ISLAMIC AZAD UNIVERSITY
SCIENCE AND RESEARCH BRANCH

THE METHODS OF VALID IMPLEMENTATION OF THE NUMERICAL ALGORITHMS

By
Mohammad Ali Fariborzi Araghi
Supervisor
Dr. Saeid Abbasbandy
Advisers
Dr. Esmail Babolian
Dr. Faezeh Toutounian

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY(Ph.D.)
IN APPLIED MATHEMATICS (NUMERICAL ANALYSIS)
FEBRUARY 2002

ISLAMIC AZAD UNIVERSITY
SCIENCE AND RESEARCH BRANCH
DEPARTMENT OF
MATHEMATICS

The undersigned hereby certify that they have read and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled "**The Methods of Valid Implementation of the Numerical Algorithms**" by **Mohammad Ali Fariborzi Araghi** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy**.

Dated: February 2002

Research Supervisor: _____
Dr. Saeid Abbasbandy

External Examiner: _____
Dr. Khosro Maleknejad

Examing Committee: _____
Dr. Gholam Reza Jahanshahloo

_____
Dr. Esmail Babolian

_____
Dr. Faezeh Toutounian

_____
Dr. Tehranian

ISLAMIC AZAD UNIVERSITY
SCIENCE AND RESEARCH BRANCH [1]

Date: **February 2002**

Author: **Mohammad Ali Fariborzi Araghi**

Title: **The Methods of Valid Implementation of the Numerical Algorithms**

Department: **Mathematics**

Degree: **Ph.D.** Convocation: **March** Year: **2002**

Signature of Author

[1]Science Faculty
Hessarak Road
Ashrafi Esfahani Ave.
Poonak
Tehran
Iran

To my Dear family

Contents

Acknowledgements

I praise God who granted me to reach this scientific degree. I would like to thank Dr. Saeid Abbasbandy, my supervisor, for his many suggestions, guidance and constant supports during this research. I am also thankful to Dr. Esmail Babolian and Dr. Faezeh Toutounian for their guidance through my researches. I appreciate my dear instructor, Dr. Golamreza Jahanshahloo, for his valuable considerations, supports and encouragement during this course.

I thank Dr. Khosro Maleknejad who accepted the refereeing of this thesis.

I am grateful to my family for their patience and favorites. Without them this work would never have come into existence literally.

Finally, I wish to thank the following:

The personnel of the Islamic Azad university, science and researches branch (for their considerations during my education);

The mathematics faculty of Mashhad Ferdowsi university (for allowing to use the computer service);

Mohammad Ali Fariborzi Araghi
March 7, 2002

Abstract

In this project, two main topics are discussed, the numerical integration and the interpolation of data. The idea of the research is to present the valid schemes in order to implement the different methods of evaluating a definite integral numerically. Also, the value of a function, by using the interpolation polynomial is evaluated and a satisfactory solution is obtained.

For this purpose, because of the restrictions of the floating-point arithmetic, the computations are performed in a new arithmetic, named 'stochastic arithmetic', which is able to estimate the reliable results. According to the idea of the CESTAC method, which is a method based on the stochastic arithmetic, a perturbation method is presented, which is used for implementing the algorithms in the stochastic arithmetic. Then, by using this method, the definite and improper integrals are computed. In this case, the optimal number of the points in the numerical integration methods is found. Also, in order to solve an improper integral, an optimal integer number is evaluated which the integral can be estimated with a definite integral with valid implementation. Some theorems are proved to show the accuracy of the numerical integration methods.

At last, it is shown that one can find the optimal degree of the interpolation polynomial which minimizes the error of it. For this purpose, the equidistant and the Chebyshev points are considered. The interpolation polynomials are compared with each other, by using these two kind of points. A theorem is proved to show the accuracy of the interpolation polynomial. In this case, the results are computed by the CADNA library.

Articles

1. Saeid Abbasbandy and Mohammad Ali Fariborzi Araghi, The Usage of the Stochastic Arithmetic in the Accuracy Estimation of the Numerical Algorithms. The Third Seminar on Numerical Analysis and its Applications, Zahedan, Iran, 2000.

2. Saeid Abbasbandy and Mohammad Ali Fariborzi Araghi, Numerical Solution of Improper Integrals with Valid Implementation. To appear in Journal of Mathematical & Computational Applications (M.C.A.), Turkey.

3. Saeid Abbasbandy and Mohammad Ali Fariborzi Araghi, The Valid Implementation of Numerical Integration Methods. To appear in Far East Journal of Applied Mathematics, India.

4. Saeid Abbasbandy and Mohammad Ali Fariborzi Araghi, The valid Implementation of Numerical Integration Methods Using Stochastic Arithmetic. 32nd Iranian Mathematics Conference, Babolsar, 2001.

Chapter 1

Introduction

The simulation of the phenomena is the development of a scientific program which translates a mathematical model and employs numerical methods to resolved it. If a scientific program is performed on a computer, all the computations are implemented with a floating-point arithmetic. In this arithmetic, the numbers, because of the round-off errors, have a finite number of significant digits. Thus, the computer arithmetic is merely an approximation of the exact arithmetic. In this case, the fundamental properties of the exact arithmetic may not be satisfied. Therefore, the results may be false. Hence, it is necessary to validate the results.

After studying and reviewing the subject of the floating-point arithmetic, a new arithmetic based on a probabilistic approach, called 'stochastic arithmetic', is proposed which is able to validate the results and perform reliable scientific computations. CESTAC [1] method is an efficient method in order to estimate the accuracy of the results on a computer. This method was developed by Prof. M. La Porte and Prof. Jean Vignes in France in 1978 and was generalized by others.

CADNA [2] means Control of Accuracy and Debugging for Numerical Applications.

[1] Controle et Estimation Stochastique des Arrondis de Calculs

[2] CADNA is a copyright of the Pierre et Marie Curie University of Paris (France)

The first goal of this software is the estimation of the accuracy of each computed result. This is done by implementing automatically the CESTAC method. Moreover, CADNA uses all the new concepts and definitions of the stochastic arithmetic, specially the definitions of order relations and equality relation. To be very short, these definitions take into account the accuracy of the operands. Then, CADNA is able to control every branching which is the second goal of the library.

CADNA works on Fortran (77 or 90), C, C++ and ADA codes. CADNA is a library which is used at the linking phase. New numerical types are available thanks to CADNA : the stochastic types. The library includes the definitions of all the elementary arithmetic operations, order relations and elementary functions defined for the classical numerical types. The round-off error control is only performed on the stochastic types and the accuracy estimation is available for any intermediate or final result. For the output, only the significant digits are displayed. When a result is a stochastic zero (i.e. is insignificant), the symbol @.0 is printed.

The last goal of CADNA is to give the users a tool for a real numerical debugging. CADNA detects numerical instabilities during the run time. It must be pointed out that this numerical debugging does not deal with the logical validity of the source code but with the capability of the computers to give correct results when the code is performed.

Of course, CADNA contains all the controls that are necessary for a good and efficient implementation of the CESTAC method. These controls, that were pointed out by the theoretical study, lead to a self-validation of the library. CADNA is able to detect when the conditions for a right estimation of the round-off errors are not satisfied anymore and when it happens, CADNA is able to advise the users.

Therefore, the numerical debugging and the self-validation of the CESTAC method are performed by systematically detecting some numerical instabilities. The users are warned by a trace that is let in a special file generated by CADNA. With this file, using the symbolic debugger, the users can find the line of the source code which is responsible for the instability.

For instance, the most important instabilities are :

1. The Instability of Division;

It means that a denominator of a division was a stochastic zero.

2. The Instability of Test;

It means that, when A <= B is tested, (A-B) is a stochastic zero. Following the stochastic definition, the answer corresponding to the equality is given but the user is advised that the mathematical answer can be the opposite. The last tool provided by CADNA is that data errors can be taken into account for the estimation of the final accuracy.

This library is based on the CESTAC which is a probabilistic approach for estimating round-off error propagation. The CADNA library defines new numerical types : stochastic types (single, double or complex). All their arithmetic operators, order relations, intrinsic or mathematical functions, have been overloaded. Consequently, the use of the CADNA library is very easy and only requires changing very few statements in Fortran source codes such as printing statements.

It is also possible with the CADNA library to take into account the physical data error in the estimation of the accuracy of the results.

CADNA library is a tool to implement the stochastic arithmetic on any Fortran code automatically. This library was written by Dr. Jean Marie Chesneaux in 1992.

According to the goals of the CADNA library, a perturbation method is proposed for performing a Fortran code in the stochastic arithmetic so that these goals are verified. In this case, the idea of the CESTAC method is considered, but the form of the perturbation of the bits is different.

In Chapter 2, at first, a review of the floating-point arithmetic is presented. Then, the idea of the CESTAC method and the stochastic arithmetic properties are discussed. Also, the quality of the implementations of the main operations in the stochastic arithmetic is explained which is used in the perturbation method. In the last section, some numerical algorithms are presented and the advantages of the stochastic arithmetic are shown in the examples.

In chapter 3, the numerical accuracy of indefinite integrals is explained. In this case, the accuracy of the Closed Newton-Cotes integration methods is proved as general. Some researches have been done about this topics which are developed them in this project. Also, the numerical accuracy of The Gauss-Chebyshev integration method is considered. In this case, by using the CADNA library, a numerical example is performed. According to the theorems, one can find the optimal number of the points in the integration methods which after this number, the accuracy of the results is fixed. For an improper integral, an optimal number is evaluated in which the value of the integral can be estimated by a definite integral. After this number, the value of the integral is negligible. In the last section, some definite and improper integrals are computed.

In chapter 4, the evaluation of optimal degree of interpolation polynomial by using the CADNA library is explained. One of the considerable discussions in data interpolation is to find the optimal number of data which minimizes the error of the

interpolation polynomial. For this purpose, at first, a theorem corresponding to the equidistant nodes is proved in order to estimate the accuracy of the interpolation polynomial, when the number of data increases. In the last section, three examples are implemented by using the Lagrange interpolation. In the numerical examples, the Runge's function is considered and is shown that, the Chebyshev nodes can provide a suitable interpolation polynomial which estimate the value of the function satisfactorily.

In this chapter, the optimal number of the points is evaluated, for the interpolation polynomial at the equidistant and Chebyshev points, such that at this number, the error is minimized. Also the accuracy of the interpolation polynomial at the optimal number of the points is shown.

Chapter 2

The Stochastic Arithmetic

2.1 Floating-Point Arithmetic

In the exact mathematics, any value $x \in \mathbb{R}$ can be represented as follows:

$$x = \epsilon m b^e, \tag{2.1.1}$$

where, ϵ is the sign of x, m is the unlimited mantissa, $b \geq 2$ is the radix and e is the exponent. If $b^{-1} \leq m < 1$, the relation (2.1.1) is called a normalized floating-point representation.

In a computer, the value x encoded in N digits. If F is the set of all the value representable in the computer then, x is in the form of $X \in F$ as follows:

$$X = \epsilon M b^E, \tag{2.1.2}$$

where, M is the limited mantissa, encoded on p digits in the radix b and E is the exponent encoded as an integer on q digits in the radix b. In general $E = e$, but the case $E \neq e$ is possible. The sign of X is encoded on one digits, therefore, $N = p+q+1$.

In the single precision case a word is represented by 32 bits which are enumerated from 0 to 31. In this case, the first bit from left is the sign bit which is 0 if X is

non-negative and 1 if it is negative. The next 7 bits are the exponent bits and the last 24 bits are the mantissa bits. Hence, $q = 7$ and $p = 24$.

In the double precision case a word is represented by 64 bits which are enumerated from 0 to 63. In this case, the first bit from left is the sign bit and the next 10 bits are the exponent bits and the last 53 bits are the mantissa bits. Hence, $q = 10$ and $p = 53$.

The values which exceed p significant digits can not be encoded exactly in the computer, consequently, they contain a relative error as follows:

$$a = \frac{X - x}{x} = \frac{M - m}{m} = \frac{-r}{m},$$

where, $r = m - M$ is the lost part of the mantissa m.

In a numerical algorithm if one wants to perform the operation $r = a\omega b, a, b, r \in \mathbb{R}$, where ω is an exact arithmetical operator, the computer performs the operation $R = A\Omega B, A, B, R \in F$, where Ω is a floating-point operator. For any Ω, the informatical results are contained rounding errors, because the results must be encoded in N digits. If $b = 2$ and $X \in F$ is a normalized floating-point value, then $0.5 \leq M < 1$ and $M = \sum_{i=1}^{p} a_i 2^{-i}$ where, $a_i \in \{0, 1\}$. Therefore;

$$m - M = \sum_{i=1}^{\infty} a_i 2^{-i} - \sum_{i=1}^{p} a_i 2^{-i} = \sum_{i=p+1}^{\infty} a_i 2^{-i} = 2^{-p} \sum_{i=1}^{\infty} a_{p+i} 2^{-i} = 2^{-p}\alpha,$$

where, $\alpha = \sum_{i=1}^{\infty} a_{p+i} 2^{-i}$. Therefore, $m - M = r = 2^{-p}\alpha$. If $\mid \alpha \mid \leq 0.5$ then, $\mid m - M \mid \leq 0.5 \times 2^{-p}$ and if $\mid \alpha \mid \leq 1$ then, $\mid m - M \mid \leq 2^{-p}$. Thus, the rounding error arising from the assignment operator is as follows:

$$x - X = \epsilon m 2^E - \epsilon M 2^E = \epsilon 2^E (m - M) = \epsilon 2^{E-p}\alpha,$$

hence,

$$X = x - \epsilon 2^{E-p}\alpha. \tag{2.1.3}$$

In relation (2.1.3), $2^{-p}\alpha$ is the lost part of the mantissa due to the round-off errors and E is the binary exponent of the result X.

The standard Institute of Electrical and Electronic Engineers (IEEE) defines four rounding modes [4]:

1. Rounding to the nearest;

   The floating-point number chosen is the nearest to the usual distance on $\mathbb{R}$. In this case, $-0.5 \leq \alpha < 0.5$.

2. Rounding to zero or Chopping;

   The floating-point number chosen is the closest to x in the interval with bounds 0 and x. In this case, $0 \leq \alpha < 1$.

3. Rounding to $+\infty$ or Rounding to front;

   The floating-point number chosen is the one immediately greater than x for the usual ordering relation on $\mathbb{R}$. In this case, $0 \leq \alpha < 1$.

4. Rounding to $-\infty$ or Rounding to back;

   The floating-point number chosen is the one immediately less than x for the usual ordering relation on $\mathbb{R}$. In this case, $-1 < \alpha \leq 0$.

Therefore, the scientific computations are used only in the bounded interval $I \subset \mathbb{R}$ and if x is a real number more than p mantissa bits, It must be approximated by a $X \in F$ in order to use in the computer. In the floating-point arithmetic with each kind of the rounding, the usual arithmetic properties are not satisfied. In other words, the computer arithmetic is an approximation of the exact arithmetic.

Let $x_1, x_2 \in \mathbb{R}$ and $X_1, X_2 \in F$, according to (2.1.3), $X_1 = x_1 - \epsilon_1 2^{E_1-p}\alpha_1$ and $X_2 = x_2 - \epsilon_2 2^{E_2-p}\alpha_2$.

The rounding errors arising from each of the arithmetical operations are as follows:

- Addition;

  Let $+$ and $\oplus$ be the exact and floating-point addition operators respectively. Therefore,

  $$X_1 \oplus X_2 = x_1 + x_2 - \epsilon_1 2^{E_1-p}\alpha_1 - \epsilon_2 2^{E_2-p}\alpha_2 - \epsilon_3 2^{E_3-p}\alpha_3,$$

  where, ϵ_3 , E_3 and α_3 are the sign, exponent and rounding error of the floating-point addition respectively.

- Subtraction;

  Let $-$ and $\ominus$ be the exact and floating-point subtraction operators respectively. Therefore,

  $$X_1 \ominus X_2 = x_1 - x_2 - \epsilon_1 2^{E_1-p}\alpha_1 + \epsilon_2 2^{E_2-p}\alpha_2 - \epsilon_3 2^{E_3-p}\alpha_3,$$

  where, ϵ_3 , E_3 and α_3 are the sign, exponent and rounding error of the floating-point subtraction respectively.

- Multiplication;

  Let $\times$ and $\otimes$ be the exact and floating-point multiplication operators respectively. Therefore,

  $$X_1 \otimes X_2 = x_1.x_2 - \epsilon_1 2^{E_1-p}\alpha_1 x_2 - \epsilon_2 2^{E_2-p}\alpha_2 x_1 + \epsilon_1\epsilon_2 2^{E_1+E_2-2p}\alpha_1\alpha_2 - \epsilon_3 2^{E_3-p}\alpha_3,$$

  where, ϵ_3 , E_3 and α_3 are the sign, exponent and rounding error of the floating-point multiplication respectively.

The forth term of the above relation is the second order in comparison with 2^{-p}, if the first approximation in comparison with 2^{-p} is considered then,

$$X_1 \otimes X_2 = x_1.x_2 - \epsilon_1 2^{E_1-p}\alpha_1 x_2 - \epsilon_2 2^{E_2-p}\alpha_2 x_1 - \epsilon_3 2^{E_3-p}\alpha_3 + O(2^{-2p}).$$

- Division;

  Let / and $\oslash$ be the exact and floating-point division operators respectively. Therefore,

$$X_1 \oslash X_2 = X_1 \otimes X_2^{-1} = (x_1 - \epsilon_1 2^{E_1-p}\alpha_1) \otimes (x_2 - \epsilon_2 2^{E_2-p}\alpha_2)^{-1} =$$

$$(x_1 - \epsilon_1 2^{E_1-p}\alpha_1) \otimes (x_2^{-1} + x_2^{-2}\epsilon_2 2^{E_2-p}\alpha_2 + ...) =$$

$$\frac{x_1}{x_2} + \epsilon_2 \frac{x_1}{x_2^2} 2^{E_2-p}\alpha_2 - \epsilon_1 \frac{\alpha_1}{x_2} 2^{E_1-p} - \epsilon_1\epsilon_2 \frac{\alpha_1\alpha_2}{x_2^2} 2^{E_1+E_2-2p} + ... - \epsilon_3 2^{E_3-p}\alpha_3.$$

  Consequently,

$$X_1 \oslash X_2 = \frac{x_1}{x_2} - \epsilon_1 2^{E_1-p}\frac{\alpha_1}{x_2} + \epsilon_2 2^{E_2-p}\frac{x_1\alpha_2}{x_2^2} - \epsilon_3 2^{E_3-p}\alpha_3 + O(2^{-2p}),$$

  where, ϵ_3, E_3 and α_3 are the sign, exponent and rounding error of the floating-point division respectively.

So, one can deduce the following theorem [8].

Theorem 2.1.1. *Let R be the informatical result of a finite sequence of elementary operations and r be the mathematical result of the same sequence of operations then,*

$$R = r + \sum_{i=1}^{n} g_i(d) 2^{E_i-p}\epsilon_i\alpha_i + O(2^{-2p}), \tag{2.1.4}$$

where, $g_i(d)$ are quantities depending only on the data and algorithm and independent from α_i. E_i, α_i and ϵ_i are the binary exponents, the lost part of the value because of the rounding and the signs of intermediate results. n is the number of the operations and p is the number of mantissa bits.

Consequently, R has the following form which is the first order approximation with respect to 2^{-p}.

$$R = r + \sum_{i=1}^{n} u_i(d)2^{-p}\alpha_i, \tag{2.1.5}$$

where $u_i(d) = \epsilon_i g_i(d)2^{E_i}, i = 1, ...n$.

Proposition 2.1.2. *Let r be a real value and R be its approximation. Then the number of the correct significant digits of R is estimated by:*

$$C_R = -log_{10} \mid \frac{R-r}{r} \mid . \tag{2.1.6}$$

Proof. It has been proved in [18], if n is the number of correct significant digits then, $\delta(R) \leq 5 \times 10^{-n}$ where, $\delta(R) = \frac{|R-r|}{|r|}$ is the relative error of R. Hence,

$$log_{10}\frac{\mid R-r \mid}{\mid r \mid} \leq log_{10}\frac{10^{-n+1}}{2} \Rightarrow log_{10}\frac{\mid R-r \mid}{\mid r \mid} \leq -n+1-log_{10}2 < -n+1$$

$$\Rightarrow n-1 < -log_{10} \mid \frac{R-r}{r} \mid, \ (*)$$

furthermore, $\delta(R) \geq 0.1 \times 10^{-n} = 10^{-n-1}$, hence, $log_{10}\frac{|R-r|}{|r|} \geq -n-1$ then,

$-log_{10} \mid \frac{R-r}{r} \mid \leq n+1 \ \ (**)$.

If we denote $C_R = -log_{10} \mid \frac{R-r}{r} \mid$, then from $(*)$ and $(**)$, C_R is an estimation of n. □

It can be proved like proposition 2.1.2, if C_R is an estimation of the number of correct significant bits then,

$$C_R = -log_2 \mid \frac{R-r}{r} \mid . \tag{2.1.7}$$

In order to prove (2.1.7), it suffices to consider $2^{-n-1} \leq \delta(R) \leq 2^{-n}$ in the proof.

