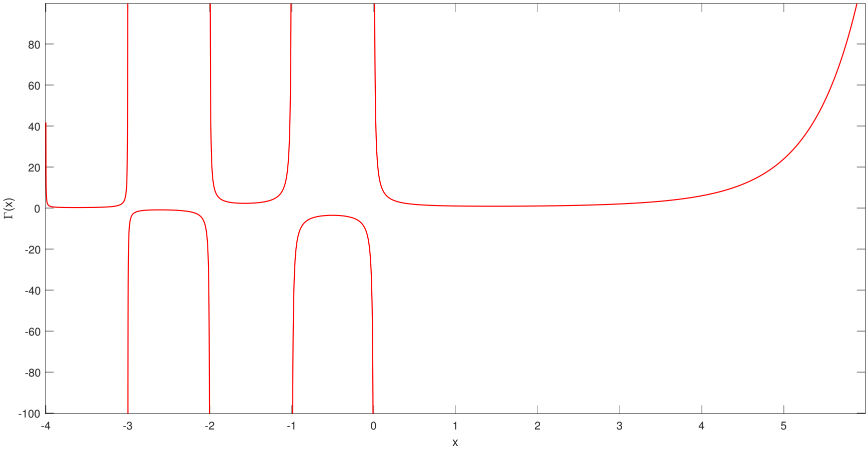
Revisiting scalar gamma

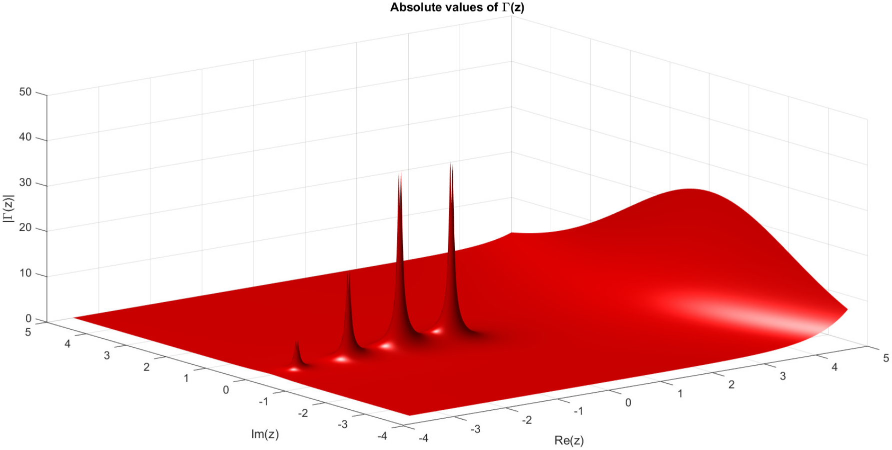
Definition

$$\Gamma(z) = \int_0^\infty e^{-t} t^{z-1} dt, \quad \mathrm{Re}(z) > 0.$$

(definition using Euler's integral)

By analytic continuation, we can extend this definition to all complex numbers except the non-positive integers $0, -1, -2, \ldots$.


80
60
40
20
0
-20
-40
-60
-80
-100
Γ(x)
-4
-3
-2
-1
0
1
2
3
4
5
x


Absolute values of Γ(z)
|Γ(z)|
50
40
30
20
10
0
5
4
3
2
1
0
-1
-2
-3
-4
Im(z)
-4
-3
-2
-1
0
1
2
3
4
5
Re(z)

Scalar gamma
Properties

- Connection with factorial

$$\Gamma(n+1) = n!$$

(*n* positive integer)

- Translation formula

$$\Gamma(z+1) = z\,\Gamma(z)$$

- Reflection formula

$$\Gamma(z) = \frac{\pi}{\Gamma(1-z)\sin(\pi z)}, \quad z \notin \mathbb{Z}$$

Computation of scalar gamma

- Lanczos method: a formula involving a partial fraction expansion due to Cornelius Lanczos (1964); most popular (available in Numerical Recipes);
- Stirling's formula: Asymptotic expansion due to Abraham de Moivre (1733); also studied by James Stirling.

$$\Gamma(A) = \int_0^\infty e^{-t} t^{A-I} \, dt,$$

where A has eigenvalues with positive real parts and

$$t^{A-I} := \exp((A - I) \log t)$$

(Jódar & Cortés, 1998)

Major difficulties

Methods for the matrix gamma based just on the substitution of the variable z by a matrix A in Lanczos and Stirling approximations give very poor results.

