

# AMS 595: Project Report

Kai Li

kai.li@stonybrook.edu

Stony Brook University — December 2, 2020

## 1 Project Objectives

As a statistics-track master's student, I am particularly interested in numerical analysis. This project's primary goal is to practice, deepen, and expand my current mathematics and programming skills. Therefore, the project is a good chance for me to do some scientific numerical approximation work using programming skills learned in AMS 595 and previous numerical analysis experience. At the end of the course, I have gained sufficient knowledge in various programming languages. In fact, I have enrolled in a course named "Scientific Computing" during the final semester of my undergraduate study. In that course, I studied introductory numerical analysis and used Matlab to do scientific computing assignments. Therefore, I chose C++ as my programming language for the project to proceed with my interest in numerical approximation to get myself familiar with C++ scientific computing.

Motivated by the scientific computing problem of  $\pi$  in AMS 595 C++ homework assignment, I will practice scientific computing skills on numerically estimating the logarithmic constant  $e$ , also called the Euler's number. In honor of Euler's and many other great mathematicians' life, as they made incredible contributions to the development of mathematics, such as the two irrationals  $\pi$  and  $e$ , I realized the importance of summarizing the existing methods of computing  $e$ . The methods are very different, depending on accuracy, convergence speed, and input. In this project, because I have already done some approximation computations on  $\pi$  in the C++ homework assignment, it is an extra opportunity to dig into numerical computing methods of  $e$ . More importantly, the project improves my ability to think analytically and creatively because I have the chance to read and possibly prove the methods of computing  $e$ .

## 2 Numerical Approximation Methods

There exist numerous approximations of the natural base of the logarithm  $e$ . For example, a simple rational fraction approximation is  $\frac{878}{323}$ , using integers less than 1000 in the denominator and the numerator [7]. Moreover,  $e$  can be approximated by repeating decimal fraction  $2.71828$  [2]. In this project, to improve programming skills in C++ and expand graduate-level applied mathematics knowledge, all approximations require inputs to determine the precision level of approximation. More specifically, the methods are functions that involve either limit, series, product, sequence, or gamma function.

I will focus on two different categories of approximations. Scientific approximation methods 2.1, 2.3, and 2.4 will be based on number  $n$  in the approximation expressions. The three methods are one-term simple expressions without summations of the other terms but depend on input  $n$ . Methods 2.5, 2.6, 2.7, and 2.8 are based on order  $p$ , or the number of terms  $n$ , in the expressions. The expressions are either a series, a product, or a continued fraction that input the number of terms wanted in the approximations. The method in 2.2 is special because it requires two inputs:  $n$  and  $p$ . For the purpose of the project, I will assume that the order is fixed for method 2.2.

## 2.1 Compound Interest Method (CIM)

The most classical method of computing  $e$  is the CIM from the study of the theory of interest [6]:

$$e = \lim_{n \rightarrow \infty} \left(1 + \frac{1}{n}\right)^n \quad (1)$$

This limit is often one of the most crucial results in undergraduate-level calculus.

## 2.2 Improved Compound Interest Method (ICIM)

Inspired by the CIM, the PSM shown in Section 2.6, and previous calculus experience, I noticed the following approximation method:

$$e = \lim_{n \rightarrow \infty} \lim_{p \rightarrow \infty} \sum_{k=0}^p \left(\frac{1}{k! n^k}\right)^n = \lim_{n \rightarrow \infty} \left(1 + \frac{1}{1! n} + \frac{1}{2! n^2} + \cdots + \frac{1}{p! n^p} + \cdots\right)^n \quad (2)$$

To be more specific, the inspiration came from a note (which cannot be found now) in my first year undergraduate real analysis course that proves Equation (2) in cases which order  $p = 2$  and  $p = 3$ . Though I have not seen proof for Equation (2) in  $p$  and infinite dimensions, it makes sense to generalize the above equation to  $p$  and infinite orders using mathematical induction.

Note that the CIM is a special case of the ICIM for  $p = 1$ . Compared to the CIM, I conjecture that the ICIM is a more precise estimate because the infinite series gives more information on the value of  $e$ . I also believe that if  $p$  increases, the more precise estimate will output. The result of my conjecture will be shown later.

On the other hand, there is a disadvantage in the number of terms  $p$  we can reach in scientific computing. More precisely, I experimented on a standard computer that  $p!$ , in Equation (2), exceeds the maximum precision level of variable type "int" when  $p \geq 13$ . Precision means the number of meaningful digits [9]. This problem is known as overflow. There are possible solutions, but I will use "int" as the variable type in calculations involving factorial and power for the project's scope.

## 2.3 Complementary Addition Method (CAM)

The CAM is another simple improvement for the CIM [1]:

$$e = \lim_{n \rightarrow \infty} \frac{1}{2} \left[ \left(1 + \frac{1}{n}\right)^n + \left(1 - \frac{1}{n}\right)^{-n} \right] \quad (3)$$

Fortunately, this method will not raise overflow issue for reasonable inputs ( $n \leq 2147483647$ ) because the two bases of the exponents are close to 1.