Proposition 2.1.3. *The lost accuracy obtained from an informatical calculations is independent from the floating-point arithmetic accuracy.*

Proof. According to (2.1.5), $\mid R-r \mid = 2^{-p} \mid \sum_{i=1}^{n} u_i(d)\alpha_i \mid$, therefore by using (2.1.7),

$$\mid \frac{R-r}{r} \mid = \frac{2^{-p}}{r} \mid \sum_{i=1}^{n} u_i(d)\alpha_i \mid \Rightarrow C_R = -log_2(\frac{2^{-p}}{r} \mid \sum_{i=1}^{n} u_i(d)\alpha_i \mid) = p - log_2 \mid \sum_{i=1}^{n} u_i(d)\frac{\alpha_i}{r} \mid .$$

The second term of this relation is the lost accuracy in the calculation of R. This term is independent of p. □

Let a and b be two real numbers as follows:

$$a = \overline{a_1 a_2 ... a_k \cdot a_{k+1} ... a_{k+l} c_1 ... c_m} \quad , \quad b = \overline{a_1 a_2 ... a_k \cdot a_{k+1} ... a_{k+l} d_1 ... d_n},$$

where, $c_1 > d_1$. If $c = 0 \cdot c_1 ... c_m$ and $d = 0 \cdot d_1 ... d_n$ then,

$$\frac{a+b}{2} = a_1 a_2 ... a_k \cdot a_{k+1} ... a_{k+l} + 10^{-l}(\frac{c+d}{2}) \ , \ 0 < \frac{c+d}{2} < 1,$$

$$a - b = 10^{-l}(c-d) \ , \ 0.1 < c - d < 1,$$

consequently,

$$\frac{a+b}{2(a-b)} = \frac{a_1 a_2 ... a_k . a_{k+1} ... a_{k+l}}{10^{-l}(c-d)} + \frac{c+d}{2(c-d)} = a_1 a_2 ... a_{k+l} . \frac{1}{c-d} + \frac{c+d}{2(c-d)},$$

hence,

$$a_1 a_2 ... a_{k+l} < \frac{a_1 a_2 ... a_{k+l}}{c-d} < \frac{a+b}{2(a-b)} < \frac{a_1 a_2 ... a_{k+l} + 1}{c-d} < \frac{a_1 a_2 ... a_{k+l} + 1}{0.1},$$

therefore,

$$log_{10}(a_1 a_2 ... a_{k+l}) < log_{10} \mid \frac{a+b}{2(a-b)} \mid < 1 + log_{10}(a_1 a_2 ... a_{k+l} + 1).$$

Thus, the following definition can be presented as the number of common significant digits between a and b. This definition has been mentioned in [9].

Definition 2.1.1. The number of common significant digits between two distinct real numbers a and b, denoted by $C_{a,b}$, can be defined by,

$$C_{a,b} = log_{10} \mid \frac{a+b}{2(a-b)} \mid = log_{10} \mid \frac{a}{a-b} - \frac{1}{2} \mid .$$

If $a = b$ then, $C_{a,b} = +\infty$. Also, if $\mid a-b \mid << \mid a+b \mid$ then, $C_{a,b} \simeq log_{10} \mid \frac{a}{a-b} \mid$.

2.2 CESTAC Method

In this section, the modelization of a computed result is explained by using a perturbed arithmetic. Let $x, y \in \mathbb{R}$ and $X, Y \in F$ and $X\Omega Y$ be the result of a binary floating-point operation with p mantissa bits then, according to (2.1.3),

$$X\Omega Y = x\omega y - 2^{E-p}\epsilon\alpha, \tag{2.2.1}$$

where, ω is the exact arithmetical result and $2^{-p}\alpha$ is the lost part of the mantissa due to round-off error.

The CESTAC method was developed by LaPorte and Vignes [16,31,32], in order to estimate the accuracy of any computed result. For chopping arithmetic, the default value for the mantissa is always kept. But, the excess value has the same validity to represent the mathematical result. The method is based on the choosing of these two values (0 or 1) with the same probability of $\frac{1}{2}$. The implementation of this method consists in adding to the last bit of the mantissa of each result of the floating-point operation a quantity h which is 0 or 1 with the same probability.

For rounding to the nearest, h is -1 or +1 with the probability of $\frac{1}{4}$ and 0 with the probability of $\frac{1}{2}$. Consequently, the floating-point arithmetic is replaced by a random arithmetic which is called 'perturbed arithmetic'.

In order to perturb the result in (2.2.1), the CESTAC method adds a random quantity $+2^{E-p}\epsilon h$, which h is defined as explained in above. Then,

$$X\Omega Y = x\omega y - 2^{E-p}\epsilon(\alpha - h). \tag{2.2.2}$$

In other words, the CESTAC method replaces the rounding error α by $\alpha - h$. By this way the last mantissa bit is perturbed stochastically. Also this hypothesis is assumed that, the exponent and the sign of the result are independent of the quantity $\alpha - h$. Therefore, the following theorem is deduced [8].

Theorem 2.2.1. *Let R be the computed result using CESTAC method of a finite sequence which contains only the four elementary operations of the perturbed arithmetic. Let r be the mathematical result obtained with the same sequence of operations, then,*

$$R = r + \sum_{i=1}^{n} u_i(d)2^{-p}(\alpha_i - h_i) + O(2^{-2p}), \tag{2.2.3}$$

where, $u_i(d)$ are quantities depending only on data and algorithm. The α_i are random variables which depend on the previous h_i.

In relation (2.2.3), two following hypothesis have been considered:

1-The exponents and the signs of any intermediate results are independent of the quantities $\alpha_i - h_i$.

2-There exist L and U such that $L < u_i(d) < U,\ i = 1, ..., n$. On the other hand the coefficients $u_i(d)$ have a regularity.

The distribution of the random variables α_i is given by experience. Hamming and Knuth [19,20], have shown that the best assumption for the distribution of the mantissa is logarithmic distribution. On this basis, Feldstein and Goodman [17], have proved that, the round-off errors α_i can be considered as random variables uniformly

distributed on [0,1] for the chopping arithmetic and on $[\frac{-1}{2}, \frac{1}{2}]$ for the rounding to the nearest arithmetic. Thus $\alpha_i - h_i$ are uniformly distributed on [-1,1] for the chopping arithmetic and on $[\frac{-3}{2}, \frac{3}{2}]$ for the rounding to the nearest arithmetic based on theorem 2.2.1. The random variable R in (2.2.3), is identified to the term of the first order in 2^{-p}. Therefore, a computer result R can be modelized by the random variable R defined by:

$$R = r + \sum_{i=1}^{n} u_i(d) 2^{-p} r_i, \tag{2.2.4}$$

where, $u_i(d)$ are constants and $r_i = \alpha_i - h_i$ are independent uniformly distributed random variables. The accuracy of random variable R depends on its mean μ and its standard deviation σ. Let $R_i, i = 1, 2, ..., N$ be the computed results which are obtained by running N times the program with this new arithmetic. Chesneaux in [5,8,11] demonstrated that,

1- The mathematical expectation of the random variable R is equal to the mathematical value r. In other words, $E(R) = r$.

2- The distribution of the R_i is quasi-Gaussian (quasi-Normal), and therefore, it can be used the T-distribution or Student law on this distribution.

If the last mantissa bit of R is changed then, R is considered as a random variable as mentioned. In this case, the following theorem can be proved, which has been mentioned in [22].

Theorem 2.2.2. *Let $r \in \mathbb{R}$ be exact value and $R \in F$ is an approximation of r in the floating-point arithmetic. Let μ and σ be the mean and standard deviation of R respectively. Then, the number of the correct significant digits of R, denoted by C_R, is estimated as follows:*

$$C_R = log_{10} \frac{\mid \mu \mid}{\sigma}. \tag{2.2.5}$$

Proof. According to (2.1.6), $10^{-C_R} = \mid \frac{R-r}{r} \mid$, hence, $(1-\frac{R}{r})^2 = 10^{-2C_R}$. Since, $E(R) = \mu = r$ and $\sigma^2 = E(R^2) - (E(R))^2$ then,

$$E(1-\frac{R}{r})^2 = E(10^{-2C_R}) \Rightarrow E(1-2\frac{R}{r}+\frac{R^2}{r^2}) = 10^{-2C_R} \Rightarrow$$

$$1-\frac{2}{r}E(R)+\frac{1}{r^2}E(R^2) = 10^{-2C_R} \Rightarrow$$

$$1+(\frac{E(R)}{r})^2-\frac{2}{r}E(R)+\frac{E(R^2)}{r^2}-(\frac{E(R)}{r})^2 = 10^{-2C_R}$$

$$\Rightarrow \frac{1}{r^2}[r^2+\sigma^2-2rE(R)+(E(R))^2] = 10^{-2C_R}$$

$$\Rightarrow \frac{\sigma^2}{\mu^2} = 10^{-2C_R} \Rightarrow \frac{\sigma}{\mid \mu \mid} = 10^{-C_R},$$

consequently, $C_R = log_{10}\frac{|\mu|}{\sigma}$. □

According to (2.2.5), if $R_1, R_2, ..., R_N$ are the samples of random variable R, and $\overline{R}$ and S are their mean and standard deviation then, the number of correct significant digits of $\overline{R}$ or the number of significant digits common to $\overline{R}$ and r, denoted by $C_{\overline{R},r}$, is estimated by:

$$C_{\overline{R},r} = log_{10}\frac{\mid \overline{R} \mid}{S}. \tag{2.2.6}$$

It has been proved in [21], if R is a Normal random variable with mean μ and variance σ^2 then, if $R_1, R_2, ..., R_N$ are random samples of R, the random variable $\frac{\overline{R}-\mu}{S/\sqrt{N}}$ has T-distribution with $N-1$ degree of freedom, where $\overline{R}$ and S are the mean and standard deviation of the samples. If it is denoted, the T-distribution value with $N-1$ degree of freedom and the probability level $1-\beta$ by τ_β then,

$$\frac{\overline{R}-\mu}{S/\sqrt{N}} = \tau_\beta \Rightarrow \overline{R}-\mu = \frac{\tau_\beta S}{\sqrt{N}} \Rightarrow \frac{\overline{R}-\mu}{\overline{R}} = \frac{\tau_\beta S}{\sqrt{N}\ \overline{R}} \Rightarrow$$

$$-log_{10}(\mid \frac{\overline{R}-\mu}{\overline{R}} \mid) = -log_{10}(\frac{\tau_\beta S}{\sqrt{N} \mid \overline{R} \mid}) = log_{10}(\frac{\sqrt{N} \mid \overline{R} \mid}{\tau_\beta S}),$$

according to (2.1.6), $C_{\overline{R}} = log_{10}(\frac{\sqrt{N}|\overline{R}|}{\tau_\beta S})$, hence, the following definition is obtained;

Definition 2.2.1. The number of correct significant digits common to $\overline{R}$ and r, denoted by $C_{\overline{R},r}$, can be estimated by:

$$C_{\overline{R},r} = log_{10}(\frac{\sqrt{N} \mid \overline{R} \mid}{\tau_\beta S}), \tag{2.2.7}$$

where, $\overline{R}$ and S are evaluated as follows:

$$\overline{R} = \frac{1}{N}\sum_{i=1}^{N} R_i, \tag{2.2.8}$$

$$S = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(R_i - \overline{R})^2}. \tag{2.2.9}$$

If $S = 0$ then, it is considered, $C_{\overline{R},r} = +\infty$.

In the CESTAC method $N = 2$ or $N = 3$. If $N = 3$ and $\beta = 0.05$, according to the T-distribution table $\tau_\beta = 4.303$. If $N = 2$ and $\beta = 0.05$ then $\tau_\beta = 12.706$. Let $N = 3$ and $\beta = 0.05$, the relation (2.2.7), is simplified as follows:

$$C_{\overline{R},r} = log_{10}\frac{\mid \overline{R} \mid}{S} + log_{10}\frac{\sqrt{3}}{4.303} = log_{10}\frac{\mid \overline{R} \mid}{S} - 0.39,$$

consequently,

$$C_{\overline{R},r} = log_{10}\frac{\mid \overline{R} \mid}{S} - 0.39. \tag{2.2.10}$$

The validity of the CESTAC method depends on the use of Student's test to R defined by (2.2.4), to estimate r. If the probabilistic approach for the α_i is not completely satisfied, the variables $\alpha_i - h_i$ are not centered and the Student's test gives a biased estimate for the mathematical result r. In CESTAC method, the formula is used for the number of the significant digits which is very robust with respect to possible bias, this is due to the small number of the realization and to the structure of the formula. The logarithm has small variations. The approximation to the terms of first order is

more important for the method. It concerns only multiplications and divisions, and it depends on the accuracy of the result.

If R is a Normal (Gaussian) random variable with mean μ and variance σ^2 then, there is a $\lambda_\beta \in \mathbb{R}$ such that [31],

$$P(R \in [\mu - \lambda_\beta\sigma, \mu + \lambda_\beta\sigma]) = 1 - \beta.$$

The interval $I_{\beta,R} = [\mu - \lambda_\beta\sigma, \mu + \lambda_\beta\sigma]$, is the confidence interval of μ at level $1-\beta$. The λ_β is the standard Normal distribution value with the probability $1-\beta$. In practice, in order to have the same confidence interval for $\overline{R}$ with N sample values $R_i, i = 1, 2, ...N$ obtain by CESTAC method, it is considered the interval $J_{\beta,R} = [\overline{R} - \tau_\beta S, \overline{R} + \tau_\beta S]$. Because, when N tends to $+\infty$, $\overline{R}$, S and τ_β tend to μ, σ and λ_β respectively, the following proposition is concluded:

Proposition 2.2.3. $\lim_{N\to\infty} J_{\beta,R} = I_{\beta,R}$.

The following value is an upper bound for the number of common significant digits between μ and each element of $I_{\beta,R}$ [6].

$$C_{\beta,R} = log_{10}(\frac{\mid \mu \mid}{\lambda_\beta\sigma}),\ \sigma \neq 0. \tag{2.2.11}$$

If $\sigma = 0$ then, $C_{\beta,R} = +\infty$. In relation (2.2.11), N has not been appeared. If 0 does not belong to $I_{\beta,R}$ then, $\mid \mu \mid > \lambda_\beta\sigma$. In this case, $C_{\beta,R} > 0$. If $\beta = 0.05$ then, $\lambda_\beta = 1.96$. Therefore, the relation (2.2.11) is simplified as follows:

$$C_{\beta,R} = log_{10}\frac{1}{\lambda_\beta} + log_{10}\frac{\mid \mu \mid}{\sigma} = log_{10}\frac{\mid \mu \mid}{\sigma} - 0.29,$$

thus,

$$C_{\beta,R} = log_{10}\frac{\mid \mu \mid}{\sigma} - 0.29. \tag{2.2.12}$$

According to (2.2.10) and (2.2.12), if $N = 3$ and $\beta = 0.05$ the values $C_{\overline{R},r}$ and $C_{\beta,R}$ are close to each other.

2.2.1 The Algorithm of the CESTAC Method

When the CESTAC method is used in order to implement an algorithm, any result $R \in F$ is a random variable with mean μ and variance σ^2. Hence, a stochastic arithmetic should be used. When the last mantissa bit is perturbed N times, the samples $R_1, R_2, ..., R_N$ are produced which are used to estimate μ and σ. Then, the mean of these samples is considered as estimation of the result and the variance of them is used to determine the accuracy of the result. The main idea of the CESTAC method is to evaluate the number of significant digits of the result R effectively. The number N can be any natural number like 2,3,5,7, but since every result r of every arithmetical operator is bounded between two floating-point values R^+ (rounding to front) and R^- (rounding to back), usually $N = 3$ is considered to estimate R. The algorithm of this method is as follows:

1. Implement the program N times synchronously by using new stochastic arithmetic. Thus, N realization, $R_i, i = 1, 2, ...N$, are obtained.

2. Evaluate $\overline{R} = \frac{1}{N} \sum_{i=1}^{N} R_i$ in order to estimate the result R.

3. Evaluate the number of common significant digits between $\overline{R}$ and the exact value $r \in \mathbb{R}$, $C_{\overline{R},r}$, using the relation (2.2.7) to find the accuracy of the result.

If $N = 3$, the relation (2.2.10) is considered to evaluate $C_{\overline{R},r}$. In the CESTAC method if $C_{\overline{R},r} \leq 0$, there is an instability in the evaluated result. In this case, the result is not reliable.

2.2.2 The Algorithm of the Perturbation Method

Let $r \in \mathbb{R}$ be the exact value and $R \in F$ be its correspond value in the floating-point arithmetic. In place of the perturbation the last mantissa bit by replacing $\alpha - h$ by α stochastically and finding the samples $R_i, i = 1, 2, ...N$, the error α can change uniformly on the interval [-1,1]. In relation (2.1.3), If $N = 3$ then, α should be changed 3 times in this interval uniformly. Thus, the algorithm of the perturbation method is as follows:

1. Produce N stochastic number on the interval [0,1] uniformly. Then, by using the variable change $\alpha_i = a + u_i(b-a), 0 \leq u_i \leq 1, i = 1, ..., N$, find N stochastic number in the interval [a,b]. If [a,b]=[-1,1] then, $\alpha_i = -1 + 2u_i$.

2. By using the relation (2.1.3), find N samples $R_i,\ i = 1, 2, ..., N$; where,

   $R_i = R - \epsilon 2^{E-p}\alpha_i$.

3. Evaluate $\overline{R} = \frac{1}{N}\sum_{i=1}^{N} R_i$ in order to estimate the result R.

4. Evaluate $C_{\overline{R},r}$ by using the relation (2.2.7) to find the accuracy of the result.

In the perturbation method it suffices for perturbing a floating-point value like X, to use the subroutine rand(X, XX, eps). In this subroutine, X is considered as a random number that must be perturbed. XX is a 3-array which is obtained by using perturbation algorithm. $'eps'$ is the magnitude of the perturbation. If $eps < 2^{-p}$ then, the last bit of the mantissa is perturbed. If one wants to perturb the previous bits, eps must be chosen so that $eps \geq 2^{-p}$. In this case, if p' is the smallest integer number where, $2^{-p} \leq 2^{-p'} \leq eps$ then, $-p' log_{10}2 \leq log_{10} eps$ therefore, if $p' = \lceil | \frac{log_{10} eps}{log_{10} 2} | \rceil$ then, $p' < 24$ in single and $p' < 53$ in double precision. $\lceil . \rceil$ means rounding to front.

In this case, the p'th bit is perturbed. It is natural, if the previous bits are perturbed the accuracy of the estimation decreases, so, this action should be done if necessary. Usually when the variance of the samples is zero, the previous bits should be perturbed.

The subroutine 'rand' and 'vari' are printed in appendix 2. In the subroutine vari(w,ave,var,cad), c, the number of significant digits of the mean value of the array w, is evaluated. If $c \leq 0$ then, the result has not any significant digits. In this case, the notation @0 is printed.

2.2.3 The Advantages of Using the CESTAC or Perturbation Method

The applying of the CESTAC or perturbation method in a scientific program has the following advantages:

1. The accuracy of any numerical result is estimated, during the running of a program.

2. The numerical instabilities are detected and the branching are checked.

3. Unnecessary iterations are eliminated which the floating-point arithmetic is not able to distinguish them. In some cases, the termination criterion of iterative methods is not suitable so that, the implementation of the algorithm is continued without improvement in the accuracy of the result. In the stochastic arithmetic, instead of the termination criterion, a criterion that directly reflects the mathematical condition, is replaced, that must be satisfied by the solution.

4. It is able to find the optimal step of the iterative methods, which after this step, the accuracy of the result does not increases or maybe decreases, because of the rounding error accumulation.

5. It is an effective and powerful tool that helps to achieve the validation of scientific programs and gives them a reliability.

2.3 The Main Definitions and Properties of the Stochastic Arithmetic

By using the CESTAC or perturbation method, the N results of each arithmetical operation can be considered as realization of a Gaussian Random variable centered on the exact mathematical result. In this case, a new number called stochastic number and a new arithmetic called stochastic arithmetic are defined. In this section, the main definitions and properties of this arithmetic are explained. The details can be found in [4,6,12,25,33].

Definition 2.3.1. The set of stochastic numbers is denoted by S which is applied to the set of Gaussian random variables. An element $X \in S$ is denoted by $X = (\mu, \sigma^2)$ where, μ is the mean and σ^2 is the variance of X.

Definition 2.3.2. $X \in S$ is a "stochastic zero", denoted by $\underline{0}$, if and only if,

$$X = (0, 0) \quad or \quad C_{\beta,X} \leq 0. \tag{2.3.1}$$

Definition 2.3.3. A computed result by using the CESTAC or perturbation method $\overline{R}$ is an "informatical zero", denoted by @0, if and only if,

$$\overline{R} = 0 \quad or \quad C_{\overline{R},r} \leq 0. \tag{2.3.2}$$

Definition 2.3.4. Let $X_1 = (\mu_1, \sigma_1^2)$ and $X_2 = (\mu_2, \sigma_2^2)$ be elements of S. The four elementary operations, denoted S^+, S^-, S^* and $S^/$, are defined as follows:

$$
\begin{aligned}
X_1 S^+ X_2 &= (\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)\,, \\
X_1 S^- X_2 &= (\mu_1 - \mu_2, \sigma_1^2 + \sigma_2^2)\,, \\
X_1 S^* X_2 &= (\mu_1 \mu_2, \mu_2^2 \sigma_1^2 + \mu_1^2 \sigma_2^2)\,, \\
X_1 S^/ X_2 &= \left(\tfrac{\mu_1}{\mu_2}, \left(\tfrac{\sigma_1}{\mu_2}\right)^2 + \left(\tfrac{\mu_1 \sigma_2}{\mu_2^2}\right)^2 \right); \mu_2 \neq 0.
\end{aligned}
$$

Definition 2.3.5. Let $X, Y \in S$, X is stochastically equal to Y, denoted $X S^= Y$ if and only if, $X S^- Y = \underline{0}$.