Scalar Lanczos method just holds for z with positive real part; if z has negative real part, the reflection gamma formula should be used.
Scalar asymptotic Stirling formula gives good results if the real part of z is sufficiently large.

How to proceed for matrices having small and large eigenvalues or eigenvalues with positive and negative real parts?

Lanczos formula for matrices

$$\log[\Gamma(A)] = 0.5\log(2\pi)I + (A - 0.5I)\log(A + (\alpha - 0.5)\, I) - (A + (\alpha - 0.5)\, I) + \log\left[c_0(\alpha)I + \sum_{k=1}^{m} c_k(\alpha)\,(A + (k-1)I)^{-1} + \epsilon_{\alpha,m}(A)\right]$$

, where $\mathrm{Re}(\lambda) > 0$.

(Logarithmic version to avoid overflow)

Lanczos method

- If A has spectrum entirely contained in the open right-half plane, compute $\Gamma(A)$ by Lanczos formula (e.g, with $m = 10$ terms and $\alpha = 9$, as recommended by Lanczos);
- If $\sigma(A)$ doesn't contain negative integers and lies entirely on the open left-half plane, use reflection property combined with Lanczos formula

(Not recommended when A has simultaneously eigvs with positive and negative real parts)

Stirling approximation for matrices

$$\Gamma(A) \approx \left(\prod_{k=1}^{s} (A + (k-1)I) \right)^{-1} e^{S_{m,s}(A)},$$

where

$$\begin{aligned} S_{m,s}(A) \quad := \quad & (A + (s-0.5)I) \log(A + (s-1)I) - (A + (s-1)I) + \\ & 0.5 \log(2\pi) + \sum_{k=1}^{m} \frac{B_{2k}}{2k(2k-1)\,(A + (s-1)I)^{2k-1}}, \end{aligned}$$

where $\mathrm{Re}(\lambda) > 0$.

(B_{2k}: Bernoulli numbers)

Matrix gamma

Stirling approximation

Algorithm. This algorithm evaluates $\Gamma(A)$ using the Stirling formula, with $m = 12$, and $A \in \mathbb{C}^{n\times n}$ is a nonsingular matrix with spectrum satisfying one and only one of the following conditions: (i) $\sigma(A)$ is contained in the closed right-half plane; or (ii) $\sigma(A)$ does not contain negative integers and lies on the open left-half plane.

`if` $\mathrm{Re}(\mathrm{trace}(A)) \geq 0$
    $z = \mathrm{trace}(A)/n$;
    `if` $\mathrm{Im}(z) \geq 8.3$ or $1 - \mathrm{Re}\left(z + \sqrt{8.3^2 - \mathrm{Im}(z)^2)}\right) \leq 0$
        $s = 0$
    `else`
        $s = \lceil 1 - \mathrm{Re}(z) + \sqrt{8.3^2 - \mathrm{Im}(z)^2} \rceil$
    `end`
    Compute $\Gamma(A)$ by Stirling formula;
`else`
    $S = \sin(\pi A)$;
    Compute $G = \Gamma(I - A)$ by Stirling formula;;
    $\Gamma(A) \approx \pi (SG)^{-1}$;
`end`