## 2.4 Power Ratio Method (PRM)

The PRM comes by analyzing the behaviors of the rate of change of the ratio between two adjacent numbers raised to their own power [1]:

$$e = \lim_{n \rightarrow \infty} \left[ \frac{(n+1)^{n+1}}{n^n} - \frac{n^n}{(n-1)^{n-1}} \right] \quad (4)$$

In this method, at first, I conjectured that there would be an overflow problem. Because  $n^n$  exceeds the maximum precision level of variable type "int" when  $n > 9$ . However, due to the fraction of two exponentiations, the decimal value drastically decreases. Thus, the overflow problem will not arise until  $n$  is bigger than 142.

## 2.5 Stirling's Formula Approximation Method (SFAM)

This method can be derived from the famous Stirling's formula  $n! \sim \sqrt{2\pi n} \left(\frac{n}{e}\right)^n$ :

$$e = \lim_{n \rightarrow \infty} \frac{n}{\sqrt[n]{n!}} \quad (5)$$

Because of the factorial in the denominator, the overflow issue restricts  $n$  to be less than 13.

## 2.6 Power Series Method (PSM)

The following method originates from the power series of  $e^x$  evaluated at  $x = 1$ :

$$e = \lim_{n \rightarrow \infty} \sum_{k=0}^n \frac{1}{k!} = \frac{1}{0!} + \frac{1}{1!} + \cdots + \frac{1}{n!} + \cdots \quad (6)$$

Similar to the ICIM and the SFAM, the factorial raises the overflow problem.

## 2.7 Continued Fraction Method (CFM)

Euler proved the following continued fraction expansion in [3]:

$$e = 2 + \frac{1}{1 + \frac{1}{2 + \frac{1}{1 + \frac{1}{1 + \frac{1}{4 + \frac{1}{1 + \frac{1}{1 + \frac{1}{6 + \ddots}}}}}}}}}. \quad (7)$$

The continued fraction is an infinite sequence [5] defined by

$$[a_0, a_1, \dots, a_i, \dots] = \begin{cases} 2 & i = 0, \\ \frac{2(i+1)}{3} & i = 3k - 1, \quad \forall i \in \mathbb{N}^0, k \in \mathbb{N}^+ \\ 1 & \text{otherwise.} \end{cases}$$

In a more simple sequence form, the expansion of  $e$  is known explicitly as [4]:

$$e = [2, 1, 2, 1, 1, 4, 1, 1, 6, \dots, 1, 1, 2n, \dots].$$

## 2.8 Pippenger Product Method (PPM)

The Pippenger Product is a striking companion to Wallis's product [8], which expresses  $e$  in terms of an infinite product:

$$\begin{aligned} e &= \lim_{n \rightarrow \infty} 2 \left(\frac{2}{1}\right)^{\frac{1}{2}} \prod_{k=2}^n \frac{2^{2k-1} [\Gamma(\frac{1}{2} + 2^{k-2})]^4}{\pi [\Gamma(\frac{1}{2} + 2^{k-1})]^2} \\ &= 2 \left(\frac{2}{1}\right)^{\frac{1}{2}} \left(\frac{2}{3}\right)^{\frac{1}{4}} \left(\frac{4}{5}\right)^{\frac{1}{8}} \left(\frac{6}{7}\right)^{\frac{1}{8}} \left(\frac{8}{9}\right)^{\frac{1}{16}} \left(\frac{10}{11}\right)^{\frac{1}{16}} \left(\frac{12}{13}\right)^{\frac{1}{16}} \left(\frac{14}{15}\right)^{\frac{1}{16}} \cdots \end{aligned} \quad (8)$$

However, the drawback is that the overflow problem will occur when  $n \geq 8$  because of the  $2^{2k-1}$  in the numerator.

### 3 Results and Conclusions

#### 3.1 Approximations Based on $n$ or Order $p/n$

The following two tables summarize the eight methods of estimating Euler's number  $e$  in 8 digits after the decimal point, fixing  $n$  to be 7:

Results Table ( $n = 7$ )			Results Table ( $n = 7$ )		
Method	Approximation	Absolute Error	Method	Approximation	Absolute Error
CIM	2.54649970	1.71782131e-1	SFAM	2.07099663	6.47285200e-1
ICIM ( $p=5$ )	2.71828163	1.98774743e-7	PSM	2.71825397	2.78602051e-5
CAM	2.74419857	2.59167369e-2	CFM	2.71830986	2.80306959e-5
PRM	2.72061297	2.33114567e-3	PPM	2.71830948	2.76515396e-5

In the left results table, among the four approximations which depend on  $n$ , the difference between the actual constant value  $e$  and the estimate in the ICIM is the smallest. Conversely, the CIM converges the slowest. Also, because CIM is a special case of the ICIM, we see that the convergence to  $e$  of the ICIM is much faster, which verifies my conjecture in 2.2.