Definition 2.3.6. Let $X_1 = (\mu_1, \sigma_1^2)$ and $X_2 = (\mu_2, \sigma_2^2)$ be elements of S. The comparative operators are defined as follows:

$$
\begin{aligned}
X_1 S^> X_2 &\Leftrightarrow \mu_1 - \mu_2 > \lambda_\beta \sqrt{\sigma_1^2 + \sigma_2^2}, \\
X_1 S^< X_2 &\Leftrightarrow \mu_2 - \mu_1 > \lambda_\beta \sqrt{\sigma_1^2 + \sigma_2^2}, \\
X_1 S^\geq X_2 &\Leftrightarrow X_1 S^= X_2 \;\; or \;\; \mu_1 \geq \mu_2, \\
X_1 S^\leq X_2 &\Leftrightarrow X_1 S^= X_2 \;\; or \;\; \mu_1 \leq \mu_2.
\end{aligned}
$$

Let $x_1, x_2 \in \mathbb{R}$, $X_1 = (\mu_1, \sigma_1^2)$ and $X_2 = (\mu_2, \sigma_2^2)$ be elements of S, the following properties can be proved, according to above definitions:

1. If $\mu_1 = \mu_2$ then, $X_1 S^= X_2$,
2. $x_1 = x_2 \Leftrightarrow X_1 S^= X_2$,
3. $X_1 S^> X_2 \Rightarrow \mu_1 > \mu_2$,
4. $\mu_1 \geq \mu_2 \Rightarrow X_1 S^\geq X_2$,

5. $X_1 S^{\geq} X_2 \Leftrightarrow X_1 S^{>} X_2$ *or* $X_1 S^{=} X_2$,

6. $X_1 S^{=} X_2$ and $X_2 S^{<} X_3 \Rightarrow X_1 S^{\leq} X_3$,

7. $X_1 S^{\leq} X_2 \;\Rightarrow\; X_1 S^{=} X_2$ *or* $X_1 S^{<} X_2$,

8. $X_1 S^{\geq} X_2$ and $X_2 S^{\geq} X_1 \Rightarrow X_1 S^{=} X_2$,

9. $\underline{0}$ is the identity element of the stochastic addition,

   $\forall X \in S,\ X\ S^{+}\ \underline{0}\ =\ X$,

10. $\forall X \in S,\ X\ S^{*}\ \underline{0} = \underline{0}$,

11. $S^{=}$ is a reflective and symmetric relation, but is not a transitive relation.

12. Let $X \in S$ then, $X = \underline{0}$ if and only if, $0 \in I_{\beta,X}$. In other words, a stochastic number is a stochastic zero, if and only if, 0 is in its confidence interval. The following statements are equivalence:
    a) $X = \underline{0}$; b) $\mid \mu \mid \leq \lambda_\beta \sigma$; c) $0 \in I_{\beta,X}$,

13. $S^{>}$ is a transitive relation, and $S^{\geq}$ is an anti-symmetric and reflexive relation, but it is not a transitive relation,

14. The associativity and distributivity rules hold for the stochastic arithmetic.

Thus, stochastic arithmetic is able to retrieve most of the properties of the exact arithmetic, which have lost in the floating-point arithmetic. Of course, there are some differences between the exact and stochastic arithmetic like the properties 11 and 13.

2.4 CADNA Library

In order to implement the CESTAC or perturbation method on a program, the floating-point arithmetic must be substituted by the stochastic arithmetic. Then, each operation is implemented $N = 2$ or 3 times synchronously, before implementing the next operation. If one wants to run a program on the stochastic arithmetic, each line of the program must be changed by an array. In other words, the program must be rewritten in the stochastic arithmetic. In this case, all of the mathematical functions and instructions must be changed. The Fortran 77 and 90 are suitable programming languages in order to use the stochastic arithmetic. Because in this new arithmetic for every floating-point number there is an N-arrays of the perturbed floating-point number, the operands are N-arrays and thus, the result of an operation is an N-array. In the CADNA library, in order to avoid the increase of execution time of the program, $N = 3$ is considered.

CADNA was designed for implementing the stochastic arithmetic in 1990. This library is able to run the CESTAC method on any Fortran program simultaneously. By using CADNA library the floating-point arithmetic is replaced by the stochastic arithmetic automatically. This library was written in ADA at first and then was provided in Fortran in 1992 [7,10,34].

In order to use the CADNA library, it suffices to place the instruction USE CADNA at the top of the Fortran program. Also it must be replaced the real and complex type of values by the stochastic type, and changed the output instruction. A sample program in order to run with CADNA library is shown in the following:

```
program sample
use cadna
type (double_st) The list of the real variables
call cadna_init(-1)
The Main Program
print *, "value=",Str(value)
end
```

The function "Str" in the output instruction shows only the significant digits of the value. In other words, in CADNA only the significant digits are printed. Hence, if the number of significant digits become zero, it is shown with the notation @0. It means that the value is an informatical zero. Also, if the real values are in single precision, it must be placed type(single_st) at the related line (similarly for the complex value). Some programs are printed, which have been run with the CADNA library in appendix 2. By using CADNA library any instabilities and detecting insignificant results have been shown. In CADNA, all the arithmetical operators and ordered relations have been overloaded and input/output instructions have been modified. If an informatical zero is detected in an instruction or a branch then, the warning message 'Instability Detected in the Branching' is written which means an instability has been detected in the related branch or line and it is shown by @0. In this case, the user should find the cause of this instability and note that the result of the algorithm is not reliable.

The program execution time by using the CADNA library is about 3 times of executing it, using the floating-point arithmetic, but this is acceptable because of the important advantages of this tool which is the validation of the result and algorithm.

2.5 The Quality of Implementation of the Elementary Operations in Stochastic Arithmetic

Let $a, b, c \in \mathbb{R}$ and $c = a\omega b$ where, ω be the exact operator. The correspond floating-point operator is $C = A\Omega B$, where, $A, B, C \in F$. In the stochastic arithmetic A, B, C are considered as random variables in S and the operation is in the form of $C = AS^{\omega}B$. In order to implement the stochastic operator S^{ω}, N random samples are obtained as mentioned, for any of the values A and B. The operation Ω is performed in the form of $c_i = a_i \Omega b_i;\ i = 1, 2, ..., N$, in this new arithmetic. The number N can be any natural number such as $N = 2, 3, 5, 7$, but in order to avoid increasing the number of arithmetical operations, it is considered $N = 3$. Let μ_a, μ_b and μ_c be the means and σ_a^2, σ_b^2 and σ_c^2 be the variance of the random samples a_i, b_i and c_i respectively. The mean and variance of c can be obtained directly by using the samples a_i and b_i so that the relations in definition 2.3.4 be established exactly or approximately. It is proved for any operation in the following. At last, μ_c is considered as the result of the stochastic operation S^{ω}. The σ_c is used for estimating the number of correct significant digits of this result.

2.5.1 Addition and Subtraction operations;

Let $c = a \pm b$ then, $c_i = a_i \pm b_i;\ i = 1, 2, ..., N$.

Hence,

$$\mu_a \pm \mu_b = \frac{1}{N}\sum_{i=1}^{N} a_i \pm \frac{1}{N}\sum_{i=1}^{N} b_i = \frac{1}{N}\sum_{i=1}^{N} (a_i \pm b_i) = \frac{1}{N}\sum_{i=1}^{N} c_i = \mu_c,$$

consequently, $\mu_a \pm \mu_b = \mu_c$. Furthermore, since the random variables a and b are independent, the covariance between them is zero. Thus, $E[(a - \mu_a)(b - \mu_b)] = 0$,

hence,

$$\sum_{i=1}^{N}[(a_i-\mu_a)(b_i-\mu_b)]P_X(x_i)=0,$$

where, $X=(a-\mu_a)(b-\mu_b)$ and $x_i=(a_i-\mu_a)(b_i-\mu_b)\,;\ i=1,2,...,N$. Because of the uniformity of random variables x_i, the probability of occurrence for each one of them is equal to $\frac{1}{N}$.

Hence,

$$\sum_{i=1}^{N}(a_i-\mu_a)(b_i-\mu_b)=0, \tag{2.5.1}$$

consequently,

$$\sigma_c^2=\frac{1}{N-1}\sum_{i=1}^{N}(c_i-\mu_c)^2=\frac{1}{N-1}\sum_{i=1}^{N}(a_i\pm b_i-(\mu_a\pm\mu_b))^2$$

$$=\frac{1}{N-1}\sum_{i=1}^{N}(a_i-\mu_a)^2+\frac{1}{N-1}\sum_{i=1}^{N}(b_i-\mu_b)^2=\sigma_a^2+\sigma_b^2,$$

hence, $\sigma_a^2+\sigma_b^2=\sigma_c^2$.

2.5.2 Multiplication operation;

Let $c=a.b$, then $c_i=a_i.b_i;\ i=1,2,...,N$. According to (2.5.1),

$$0=\frac{1}{N}\sum_{i=1}^{N}(a_i-\mu_a)(b_i-\mu_b)=\frac{1}{N}\sum_{i=1}^{N}a_ib_i-\frac{\mu_b}{N}\sum_{i=1}^{N}a_i-\frac{\mu_a}{N}\sum_{i=1}^{N}b_i+\mu_a\mu_b,$$

consequently, $\mu_a\mu_b=\mu_c$. Besides,

$$\sigma_c^2=\frac{1}{N-1}\sum_{i=1}^{N}(c_i-\mu_c)^2=\frac{1}{N-1}\sum_{i=1}^{N}(a_ib_i-\mu_a\mu_b)^2$$

$$=\frac{1}{N-1}\sum_{i=1}^{N}\left(a_i^2b_i^2-\mu_a^2\mu_b^2\right)=\frac{1}{N-1}\sum_{i=1}^{N}[a_i^2b_i^2-\mu_a^2\mu_b^2+a_i^2\mu_b^2-a_i^2\mu_b^2]$$

$$= \frac{1}{N-1}\sum_{i=1}^{N}\left(a_i^2 - \mu_a^2 + \mu_a^2\right)\left(b_i^2 - \mu_b^2\right) + \frac{1}{N-1}\mu_b^2\sum_{i=1}^{N}\left(a_i^2 - \mu_a^2\right)$$

$$= \frac{1}{N-1}\sum_{i=1}^{N}\left(a_i^2 - \mu_a^2\right)\left(b_i^2 - \mu_b^2\right) + \frac{1}{N-1}\mu_a^2\sum_{i=1}^{N}\left(b_i^2 - \mu_b^2\right) + \frac{1}{N-1}\mu_b^2\sum_{i=1}^{N}\left(a_i^2 - \mu_a^2\right).$$

The first term of the last relation is almost zero, because, $\mu_a \simeq a_i$ and $\mu_b \simeq b_i$. Therefore,

$$\sigma_c^2 \simeq \frac{1}{N-1}\mu_a^2\sum_{i=1}^{N}\left(b_i^2 - \mu_a^2\right) + \frac{1}{N-1}\mu_b^2\sum_{i=1}^{N}\left(a_i^2 - \mu_a^2\right).$$

On the other hand,

$$\mu_a^2\sigma_b^2 + \mu_b^2\sigma_a^2 = \frac{1}{N-1}[\mu_a^2\sum_{i=1}^{N}(b_i - \mu_b)^2 + \mu_b^2\sum_{i=1}^{N}(a_i - \mu_a)^2]$$

$$= \frac{1}{N-1}\mu_a^2\sum_{i=1}^{N}b_i^2 - \frac{N}{N-1}\mu_a^2\mu_b^2 + \frac{1}{N-1}\mu_b^2\sum_{i=1}^{N}a_i^2 - \frac{N}{N-1}\mu_a^2\mu_b^2$$

$$= \frac{1}{N-1}\mu_a^2\sum_{i=1}^{N}\left(b_i^2 - \mu_b^2\right) + \frac{1}{N-1}\mu_b^2\sum_{i=1}^{N}\left(a_i^2 - \mu_a^2\right),$$

hence, $\mu_a^2\sigma_b^2 + \mu_b^2\sigma_a^2 \simeq \sigma_c^2$.

2.5.3 Division operation;

Let $c = \frac{a}{b}$, then $c_i = \frac{a_i}{b_i};\ i = 1, 2, ..., N$. Hence,

$$\frac{\mu_a}{\mu_b} = \frac{\frac{1}{N}\sum_{i=1}^{N}a_i}{\frac{1}{N}\sum_{i=1}^{N}b_i} = \frac{\sum_{i=1}^{N}a_i}{b_1b_2...b_N} \Big/ \frac{\sum_{i=1}^{N}b_i}{b_1b_2...b_N}$$

$$= \frac{a_1}{b_1}\left[\frac{1}{1 + \frac{b_2}{b_1} + \frac{b_3}{b_1} + ... + \frac{b_N}{b_1}}\right] + \frac{a_2}{b_2}\left[\frac{1}{1 + \frac{b_1}{b_2} + \frac{b_3}{b_2} + ... + \frac{b_N}{b_2}}\right]$$

$$+ \cdots + \frac{a_N}{b_N}\left[\frac{1}{1 + \frac{b_1}{b_N} + \frac{b_2}{b_N} + ... + \frac{b_{N-1}}{b_N}}\right] \simeq \frac{1}{N}\left(\frac{a_1}{b_1} + \frac{a_2}{b_2} + \cdots + \frac{a_N}{b_N}\right)$$

$$= \frac{1}{N}\left(c_1 + c_2 + ... + c_N\right) = \mu_c,$$

consequently, $\frac{\mu_a}{\mu_b} \simeq \mu_c$ (We note that, $b_1 \simeq b_2 \simeq \cdots \simeq b_N$). Also for the variance of c, since $\sum_{i=1}^{N} \frac{a_i}{b_i} \simeq N\frac{\mu_a}{\mu_b}$, it is concluded that,

$$\sigma_c^2 = \frac{1}{N-1}\sum_{i=1}^{N}\left(c_i - \mu_c\right)^2 = \frac{1}{N-1}\sum_{i=1}^{N}\left(\frac{a_i}{b_i} - \frac{\mu_a}{\mu_b}\right)^2$$

$$\simeq \frac{1}{N-1}\sum_{i=1}^{N}\left[\frac{a_i^2}{b_i^2} - \frac{\mu_a^2}{\mu_b^2}\right].$$

Besides,

$$\frac{\sigma_a^2}{\mu_b^2} + \frac{\mu_a^2\sigma_b^2}{\mu_b^4} = \frac{1}{(N-1)\mu_b^2}[\sum_{i=1}^{N}\left(a_i^2 - \mu_a^2\right) + \frac{\mu_a^2}{\mu_b^2}\sum_{i=1}^{N}\left(b_i^2 - \mu_b^2\right)]$$

$$= \frac{1}{N-1}\sum_{i=1}^{N}[\frac{a_i^2}{\mu_b^2} - \frac{\mu_a^2}{\mu_b^2} + \frac{\mu_a^2 b_i^2}{\mu_b^4} - \frac{\mu_a^2}{\mu_b^2}]$$

$$= \frac{1}{N-1}\sum_{i=1}^{N}\left[\frac{a_i^2\mu_b^2 + \mu_a^2\left(b_i^2 - \mu_b^2\right)}{\mu_b^4} - \frac{\mu_a^2}{\mu_b^2}\right].$$

Since, $b_i \simeq \mu_b;\ i = 1, 2, ..., N$, thus the term $\mu_a^2\left(b_i^2 - \mu_b^2\right)$ is negligible. Therefore,

$$\frac{\sigma_a^2}{\mu_b^2} + \frac{\mu_a^2\sigma_b^2}{\mu_b^4} \simeq \frac{1}{N-1}\sum_{i=1}^{N}\left(\frac{a_i^2}{\mu_b^2} - \frac{\mu_a^2}{\mu_b^2}\right)$$

$$\simeq \frac{1}{N-1}\sum_{i=1}^{N}\left(\frac{a_i^2}{b_i^2} - \frac{\mu_a^2}{\mu_b^2}\right),$$

consequently,

$$\sigma_c^2 \simeq \left(\frac{\sigma_a}{\mu_b}\right)^2 + \left[\frac{\mu_a\sigma_b}{\mu_b^2}\right]^2.$$

2.6 The Termination Criteria In The Stochastic Arithmetic

In this section, two kinds of the numerical methods are considered in order to implement on the stochastic arithmetic, which are finite and iterative methods. In the finite methods there are an ordered and finite sequence of the scientific calculations and related branching. When these methods are used to run in the floating-point arithmetic, the results may be false without awareness of the user. But using stochastic arithmetic, the validation of the results is determined and if the results are not reliable, a warning is flagged for the user. When the number of the significant digits of a result is zero, the result is not valid. In this case, the result is called an informatical zero. The floating-point arithmetic is not able to detect an informatical zero.

In the iterative methods, a sequence like x_n is made such that it tends to the exact value x when n tends to infinity. The usual termination criteria in these methods are:

1- If $\| x_n - x_{n-1} \| \leq \epsilon$ then stop,

2- If $\| x_n - x_{n-1} \| \leq \epsilon \| x_n \|$ then stop,

where, ϵ is an arbitrary positive value. These criteria may not be acceptable. If ϵ is chosen very large, the iterations are stopped before getting access to a suitable approximation. If ϵ is chosen very small then, unnecessary iterations are done without improving the accuracy of the results. In this case, x_n has finite significant digits and the termination criteria is meaningless. Furthermore,

1- $\| x_n - x_{n-1} \| \leq \epsilon \ \Rightarrow \| x_n - x \| \leq \epsilon$,

2- $\| x_n - x_{n-1} \| \leq \epsilon \| x_n \| \Rightarrow \| x_n - x \| \leq \epsilon \| x \|$,

In the stochastic arithmetic, the following criteria are used:

1-If $(x_n == x_{n-1})$ then stop,

2-If $(f(x_n) == \underline{0})$ then stop.

These criteria mean,

1-If $\| x_n - x_{n-1} \| S^{=}\underline{0}$ then stop,

2-If $(f(x_n)S^{=}\underline{0})$ then stop.

When these criteria are happened, the iterative process is stopped at nth iteration. In this case, the algorithm is stationary and x_n is a satisfactory informatical solution of the algorithm.

Some of the problems like evaluating the series $\sum_{k=0}^{\infty} \frac{x^k}{k!}$, are not controllable directly. In this case, the termination criterion is:

If $\mid s_k - s_{k-1} \mid \leq \epsilon \mid s_k \mid$ then stop,

where, s_k is the sum of the first k terms of the series.

In the stochastic arithmetic, the termination criterion is defined as follows:

If $(s_k == s_{k-1})$ then stop.

It means that, when any improvement is not obtained in the results, the evaluations are stopped.

For the problems which are controllable directly, like evaluating the roots of a nonlinear equation with Newton-Raphson method, the termination criterion is:

If $\mid f(x_n) \mid \leq \epsilon$, then stop.

By using stochastic arithmetic, when a satisfactory solution is obtained, the iterative process is stopped. In this case, one of the following termination criteria is considered:

1- If $(f(x_n) == \underline{0})$ then stop.

It means that, if $f(x_n)S^{=}\underline{0}$ then stop. This is an optimal criterion, which the

value of the function in the nth iteration is equal to informatical zero. If the method is convergent then, this criterion is happened and x_n is a satisfactory solution.

2- If $(x_n == x_{n-1})$ then stop.

It means that, when $\| x_n - x_{n-1} \| = \underline{0}$ the evaluations are stopped. In this case, the continuation of the iterations is not necessary, and n is an optimal value for the number of iterations.

2.7 Numerical Examples

In this section, some numerical algorithms are shown which are performed in the stochastic arithmetic by using the perturbation method. The following examples have been provided with Fortran 90 in single precision.

Example 2.7.1. *The stochastic arithmetic is able to save computer time because many useless iteration are not performed. In this example, the sum of the terms of the series* $e^x = \sum_{i=0}^{\infty} \frac{x^i}{i!}$ *is calculated for some special x [30]. The values n in the table 2.1 are the optimal values of n in calculating the series terms. The program is terminated when the n-th term is equal to zero stochastically. In the third row of the table, the solution is accompanied by formation of an informatical zero in optimal step. Therefore, this result is not reliable.*

row	x	n	e^x	exact solution	explanations
1	1.3	12	3.669296	3.669296668	
2	-5.8	31	3.02655e-03	3.027554e-03	
3	-9.7	45	1.176382e-04	6.1283e-05	informatical zero has been detected.

Table 2.1

Example 2.7.2. *In the floating-point arithmetic, when some programs are performed on a computer, because of the accumulation of rounding error, the false results maybe furnished to the user without his knowledge. But, when these programs are performed with stochastic arithmetic, the numerical instabilities are detected and the results are shown correctly. In this example, solving quadratic equation* $0.3x^2-2.1x+3.675=0$ *is considered [34]. The true solution is* $x_1=x_2=3.5$. *Because of detecting a stochastic zero in the branch related to discriminant of the equation, the results show the true solutions.*

results in stochastic arithmetic	results in floating-point arithmetic
x_1=3.499999	x_1=(3.5,1.200186e-03)
x_2=3.499999	x_2=(3.5,-1.200186e-03)
stochastic zero has been detected.	complex solutions

Table 2.2

Example 2.7.3. *Stochastic arithmetic is able to eliminate epsilon in the termination criteria, and stop the iterative process, as soon as, a satisfactory solution is reached. In this example, solving the following equation is considered [34]:*

$$f(x)=1.47x^3+1.19x^2-1.83x+0.45,$$

the exact solution is $x=0.428571$. *It is solved by the Newton-Raphson method with* $x_0=0.5$. *In the floating-point arithmetic, when epsilon is chosen too small , many useless iterations are performed without improving the accuracy of the approximate solutions. In the stochastic arithmetic, the condition* $x_{n+1}-x_n=\underline{0}$ *is used in place of* $|\ f(x_n)\ |<\epsilon$ *in the termination criterion, so the solution is provided in an optimal* n.

stochastic arithmetic		floating-point arithmetic		
approximate solution	optimal No. of iteration	approximate solution	No. of iteration	epsilon
4.286512e-01	10	4.377610e-01	3	1e-3
(single precision)		4.288664e-01	8	1e-6
4.286469117260276e-001	15	4.286470e-01	12	1e-9
(double precision)		4.286469e-01	13	1e-11
		4.286469e-01	500	1e-12

Table 2.3

Example 2.7.4. *One of the main problems in the interpolation of data, is that it seems the error decreases, when the number of the data ,n, increases. But, this is not correct numerically. Stochastic arithmetic is able to find the suitable number of data, which after this number the interpolation error may not vanish. In this example, by using the Neville interpolation, the value of infinite norm of the error is calculated for the function* $f(x) = \frac{1}{1+x^2}$ *in which* $-1 \leq x \leq 1$. *The nodes are considered the zeros of the Chebyshev polynomial* $T_n(x)$ *in the form of* $t_i = cos\frac{(2i+1)\pi}{2n},\ i = 0, 1, 2, ..., n-1$. *The value* $ermax = Max \mid f(t_i) - p_{1n}(t_i) \mid$ *is calculated over 101 equidistant points* $t_i = -1 + 0.02i;\ i = 0, 1, \ldots, 100$ *and determined in the table 2.4. The accuracy is* $ermax \leq 10^{-5}$. *As it is observed, when* $n = 14$ *this accuracy is furnished, but when* $n = 20$ *the value ermax has not any significant digit.*

n	ermax	explanations
2	3.333333e-01	
4	5.882353e-02	
6	1.010096e-02	
8	1.733124e-03	
10	2.973676e-04	
12	5.102158e-05	
14	8.642673e-06	$ermax \leq 10^{-5}$
16	1.490116e-06	
18	3.576279e-07	
20	8.940697e-08	$ermax = @0$

Table 2.4

In the following examples the value c is an estimation of the number of significant digits which is evaluated by relation (2.2.10).