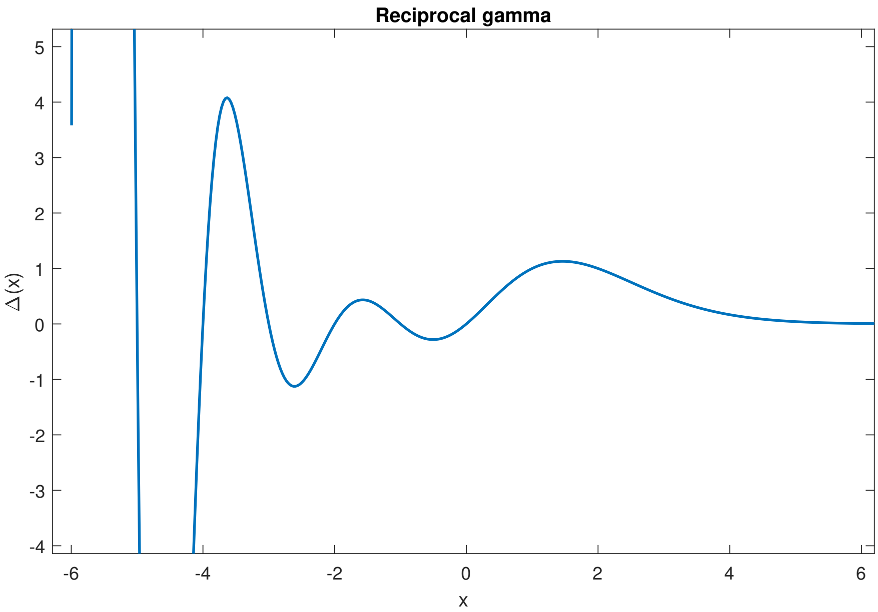
Matrix reciprocal gamma

$$\Delta(A) = (\Gamma(A))^{-1} = \sum_{k=0}^{\infty} a_k A^k,$$

$a_1 = 1,\ a_2 = \gamma$ (here γ stands for the Euler-Mascheroni constant) and the coefficients a_k ($k \geq 2$) are given recursively by

$$a_k = \frac{a_2 a_{k-1} - \sum_{j=2}^{k-1} (-1)^j \zeta(j) a_{k-j}}{k-1},$$

with $\zeta(.)$ being the Riemann zeta function.



Reciprocal gamma
Δ(x)
x
5
4
3
2
1
0
-1
-2
-3
-4
-6
-4
-2
0
2
4
6

Matrix reciprocal gamma
Error bound

Lemma

If $A \in \mathbb{C}^{n\times n}$ *with* $\|A\| \leq 1$, *then*

$$\left\| \Delta(A) - \sum_{k=1}^{m} a_k A_k \right\| \lesssim \frac{4}{\pi^2} \sum_{k=m+1}^{\infty} \frac{\sqrt{k!}}{(m+1)!(k-m-1)!}.$$

Matrix reciprocal gamma

Algorithm. This algorithm approximates $\Gamma(A)$, where A is a non-singular matrix with no negative integers eigenvalues, by the reciprocal gamma function series combined with the Gauss multiplication formula.

1. $\mu = 3$;
2. `if` $\rho(A) \leq \mu$
3. $\quad \widetilde{\Delta} = \sum_{k=1}^{50} a_k A^k$;
4. $\quad \Gamma(A) \approx (\widetilde{\Delta})^{-1}$;
5. `else`
6. $\quad$ Compute $r = \left\lceil \frac{\rho(A)-1}{\mu-1} \right\rceil$;
7. $\quad \widetilde{\Delta} = \sum_{k=0}^{50} a_k \left(\frac{A}{r}\right)^k$;
8. $\quad$ `for` $p = 1 : r-1$
9. $\quad\quad$ Compute $\widetilde{\Delta} = \widetilde{\Delta} \sum_{k=0}^{50} a_k \left(\frac{A+pI}{r}\right)^k$;
10. $\quad$ `end`
11. $\quad \widetilde{\Delta} = (2\pi)^{\frac{r-1}{2}} \, r^{0.5\, I - A} \, \widetilde{\Delta}$;
12. $\quad \Gamma(A) \approx (\widetilde{\Delta})^{-1}$;
13. `end`

Schur-Parlett method

Algorithm. This algorithm approximates $\Gamma(A)$ by Schur-Parlett method combined with Lanczos, Stirling and reciprocal algorithms.

1. Compute a Schur decomposition $A = UTU^*$, where the blocks T_{ii} in the diagonal of T are well separated ($\delta = 0.1$); (Some codes available in Higham's MFTOOLBOX)
2. Approximate $G_{ii} = \Gamma(T_{ii})$ by one of the previous three algs;
3. Solve the Sylvester equations arising in Parlett's recurrence, in order to compute all the blocks G_{ij}, with $i < j$;
4. $\Gamma(A) \approx UGU^*$, where $G = [G_{ij}]$.

