Journal of Applied Computer Science Methods
Published by University of Social Sciences

ACSM

Volume 7 Number 1 2015
University of Social Sciences, IT Institute
Społeczna Akademia Nauk
Łódź
Instytut Technologii Informatycznych
ISSN 1689-9636

International
Journal of Applied Computer Science Methods

Journal of Applied Computer Science Methods

Published by University of Social Sciences

ACSM

Volume 7 Number 1 2015

University of Social Sciences, IT Institute

SPOŁECZNA AKADEMIA NAUK
ŁÓDŹ

Instytut Technologii Informatycznych
iti

INTERNATIONAL JOURNAL OF APPLIED COMPUTER SCIENCE METHODS (JACSM) is a semi-annual periodical published by the University of Social Sciences (SAN) in Lodz, Poland.

PUBLISHING AND EDITORIAL OFFICE:
University of Social Sciences (SAN)
Information Technology Institute (ITI)
Sienkiewicza 9
90-113 Lodz
Tel.: +48 42 6646654
Fax.: +48 42 6366251
E-mail: acsm@spoleczna.pl
URL: http://acsm.spoleczna.pl

Print: Mazowieckie Centrum Poligrafii, ul. Duża 1, 05-270 Marki, www.c-p.com.pl, biuro@c-p.com.pl

AIMS AND SCOPE:

The **International Journal of Applied Computer Science Methods is** a semi-annual, refereed periodical, publishes articles describing recent contributions in theory, practice and applications of computer science. The broad scope of the journal includes, but is not limited to, the following subject areas:
Knowledge Engineering and Information Management: *Knowledge Processing, Knowledge Representation, Data Mining, Machine Learning, Knowledge-based Systems, Knowledge Elicitation, Knowledge Acquisition, E-learning, Web-intelligence, Collective Intelligence, Language Processing, Approximate Reasoning, Information Archive and Processing, Distributed Information Systems.*
Intelligent Systems: *Intelligent Database Systems, Expert Systems, Decision Support Systems, Intelligent Agent Systems, Artificial Neural Networks, Fuzzy Sets and Systems, Evolutionary Methods and Systems, Hybrid Intelligent Systems, Cognitive Systems, Intelligent Systems and Internet, Complex Adaptive Systems.*
Image Understanding and Processing: *Computer Vision, Image Processing, Computer Graphics, Pattern Recognition, Virtual Reality, Multimedia Systems.*
Computer Modeling, Simulation and Soft Computing: *Applied Computer Modeling and Simulation, Intelligent Computing and Applications, Soft Computing Methods, Intelligent Data Analysis, Parallel Computing, Engineering Algorithms.*
Applied Computer Methods and Computer Technology: *Programming Technology, Database Systems, Computer Networks Technology, Human-computer Interface, Computer Hardware Engineering, Internet Technology, Biocybernetics.*

DISTRIBUTION:

Apart from the standard way of distribution (in the conventional paper format), on-line dissemination of the JACSM is possible for interested readers.

Contents

APPLICATION OF NUMERICAL LAPLACE INVERSION METHODS IN CHEMICAL ENGINEERING WITH MAPLE®

Małgorzata Wójcik[1], Mirosław Szukiewicz[1], Paweł Kowalik[2]

[1]Department of Chemical and Process Engineering
Rzeszow University of Technology, Rzeszow, Poland
wojcik.mm@op.pl, ichms@prz.edu.pl

[2]Institute of New Chemical Synthesis, Pulawy, Poland
pawel.kowalik@ins.pulawy.pl

Abstract

In this paper, the procedure of choice of the effective numerical inversion method of Laplace transform for solving of the mathematical model of gas flow was shown. Two inversion methods are evaluated in this work: the Gaver-Stehfest method, and the Dubner and Abate method. The numerical tests has been made using test functions. Based on the test results the fast and accurate method was selected and applied for modeling of the gas flow. The Gaver-Stehfest method has proven the most effective. The model was developed using the computer algebra system Maple®. Implementation of the methods were done by authors.