Example 2.7.5. *Stochastic arithmetic is able to determine the divergence of an integral. Let consider* $\int_{-1}^{1} \frac{dx}{x}$. *In this case,* $x = 0$ *is the singular point of the function* $f(x) = \frac{1}{x}$. *So, this integral has not any finite value. But, in floating point arithmetic, the finite numerical results are obtained. The results in table 2.5 have obtained by using Trapezoidal rule (Th).*

In stochastic arithmetic, the result Th is an informatical zero. Therefore, this result is not reliable, (Table 2.5).

Trapezoidal rule				
floating-point arithmetic		stochastic arithmetic		
n	Th	n	Th	c
5	-3.443824e-07	5	9.457270e-07	-0.8518668
10	1.342177e07	10	-532609.800000	-1.043534
20	6710886.00000	20	-266305.00000	-1.043534
200	-447392.800000	200	-1.041667e-02	-0.7459888
1000	42107.790000	1000	-4557.025000	-1.337839

Table 2.5

Example 2.7.6. *In this example, the value of* $\int_0^{0.09} \frac{dx}{\sqrt{x}}$ *is calculated. The exact value of this integral is* 0.6. *In table (2.6), the approximation of this integral is shown by using the Midpoint rule (*Md*).*

One can validate these results by using the stochastic arithmetic. The table 2.6 shows the approximate value of this integral in the stochastic arithmetic. The value c shows that, how many digits of the results in floating point arithmetic are reliable.

Mid-point rule				
floating-point arithmetic		stochastic arithmetic		
n	Mh	n	Mh	c
10	5.426768e-01	10	5.426760e-01	5.351179
20	5.594378e-01	20	5.594367e-01	5.215071
100	5.818536e-01	100	5.818509e-01	4.887663
200	5.871685e-01	200	5.871648e-01	4.734209
500	5.918841e-01	500	5.918784e-01	4.544301
1000	5.942608e-01	1000	5.942528e-01	4.396454
2000	5.959424e-01	2000	5.959309e-01	4.243607

Table 2.6

Example 2.7.7. *In this example, the numerical solution of the improper integral* $\int_0^\infty e^{-x^2}dx$ *is considered. The exact solution is* $\frac{\sqrt{\pi}}{2} = 0.886226925$*,[13]. The results have obtained by using Trapezoidal* (Th) *and Simpson* (Sh) *rules in the stochastic arithmetic. At first,* $\int_0^m e^{-x^2}dx$ *is solved with* $m = 1$*. Then,* m *increases one by one and this integral should be evaluated again. This calculation must be continued until an informatical zero is detected in the difference between two sequential results. The values* m *in table 2.7 are the final values. Values of* c *show the accuracy of the results for each* n*. We observe that, the results are satisfactory solutions. In the Trapezoidal and Simpson rules, when* $n = 2000$ *and* $n = 3000$ *respectively, the value of the function is an informatical zero in some points. This means* n *should not be very large.*

Trapezoidal (Th) rule				Simpson (Sh) rule			
n	Th	c	m	n	Sh	c	m
20	.8862269	6.9328	8	20	.8862270	6.6942	5
50	.8862267	6.9328	5	50	.8862268	6.5437	6
100	.8862270	6.9327	6	100	.8862270	6.5437	5
500	.8862269	6.9328	8	500	.8862270	6.6942	5
1000	.8862265	6.7823	10	1000	.8862271	6.7823	6
2000	.8862264	6.5833	14	2000	.8862261	6.5833	6
				3000	.8862274	6.6942	11

Table 2.7

Chapter 3

The Valid Implementation of Numerical Integration Methods

3.1 Introduction

In this chapter, the accuracy of the definite integrals is explained. At first, by using the Closed Newton-Cotes integration rules, the definite integral $I = \int_a^b f(x)dx$ is solved numerically in the stochastic arithmetic, by using the perturbation method. In this case, the optimal number of the points in these methods is evaluated. This idea is developed for the Gauss-Chebyshev integration method. Then, this arithmetic is used in order to find the optimal value m such that; $I = \int_a^\infty f(x)dx = \int_a^m f(x)dx + O(\frac{1}{2^m})$.

3.2 The Numerical Accuracy of Definite Integrals

In this section, a theorem is explained which is used in order to solve the definite integral $I = \int_a^b f(x)dx$ numerically. It has been proved in [9], one can use the stochastic arithmetic in order to estimate I by using the Trapezoidal and Simpson rules. This idea is developed generally for the Closed Newton-Cotes integration rules. Then,

the stochastic arithmetic is used to validate the results and implement the numerical examples by using the perturbation method.

If these methods are used to estimate $I = \int_a^b f(x)dx$, if it exists, then, one can find the optimal number of the points, which minimizes the global error. For this purpose, the stochastic arithmetic and the CESTAC method can be used to guarantee the number of the exact significant digits and to find the accuracy of these methods.

At first, the error of the Closed Newton-Cotes integration Rules is recalled.

Theorem 3.2.1. *Let $I = \int_{x_0}^{x_p} f(x)dx$, $p \geq 1$, be the exact value and $I_p = \sum_{j=0}^{p} a_j f(x_j)$ be the approximation of I by using the Closed Newton-Cotes integration Rule with $p+1$ points, where, $x_j = x_0 + jh$, $j = 0, 1, ..., p$ and $h = \frac{x_p - x_0}{p}$. Then, there exists a point like ξ in $]x_0, x_p[$ such that [3],*

$$I = I_p + \frac{h^{p+3} f^{(p+2)}(\xi)}{(p+2)!} \int_0^p t^2(t-1)...(t-p)dt \quad f \in \mathcal{C}^{p+2}[x_0, x_p] \ , \quad p \text{ is even}, \tag{3.2.1}$$

$$I = I_p + \frac{h^{p+2} f^{(p+1)}(\xi)}{(p+1)!} \int_0^p t(t-1)...(t-p)dt \quad f \in \mathcal{C}^{p+1}[x_0, x_p] \ , \quad p \text{ is odd}. \tag{3.2.2}$$

The following proposition is concluded from the theorem.

Proposition 3.2.2. *Let $I = \int_a^b f(x)dx$, and $I_N^p = \sum_{j=0}^{N} a_j f(x_j)$ be the approximation of I by using the Composite Closed Newton-Cotes integration Rule with $p+1$ points, where, $x_j = a + jh$, $j = 0, 1, ..., N$ and $h = \frac{b-a}{N}$, $N = pl$, $l \geq 1$. Then, there exists a point like ξ in $]a, b[$ such that,*

$$I = I_N^p + \frac{(b-a)}{p} \frac{h^{p+2} f^{(p+2)}(\xi)}{(p+2)!} \int_0^p t^2(t-1)...(t-p)dt \quad f \in \mathcal{C}^{p+2}[a, b] \ , \quad p \text{ is even}, \tag{3.2.3}$$

$$I = I_N^p + \frac{(b-a)}{p} \frac{h^{p+1} f^{(p+1)}(\xi)}{(p+1)!} \int_0^p t(t-1)...(t-p)dt \quad f \in \mathcal{C}^{p+1}[a, b] \ , \quad p \text{ is odd}. \tag{3.2.4}$$

Proof. Let p be even and $f \in \mathcal{C}^{p+2}[a,b]$, according to the theorem 3.2.1,

$$I = \sum_{i=1}^{l} \int_{x_{pi-p}}^{x_{pi}} f(x)dx = I_N^p + \frac{h^{p+3}}{(p+2)!} \int_0^p t^2(t-1)...(t-p)dt \times \sum_{i=1}^{l} f^{(p+2)}(\xi_i),$$

where, $x_{pi-p} < \xi_i < x_{pi}$. Furthermore,

$$\min f^{(p+2)}(x) \leq f^{(p+2)}(\xi_i) \leq \max f^{(p+2)}(x),$$

where, $a \leq x \leq b$. Thus,

$$\min f^{(p+2)}(x) \leq \frac{1}{l} \sum_{i=1}^{l} f^{(p+2)}(\xi_i) \leq \max f^{(p+2)}(x).$$

According to the Mean-value theorem, there exists a point like ξ in $[a,b]$ such that $\sum_{i=1}^{l} f^{(p+2)}(\xi_i) = l f^{(p+2)}(\xi)$. Since $lh = \frac{b-a}{p}$, hence,

$$I = I_N^p + \frac{b-a}{p} \frac{h^{p+2} f^{(p+2)}(\xi)}{(p+2)!} \int_0^p t^2(t-1)...(t-p)dt.$$

The similar proof is presented if p is odd. □

By using this proposition, the following lemma is concluded, which is a development up to order $m+3$ due to the Closed Newton-Cotes methods with $p+1$ points ($m = p$ or $p+1$).

Lemma 3.2.3. *Let I_N^p be the approximate value of I by using the Composite Closed Newton-Cotes integration Rule with $p+1$ points. If $f \in \mathcal{C}^{m+3}[a,b]$, $m \geq 1$ and $h = \frac{b-a}{N}$, $N = pl$, $l \geq 1$ then,*

$$I_N^p - I = K_p h^{m+1} [f^{(m)}(b) - f^{(m)}(a)] + O(h^{m+3}), \tag{3.2.5}$$

where, K_p is a constant which depends only on p, $f^{(m)}(b) \neq f^{(m)}(a)$, $m = p+1$ if p is even and $m = p$, if p is odd.

Proof. Let p be even, then $m = p + 1$. According to (3.2.3),

$$I_N^p - I = -A_p \frac{(b-a)}{p} \frac{h^{p+2} f^{(p+2)}(\xi)}{(p+2)!}, \tag{3.2.6}$$

where, $A_p = \int_0^p t^2(t-1)...(t-p)dt$. Let,

$$I_N^p - I = \alpha h^{p+2} f^{(p+1)}(b) + \beta h^{p+2} f^{(p+1)}(a) + O(h^{p+4}). \tag{3.2.7}$$

The values α and β are evaluated so that this formula be exact for the polynomials of degree of at most $p + 3$. If $f(x) = x^{p+2}$ and $f(x) = x^{p+1}$, from (3.2.6) and (3.2.7),

$$-A_p \frac{b-a}{p} h^{p+2} \frac{(p+2)!}{(p+2)!} = \alpha h^{p+2}.b(p+2)! + \beta h^{p+2}.a(p+2)!,$$

$$0 = \alpha h^{p+2}(p+1)! + \beta h^{p+2}(p+1)!.$$

Therefore, the following system is obtained:

$$\begin{cases} \alpha b + \beta a = -A_p \frac{b-a}{p(p+2)!} \\ \alpha + \beta = 0 \end{cases}$$

After solving the system,

$$\alpha = -A_p \frac{1}{p(p+2)!} \quad , \quad \beta = A_p \frac{1}{p(p+2)!} \ . \tag{3.2.8}$$

If $K_p = -A_p \frac{1}{p(p+2)!}$ is considered then,

$$I_N^p - I = K_p h^{p+2}[f^{(p+1)}(b) - f^{(p+1)}(a)] + O(h^{p+4}).$$

One can observe, this formula is also exact if $f(x) = x^{p+3}$. For this purpose, it is sufficient to prove,

$$I_p - \int_{-h}^{h} f(x)dx + A_p.\frac{1}{p(p+2)!} h^{p+2}[f^{(p+1)}(-h) - f^{(p+1)}(h)] = 0.$$

Since, $x_i = -h + i.\frac{2h}{p}$, $a_{p-i} = a_i$, $i = 0, 1, ..., p$ and $a_{\frac{p}{2}} = 0$, hence, $I_p = \sum_{i=0}^{p} a_i x_i^{p+3} = 0$. Besides, $\int_{-h}^{h} x^{p+3} dx = 0$ and $f^{(p+1)}(x) = \frac{(p+3)!}{2} x^2$, therefore, $[f^{(p+1)}(-h) - f^{(p+1)}(h)] = 0$. so, the relation can be concluded for the interval $[x_0, x_N]$.

If p is odd then, the similar proof is presented. In this case,

$x_i = \frac{-ph}{2} + ih$, $a_{p-i} = a_i$, $i = 0, 1, ..., p$ and $\alpha = -\frac{1}{p(p+1)!} A'_p = -\beta$ where,

$A'_p = \int_0^p t(t-1)...(t-p)dt$. The interval $[-\frac{ph}{2}, \frac{ph}{2}]$ for the error term is considered. Thus, $K_p = \frac{-1}{p(p+1)!} A'_p$. □

The following table gives the value K_p for $1 \leq p \leq 4$.

p	m	K_p	I_N^p
1	1	$\frac{1}{12}$	Trapezoidal rule
2	3	$\frac{1}{180}$	Simpson rule
3	3	$\frac{1}{80}$	$\frac{3}{8}$ rule
4	5	$\frac{2}{945}$	Boolean rule

According to (3.2.5), since, $h = \frac{b-a}{N}$ hence,

$$I_N^p - I = K_{p,m} \frac{(b-a)^{m+1}}{N^{m+1}} + O(\frac{1}{N^{m+3}}), \tag{3.2.9}$$

where, $K_{p,m} = K_p.(f^{(m)}(b) - f^{(m)}(a))$. The value $K_{p,m}$ is small in comparison with N, as $N \to \infty$.

Theorem 3.2.4. *Let I_N^p be the approximate value of $I = \int_a^b f(x)dx$ computed using the Closed Newton-Cotes integration rule with $p+1$ points over [a,b] and step size $h = \frac{b-a}{N}$. If $f \in \mathcal{C}^{m+3}[a, b]$, $m \geq 1$, then,*

$$C_{I_N^p, I_{2N}^p} \simeq C_{I, I_N^p} + \log_{10} \frac{2^{m+1}}{2^{m+1} - 1} + O(\frac{1}{N^{m+1}}), \tag{3.2.10}$$

where, $m = p + 1$ if p is even and $m = p$, if p is odd.

Proof. According to (3.2.9),

$$I_N^p - I_{2N}^p = I_N^p - I - (I_{2N}^p - I) = K_{p,m}\frac{(b-a)^{m+1}}{N^{m+1}} - K_{p,m}\frac{(b-a)^{m+1}}{(2N)^{m+1}} + O(\frac{1}{N^{m+3}})$$
$$= (1 - \frac{1}{2^{m+1}})(K_{p,m}\frac{(b-a)^{m+1}}{N^{m+1}}) + O(\frac{1}{N^{m+3}}),$$

hence,

$$I_N^p - I_{2N}^p = \frac{2^{m+1}-1}{2^{m+1}}(I_N^p - I) + O(\frac{1}{N^{m+3}}). \tag{3.2.11}$$

Furthermore,

$$\frac{I_N^p + I}{2(I_N^p - I)} = \frac{I_N^p}{I_N^p - I} - \frac{1}{2} = \frac{I_N^p}{K_{p,m}\frac{(b-a)^{m+1}}{N^{m+1}}(1 + O(\frac{1}{N^2}))} + O(1)$$
$$= \frac{I_N^p N^{m+1}}{K'_m}(1 - O(\frac{1}{N^2})) + O(1) \simeq \frac{I_N^p N^{m+1}}{k'_m} + O(1),$$

where, $K'_m = K_{p,m}(b-a)^{m+1}$. Also, from (3.2.11),

$$\frac{I_N^p + I_{2N}^p}{2(I_N^p - I_{2N}^p)} = \frac{I_N^p}{I_N^p - I_{2N}^p} - \frac{1}{2} \simeq \frac{I_N^p N^{m+1}}{K''_m} + O(1),$$

where, $K''_m = \frac{2^{m+1}-1}{2^{m+1}}K'_m$. Therefore, according to definition 2.1.1,

$$C_{I,I_N^p} \simeq \log_{10} \mid \frac{I_N^p N^{m+1}}{K'_m} + O(1) \mid = \log_{10} \mid \frac{I_N^p N^{m+1}}{K'_m} \mid$$
$$+ \log_{10}(1 + O(\frac{1}{N^{m+1}})) = \log_{10} \mid \frac{I_N^p N^{m+1}}{K'_m} \mid + O(\frac{1}{N^{m+1}}),$$

and,

$$C_{I_N^p,I_{2N}^p} \simeq \log_{10} \mid \frac{I_N^p N^{m+1}}{K''_m} + O(1) \mid = \log_{10} \mid \frac{I_N^p N^{m+1}}{K''_m} \mid$$
$$+ \log_{10}(1 + O(\frac{1}{N^{m+1}})) = \log_{10} \mid \frac{I_N^p N^{m+1}}{K''_m} \mid + O(\frac{1}{N^{m+1}}).$$

Hence,

$$C_{I_N^p,I_{2N}^p} \simeq C_{I,I_N^p} + \log_{10}\frac{2^{m+1}}{2^{m+1}-1} + O(\frac{1}{N^{m+1}}).$$

□

The relation (3.2.10) shows that:

1. If we are able to compute the number of significant digits in common between I_N^p and I_{2N}^p, we are sure that, these digits are also in common with the exact value of integral. The successive values $I_N^p, N \geq 1$ are computed and at each iteration, the number of significant digits of I_N^p , I_{2N}^p and $I_N^p - I_{2N}^p$ can be estimated. The computations of the sequence I_N^p's are stopped when for an index like N_{opt}, the number of common significant digits in the difference between $I_{N_{opt}}^p$ and $I_{2N_{opt}}^p$ become zero. In this case, one can say, before N_{opt}'th iteration, $I_N^p - I_{2N}^p$ has exact significant digits. But, the computations after N_{opt}'th iteration are useless. In other words, the number of iteration in N_{opt} has been optimized. Also, according to the theorem 3.2.4, the significant digits of the last approximation $I_{N_{opt}}^p$ are in common with the mathematical value of the integral I. Therefore, $I_{N_{opt}}^p$ is an approximation of I with optimal step size $h_{opt} = \frac{b-a}{N_{opt}}$.

2. Since, $0 < \log_{10} \frac{2^{m+1}}{2^{m+1}-1} < 1$ for $m \geq 1$, if this term is neglected then, the significant bits in common between I_N^p and I_{2N}^p are also in common with I up to less than 1 bit.

3. The previous remarks are valid if N is large enough. In this case, $O(\frac{1}{N^{m+1}}) < 1$.

According to (3.2.10), the following results are obtained:

- If $N = 2^n,\ n \geq 0$ and $p = 1$ then,

$$C_{I_N^1, I_{2N}^1} = C_{I, I_N^1} + \log_{10} \frac{4}{3} + O(\frac{1}{4^n}), \tag{3.2.12}$$

where, $f \in \mathcal{C}^4[a, b]$. This shows the numerical accuracy of the Trapezoidal rule.

- If $N = 2^n,\ n \geq 1$ and $p = 2$ then,

$$C_{I_N^2, I_{2N}^2} \simeq C_{I, I_N^2} + \log_{10} \frac{16}{15} + O(\frac{1}{16^n}), \tag{3.2.13}$$

  where, $f \in \mathcal{C}^6[a, b]$. This shows the numerical accuracy of the Simpson rule.

- If $N = 2^n,\ n \geq 2$ and $p = 4$ then,

$$C_{I_N^4, I_{2N}^4} \simeq C_{I, I_N^4} + \log_{10} \frac{64}{63} + O(\frac{1}{64^n}), \tag{3.2.14}$$

  where, $f \in \mathcal{C}^8[a, b]$. This shows the numerical accuracy of the Boolean rule. One can observe that the accuracy is more than two other rules.

3.3 Numerical Accuracy of The Gauss-Chebyshev Rule

In this section, the numerical accuracy of the Gauss-Chebyshev integration method is explained. Then, one can use the CADNA library to find the optimal number of points in this method.

Let f be a function which its Chebyshev expansion is rapidly convergent. Let $I = \int_{-1}^{1} \frac{f(x)}{\sqrt{1-x^2}} dx$ and I_N be an approximation of I using closed or opened Gauss-Chebyshev integration method. The aim is to find N_{opt} such that, $I_N - I_{2N} = @0$.

It has been proved in [15], the errors of I_N and I_{2N} are:

$$E_N f = I_N - I = \pi a_{2N} - \pi a_{4N} + \pi a_{6N} - ...,$$

$$E_{2N} f = I_{2N} - I = \pi a_{4N} - \pi a_{8N} + \pi a_{12N} - ...,$$

where, $a_i,\ i = kN,\ k = 2, 4, 6, ...,$ are the coefficients of Chebyshev expansion of f. Since, f has a rapidly convergent Chebyshev expansion, the error can be found by

$$E_N f \approx \pi a_{2N} \quad , \quad E_{2N} f \approx \pi a_{4N}. \tag{3.3.1}$$

It has been explained in [15], if $a_i,\ i = 0, 1, ...,$ are the coefficients of Chebyshev expansion of f, the real numbers $r > 1$ and C_f exist so that:

$$\mid a_i \mid \leq C_f \hat{i}^{-r}, \tag{3.3.2}$$

where, $\hat{i} = Max\{i, 1\},\ i \geq 0$. Therefore, $a_i = O(i^{-r})$. Hence,

$$E_N f = O((2N)^{-r}) \quad , \quad E_{2N} f = O((4N)^{-r}). \tag{3.3.3}$$

The following theorem shows the numerical accuracy of the Gauss-Chebyshev integration method.