For the methods which depend on orders or the number of terms, the differences are very small among the PSM, the CFM, and PPM. On the other hand, the SFAM provides an estimate with a much larger absolute error, relatively speaking, given the same number of terms  $n$ .

The next two tables show the results for approximations using the eight methods with a larger  $n$ :

Results Table ( $n = 100$ )			Results Table ( $n = 100$ )		
Method	Approximation	Absolute Error	Method	Approximation	Absolute Error
CIM	2.70481383	1.34679990e-2	SFAM	N/A	N/A
ICIM ( $p=5$ )	2.71828183	3.74131715e-13	PSM	N/A	N/A
CAM	2.71840643	1.24599466e-4	CFM	2.71828183	1.44632570e-16
PRM	2.71829316	1.13266405e-5	PPM	N/A	N/A

The conclusions obtained in the first two tables ( $n = 7$ ) match exactly with the results for  $n = 100$  for the methods depend on  $n$ . In the results table on the right, I only obtained a result for the CFM for  $n = 100$  because no overflow problem occurs in this calculation. Yet, I speculate that the pattern follows for the three methods depend on orders  $n$  for orders  $n = 7$  to  $n = 100$  as well. That is, I claim that the SFAM converges at a terrible speed compared to the other three methods. Also, there is a probability that the convergence rate for the PSM, the CFM, and PPM is similar. However, because results cannot be obtained for the four methods except for the SFAM, there may be a coincidence that the convergence speed changes for any one of the three methods in any order  $n > 7$ .

#### 3.2 Approximations Based on Absolute Error

Finally, the following tables show the number of  $n$  or the number of terms needed to obtain certain levels of precisions:

Results Table (Error $< 10^{-4}$ )			Results Table (Error $< 10^{-4}$ )		
Method	$n$	Absolute Error	Method	$n$	Absolute Error
CIM	$\approx 13591$	9.99962686e-5	SFAM	$> 12$	N/A
ICIM ( $p=5$ )	2	7.70080380e-5	PSM	7	2.78602051e-5
CAM	112	9.93280642e-5	CFM	7	2.80306959e-5
PRM	34	9.80122646e-5	PPM	7	2.76515396e-5
Results Table (Error $< 10^{-7}$ )			Results Table (Error $< 10^{-7}$ )		
Method	$n$	Absolute Error	Method	$n$	Absolute Error
CIM	$\approx 13,000,000$	9.90995765e-8	SFAM	$> 12$	N/A
ICIM ( $p=5$ )	9	5.81338827e-8	PSM	10	2.73126606e-8
CAM	$\approx 3530$	9.99831204e-8	CFM	11	6.74694742e-9
PRM	$> 142$	N/A	PPM	$> 7$	N/A

Most of the results gained here match the conclusions in Section 3.1. An additional finding is that the number  $n$  needed to obtain a more accurate result for the CIM increases drastically. That again suggests that the classical method converges very slowly to the true value  $e$ , though eventually the limit as  $n$  goes to infinity will be  $e$ . In the above tables, the  $n$ 's obtained for the CIM and the CAM are approximated. These  $n$ 's may not be the first  $n$ 's that reach the precision levels, as I observed in my trial and error experimentation. The reason is that for a large  $n$ , the approximation is not strictly, but in general, increasing or decreasing. However, because the differences between the approximations for the numbers near  $n$ ,  $n$  being large, are small enough to be negligible. Therefore, I used  $\approx$  to denote the desired  $n$  to reach the absolute error level, though the first actual  $n$  with respect to its absolute error that is less than the wanted absolute error can be a few numbers before the  $n$  approximated.

## References

- [1] H. J. Brothers and J. A. Knox. New closed-form approximations to the logarithmic constant  $e$ . *The Mathematical Intelligencer*, 20(4):25–29, 1998.
- [2] P. R. Brown. Approximations of  $e$  and  $\pi$ : an exploration. *International Journal of Mathematical Education in Science and Technology*, pages S30–S39, 2017.
- [3] L. Euler. De fractionibus continuis dissertatio. *Commentarii academiae scientiarum Petropolitanae*, 9:98–137, 1744.
- [4] P. Eymard and J.-P. Lafon. *The number  $\pi$* . American Mathematical Society, Providence, 2004.
- [5] S. R. Finch. *Mathematical Constants*. Cambridge University Press, New York, NY, USA, 2003.
- [6] S. Iyanaga and Y. Kawada. *Encyclopedic dictionary of mathematics*. MIT Press, Cambridge, MA, USA, 1977.
- [7] E. Maor.  *$e$  : the story of a number*. Princeton University Press, Princeton, NJ, USA, 2015.
- [8] N. Pippenger. An infinite product for  $e$ . *The American Mathematical Monthly*, 87(5):291, 5 1980.
- [9] W. Savitch. *Problem solving with C++*. Pearson, 10 edition, 2018.