Key words: numerical inversion of Laplace transform, gas flow model, Maple® program

1 Introduction

The Laplace transformation plays a significant role in mathematical applications in physics, mechanics, mathematics, economy, and computational sciences. It is a powerful tool for solving and analyzing of linear differential and linear integral equations. It is applicable in many areas of engineering mathematics and engineering systems e.g. for fluid flow and mass and energy transport in porous media were reported. Using of Laplace method technique greatly simplifies the solution of models representing sophisticated physical systems [1, 2, 3, 4]. The inverse transform of a transfer function can be made by employing properties of Laplace transform and using tables of Laplace transform. A recent alternative are Computer Algebra Systems. Unfortunately, often it is impossible to find an inverse Laplace transform function analytically. Then one can need a numerical approach, but it was reported that numerical inversion of Laplace transform can by numerically instable. On the other hand successful applications were also presented. For example, Chiang [5] suggested that best results, for solute the transport problems with groundwater, one can obtain by employing – the Crump method. Kocabas [6] presented two procedures of inverse Laplace transform, the Gaver-Stehfest method and the Dubner and Abate method to modeling of tracer transport in heterogeneous porous media. Author showed that the Dubner and Abate method was in this case the most powerful but also the most computationally expensive, while the Gaver-Stehfest method found the exact solution only for highly dispersive systems. Many scientists emphasized that the efficiency and accuracy of numerical Laplace inversion methods are different for various types of functions. Hassanzadeh and Pooladi-Darvish [7] found that the Gaver-Stehfest method can be applied for exp(-t) type of functions but this method fails when the solution is like exp(t), sinusoid and wave type of functions. The presented facts suggest that some algorithms can work best for certain types of problems, and they can fail for other types of problems. So, application of the method should be preceded by investigations.

In this work, we have undertaken an attempt to use numerical methods for inverse Laplace transform to solve a mathematical model related with gas flow through the measuring system. We examined, following Kocabas, two methods for the numerical inversion of transform function: the Gaver-Stehfest method, the Dubner and Abate method. The accuracy and computational efficiency of methods will be tested for their applicability for gas flow modeling. Methods was implemented in the system Maple®.

The paper is organized as follows. In section 2 we show information about the numerical inversion Laplace methods. In Section 3, we examine the accuracy of the methods. The method chosen is examined for defined problem in Section 4. A discussion of the results and conclusions are presented in Section 5.

2 The numerical methods

There are many numerical inversion methods for Laplace transform. Each method has its own applications and is suitable for various problems in many domains, such as geology, mathematics, circuit theory, process control, and medicine. There are four main groups: (i) the Fourier series method, which is based on the Poisson summation formula, (ii) the Gaver-Stehfest algorithm, which is based on combinations of Gaver functional, (iii) the Weeks method, which is based on bilinear transformations and Laguerre expansions, (iv) the Talbot method, which is based on deforming the contour in the Bromwich inversion integral [8]. Based on results obtained by Kocabas [6] (similarity of problem considered by him), the Gaver-Stehfest method and the Dubner and Abate method were chosen for further tests.

2.1 The Gaver-Stehfest method

This method was developed in the late 1960s. It is very simple numerical inverse Laplace transform method which has been used in such diverse areas as chemistry, economics, mathematics, computational physics and engineering. This method is also employed for fluid flow problems [5, 6, 7, 9].

This method approximates the time domain solution as [10]

$$f(t) = \frac{ln\,2}{t}\sum_{k=1}^{N} V_k \cdot F(k \cdot \frac{ln\,2}{t}) \tag{1}$$

where V_k is described by the following equation

$$V_k = (-1)^{k+\frac{N}{2}} \sum_{j=(\frac{k+1}{2})}^{min(k,\frac{N}{2})} \frac{j^{\frac{N}{2}}(2j)!}{\left(\frac{N}{2}-j\right)!\,j!\,(j-1)!\,(k-j)!\,(2j-k)!} \tag{1a}$$

The parameter N is the number of terms used in Eq. (1). N must be an even integer and should be chosen by trial and error method. The precision of this method depends on the parameter N. If N rises, accuracy of results, increases first and then the accuracy declines due to round-off errors [7]. Many authors propose a different value of the parameter N to obtain the best accuracy of this method. For example, Cheng and Sidauruk suggested that optimal choice of N should be in range from 6 to 20 [10], while Knight and Raiche recommended that N should be approximately equal to the number of decimal digits used by the computer in the calculation [11].