Theorem 3.3.1. *Let f be a function which its Chebyshev expansion be rapidly convergent. Let $I = \int_{-1}^{1} \frac{f(x)}{\sqrt{1-x^2}} dx$ and I_N be an approximation of I by using closed or opened Gauss-Chebyshev integration method then,*

$$C_{I_N, I_{2N}} = C_{I_N, I} - log_{10} \mid 1 - \frac{a_{4N}}{a_{2N}} \mid + O(\frac{1}{(2N)^r}), \tag{3.3.4}$$

where, a_{2N}, a_{4N} are the first coefficients of the Chebyshev expansion of f and $r > 1$ is a real value.

Proof. According to definition 2.1.1,

$$C_{I_N, I_{2N}} = log_{10} \mid \frac{I_N}{I_N - I_{2N}} - \frac{1}{2} \mid = log_{10} \mid \frac{I_N}{I_N - I_{2N}} \mid + log_{10} \mid 1 - \frac{1}{2I_N}(I_N - I_{2N}) \mid$$

$$= \log_{10} \mid \frac{I_N}{I_N - I_{2N}} \mid + O(I_N - I_{2N}).$$

Since, $I_N - I_{2N} = I_N - I - (I_{2N} - I) = E_N f - E_{2N} f$,

thus,

$$O(I_N - I_{2N}) = O(\frac{1}{(2N)^r}) + O(\frac{1}{(4N)^r}) = O(\frac{1}{(2N)^r}).$$

Therefore,

$$C_{I_N,I_{2N}} = log_{10} \mid \frac{I_N}{I_N - I_{2N}} \mid + O(\frac{1}{(2N)^r}). \tag{3.3.5}$$

Furthermore,

$$C_{I_N,I} = log_{10} \mid \frac{I_N}{I_N - I} - \frac{1}{2} \mid = log_{10} \mid \frac{I_N}{I_N - I} \mid + O(I_N - I) = log_{10} \mid \frac{I_N}{I_N - I} \mid + O(\frac{1}{(2N)^r}).$$

According to (3.3.5),

$$C_{I_N,I_{2N}} = log_{10} \mid \frac{I_N}{I_N - I - (I_{2N} - I)} \mid + O(\frac{1}{(2N)^r}) = log_{10} \mid \frac{I_N}{(I_N - I)(1 - \frac{I_{2N} - I}{I_N - I})} \mid$$

$$+ O(\frac{1}{(2N)^r}) = log_{10} \mid \frac{I_N}{I_N - I} \mid - log_{10} \mid 1 - \frac{I_{2N} - I}{I_N - I} \mid + O(\frac{1}{(2N)^r}).$$

Consequently,

$$C_{I_N,I_{2N}} = C_{I_N,I} - log_{10} \mid 1 - \frac{I_{2N} - I}{I_N - I} \mid + O(\frac{1}{(2N)^r}).$$

Since, $lim_{N \to \infty} \frac{I_{2N} - I}{I_N - I} = lim_{N \to \infty} \frac{a_{4N}}{a_{2N}}$, and $0 < \frac{a_{4N}}{a_{2N}} < 1$, the second term of the relation is small, as N tends to ∞.

Consequently,

$$C_{I_N,I_{2N}} = C_{I_N,I} - log_{10} \mid 1 - \frac{a_{4N}}{a_{2N}} \mid + O(\frac{1}{(2N)^r}).$$

□

The relation (3.3.4) shows that, if the Gauss-Chebyshev integration method is used in order to estimate I then, for N large enough, the number of common significant

digits between I_N and I_{2N} are almost equal to the number of common significant digits between I and I_N. Therefore, if the CESTAC method is used then, the computations of the sequence I_N's are stopped when for an index like N_{opt}, $I_{N_{opt}} - I_{2N_{opt}} = @0$. In this case, $I_{N_{opt}}$ is an approximation of I.

3.4 Numerical Solution Of Improper Integrals

In this section, two theorems are explained, which are used in order to solve the improper integral $I = \int_a^\infty f(x)dx$. It has been proved in [9] that, one can use the Trapezoidal and Simpson rules to evaluate the definite integral $I_m = \int_a^m f(x)dx,\ m \geq 1$, numerically by using the CESTAC method. These theorems are developed for the improper integrals. Then, the CESTAC method and stochastic arithmetic are used to validate the results and implement the numerical examples. By using this method, one can find the optimal number m such that $I \simeq I_m$.

Let $\mid I - I_m \mid \leq M\frac{1}{2^m}$, where, M is a positive constant. One can use Trapezoidal or Simpson rule to approximate I_m.

According to (3.2.12) and (3.2.13), if $I_{m,n}$ is the approximation of I_m with step size $h = \frac{m-a}{2^n}$ by using Trapezoidal rule then,

$$C_{I_{m,n},I_{m,n+1}} = C_{I_{m,n},I_m} + \log_{10}\frac{4}{3} + O(\frac{1}{4^n}), \tag{3.4.1}$$

and with Simpson rule,

$$C_{I_{m,n},I_{m,n+1}} \simeq C_{I_{m,n},I_m} + \log_{10}\frac{16}{15} + O(\frac{1}{16^n}). \tag{3.4.2}$$

The relations (3.4.1) and (3.4.2) can be developed for Improper Integral I, according to the following theorems. In these theorems, $m \geq 1$, is an integer number such that,

$$I = I_m + O(\frac{1}{2^m}). \tag{3.4.3}$$

Let f be a real function and $f \in \mathcal{C}^k[a, m], k \geq 4$. In the following theorem the accuracy of the Trapezoidal rule is considered.

Theorem 3.4.1. *Let $I_{m,n}$ be the approximate value of $I_m = \int_a^m f(x)dx$ computed using the Trapezoidal rule with step size $h = \frac{m-a}{2^n}$. If $I = \int_a^\infty f(x)dx$ and $m > 2n$ then,*

$$C_{I_{m,n},I_{m,n+1}} \simeq C_{I,I_{m,n}} + \log_{10}\frac{4}{3} + O(\frac{1}{2^{m-2n}}) + O(\frac{1}{4^n}). \tag{3.4.4}$$

Proof. According to (3.4.3), $I - I_{m,n} = I_m - I_{m,n} + O(\frac{1}{2^m})$, also from (3.2.5),

$$I_{m,n} - I_m = \frac{h^2}{12}[f'(m) - f'(a)] + O(h^4).$$

Because, $h = \frac{m-a}{2^n}$,

$$I_{m,n} - I_m = \frac{1}{4^n}k_m + O(\frac{1}{16^n}),$$

where $k_m = \frac{(m-a)^2}{12}[f'(m) - f'(a)]$.

Hence,

$$\frac{I_{m,n}}{I_{m,n} - I} = \frac{I_{m,n}}{I_{m,n} - I_m + O(\frac{1}{2^m})} = \frac{I_{m,n}}{I_{m,n} - I_m}(\frac{1}{1 + \frac{1}{\frac{1}{4^n}k_m + O(\frac{1}{16^n})}O(\frac{1}{2^m})})$$

$$= \frac{I_{m,n}}{I_{m,n} - I_m}(\frac{1}{1 + O(\frac{1}{2^{m-2n}})}) = \frac{I_{m,n}}{I_{m,n} - I_m}(1 - O(\frac{1}{2^{m-2n}})),$$

and consequently,

$$C_{I,I_{m,n}} = \log_{10} \mid \frac{I_{m,n}}{I_{m,n} - I} - \frac{1}{2} \mid \simeq \log_{10} \mid \frac{I_{m,n}}{I_{m,n} - I} \mid = \log_{10} \mid \frac{I_{m,n}}{I_{m,n} - I_m} \mid$$

$$+log_{10} \mid 1 - O(\frac{1}{2^{m-2n}}) \mid = \log_{10} \mid \frac{I_{m,n}}{I_{m,n} - I_m} \mid -O(\frac{1}{2^{m-2n}}) = C_{I_{m,n},I_m} - O(\frac{1}{2^{m-2n}}).$$

According to (3.4.1),

$$C_{I_{m,n},I_{m,n+1}} \simeq C_{I,I_{m,n}} + \log_{10} \frac{4}{3} + O(\frac{1}{2^{m-2n}}) + O(\frac{1}{4^n}).$$

□

The relation (3.4.4) shows that, an optimal m where, $m > 2n$ exists such that, $C_{I_{m,n},I_{m,n+1}} \simeq C_{I,I_{m,n}}$.

Therefore, if m is large enough, then, the number of common significant digits between $I_{m,n}$ and $I_{m,n+1}$ are almost equal to the number of the common significant digits between the exact value I and the approximate value $I_{m,n}$ in company with the term $\log_{10} \frac{4}{3} + O(\frac{1}{4^n})$ which is negligible when n increases and the term $O(\frac{1}{2^{m-2n}})$ which is small because, $m > 2n$. Hence, one can find an optimal m such that, the number of the significant digits in the error term $\mid I_{m,n} - I_{m,n+1} \mid$ become zero when n increases from 1 to $\frac{m}{2} - 1$. In this case, $I_{m,n}$ is an approximate value for I.

Let f be a real function and $f \in \mathcal{C}^k[a, m]$ for $k \geq 6$. In the following theorem, the accuracy of the Simpson rule is considered.

Theorem 3.4.2. *Let $I_{m,n}$ be the approximate value of $I_m = \int_a^m f(x)dx$ computed using the Simpson rule with step size $h = \frac{m-a}{2^n}$. If $I = \int_a^\infty f(x)dx$ and $m > 4n$ then,*

$$C_{I_{m,n},I_{m,n+1}} \simeq C_{I,I_{m,n}} + \log_{10} \frac{16}{15} + O(\frac{1}{2^{m-4n}}) + O(\frac{1}{16^n}). \tag{3.4.5}$$

Proof. According to (3.2.5), $I_{m,n} - I_m = \frac{h^4}{180}[f^{(3)}(m) - f^{(3)}(a)] + O(h^6)$. Therefore,

$$I_{m,n} - I_m = \frac{1}{16^n} k'_m + O(\frac{1}{64^n}),$$

where, $k'_m = \frac{(m-a)^4}{180}[f^{(3)}(m) - f^{(3)}(a)]$.

Hence,

$$\frac{I_{m,n}}{I_{m,n}-I}=\frac{I_{m,n}}{I_{m,n}-I_m+O(\frac{1}{2^m})}=\frac{I_{m,n}}{I_{m,n}-I_m}(\frac{1}{1+\frac{1}{\frac{1}{16^n}k'_m+O(\frac{1}{64^n})}O(\frac{1}{2^m})})$$

$$=\frac{I_{m,n}}{I_{m,n}-I_m}(\frac{1}{1+O(\frac{1}{2^{m-4n}})})=\frac{I_{m,n}}{I_{m,n}-I_m}(1-O(\frac{1}{2^{m-4n}})),$$

consequently,

$$C_{I,I_{m,n}}=\log_{10}\mid\frac{I_{m,n}}{I_{m,n}-I}-\frac{1}{2}\mid\simeq\log_{10}\mid\frac{I_{m,n}}{I_{m,n}-I_m}\mid-O(\frac{1}{2^{m-4n}})=C_{I_{m,n},I_m}+O(\frac{1}{2^{m-4n}}).$$

According to (3.4.2),

$$C_{I_{m,n},I_{m,n+1}}\simeq C_{I,I_{m,n}}+\log_{10}\frac{16}{15}+O(\frac{1}{2^{m-4n}})+O(\frac{1}{16^n}).$$

□

The relation (3.4.5) shows that, an optimal m where $m>4n$ exists such that $I_{m,n}\simeq I$. Similar to the previous theorem there is an integer number $m>1$ which, the significant digits in the error term $\mid I_{m,n}-I_{m,n+1}\mid$ become zero when n increases from 1 to $\frac{m}{4}-1$.

3.5 Numerical Examples

In this section, some numerical examples are computed according to the discussions in this chapter. The programs have been provided with Fortran 90 programming language. The values C in the tables show the estimation of the number of significant digit in the error.

In the first two examples, two definite integrals are evaluated. The values have obtained by using the Trapezoidal, Simpson and Boolean rules in the stochastic arithmetic with step size $h=\frac{b-a}{2^n},\ n\geq 0$. When the informatical zero is detected in the

difference between two sequential results, the computations are terminated. The last values n in the tables are the optimal numbers of the points. One can observe the accuracy of the Boolean rule is better and the optimal n is less than two other rules. The results are shown in single precision.

Example 3.5.1. *In this example, the numerical solution of the integral* $I = \int_0^1 e^{-x^2} dx$ *is considered. The results are obtained by using the Simpson, Trapezoidal and Boolean rules in stochastic arithmetic. The last values of* n *in tables 3.1, 3.2 and 3.3 are the optimal values. One concludes that, the approximate value for* I *is* $7.46824E-01$*. Because, the accuracy of the Boolean rule is better than two other rules, the optimal value of* n *in the Boolean rule, is less than or equal to the others.*

n	I_n	$\mid I_n - I_{n+1} \mid$	$C_{\mid I_n - I_{n+1} \mid}$
1	7.471804E-01	3.250639E-04	3.6
2	7.468553E-01	2.930562E-05	2.5
3	7.468260E-01	1.827876E-06	1.0
4	7.468242E-01	@0	-3.3

Table 3.1 (Simpson rule)

n	I_n	$\mid I_n - I_{n+1} \mid$	$C_{\mid I_n - I_{n+1} \mid}$
0	6.839351E-01	4.742934E-02	4.1
1	7.313644E-01	1.161353E-02	4.1
2	7.429780E-01	2.881448E-03	4.0
3	7.458594E-01	7.189711E-04	3.9
4	7.465784E-01	1.796285E-04	2.9
5	7.467580E-01	4.498164E-05	2.4
6	7.468030E-01	1.118581E-05	1.3
7	7.468142E-01	2.821286E-06	0.97
8	7.468171E-01	1.053015E-06	0.18
9	7.468181E-01	4.768372E-07	0.04
10	7.468176E-01	@0	-1.4

Table 3.2 (Trapezoidal rule)

n	I_n	$\mid I_n - I_{n+1} \mid$	$C_{\mid I_n - I_{n+1} \mid}$
2	7.468336E-01	9.497006E-06	2.1
3	7.468241E-01	@0	-3.3

Table 3.3 (Boolean rule)

Example 3.5.2. *In this example, the integral* $I = \int_1^3 \frac{(sin(x))^2}{x} dx$ *is evaluated. This integral is computed like previous example. The results are determined in tables 3.4, 3.5 and 3.6. The last values of* n *in these tables are the optimal numbers of the points.*

n	I_n	$\mid I_n - I_{n+1} \mid$	$C_{\mid I_n - I_{n+1} \mid}$
1	7.894511E-01	5.202631E-03	4.5
2	7.946537E-01	1.616081E-04	3.3
3	7.948154E-01	8.523464E-06	1.5
4	7.948239E-01	@0	-2.3

Table 3.4 (Simpson rule)

n	I_n	$\mid I_n - I_{n+1} \mid$	$C_{\mid I_n - I_{n+1} \mid}$
0	7.147076E-01	5.605398E-02	4.1
1	7.707615E-01	1.791537E-02	4.5
2	7.886769E-01	4.600028E-03	4.1
3	7.932768E-01	1.156370E-03	3.7
4	7.944332E-01	2.894600E-04	3.0
5	7.947227E-01	7.237991E-05	3.0
6	7.947950E-01	1.815955E-05	1.5
7	7.948132E-01	4.669031E-06	0.8
8	7.948179E-01	@0	-0.005

Table 3.5 (Trapezoidal rule)

n	I_n	$\mid I_n - I_{n+1} \mid$	$C_{\mid I_n - I_{n+1} \mid}$
2	7.950013E-01	1.745423E-04	3.0
3	7.948267E-01	1.629194E-06	1.3
4	7.948251E-01	@0	-6.3

Table 3.6 (Boolean rule)

In the following example, the numerical solution of a definite integral which has been computed by CADNA library in [9], is considered. The following results are obtained by using the perturbation method in double precision with step size $h = \frac{b-a}{2^n},\ n \geq 0$. One can conclude that, the Simpson rule is faster than the Trapezoidal rule. Also the Boolean rule is faster than the Simpson rule.

Example 3.5.3. *In this example, the numerical solution of the integral* $I = \int_{-1}^{1} 20cos(20x)(2.7x^2 - 3.3x + 1.2)dx$ *is considered. The exact value is* $I = 0.7316687747285081E + 001$. *The tables 3.7, 3.8 and 3.9 show the results by using the Trapezoidal, Simpson and Boolean rules.*

n	I_n	$\mid I_n - I \mid$	$C_{\mid I_n - I_{n+1} \mid}$
1	55.830400821443420	48.513713074158340	13.4
2	-3.549981929644979	10.866669676930060	13.9
3	-18.546379932143980	25.863067679429060	13.4
4	3.322208564406735	3.994479182878346	13.5
5	6.395763316297033	9.209244309880482E-001	13.3
6	7.090693727981386	2.259940193036947E-001	13.2
7	7.260445687519884	5.624205976519647E-002	12.6
8	7.302643133738241	1.404461354684017E-002	12.7
9	7.313177585776868	3.510161508213372E-003	11.2
10	7.315810268869929	8.774784151519555E-004	10.4
11	7.316468381553480	2.193657316009734E-004	9.2
12	7.316632906094228	5.484119085341632E-005	8.9
13	7.316674037002549	1.371028253149831E-005	8.7
14	7.316684319715416	3.427569664538055E-006	7.5
15	7.316686890392780	8.568923011154084E-007	7.0
16	7.316687533061931	2.142231494417501E-007	6.7
17	7.316687693729286	5.355579535404331E-008	5.9
18	7.316687733896178	1.338890243118840E-008	5.5
19	7.316687743937936	3.347144783560907E-009	4.6
20	7.316687746448042	8.370388826506314E-010	4.1
21	7.316687747075711	2.093702988759105E-010	4.8
22	7.316687747232461	5.262013047513392E-011	2.9
23	7.316687747271778	1.330313637026848E-011	2.0
24	7.316687747283422	1.659117287999834E-012	1.8
25	7.316687747285317	@0	0.0

Table 3.7 (Trapezoidal rule)

n	I_n	$\mid I_n - I \mid$	$C_{\mid I_n - I_{n+1} \mid}$
1	53.220267214295650	45.903579467010570	13.6
2	-23.343442846674450	30.660130593959530	11.3
3	-23.545175266310300	30.861867013595380	13.5
4	10.611738063256970	3.295050315971889	13.2
5	7.420281566927133	1.035938196420521E-001	12.7
6	7.322337198542837	5.649451257756510E-003	11.9
7	7.317029674032720	3.419267476392207E-004	10.5
8	7.316708949144362	2.120185928156104E-005	9.1
9	7.316689069789740	1.322504658674006E-006	8.1
10	7.316687829900958	8.261587680635785E-008	6.2
11	7.316687752448002	5.162920757584288E-009	5.2
12	7.316687747607818	3.227373923664345E-010	3.4
13	7.316687747305317	2.023625711444765E-011	2.4
14	7.316687747286399	1.318056774834986E-012	1.6
15	7.316687747285207	@0	0.0

Table 3.8 (Simpson rule)

n	I_n	$\mid I_n - I \mid$	$C_{\mid I_n - I_{n+1} \mid}$
2	-28.447690184072450	7.316687747285081	13.9
3	-23.558628360952720	35.764377931357540	12.6
4	12.888865885228130	30.875316108237800	13.4
5	7.207517800505142	5.572178137943047	13.4
6	7.315807573983889	1.091699467799394E-001	12.5
7	7.316675839065375	8.801733011916113E-004	10.6
8	7.316687567485130	1.190821970631362E-005	8.9
9	7.316687744499432	1.797999509278725E-007	7.0
10	7.316687747241705	2.785649044767524E-009	5.3
11	7.316687747284455	4.337596948289502E-011	3.4
12	7.316687747285129	6.261657858885883E-013	1.5
13	7.316687747285133	@0	-1.1

Table 3.9 (Boolean rule)

The optimal number of the points in the Trapezoidal rule is $n_{opt} = 25$ with approximate value $I_{n_{opt}} = 7.316687747285317$ The optimal number of the points in the Simpson rule is $n_{opt} = 15$ with approximate value $I_{n_{opt}} = 7.316687747285207$ The optimal number of the points in the Boolean rule is $n_{opt} = 13$ with approximate value $I_{n_{opt}} = 7.316687747285133$ The computed values have 13 correct significant digits.