Experiments

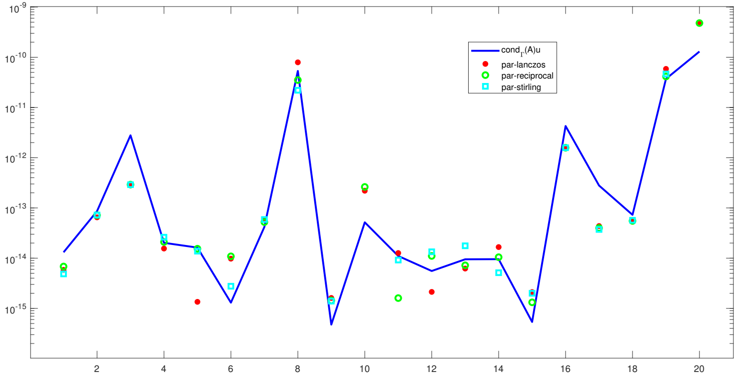


Figure: Relative errors for 20 test matrices together with the relative condition number of $\Gamma(A)$ times the unit roundoff of MATLAB.

Experiments

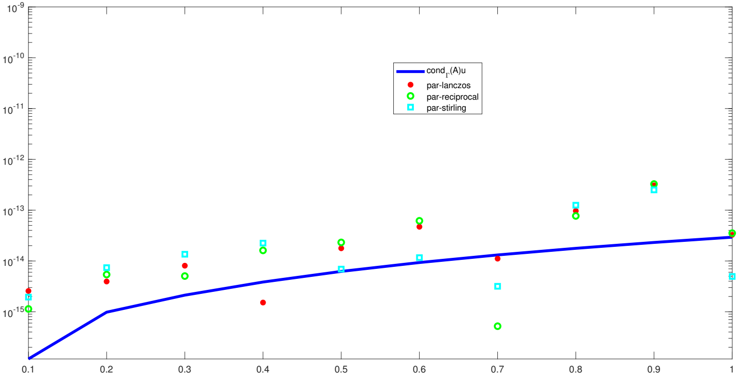


Figure: Relative errors for `gallery('moler',12,a)`, by varying a from 0.1 to 1.

Experiments

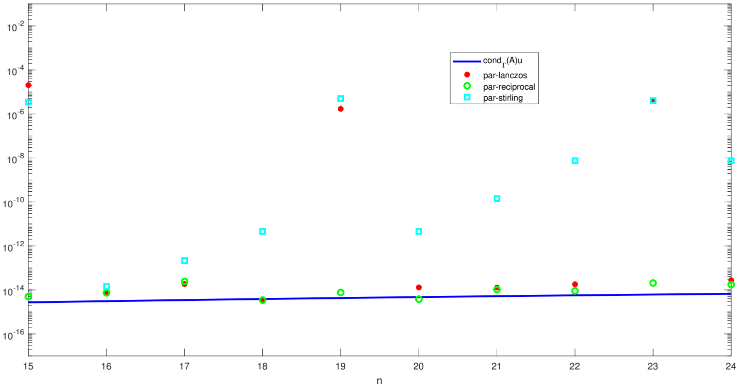


Figure: Relative errors for `wilkinson(n)` with size n increasing as $n = 15, \ldots, 24$.

Conclusions

- Likewise the scalar case, the Lanczos methods shows a good performance in terms of a combination between accuracy and computational cost;
- However, the technique based on the reciprocal gamma function combined with the Gauss multiplication formula, gives very good results in terms of accuracy, with the advantage of being rich in matrix-matrix multiplications;
- To make the three approximations of matrix gamma effective, they were combined with the Schur-Parlett method.

Further details in

João R. Cardoso, Amir Sadeghi, Computation of matrix gamma function, BIT Numerical Mathematics, (2019)
(article in press, available online at
https://doi.org/10.1007/s10543-018-00744-1).