2.2 The Fourier series method

There are about 40 methods based on the Fourier series. These methods were first used by Dubner and Abate in 1968. Next, Durbin modified the method in 1973. The Fourier series technique for the Laplace inversion is based on choosing the contour of integration in the inversion integral, next converting the inversion integral into the Fourier transform, and finally approximating the transform by a Fourier series. The Fourier series method proposed by Dubner and Abate [9], is given by

$$f(t) = \frac{2e^{at}}{T}\{\frac{1}{2}Re\big(F(s)\big) + \sum_{k=1}^{N} Re\left(F\left(a + \frac{k\pi i}{T}\right)\right)cos\left(\frac{k\pi t}{T}\right)\} \tag{2}$$

T is the time interval; *Re* means the real part of a complex function.

It is known that the infinite series in equation converges very slowly which is the main obstacle in the efficient evaluation of the series.

The parameter *a* is chosen as

$$a = \alpha - \frac{ln\,E}{2T} \tag{3}$$

where *E* is the error tolerance, *α* is the real part of the leading pole of the function F(s) [10]. Lee et al. suggested values of *at* between 4 and 5 [7].

3 Numerical examples

This section presents results obtained for test functions. The numerical results were compared with the analytical solution in the real time domain. Different types of mathematical function was used for tests. The function with Example A is an periodic function. The function with Example B is an rapidly decreasing exponential function. The presence of this type functions of is expected in the solution of gas flow model. All numerical computations were performed using Maple 17 program. Calculations were made with precision up to 48 decimal digits. The results are presented in Tables.

3.1 Example A

In this example, we discuss the numerical inversion of Laplace transform for equation

$$\mathcal{L}\{f(t)\} = F(s) = \frac{1}{s^2 + s + 1} \tag{4}$$

The analytical form of inverse Laplace transform of (4) is

$$f(t) = \frac{2}{3}\sqrt{3}\cdot e^{-\frac{1}{2}t}\cdot sin(\frac{1}{2}\sqrt{3}\cdot t) \tag{5}$$

The results are presented in Table 1, 2 and 3. For example, 5.335071E-01 means $5.335071 \cdot 10^{-1}$.

Table 1. Exact and numerical solution for the function (4) for the Gaver-Stehfest method.

t	Approximate solution			Exact solution
	N=18	N=30	N=32	
1	5.335071E-01	5.335071E-01	5.335071E-01	5.335071E-01
2	4.192796E-01	4.192796E-01	4.192795E-01	4.192796E-01
3	1.332426E-01	1.332426E-01	1.332426E-01	1.332426E-01
4	-4.952987E-02	-4.952987E-02	-4.952987E-02	-4.952987E-02
5	-8.794241E-02	-8.794242E-02	-8.794242E-02	-8.794242E-02
6	-5.089245E-02	-5.089231E-02	-5.089229E-02	-5.089231E-02
7	-7.643044E-03	-7.643713E-03	-7.643737E-03	-7.643713E-03
8	1.271483E-02	1.271509E-02	1.271522E-02	1.271509E-02
9	1.280100E-02	1.280467E-02	1.280464E-02	1.280467E-02
10	5.387351E-03	5.385480E-03	5.385522E-03	5.385480E-03

Table 2. Exact and numerical solution for the function (4) for the Dubner and Abate method.

t	Approximate solution			Exact solution
	N=500	N=1000	N=10000	
1	5.335231E-01	5.335031E-01	5.335071E-01	5.335071E-01
2	4.193116E-01	4.192715E-01	4.192795E-01	4.192796E-01
3	1.332906E-01	1.332305E-01	1.332425E-01	1.332426E-01
4	-4.946593E-02	-4.954597E-02	-4.953004E-02	-4.952987E-02
5	-8.786249E-02	-8.796254E-02	-8.794262E-02	-8.794242E-02
6	-5.079641E-02	-5.091646E-02	-5.089256E-02	-5.089231E-02
7	-7.531829E-03	-7.671887E-03	-7.643997E-03	-7.643713E-03
8	1.284295E-02	1.268289E-02	1.271477E-02	1.271509E-02
9	1.294850E-02	1.276844E-02	1.280430E-02	1.280467E-02
10	5.545293E-03	5.345234E-03	5.385075E-03	5.385480E-03