Example 3.5.4. *In this example, the integral* $I = \int_{-1}^{1} \frac{x}{1+\sin x} dx = -1.249249093594816$ *is computed. The results are shown in the tables 3.10, 3.11 and 3.12 in double precision.*

n	I_n	$\mid I_n - I \mid$	$C_{\mid I_n - I_{n+1} \mid}$
0	-2.882474695629098	1.633225602034282	13.0
1	-1.752491532644204	5.032424390493873E-001	13.0
2	-1.386642811512912	1.373937179180960E-001	12.9
3	-1.284585889758802	3.533679616398544E-002	12.9
4	-1.258152205685602	8.903112090785204E-003	12.9
5	-1.251479315852714	2.230222257897152E-003	12.5
6	-1.249806929204200	5.578356093836057E-004	11.9
7	-1.249388570036224	1.394764414075489E-004	11.7
8	-1.249283963801965	3.487020714820766E-005	10.9
9	-1.249257811215199	8.717620382459046E-006	9.9
10	-1.249251273004238	2.179409421820822E-006	9.8
11	-1.249249638447476	5.448526598783587E-007	8.8
12	-1.249249229808041	1.362132246995884E-007	8.0
13	-1.249249127648156	3.405333948158784E-008	7.3
14	-1.249249102108183	8.513366678286616E-009	6.4
15	-1.249249095723215	2.128398346457061E-009	5.6
16	-1.249249094126915	5.320988094581480E-010	5.3
17	-1.249249093727931	1.331146304295316E-010	5.1
18	-1.249249093698089	3.327271791420117E-011	4.3
19	-1.249249093603133	8.316902722071973E-012	3.5
20	-1.249249093596952	2.135847054773876E-012	2.9
21	-1.249249093595424	6.075140390748857E-013	2.7
22	-1.249249093594912	9.547918011776346E-014	2.3
23	-1.249249093594801	1.554312234475219e-014	1.1
24	-1.249249093594785	@0	0.0

Table 3.10 (Trapezoidal rule)

n	I_n	$\mid I_n - I \mid$	$C_{\mid I_n - I_{n+1} \mid}$
1	-1.921649797086070	6.724007034912538E-001	12.9
2	-1.375830478315905	1.265813847210890E-001	12.9
3	-1.264693237802482	1.544414420766538E-002	12.9
4	-1.250566915840765	1.317822245948896E-003	12.6
5	-1.249340977661201	9.188406638438451E-005	12.0
6	-1.249255019241749	5.925646932691819E-006	10.6
7	-1.249249466988030	3.733932136817941E-007	9.2
8	-1.249249116980234	2.338541760238400E-008	7.5
9	-1.249249095057214	1.462397092311107E-009	6.7
10	-1.249249093686279	9.146239321466965E-011	5.8
11	-1.249249093600590	5.773159728050814E-012	4.4
12	-1.249249093595230	4.138911435802584E-013	3.1
13	-1.249249093594893	7.682743330406083E-014	2.2
14	-1.249249093594870	5.373479439185758E-014	0.91
15	-1.249249093594876	@0	-0.52

Table 3.11 (Simpson rule)

n	I_n	$\mid I_n - I \mid$	$C_{\mid I_n - I_{n+1} \mid}$
2	-1.339442523731228	1.249249093594816	12.9
3	-1.257284088434921	9.019343013641157E-002	12.8
4	-1.249625161043317	8.034994840104170E-003	12.8
5	-1.249259248449230	3.760674485009385E-004	11.5
6	-1.249249288680454	1.015485441380193E-005	10.0
7	-1.249249096837782	1.950856372001653E-007	8.0
8	-1.249249093646378	3.242965895822181E-009	6.6
9	-1.249249093595677	5.156142179885137E-011	4.7
10	-1.249249093594881	8.606448886894214E-013	2.5
11	-1.249249093594869	6.483702463810914E-014	0.98
12	-1.249249093594871	@0	-0.7

Table 3.12 (Boolean rule)

Example 3.5.5. *In this example, the integral* $I = \int_{-1}^{1} \frac{\sqrt{(1+x)^3}}{\sqrt{1-x^2}} dx = \frac{8\sqrt{2}}{3} \approx 3.771236166$ *[15], is solved using the Open Gauss-Chebyshev method,* $I_N = \frac{\pi}{N} \sum_{i=1}^{N} f(cos\frac{(2i-1)\pi}{2N})$, *and Closed Gauss-Chebyshev method,* $I_N = \frac{\pi}{N} \sum_{i=1}^{''N} f(cos\frac{i\pi}{N})$, *where,* $N = 2^n,\ n \geq 1$. *For the function* $f(x) = \sqrt{(1+x)^3}, E_N f = O((2N)^{-4})$. *The results in tables 3.13 and 3.14, are obtained with the CADNA library.*

n	I_N	$\mid I_N - I_{2N} \mid$	$\mid I_N - I \mid$
1	0.375256199832479E+001	0.375256199832479E+001	0.186741680034537E-001
2	0.377021386720234E+001	0.1765186887754E-001	0.102229912591E-002
3	0.377117422901932E+001	0.960361816977E-003	0.6193730893E-004
4	0.377123232478378E+001	0.5809576446E-004	0.384154447E-005
5	0.377123592668936E+001	0.36019056E-005	0.2396389E-006
6	0.37712361513579E+001	0.2246686E-006	0.149703E-007
7	0.37712361653927E+001	0.14035E-007	0.9355E-009
8	0.37712361662698E+001	0.8771E-009	0.585E-010
9	0.3771236166325E+001	0.55E-010	0.4E-011
10	0.3771236166328E+001	0.3E-011	@.0
11	0.3771236166328E+001	@.0	@.0

Table 3.13 (Opened Gauss-Chebyshev)

n	I_N	$\mid I_N - I_{2N} \mid$	$\mid I_N - I \mid$
1	0.379223779587408E+001	0.379223779587408E+001	0.21001629545826E-001
2	0.377239989709944E+001	0.1983789877464E-001	0.116373077119E-002
3	0.377130688215089E+001	0.109301494855E-002	0.70715822639E-004
4	0.377124055558511E+001	0.66326565786E-004	0.438925685E-005
5	0.377123644018444E+001	0.411540066E-005	0.27385619E-006
6	0.37712361834369E+001	0.2567475E-006	0.171086E-007
7	0.37712361673974E+001	0.160395E-007	0.10692E-008
8	0.37712361663951E+001	0.10024E-008	0.668E-010
9	0.3771236166332E+001	0.626E-010	0.42E-011
10	0.3771236166329E+001	0.4E-011	@.0
11	0.3771236166328E+001	@.0	@.0

Table 3.14 (Closed Gauss-Chebyshev)

In the following examples, three improper integrals are computed. The values have obtained using Trapezoidal (Th) and Simpson (Sh) rules in the stochastic arithmetic

by using the perturbation method. The results are shown in double precision. At first, $m = 4$ in Trapezoidal and $m = 8$ in Simpsom rule are considered. Then, m increases with step 2. According to the relations (3.4.4) and (3.4.5), the conditions $n < \frac{m}{2}$ for Trapezoidal and $n < \frac{m}{4}$ for Simpson rule are considered. For each m , n increases, until the number of common significant digits between two approximate values $I_{m,n}$ and $I_{m,n+1}$ become zero. In other words, if $I_{m,n+1} - I_{m,n} = @0$ then, m is an optimal value, which yields the approximate value of I.

Example 3.5.6. *In this example, the numerical solution of the improper integral* $I = \int_0^\infty e^{-x^2} dx$ *is considered. The exact value is* $I = \frac{\sqrt{\pi}}{2} = 8.862269254527579E{-}001$*, [2]. In this case,* $I_m = \int_0^m e^{-x^2} dx$.

Hence,

$$| I - I_m | = | \int_m^\infty e^{-x^2} dx | \leq \int_m^\infty e^{-x^2} dx \leq \int_m^\infty e^{-x} dx = \frac{1}{e^m} < \frac{1}{2^m},$$

therefore, $I - I_m = O(\frac{1}{2^m})$. *According to theorems (3.4.1) and (3.4.2),* I*, can be estimated with* I_m *in an optimal* m.

The results have obtained by using Trapezoidal (Th) *and Simpson* (Sh) *rules in stochastic arithmetic.*

At first, $\int_0^m e^{-x^2} dx$ *is evaluated with a determined* m. *Then* m *increases and the integral is evaluated again. The calculation is continued until an informatical zero is detected in the difference between two sequential results. The last values* m *in tables 3.15 and 3.16 are the optimal values.*

m	n_{max}	Th
4	1	8.863185461318505E-001
6	2	8.862269679596997E-001
8	3	8.862269254527581E-001
10	4	8.862269254527577E-001
12	5	8.862269254527581E-001

Table 3.15 (Trapezoidal rule)

m	n_{max}	Sh
8	1	7.155085204168580E-001
10	2	7.676289922918276E-001
12	2	7.109218621382717E-001
14	3	8.626895370294552E-001
16	3	8.362143022684955E-001
18	4	8.859844119802730E-001
20	4	8.851598079412636E-001
22	5	8.862269249482280E-001
24	5	8.862269112837772E-001
26	6	8.862269254527581E-001
28	6	8.862269254527578E-001
30	7	8.862269254527581E-001

Table 3.16 (Simpson rule)

Example 3.5.7. *In this example, the improper integral* $I = \int_0^\infty e^{-2x}cos(3x)dx$ *is evaluated. The exact value is* $I = \frac{2}{13} = 0.\overline{153846}$. *In this case,* $I_m = \int_0^m e^{-2x}cos(3x)dx$.

Hence,

$$\mid I - I_m \mid = \mid \int_m^\infty e^{-2x}cos(3x)dx \mid \leq \int_m^\infty \mid e^{-2x}cos(3x) \mid dx \leq \int_m^\infty e^{-2x}dx = \frac{1}{2e^{2m}} < \frac{1}{2}\frac{1}{2^m},$$

therefore, $I - I_m = O(\frac{1}{2^m})$. *According to theorems (3.4.1) and (3.4.2),* I, *can be estimated with* I_m *in an optimal* m.

This integral is evaluated like previous example. The results are determined in tables 3.17 and 3.18. The last values m in these tables are the optimal values.

m	n_{max}	Th
4	1	2.148238054633220E-001
6	2	1.741420015666767E-001
8	3	1.578793399979208E-001
10	4	1.544586939344967E-001
12	5	1.539252849997985E-001
14	6	1.538553038285306E-001
16	7	1.538471289524367E-001
18	8	1.538462514505428E-001
20	9	1.538461631434344E-001
22	10	1.538461546968986E-001
24	11	1.538461539214598E-001
26	12	1.53846153852634E-001
28	13	1.538461538466960E-001
30	14	1.538461538461965E-001
32	15	1.538461538461740E-001
34	16	1.538461538461722E-001
36	17	1.538461538461748E-001
38	18	1.538461538460303E-001
40	18	1.538461538461714E-001

Table 3.17 (Trapezoidal rule)

m	n_{max}	Sh
8	1	4.802812721344164
10	2	9.304777710897865E-001
12	2	1.77065207907418
14	3	2.739722708234744E-001
16	3	4.191053271683907E-001
18	4	1.571867720725508E-001
20	4	1.627716060819796E-001
22	5	1.538149368929772E-001
24	5	1.538305537812287E-001
26	6	1.538418773972054E-001
28	6	1.538398498310364E-001
30	7	1.538459613559010E-001
32	7	1.538458737415856E-001
34	8	1.538461470999883E-001
36	8	1.538461443698804E-001
38	9	1.538461536370679E-001
40	9	1.53846153561917E-001
42	10	1.538461538401618E-001
44	10	1.538461538382356E-001
46	11	1.538461538459887E-001
48	11	1.538461538459413E-001
50	12	1.538461538461459E-001
52	12	1.538461538461445E-001
54	13	1.538461538461504E-001
56	13	1.538461538461510E-001
58	14	1.538461538461491E-001
60	14	1.538461538461502E-001

Table 3.18 (Simpson rule)

Example 3.5.8. *In this example, the improper integral* $I = \int_0^\infty \frac{cosx}{(1+x^2)^2}dx$ *is computed. The exact value is* $I = \frac{\pi}{2e} = 5.77863674895460E - 001$, *[13]. In this case,* $I_m = \int_0^m \frac{cosx}{(1+x^2)^2}dx$.

Hence,

$$\mid I - I_m \mid = \mid \int_m^\infty \frac{\cos x}{(1+x^2)^2}dx \mid \leq \int_m^\infty \mid \frac{\cos x}{(1+x^2)^2} \mid dx \leq \int_m^\infty \frac{1}{x^4}dx = \frac{1}{3m^3}.$$

The relation, $\frac{1}{3m^3} < \frac{1}{2^m}$ *is true for* $1 \leq m \leq 12$. *If* $\frac{1}{3m^3} < M.\frac{1}{2^m}$ *is considered where,* $M > 1$, *the restriction of* m *can be increased.*

Therefore, for m, *not very large,* $I - I_m = O(\frac{1}{2^m})$, I, *can be estimated with* I_m *in an optimal* m.

The results are determined in tables 3.19 and 3.20. The last values m *in these tables are the optimal values. Because, the relation* $I - I_m = O(\frac{1}{2^m})$ *is not true, when* m *increases, the accuracy of the results is fixed after a value of* m.

m	n_{max}	Th
4	1	6.073989067227689E-001
6	2	5.822252574845663E-001
8	3	5.781670757805998E-001
10	4	5.778449272437490E-001
12	5	5.778297353499192E-001
14	6	5.778861289578933E-001
16	7	5.778628731824415E-001
18	8	5.778556797692490E-001
20	9	5.778686221848727E-001
22	10	5.778643671400727E-001
24	11	5.778608409337444E-001
26	12	5.778650844201345E-001
28	13	5.778643188001559E-001
30	14	5.778624591113437E-001
32	15	5.778640935095384E-001
34	16	5.778641362094003E-001
36	17	5.778631023887175E-001
38	18	5.778637677709927E-001
40	16	5.778639875869761E-001

Table 3.19 (Trapezoidal rule)

m	n_{max}	Sh
8	1	6.211093651259351E-001
10	2	4.791966356984850E-001
12	2	5.032810782059101E-001
14	3	5.060642694218668E-001
16	3	4.888465119135796E-001
18	4	5.605837668411248E-001
20	4	5.510278958691134E-001
22	5	5.770513360182029E-001
24	5	5.762621516184915E-001
26	6	5.778626495100095E-001
28	6	5.778574861539080E-001
30	7	5.778624590603949E-001
32	7	5.778640932613278E-001
34	8	5.778641362094233E-001
36	8	5.778631023886115E-001
38	9	5.778637677709814E-001
40	9	5.778639875869904E-001
42	10	5.778633959637324E-001
44	10	5.778636557167814E-001
46	11	5.778638825139500E-001
48	11	5.778635414376049E-001
50	10	5.778636210867866E-001

Table 3.20 (Simpson rule)

Chapter 4

The Evaluation of Optimal Degree of Interpolation Polynomial Using CADNA Library

4.1 Introduction

One of the considerable discussions in data interpolation is to find the optimal number of data which minimizes the error of the interpolation polynomial. In this chapter, at first, two theorems corresponding to the equidistant nodes and the roots of the Chebyshev polynomials are proved in order to estimate the accuracy of the interpolation polynomial, when the number of data increases. Based on these theorems, then, we show that, how by using the CADNA library, it is possible to find the optimal degree of the interpolation polynomial.

In the last section three examples are implemented by using the Lagrange interpolation. In the numerical examples, the Runge's function is considered and is shown that it can be a bad-behaved function in some points.

An important question in data interpolation is, "Can one be sure that, the error of the interpolation polynomial $p_n(x)$ decreases when n increases?". C. Runge [1901]

investigated a function which, the sequence of $p_n(x)$ may not converge to $f(x)$ at equidistant points [23]. But, if the roots of the Chebyshev polynomials are used, as interpolation points, then, the convergence is happened.

The subject of this chapter is to answer the following questions:

- What is the optimal number of the points for the interpolation polynomial at the equidistant and Chebyshev points, such that at this number, the error is minimized?

- What is the accuracy of the interpolation polynomial at the optimal number of the points?

In section 4.2.1, We show that, if the ratio of the successive derivatives of the function f is bounded then, we can estimate the number of significant digits common between $f(x)$ and the interpolation polynomial $p_n(x)$ at equidistant points.

We also show that, at a given point $\overline{x}$, the significant digits common between $p_n(\overline{x})$ and $p_{n+1}(\overline{x})$ are almost equal to the significant digits common between $p_n(\overline{x})$ and the exact value $f(\overline{x})$. According to these results, it is necessary to estimate the accuracy of the computed values $p_n(\overline{x})$ and $p_{n+1}(\overline{x})$.

4.2 Numerical Accuracy of the Interpolation Polynomial

The Lagrange interpolation polynomial at the nodes $a = x_0 < x_1 < x_2 < ... < x_n = b$ has been defined as follows [18,27]

$$p_n(x) = \sum_{i=0}^{n} l_i(x) f(x_i), \tag{4.2.1}$$

where, $l_i(x) = \frac{w(x)}{(x-x_i)w'(x_i)}$ and $w(x) = \prod_{i=0}^{n}(x - x_i)$.

The error of the interpolation polynomial $p_n(x)$ is evaluated by:

$$f_x - p_n(x) = (x - x_0)(x - x_1)(x - x_2)\dots(x - x_n)\frac{f^{(n+1)}(\xi)}{(n+1)!}, \tag{4.2.2}$$

where, $a \leq \xi \leq b$ and $f \in \mathcal{C}^{n+1}[a, b]$.

Let f be a real function which is in $\mathcal{C}^{n+1}$ over [a,b].

Theorem 4.2.1. *Let $p_n(x)$ and $p_{n+1}(x)$ be the interpolation polynomials of the function $f(x)$ at the equidistant points $a = x_0 < x_1 < x_2 < ... < x_n = b$ and $a = x'_0 < x'_1 < x'_2 < ... < x'_{n+1} = b$ respectively. Also let $\overline{x}$ be a value in $[a, b]$ and $x_l \leq \overline{x} < x_{l+1}$ and $x'_k \leq \overline{x} < x'_{k+1}$ then, if there exists a positive real number like M such that $|\frac{f^{(n+1)}(x)}{f^{(n)}(x)}| \leq M$, for all $x \in [a, b]$ and $n \geq 1$ then, the number of common significant digits between $f(x)$ and $p_n(x)$ is estimated as follows:*

$$C_{p_n(\overline{x}),f(\overline{x})} = C_{p_n(\overline{x}),p_{n+1}(\overline{x})} + \log_{10} | \frac{2f(x_l)}{f(x_l) + f(x'_k)} | + O(\frac{1}{n}), \tag{4.2.3}$$

Proof. Let $h = \frac{b-a}{n}$ and $h' = \frac{b-a}{n+1}$ hence, $x_i = x_0 + ih$, $x'_j = x'_0 + jh'$ and $\overline{x} = x_l + \theta h = x'_k + \theta' h'$, $0 \leq \theta, \theta' < 1$, $0 \leq l \leq n, 0 \leq k \leq n+1$, $0 \leq i \leq n$, $0 \leq j \leq n+1$. According to (4.2.2), the error of $p_n(x)$ at $x = \overline{x}$ is,

$$f(\overline{x}) - p_n(\overline{x}) = (\overline{x} - x_0)(\overline{x} - x_1)(\overline{x} - x_2)\dots(\overline{x} - x_n)\frac{f^{(n+1)}(\xi)}{(n+1)!} \tag{4.2.4}$$

where, $a \leq \xi \leq b$. The error of $p_{n+1}(x)$ at $x = \overline{x}$ is,

$$f(\overline{x}) - p_{n+1}(\overline{x}) = (\overline{x} - x'_0)(\overline{x} - x'_1)\dots(\overline{x} - x'_n)(\overline{x} - x'_{n+1})\frac{f^{(n+2)}(\eta)}{(n+2)!} \tag{4.2.5}$$

where, $a \leq \eta \leq b$. From definition 2.1.1,

$$C_{p_n(\overline{x}),f(\overline{x})} - C_{p_n(\overline{x}),p_{n+1}(\overline{x})} = \log_{10} | \frac{p_n(\overline{x}) + f(\overline{x})}{2(p_n(\overline{x}) - f(\overline{x}))} | - \log_{10} | \frac{p_n(\overline{x}) + p_{n+1}(\overline{x})}{2(p_n(\overline{x}) - p_{n+1}(\overline{x}))} |$$

$$= \log_{10} | \frac{p_n(\overline{x}) + f(\overline{x})}{p_n(\overline{x}) + p_{n+1}(\overline{x})} | + \log_{10} | \frac{p_n(\overline{x}) - p_{n+1}(\overline{x})}{p_n(\overline{x}) - f(\overline{x})} | . \tag{4.2.6}$$

According to the hypothesis since, $p_n(x_l) = f(x_l)$ and $p_{n+1}(x'_k) = f(x'_k)$, the first term of (4.2.6) is,

$$\log_{10} | \frac{p_n(\overline{x}) + f(\overline{x})}{p_n(\overline{x}) + p_{n+1}(\overline{x})} | = \log_{10} | \frac{p_n(x_l + \theta h) + f(x_l + \theta h)}{p_n(x_l + \theta h) + p_{n+1}(x'_k + \theta' h')} |$$

$$= \log_{10} \mid \frac{p_n(x_l) + f(x_l) + O(\frac{1}{n})}{p_n(x_l) + p_{n+1}(x'_k) + O(\frac{1}{n})} \mid = \log_{10} \mid \frac{2f(x_l) + O(\frac{1}{n})}{f(x_l) + f(x'_k) + O(\frac{1}{n})} \mid$$

$$= \log_{10} \mid \frac{2f(x_l) + O(\frac{1}{n})}{(f(x_l) + f(x'_k))(1 + O(\frac{1}{n}))} \mid = \log_{10} \mid \frac{2f(x_l) + O(\frac{1}{n})}{(f(x_l) + f(x'_k))}(1 + O(\frac{1}{n})) \mid$$

$$= \log_{10} \mid \frac{2f(x_l)}{f(x_l) + f(x'_k)} + O(\frac{1}{n}) \mid = \log_{10} \mid \frac{2f(x_l)}{f(x_l) + f(x'_k)} \mid + O(\frac{1}{n}).$$

Since, the points x_l and x'_k are near to each other, as n increases, the fraction $\mid \frac{2f(x_l)}{f(x_l)+f(x'_k)} \mid$ is almost 1. Therefore, the logarithm of this value vanishes, as $n \to \infty$.
Also the second term of (4.2.6) is,

$$\log_{10} \mid \frac{p_n(\overline{x}) - f(\overline{x}) - (p_{n+1}(\overline{x}) - f(\overline{x}))}{(p_n(\overline{x}) - f(\overline{x}))} \mid = \log_{10} \mid 1 - \frac{p_{n+1}(\overline{x}) - f(\overline{x})}{p_n(\overline{x}) - f(\overline{x})} \mid .$$

Furthermore, according to (4.2.4) and (4.2.5) since, $\overline{x} - x_i = (l + \theta - i)h$ and $\overline{x} - x'_j = (k + \theta' - j)h'$, thus,

$$\frac{p_{n+1}(\overline{x}) - f(\overline{x})}{p_n(\overline{x}) - f(\overline{x})} = \frac{(k + \theta')(k + \theta' - 1)...(k + \theta' - n)}{(l + \theta)(l + \theta - 1)...(l + \theta - n)} \frac{(\overline{x} - b)h'}{n + 2} (\frac{h'}{h})^{n+1} \frac{f^{(n+2)}(\eta)}{f^{(n+1)}(\xi)}.$$

Since, in this relation $k = l$ or $k = l + 1$ and $0 \leq \theta, \theta' < 1$, thus, the first fraction tends to 1 when n increases. Also, $\overline{x} - b$ is constant. Therefore, the fraction $\frac{(\overline{x}-b)h'}{n+2}(\frac{h'}{h})^{n+1} = \frac{(\overline{x}-b)(b-a)}{(n+1)(n+2)} \frac{1}{(1+\frac{1}{n})^{n+1}}$ vanishes, when n increases. Since, the ratios of successive derivatives of f is bounded, the second term of the relation (4.2.6) has the order $O(\frac{1}{n^2})$. According to these results, it is concluded that,

$$C_{p_n(\overline{x}),f(\overline{x})} = C_{p_n(\overline{x}),p_{n+1}(\overline{x})} + \log_{10} \mid \frac{2f(x_l)}{f(x_l) + f(x'_k)} \mid + O(\frac{1}{n}).$$

□

According to (4.2.3), if the ratios of successive derivatives of the function f is bounded in the interval $[a, b]$ then, the significant digits common between $p_n(\overline{x})$ and $p_{n+1}(\overline{x})$ are almost equal to the significant digits common between $p_n(\overline{x})$ and the exact value $f(\overline{x})$. In this case, if n is large enough $C_{p_n(\overline{x}),p_{n+1}(\overline{x})} \simeq C_{p_n(\overline{x}),f(\overline{x})}$. Thus, one can use the CESTAC method to estimate the error of $p_n(\overline{x})$ and find an optimal

number for data interpolation. When, $p_n(\overline{x}) - p_{n+1}(\overline{x})$ has not any significant digits, the increase of n is unnecessary and $p_n(\overline{x})$ or $p_{n+1}(\overline{x})$ is a valid approximation for $f(\overline{x})$. In this case, $\frac{1}{2}(p_n(\overline{x}) + p_{n+1}(\overline{x}))$ is a suitable approximation for $f(\overline{x})$.

4.3 Numerical Examples

In this section, three examples are considered. The Lagrange interpolation is used in order to find $p_n(\overline{x})$. The programs have been provided by using the CADNA library in double precision.

Example 4.3.1. *In this example,* $f(x) = exp(x),\ -1 \leq x \leq 1$ *is considered. In this case,* $f^{(n)}(x) = exp(x)$*, then* $\frac{f^{(n+1)(x)}}{f^{(n)}(x)} = O(1)$*. In the tables 4.1 and 4.2, the results are determined at the given point* $\overline{x} = -0.9$ *by using equidistant and Chebyshev nodes respectively. One can observe, the difference between* $p_n(\overline{x}) - p_{n+1}(\overline{x})$ *and* $p_n(\overline{x}) - f(\overline{x})$ *has the same reduction cost as n increases.*

n	$p_n(\overline{x})$	$\lvert p_n(\overline{x}) - p_{n+1}(\overline{x})\rvert$	$\lvert p_n(\overline{x}) - f(\overline{x})\rvert$
2	0.000000000000000E+000	0.382214239920926E+000	0.406569659740599E+000
3	0.382214239920926E+000	0.292091821925583E-001	0.243554198196729E-001
4	0.411423422113484E+000	0.55675233696111E-002	0.485376237288532E-002
5	0.405855898743873E+000	0.796746030555E-003	0.7137609967258E-003
6	0.406652644774429E+000	0.909409995207E-004	0.829850338297E-004
7	0.406561703774908E+000	0.8602976269E-005	0.79559656910E-005
8	0.40657030675118E+000	0.692552470E-006	0.647010578E-006
9	0.40656961419871E+000	0.4835808E-007	0.4554189E-007
10	0.40656966255678E+000	0.297089E-008	0.281619E-008
11	0.40656965958589E+000	0.1623E-009	0.1547E-009
12	0.4065696597482E+000	0.795E-011	0.761E-011
13	0.4065696597403E+000	0.4E-012	0.3E-012
14	0.4065696597406E+000	@0	@.0

Table 4.1 (equidistant points)

n	$p_n(\overline{x})$	$\lvert p_n(\overline{x}) - p_{n+1}(\overline{x})\rvert$	$\lvert p_n(\overline{x}) - f(\overline{x})\rvert$
2	0.000000000000000E+000	0.414158983761299E+000	0.406569659740599E+000
3	0.414158983761299E+000	0.6537067131354E-002	0.75893240207001E-002
4	0.407621916629945E+000	0.1346743998849E-002	0.1052256889346E-002
5	0.406275172631095E+000	0.330266983769E-003	0.294487109504E-003
6	0.406605439614865E+000	0.386328187386E-004	0.35779874266E-004
7	0.406566806796126E+000	0.3013717130E-005	0.2852944473E-005
8	0.406569820513257E+000	0.166895868E-006	0.160772658E-006
9	0.406569653617389E+000	0.6224968E-008	0.6123209E-008
10	0.406569659842357E+000	0.9605347E-010	0.101758E-009
11	0.406569659746304E+000	0.6329E-011	0.5705E-011
12	0.406569659739975E+000	0.6586E-012	0.624E-012
13	0.406569659740633E+000	0.36E-013	0.34E-013
14	0.406569659740598E+000	0.1E-014	0.1E-014
15	0.406569659740599E+000	@.0	@.0

Table 4.2 (Chebyshev points)

It is concluded that, the optimal degree of interpolation polynomial is $n = 14$ at equidistant points, and $n = 15$ at the Chebyshev points with approximate value $p_n(-0.9) = \frac{1}{2}(p_{14}(-0.9) + p_{15}(-0.9)) = 0.40656965974059$.

Example 4.3.2. *In this example,* $f(x) = xsin(x),\ -1 \leq x \leq 1$ *is considered. In this case,*

$f^{(n)}(x) = xsin(x + \frac{n\pi}{2}) + nsin(x + \frac{(n-1)\pi}{2})$,

then, $\frac{f^{(n+1)(x)}}{f^{(n)}(x)} = O(1)$ *. In the tables 4.3 and 4.4 the results are determined at the given point* $\overline{x} = -0.9$ *by using equidistant and Chebyshev nodes respectively. One can observe the difference between* $p_n(\overline{x}) - p_{n+1}(\overline{x})$ *and* $p_n(\overline{x}) - f(\overline{x})$ *has the same reduction cost as n increases.*

n	$p_n(\overline{x})$	$\lvert p_n(\overline{x}) - p_{n+1}(\overline{x})\rvert$	$\lvert p_n(\overline{x}) - f(\overline{x})\rvert$
2	0.000000000000000E+000	0.681591497694396E+000	0.704994218664735E+000
3	0.681591497694396E+000	0.3327686257538E-002	0.23402720970339E-001
4	0.684919183951934E+000	0.20758708703046E-001	0.20075034712801E-001
5	0.705677892654981E+000	0.163318978884E-003	0.6836739902458E-003
6	0.705514573676096E+000	0.527903689725E-003	0.520355011361E-003
7	0.704986669986371E+000	0.2152489879E-005	0.754867836361139E-005
8	0.704988822476249E+000	0.5439066719E-005	0.5396188485E-005
9	0.704994261542969E+000	0.1363742E-007	0.42878234E-007
10	0.704994247905554E+000	0.2938562E-007	0.2924082E-007
11	0.704994218519937E+000	0.5029E-010	0.14479E-009
12	0.704994218570230E+000	0.9482E-010	0.94505E-010
13	0.70499421866505E+000	0.1E-012	0.32E-012
14	0.70499421866493E+000	0.2E-012	0.19E-012
15	0.70499421866473E+000	@.0	@.0

Table 4.3 (equidistant points)

n	$p_n(\overline{x})$	$\lvert p_n(\overline{x}) - p_{n+1}(\overline{x})\rvert$	$\lvert p_n(\overline{x}) - f(\overline{x})\rvert$
2	0.000000000000000E+000	0.712479775132298E+000	0.704994218664735E+000
3	0.712479775132298E+000	0.11879530216013E-001	0.7485556467563E-002
4	0.700600244916285E+000	0.4676047581938E-002	0.439397374844963E-002
5	0.705276292498224E+000	0.58668966378E-004	0.282073833489E-003
6	0.705217623531845E+000	0.226096999944E-003	0.2234048671102E-003
7	0.704991526531902E+000	0.136090889E-005	0.2692132833E-005
8	0.704992887440799E+000	0.133694952E-005	0.1331223935E-005
9	0.704994224390319E+000	0.4677194E-008	0.5725584E-008
10	0.704994219713125E+000	0.104309E-008	0.1048389E-008
11	0.704994218670037E+000	0.239E-011	0.5302E-011
12	0.704994218672424E+000	0.7721E-011	0.7689E-011
13	0.704994218664703E+000	0.1E-013	0.32E-013
14	0.704994218664716E+000	0.2E-013	0.19E-013
15	0.704994218664735E+000	@.0	@.0

Table 4.4 (Chebyshev points)

It is concluded that, the optimal degree of interpolation polynomial is $n = 15$ with approximate value $p_n(-0.9) = \frac{1}{2}(p_{14}(-0.9) + p_{15}(-0.9)) = 0.704994218664$.

Example 4.3.3. *The hypothesis of* $|\frac{f^{(n+1)}(x)}{f^{(n)}(x)}| \leq M$ *in the theorem 4.2.1 is important. In this example, the Runge's function* $f(x) = \frac{1}{1+12x^2}$ *for* $-1 \leq x \leq 1$ *is considered [23]. For this function this hypothesis is not satisfied.*

For this reason, the following lemma is used which is proved in Appendix 1.

Lemma 4.3.4. *The nth derivative of the function* $f(x) = \frac{1}{1+12x^2}$ *is evaluated as follows,*

$$f^{(n)}(x) = \sum_{i=0}^{[\frac{n}{2}]} \frac{\alpha_i x^{n-2i}}{(1+12x^2)^{n+1-i}}, \tag{4.3.1}$$

where, $\alpha_0 = (-1)^n 24^n n!$ *and* $\alpha_i,\ i = 1, ..., [\frac{n}{2}]$, *are integer numbers such that,*
$\mid \alpha_i \mid < 24^n n!$.

According to lemma 4.3.4, since $\mid x \mid \leq 1$, $\mid f^{(n)}(x) \mid \leq 24^n n!([\frac{n}{2}]+1)$. *Hence,* $\frac{f^{(n+1)}(x)}{f^{(n)}(x)} = O(n)$, *thus, the ratio of the derivative is not bounded. In this case, the Lagrange interpolation polynomial does not converge to* f *as* n *increases. This situation is more evident at the end-points. In the table 4.5, the point* $\overline{x} = -0.9$ *is chosen to show this problem.*

One can observe the number of significant digits of the $p_n(\overline{x})$ *decreases as* n *increases. When* $n = 81$ *the value* $p_n(-0.9)$ *is an informatical zero. In this case, the result is not reliable.*

n	$p_n(\overline{x})$	$\lvert p_n(\overline{x}) - p_{n+1}(\overline{x})\rvert$	$\mid p_n(\overline{x}) - f(\overline{x}) \mid$
2	0.000000000000000E+000	0.252307692307692E+000	0.932835820895522E-001
3	0.252307692307692E+000	0.100219780219780E+000	0.159024110218140E+000
4	0.152087912087912E+000	0.325964835164835E+000	0.588043299983598E-001
5	-0.173876923076923E+000	0.163481912681913E+000	0.267160505166475E+000
6	-0.10395010395010E-001	0.499571625779626E+000	0.103678592484563E+000
.	.	.	.
.	.	.	.
.	.	.	.
75	-0.89E+002	0.62E+002	0.89E+002
76	-0.3E+002	0.10E+003	0.3E+002
77	0.76E+002	0.6E+002	0.76E+002
78	0.2E+002	0.7E+002	0.2E+002
79	-0.5E+002	0.4E+002	0.5E+002
80	-0.8E+001	0.8E+001	0.8E+001
81	@.0	@.0	@.0

Table 4.5 (equidistant points)

If the Chebyshev nodes are used for this function the convergence is guaranteed. The table 4.6 shows the results.

n	$p_n(\overline{x})$	$\lvert p_n(\overline{x}) - p_{n+1}(\overline{x})\rvert$	$\lvert\, p_n(\overline{x}) - f(\overline{x})\,\rvert$
2	0.000000000000000E+000	0.279999999999998E-001	0.932835820895522E-001
3	0.279999999999998E-001	0.778064516129034E-001	0.652835820895524E-001
4	0.105806451612903E+000	0.106488418826018E+000	0.12522869523351E-001
5	-0.6819672131149E-003	0.643079412390889E-001	0.939655493026671E-001
6	0.636259740259739E-001	0.110364387991830E+000	0.296576080635782E-001
7	0.173990362017804E+000	0.638152030691183E-001	0.807067799282519E-001
8	0.110175158948686E+000	0.442894558768628E-001	0.168915768591337E-001
9	0.658857030718230E-001	0.25234567562833E-001	0.273978790177292E-001
10	0.911202706346565E-001	0.4094680090213E-002	0.21633114548957E-002
.	.	.	.
.	.	.	.
.	.	.	.
110	0.932835820895558E-001	0.14E-013	0.4E-014
111	0.932835820895416E-001	0.82E-014	0.11E-013
112	0.932835820895497E-001	0.72E-014	0.25E-014
113	0.932835820895569E-001	0.4E-014	0.5E-014
114	0.932835820895528E-001	@.0	@.0

Table 4.6 (Chebyshev points)

It is concluded that, the optimal degree of interpolation polynomial is $n = 114$ with approximate value $p_n(-0.9) = \frac{1}{2}(p_{113}(-0.9) + p_{114}(-0.9)) = 0.9328358220895$.

Conclusion and Comments

In this research, the valid methods are explained to implement the numerical algorithms. It is shown that, the CESTAC method can estimate the accuracy of the results. By using the stochastic arithmetic, one is able to validate the results and obtain reliable values. Also, one conclude that, the CADNA library is a powerful tool that helps to achieve validity and safety in scientific computations.

The following topics are explained:

1. The numerical solution of a definite integral $I = \int_a^b f(x)dx$ where, $f \in \mathcal{C}^m[a,b]$ and $a, b \in \mathbb{R}$; if it exists. By using the CESTAC method, based on the stochastic arithmetic, one can use the Closed Newton-Cotes methods like the Trapezoidal, Simpson or Boolean rule to approximate it. According to theorem 3.2.4 one can find an optimal value N_{opt} for the number of the points in these rules.

2. The numerical solution of the integral $I = \int_{-1}^{1} \frac{f(x)}{\sqrt{1-x^2}}dx$; one can use the Gauss-Chebyshev rule in the stochastic arithmetic and approximate this integral with a valid implementation and find the optimal step.

3. Finding divergence and instabilities of a definite integral; The methods of numerical integration may show inaccurate or false results, because of a singular point in the interval [a,b]. In this case, the floating-point arithmetic may not

distinguish it. But, the stochastic arithmetic is able to find the instability and show the invalidity of the result.

4. The numerical solution of an improper integral, $I = \int_a^{\infty} f(x)dx$ where $a \in \mathbb{R}$; if it exists. By using the CESTAC method , one can use the Trapezoidal and Simpson rules to approximate it with a definite integral such as $I_m = \int_a^m f(x)dx$ where, $m \in \mathbb{N}$ and $I = I_m + O(\frac{1}{2^m})$. It is shown in theorems 3.4.1 and 3.4.2, one can find an optimal value m such that, $I \simeq I_{m,n}$. Therefore, the stochastic arithmetic is able to approximate a converge improper integral with a valid implementation and obtain the optimal step.

5. Finding the value of interpolation polynomial with an optimal degree; in this case, the error of interpolation polynomial is minimized. It is proved, in the theorem 4.2.1, the accuracy of this polynomial depends on the nodes and the ratio of the successive derivatives of the function $f(x)$. For some function like the Runge's example, the choice of equidistant nodes, maybe decreases the number of significant digits of the interpolation polynomial $p_n(x)$ at a given point $x = \overline{x}$. But, the Chebyshev nodes are suitable for this purpose. One can observe that the CADNA library is able to find the optimal number of data and a satisfactory solution can be obtained for the value of the interpolation polynomial. In this case, the numerical instabilities are detected, the accuracy of the solution is estimated and the computer execution time is saved. Consequently, CADNA library is an efficient tool for finding the value of the interpolation polynomial of a given function with the number of optimal data.

So, these advantages guide us to use stochastic arithmetic to find the solution of

the numerical methods and validate the results and algorithms.

For the present time, the researches about the different applications of the stochastic arithmetic is continued. The scientific groups in the Lip6 laboratory [34] in France, are working on the use of the CADNA library. Some open works which one can do by using the CESTAC method and the stochastic arithmetic is as follows:

1. Finding the value of a definite integral by using the other integration methods like Gauss and Romberg rules, also evaluating of the improper integrals generally. In this case, the aim is to find an optimal number which the integral is computed with a definite integral. For this purpose, a suitable algorithm should be considered for all kinds of the functions. The evaluating of integrals which have some poles in the interval of integration,

2. The solving a linear or nonlinear system of equations, by using the iterative methods, specially the sparse systems. About this topics, some works have been done by using the CADNA library [28,29],

3. The solving of the linear integral equation, especially of the first kind,

4. Finding the optimal number of data by using the other interpolation polynomials like splines,

5. Determining the optimal meshes, in the P.D.E. topics and the approximation methods such as finite-difference methods, also in the different subjects in O.D.E.

In any case, the applying of the CESTAC method has some problems which should be considered.

- The method is based on the probabilistic approach and therefore, the theory of the method is not accurate mathematically. Also, the validity of the CESTAC method and the quality of the estimates should be considered. The assumptions which are used in the theorems are very strong, so that the method is restricted. The evaluation of the number of significant digits is highly dependent of Gaussian behavior which apply for the linear computations.

  So, the improvement of the method is suggested which is based on the mathematical justifications.

- The choice $N = 3$ is not acceptable statistically. According to the central limit theorem and the topics in probability theory [21], when N tends to infinity, the distribution of the random variable is Normal. Hence, the number of the samples must increase. For this purpose, it is suggested, the parallel computers are used, in order to implement the CESTAC method. In this case, the computer execution time does not increase and is economical from the point of view numerical analysis. In [1], the parallel processing is used over the CADNA library. Of course, if one use the parallel computers, this number can be chosen a greater value.

- In the evaluating of the number 'C', the number of the significant digits, the mean of the random samples is a value which most of the time is not equal to the exact result. Therefore, this number can not be used to say anything about the exact result. Also, if the variance of the samples in the CESTAC method is zero, which is happened when the value is too small, the formula of finding 'C' may not able to show the expectation result. Also, the termination criteria

may not satisfied. This is because of the nature of the method which perturbs only the last mantissa bit. In this case, it is suggested, the method is developed so that the previous bits can be changed like in the perturbation method.

Of course, if the other relation for estimating the number of the significant digits is presented then, this problem may be solved.

- The name 'stochastic zero' does not mean a 'zero' has been detected necessarily. This name may mislead the one who wants to use the method. In some cases, a very large number, is a stochastic or informatical zero. The aim of this name is only to show that a numerical instability is detected. In this case, it is suggested this name is changed. for example, It is called, 'instable mode' or 'insignificant value'.

- The stochastic arithmetic is not the only way to validate and rely the algorithms and their results. One can use the interval arithmetic or other schemes to implement the code and know the validity of the results.

Bibliography

1. Asserrhine J., Chesneaux J. M., Lamotte J. L., 1998, *Estimation of Round-off Errors On Several Computers Architecture*, Journal of Universal Computer Science, Vol. 1, No. 7, 454-468.

2. Bartle R. G., 1976, *The Elements of Real Analysis*, John Wiley and Sons, Second Edition.

3. Burden R. L., Faires J. D., 1985, *Numerical Analysis*, P.W.S., third Edition, Boston.

4. Chesneaux J. M., 1994, *The equality relations in scientific computing*, Numerical Algorithms, Vol. 7, 129-143.

5. Chesneaux J. M., 1988, *Modelisation et conditions de validite de la methode CESTAC*, C.R. Acad. Sci. Paris, Serie I, Vol. 307, 417-422.

6. Chesneaux J. M., 1992, *Stochastic Arithmetic Properties*, Computational and Applied Mathematics, IMACS, 81-91.

7. Chesneaux J. M., 1990, *CADNA, An ADA Tool for Round-off Error Analysis and for Numerical Debugging*, Proceedings Congress on ADA in Aerospace, Barcelona.

8. Chesneaux J. M., 1990, *Study of the computing accuracy by using probabilistic approach*, Contribution to Computer Arithmetic and Self-validating Numerical Methods, Edited by C.Ullrich., IMACS, 19-30.

9. Chesneaux J. M., Jezequel F., 1998, *Dynamical Control of Computations Using the Trapezoidal and Simpson's Rules*, Journal of Universal Computer Science, Vol. 4, No. 1, 2-10.

10. Chesneaux J. M., Salhi Y., 1989, *Theoretical study of C.E.S.T.A.C. method and ADA Implementation*, Numerical And Applied Mathematics, IMACS, 617-621.

11. Chesneaux J. M., Vignes J., 1988, *Sur la robustesse de la methode CESTAC*, C.R. Acad. Sci. Paris, Serie I, Vol. 307, 855-860.

12. Chesneaux J. M., Vignes J., 1992, *Les fondements de Larithmetique Stochastique*, C.R. Acad. Sci. Paris, Serie I, Vol. 315, 1435-1440.

13. Churchill R. V., Brown J. W., Verhey R. F., 1974, *Complex Variables and Applications*, McGraw-Hill, Third Edition.

14. Conte S. D., De Boor C., 1981, *Elementry Numerical Analysis, An Algorithmic Approach*, International Student Edition, Third Edition.

15. Delves L. M., Mohamed J. L., 1985, *Computational Methods for Integral Equations*, Cambridge University.

16. Faye J. P., Vignes J., 1985, *Stochastic Approach of the Permutation-Perturbation Method for Round-off Error Analysis*, Applied Numerical Mathematics-Transaction of IMACS, Vol. 1, No. 4, 349-362.

17. Feldstein A., Goodman R., 1976, *Convergence Estimates for the Distribution of Trailing Digits*, Journal of A.C.M., Vol. 23, 287-297.

18. Hammerlin G., Hoffmann H. 1991, *Numerical Mathematics*, Springer-Verlag, NewYork Inc..

19. Hamming R. W., 1970, *On the Distribution of Numbers*, The Bell System Technical Journal, 1609-1625.

20. Knuth D. E., 1969, *The Art of Computer Programming*, Vol. 2, Addison-Wesley.

21. Larson H. J., 1974, *Introduction to Probability and Statistical Inference*, John Wiley and Sons.

22. Maille M., 1982, *Some methods to estimate accuracy of measurements or numerical Computations*, Proc. Math. for Computation Congress AFCET Paris, 495-503.

23. Mathews J. H., Translated by Toutounian F., 2000, *Numerical Methods for Mathematics, Science and Engineering*, Ferdowsi University, Vol. 2, Second Edition, Mashhad, Iran.

24. Mossaheb G. H., 1984, *Mathematical Analysis, Theory of Real Numbers*, Amir Kabir Inc., Vol. 1, Fifth Edition, Tehran, Iran.

25. Pichat M., 1994, *Chaotic evaluation and Stochastic arithmetic*, Proc. 14th, IMACS world congress, Atlanta.

26. Rivlin T. J., 1990, *Cheyshev Polynomials*, John Wiley and Sons, Second Edition.

27. Stoer J., Bulirsch R., 1980, *Introduction to Numerical Analysis*, Springer-Verlag, NewYork Inc.

28. Toutounian F., 1998, *The stable $A^T A$-orthogonal S-step Orthomin(k) Algorithm with CADNA library*, Numerical Algorithms, Vol. 17, 105-119.

29. Toutounian F., 1997, *The Use of the CADNA library for validating the numerical results of the hybrid GMRES algorithm*, Applied Numerical Mathematics, Vol. 23, 275-289.

30. Toutounian F., 1997, *A stochastic method for the accumulation of round-off error Analysis*, Farhang va Andishe-ye Riazi, Vol. 16, No. 2, 23-44.

31. Vignes J., 1993, *A stochastic arithmetic for reliable scientific computation*, Mathematics and Computers in Simulation, Vol. 35, 233-261.

32. Vignes J., 1978, *New Methods For Evaluating the Validity of the Results of Mathematical Computations*, Mathematics and Computers in Simulation, IMACS, Vol. XX, No. 4, 227-249.

33. Vignes J., 1986, *Informatique Theorique-Zero Mathematique et zero Informatique*, C.R. Acad. Sci. Paris, Serie I, Vol. 303, No. 20, 997-1000.

34. http://www.anp.lip6.fr/english/cadna/

Appendix 1

4.4 The Properties of the Notation 'O'

Definition Let f_n and g_n be two sequences. One says that, f_n is of order g_n (or f_n is 'big-oh' of g_n), and writes $f_n = O(g_n)$ in case

$$\exists M > 0,\ \exists N \in \mathbb{N},\ \forall n > N, \mid f_n \mid \leq M \mid g_n \mid,$$

where, M is a real constant value, independent from n. In this case, the order of f_n is less than or equal to the order of g_n when n increases. The notation O has the following properties which have been explained in [14,24]:

1. If f_n is a sequence then, $f_n = O(f_n)$.

2. The sequence f_n is bounded, if and only if $f_n = O(1)$.

3. The equalities $f_n = O(g_n)$ and $O(g_n) = f_n$ are different. In other words, $'='$ in this case, is not symmetric.

4. The sum of two bounded sequences is a bounded sequence, or $O(1) + O(1) = O(1)$.

5. If $a_n = O(b_n)$ and c_n is a nonnegative sequence, from an index n then, $a_n c_n = O(b_n c_n)$.

6. If α is a constant value and f_n is a sequence then, $O(\alpha f_n) = O(f_n)$.

7. If $\alpha > 0$ and $f_n = O(g_n)$ then, $\mid f_n \mid^{\alpha} = O(\mid g_n \mid^{\alpha})$.

8. If $\lambda_1, \lambda_2, ..., \lambda_k,\ k \in \mathbb{N}$, are constant values, and $f_n^{(1)}, ..., f_n^{(k)},\ g_n^{(1)}, ..., g_n^{(k)}$, are sequences such that, $f_n^{(i)} = O(g_n^{(i)}),\ i = 1, 2, ..., k$, then,

$$\sum_{i=1}^{k} \lambda_i f_n^{(i)} = O(\sum_{i=1}^{k} \mid \lambda_i g_n^{(i)} \mid) \quad , \quad \prod_{i=1}^{k} f_n^{(i)} = O(\prod_{i=1}^{k} g_n^{(i)}).$$

9. $O(O(f_n)) = O(f_n)$.

10. $O(f_n) + O(f_n) = O(f_n)$.

11. $O(f_n).O(g_n) = O(f_n g_n)$.

12. If $f_n = O(g_n)$ then, $O(f_n) + O(g_n) = O(g_n)$.

13. If $f_n = O(g_n)$ and $f'_n = O(g'_n)$ then,

$$f_n \pm f'_n = O(g_n \pm g'_n),$$

$$f_n \times f'_n = O(g_n \times g'_n),$$

$$f_n / f'_n = O(g_n / g'_n),\ f'_n, g'_n \neq 0,$$

14. If $\mid f_n \mid \leq 1$ then, $\frac{1}{1 \pm f_n} = 1 \mp O(f_n)$.

15. $log_{10} \mid f_n + g_n \mid = log_{10} \mid f_n \mid + O(\frac{g_n}{f_n}),\ f_n \neq 0$.

16. If $\alpha_n = \alpha + O(f_n)$ such that, $\lim_{n \to \infty} f_n = 0$ and C is a constant, then, $C\alpha_n = C\alpha + O(f_n)$, if also $\beta_n = \beta + O(g_n)$ such that $g_n = O(f_n)$ then, $\alpha_n + \beta_n = \alpha + \beta + O(f_n)$. $\alpha_n \times \beta_n = \alpha \times \beta + O(f_n)$. $\alpha_n / \beta_n = \alpha / \beta + O(f_n),\ \beta \neq 0$, in this case, if $\alpha = \beta = 0$ then, $\alpha_n.\beta_n = O(f_n.g_n)$.

4.5 The n-th derivative of the Runge's function

Lemma The nth derivative of the function $f(x) = \frac{1}{1+12x^2}$ is evaluated as follows:

$$f^{(n)}(x) = \sum_{i=0}^{[\frac{n}{2}]} \frac{\alpha_i x^{n-2i}}{(1+12x^2)^{n+1-i}}, \tag{4.5.1}$$

where, $\alpha_0 = (-1)^n 24^n n!$ and $\alpha_i,\ i = 1, ..., [\frac{n}{2}]$, are integer numbers such that,

$$\mid \alpha_i \mid < 24^n n!.$$

Proof. According to mathematical induction on n, if $n = 1$ then $f'(x) = \frac{-24x}{(1+12x^2)^2}$. Thus, $\alpha_0 = -24$. So, for $n = 1$ the relation (4.5.1) is true. Now let the relation (4.5.1) be true, it must be proved that,

$$f^{(n+1)}(x) = \sum_{i=0}^{[\frac{n+1}{2}]} \frac{\alpha'_i x^{n+1-2i}}{(1+12x^2)^{n+2-i}},$$

where, $\alpha'_0 = (-1)^{n+1} 24^{n+1}(n+1)!$ and $\alpha'_i,\ i = 1, ..., [\frac{n+1}{2}]$, are integer numbers such that $\mid \alpha'_i \mid < 24^{n+1}(n+1)!$. In order to prove, let n be even. Thus, $[\frac{n}{2}] = \frac{n}{2}$ and $[\frac{n+1}{2}] = \frac{n}{2}$ then,

$$f^{(n+1)}(x) = (f^{(n)}(x))' =$$

$$\sum_{i=0}^{\frac{n}{2}} \frac{\alpha_i(n-2i)x^{n-2i-1}(1+12x^2)^{n+1-i} - \alpha_i x^{n-2i} 24x(n+1-i)(1+12x^2)^{n-i}}{(1+12x^2)^{2(n+1-i)}}$$

$$= \sum_{i=0}^{\frac{n}{2}} \frac{\alpha_i(n-2i)(1+12x^2)x^{n-1-2i} - 24\alpha_i(n+1-i)x^{n-2i+1}}{(1+12x^2)^{n-i+2}}$$

$$= \frac{\alpha_0 n x^{n-1}}{(1+12x^2)^{n+1}} + \frac{-24\alpha_0(n+1)x^{n+1}}{(1+12x^2)^{n+2}} + \frac{\alpha_1(n-2)x^{n-3}}{(1+12x^2)^n} + \frac{-24\alpha_1 n x^{n-1}}{(1+12x^2)^{n+1}} +$$

$$\frac{\alpha_2(n-4)x^{n-5}}{(1+12x^2)^{n-1}} + \frac{-24\alpha_2(n-1)x^{n-3}}{(1+12x^2)^n} + ... + \frac{-24\alpha_{\frac{n}{2}} x(\frac{n}{2}+1)}{(1+12x^2)^{\frac{n}{2}+2}} = \sum_{i=0}^{\frac{n}{2}} \frac{\alpha'_i x^{n+1-2i}}{(1+12x^2)^{n+2-i}},$$

where, $\alpha'_0 = -24\alpha_0(n+1)$ and $\alpha'_i = \alpha_{i-1}(n-2i+2) - 24\alpha_i(n-i+1);\ i = 1, 2, ..., \frac{n}{2}$. According to induction hypothesis, $\alpha'_0 = (-1)^{n+1} 24^{n+1}(n+1)!$. Besides,

$$\mid \alpha'_i \mid < \mid \alpha_{i-1} \mid n + 24 \mid \alpha_i \mid (\frac{n}{2}+1) \leq 24^n n!.n + 24^{n+1} n!(\frac{n}{2}+1) < 24^{n+1}(n+1)!,$$

consequently, $\mid \alpha'_i \mid < 24^{n+1}(n+1)!$. If n is odd, then, the similar proof is presented. □

Appendix 2

4.6 Computer Programs

```
!*****************************************************************
    ! The subroutine 'rand' perturbs the last mantissa bit or previous
    ! bits, if necessary, of the value aa and finds n=3 random samples
    ! for this value by using the perturbation method.
    !*****************************************************************
    subroutine rand(aa,a1,eps)
    integer,parameter:: n=3
    real(8):: x1(n),x,eps
    real(8):: aa,a1(n)
    integer:: e,d,p1
    e=exponent(aa)
    d=digits(aa)
    if (eps.gt.2.d0**(-d)) then
    p1=ceiling(abs(log10(eps)/log10(2.d0)))
    d=p1
    print *,"Perturbation in the",d,"th bit "
    endif
```

```
do i=1,n
call random_number(x)
x1(i)=x
enddo
x1=-1.d0+2.d0*x1
if(aa==0.d0) then
a1=-(2.d0**(e-d))*(x1)
else
a1=aa-(abs(aa)/aa)*(2.d0**(e-d))*(x1)
endif
endsubroutine rand
!*************************************************
The subroutine 'vari' calculates the number of significant
digits for the mean value of the array w.
!*************************************************
subroutine vari(w,ave,cad,c)
integer,parameter:: n=3
real (8)::var,summ,sum1,w(n),ave,c
character(5)::cad
summ=sum(w)
ave=summ/real(n)
sum1=sum((w-ave)**2)
var=(1.d0/real(n-1))*sum1
if(ave==0.d0) then
```

```
print *,"instability detected in the branching ","ave=",ave &
," " , adjustr(cad)," =@0"
else
if (var==0.d0) then
print *,"warning–The variance is zero"
return
else
c=log10(abs(ave)/sqrt(var))-.39d0
endif
endif
if(c < = 0.d0) print *,"instability detected in the branching"," c= ",c &
," " ,adjustr(cad)," =@0"
endsubroutine vari
!***********************************************
! Lagrange Interpolation
! data interpolation by using CADNA library
! for the function: f(x)=1./(1.+12.*x**2)
! Chebyshev Polynomials
!***********************************************
program interpol
use cadna
type (double_st) :: a,b,prod,erroptf,l(500),x(500),erroptn,pnx
type (double_st) :: fx(500),sum,xbar,fxbar,pnxbar
open (3,file='erinchc.res')
```

```
call cadna_init(0)
a=-1.d0
b=1.d0
n=1
sum=0.d0
prod=1.d0
!********** xbar  **************
xbar=-0.9d0
fxbar=1.d0/(1.d0+12.d0*xbar**2)
1   n=n+1
do i=0,n
x(i+1)=cos((2.d0*i+1.d0)*4.d0*atan(1.d0)/(2.d0*(n+1)))
fx(i+1)=1.d0/(1.d0+12.d0*x(i+1)**2)
enddo
do i=0,n
do j=0,n
if (i==j) cycle
prod=prod*((xbar-x(j+1))/(x(i+1)-x(j+1)))
enddo
l(i+1)=prod
prod=1.d0
enddo
do i=0,n
sum=sum+l(i+1)*fx(i+1)
```

```
enddo
pnxbar=sum
sum=0.d0
erroptn=abs(pnx-pnxbar)
erroptf=abs(fxbar-pnx)
write(3,*) "n=",n,"erroptn=",str(erroptn),"erroptf=",str(erroptf)
write(3,*) "xbar=",str(xbar),"pnx=",str(pnx)
if(erroptn==0.) then
write(3,*) "optimal n=",n-1,"erroptn=",str(erroptn),"erroptf=",str(erroptf)
write(3,*) "xbar=",str(xbar),"pnx=",str(pnx)
stop
endif
pnx=pnxbar
goto 1
end
!************************************************************ !
The calculation of an improper integral in stochastic arithmetic
by using the Trapezoidal rule (f(x)=exp(-x**2))
!************************************************************
integer,parameter :: n=3
character(5) :: cad
real(8) :: ave2,a,um,umm(n),aa(n),hh(n),Thh(n),ssumm(n)&
,x(n),f(n),eps,c,cc,ave,fa(n),fm(n),Th(n),Thh1(n)
write (*,*) "please enter a and eps"
```

```
read (*,*) a,eps
call rand(a,aa,eps)
um=2.d0
1  um=um+2.d0
call rand(um,umm,eps)
Thh1=0.d0
do n1=1,int(um)/2
hh=(umm-aa)/2**n1
cad="hh"
call vari(hh,ave,cad,c)
ssumm=0.
do i=1,2**n1-1
x=aa+i*hh
! ***************
f=exp(-x**2)
cad="f"
call vari(f,ave,cad,c)
ssumm=ssumm+2*f
cad="ssumm"
call vari(ssumm,ave,cad,c)
if (c < = 0.d0 .or. ave==0.d0) then
write (*,*) "The Integral has an instability ","ssum",ave,"c=",c,"mm=",umm
pause
endif
```

```
enddo
!*********************
fa=exp(-aa**2)
cad="fa"
call vari(fa,ave,cad,c)
!*********************
fm=exp(-umm**2)
cad="fm"
call vari(fm,ave,cad,c)
!*********************
Thh=(hh/2.d0)*(fa+ssumm+fm)
cad="Thh"
call vari(Thh,ave,cad,c)
ave2=ave
ccThh=c
Th=Thh-Thh1
cad="Th"
call vari(Th,ave,cad,c)
cc=c
if (cc < = 0.d0 .or. ave==0.d0) then
write (*,*) "The result is ","ccerror=",cc,"m=",um,"Thh=",ave2,"cc=",cc,&
"n=",n1 − 1,"error=",abs(ave),"ccThh=",ccThh
stop
else
```

```
print *,"Thh=",ave2,"ccerror=",cc,"m=",um,"n=",n1-1,&
"error=",abs(ave),"ccThh=",ccThh
pause
Thh1=Thh
endif
nn1=n1
enddo
print *,"Thh=",ave2,"ccerror=",cc,"m=",um,"n=",nn1-1,
&"ccThh=",ccThh,"error=",abs(ave)
print *,"**********************************"
pause
goto 1
end
!*********************************************************
! The use of CESTAC method in order to calculate the improper integral of
! a function by using Simpson rule (f(x)=exp(-x**2))
integer,parameter :: n=3
character(5) :: cad
integer :: n1,m,l,ir
real(8) :: ave2,a,aa(n),um,umm(n),hh(n),hh2(n),shh(n),ssumm(n)&
,eps,c,cc,ave,Sh(n),shh1(n),xx(n),yy(n)
write (*,*) "please enter a and eps"
read (*,*) a,eps
call rand(a,aa,eps)
```

```
pause
um=6.d0
ave1=0.d0
shh1=0.d0
1  um=um+2.d0
call rand(um,umm,eps)
ir=(int(um)/4)*4-int(um)
if (ir==0) then
l=um/4
else
l=int(um/4)+1
endif
do n1=1,l
hh=(umm-aa)/2**n1
cad="hh"
call vari(hh,ave,cad,c)
m=(2**n1)/2-1
hh2=2*hh
cad="hh2"
call vari(hh2,ave,cad,c)
!**************************************
ssumm=exp(-aa**2)+exp(-umm**2)
cad="ssumm"
call vari(ssumm,ave,cad,c)
```

```
xx=aa-hh
cad="xx"
call vari(xx,ave,cad,c)
yy=aa
do i=1,m
xx=xx+hh2
cad="xx"
call vari(xx,ave,cad,c)
yy=yy+hh2
cad="yy"
call vari(yy,ave,cad,c)
!****************************************
ssumm=ssumm+4.d0*(exp(-xx**2))+2.d0*(exp(-yy**2))
cad="ssumm"
call vari(ssumm,ave,cad,c)
enddo
!**********************************
ssumm=ssumm+4.d0*(exp(-(umm-hh)**2))
cad="ssumm"
call vari(ssumm,ave,cad,c)
shh=ssumm*hh/3.d0
cad="shh"
call vari(shh,ave,cad,c)
if (c < = 0.d0 .or. ave==0.d0) then
```

```
write (*,*) "The Integral has an instability ","shh",ave,"c=",c,"um=",um,"n=",n1
pause
endif
ave2=ave
cc=c
Sh=shh-shh1
cad="Sh"
call vari(Sh,ave,cad,c)
if (c < = 0.d0 .or. ave==0.d0) then
write (*,*) "The result is","cc=",cc,"shh=",&
ave2,"um=",um,"n",n1-1,"error=",abs(Sh)
stop
else
write (*,*) "Shh=",ave2,"cc=",cc,"um=",um,"n=",n1-1,"error=",abs(Sh)
pause
shh1=shh
endif
nn1=n1
enddo
write (*,*) "Shh=",ave2,"cc=",cc,"um=",um,"n=",nn1-1,"error=",ave
print *,"**************************************"
pause
goto 1
end
```

```
!*****************************************************************
! Open Gauss-Chebychev integration rule by using the CADNA library
integer :: n
Type (double_st) :: x(100),In,Inn,sum,f
open(23,file='pgcheb.res')
call cadna_init(0)
n=1
pi=4.d0*atan(1.d0)
Inn=0.d0
In=0.d0
do while (In==Inn)
n=2*n
do i=1,n
x(i)=cos((2*i-1)*pi/real(2*n))
enddo
sum=0.d0
do i=1,n
sum=sum+f(x(i))
enddo
In=(pi/n)*sum
write (23,*) "The approximate value for the integral of the function in the &
& interval [-1,1] is In=",str(In),"n=",n
Inn=In
enddo
```

```
end
function  f(x)
use cadna
type (double_st) :: x,f
f=(1+x)**(1.5d0)
end
!*********************************************************
! The use of the stochastic arithmetic in order to calculate
! definite integral of a function by using Trapezoidal rule"
integer,parameter :: n=3
character(5) :: cad
integer :: n1,M
real(4) :: a,b,aa(n),bb(n),hh(n),Thh(n),Th(n),Thh1(n),ssumm(n),x(n)&
,f(n),eps,c,ave
write (*,*) "please enter eps "
read (*,*) eps
!********************************* ! the limits of integral
a=0.
b=1.
!*********************************
n1=0
Thh1=0.
call rand(a,aa,eps)
call rand(b,bb,eps)
```

```
1 M=2**n1
hh=(bb-aa)/(M)
cad="hh"
call vari(hh,ave,cad,c)
ssumm=0.
do i=1,M-1
x=aa+i*hh
cad="x"
call vari(x,ave,cad,c)
! ******* introducing a function *******
f=exp(-x**2)
!****************************************
cad="f"
call vari(f,ave,cad,c)
ssumm=ssumm+2.*f
cad="ssumm"
call vari(ssumm,ave,cad,c)
enddo
!*****************************************
Thh=hh/2*(exp(-aa**2)+ssumm+exp(-bb**2))
cad="Thh"
call vari(Thh,ave,cad,c)
ave1=ave
c1=c
```

```
!************************************
print *,"Thh=",ave,"c=",c
if (c < = 0 .or. ave==0.) then
write (*,*) "The Integral does not exit or the solution is not stable"&
,"c=",c,"Thh=",ave
stop
else
Th=Thh1-Thh
cad="Th"
call vari(Th,ave,cad,c)
print *,"Th=",ave,"n=",n1
if (c < = 0. .or. ave==0.) then
write (*,*) "The approximate value for the integral of the function in the &
& interval [a,b] is=",ave1 ,"c=",c,"n=",n1
else
Thh1=Thh
print *,"n=",n1,"Thh1=",ave1,"c=",c,"c1=",c1,"**Th=In-I",ave
n1=n1+1
pause
goto 1
endif
endif
end
!*****************************************
```