Table 3. Comparison of numerical calculations for the function (4)

The Gaver-Stehfest method			
	N=18	N=30	N=32
Standard deviation	5.90684E-07	3.7104E-11	8.4732E-09
Time of calculations [s]	0.219	0.516	0.531
The Dubner and Abate method			
	N=500	N=1000	N=10000
Standard deviation	4.4350E-05	1.1168E-05	1.1245E-07
Time of calculations [s]	30.875	66.125	1116.046

Tests of the Gaver-Stehfest method were made for many values of parameter N. Accuracy of the method and time of computations increase as N increases up to N=30. For larger value of N accuracy drops. For N=30 time of calculations was acceptable.

Tests of the Dubner and Abate method show that accuracy of computations is lower than in previous tests. Similar (but still lower) accuracy as the Gaver-Stehfest method was obtained for N=10000 for which time of computations was approximately 2000 times longer (about 20 minutes). Obtained results show that for case A better method of solution finding is the Gaver-Stehfest method.

3.2 Example B

In this example, both the Gaver-Stehfest method and the Dubner and Abate method are used to calculate the inverse Laplace transformation of equation

$$\mathcal{L}\{f(t)\} = F(s) = \frac{1}{\sqrt{s} + \sqrt{s+1}} \tag{6}$$

The analytical inverse Laplace transform of (6) is

$$f(t) = \frac{1 - e^{-t}}{\sqrt{4\pi t^3}} \tag{7}$$

The results are presented in Table 4, 5 and 6.

Table 4. Exact and numerical solution for the function (6) the Gaver-Stehfest method

t	Approximate solution			Exact solution
	N=10	N=28	N=30	
1	1.783179E-01	1.783179E-01	1.783179E-01	1.783179E-01
2	8.623782E-02	8.623782E-02	8.623798E-02	8.623782E-02
3	5.158626E-02	5.158626E-02	5.158626E-02	5.158626E-02
4	3.461599E-02	3.461600E-02	3.461601E-02	3.461600E-02
5	2.506131E-02	2.506131E-02	2.506131E-02	2.506131E-02
6	1.914654E-02	1.914654E-02	1.914655E-02	1.914654E-02
7	1.521782E-02	1.521779E-02	1.521779E-02	1.521779E-02
8	1.246279E-02	1.246276E-02	1.246274E-02	1.246276E-02
9	1.044669E-02	1.044666E-02	1.044666E-02	1.044666E-02
10	8.920229E-03	8.920215E-03	8.920219E-03	8.920215E-03

Table 5. Exact and numerical solution for the function (6)
the Dubner and Abate method

T	Approximate solution			Exact solution
	N=500	N=1000	N=10000	
1	5.335231E-01	5.335031E-01	5.335071E-01	5.335071E-01
2	4.193116E-01	4.192715E-01	4.192795E-01	4.192796E-01
3	1.332906E-01	1.332305E-01	1.332425E-01	1.332426E-01
4	-4.946593E-02	-4.954597E-02	-4.953004E-02	-4.952987E-02
5	-8.786249E-02	-8.796254E-02	-8.794262E-02	-8.794242E-02
6	-5.079641E-02	-5.091646E-02	-5.089256E-02	-5.089231E-02
7	-7.531829E-03	-7.671887E-03	-7.643997E-03	-7.643713E-03
8	1.284295E-02	1.268289E-02	1.271477E-02	1.271509E-02
9	1.294850E-02	1.276844E-02	1.280430E-02	1.280467E-02
10	5.545293E-03	5.345234E-03	5.385075E-03	5.385480E-03

Table 6. Comparison of numerical calculations for the function (6)

The Gaver-Stehfest method			
	N=10	N=28	N=30
Standard deviation	7.8655E-09	1.4790E-11	2.0181E-08
Time of calculations [s]	0.219	0.641	0.734
The Dubner and Abate method			
	N=500	N=1000	N=10000
Standard deviation	1.0812E-02	7.6397E-03	2.4141E-03
Time of calculations [s]	91.484	185.875	3193.500

Tests of the Gaver-Stehfest method was applied for many values of parameter N. Accuracy of the method and time of increase as N increases up to N=28. For next value of N accuracy drops. For N=28 time of calculations was acceptable (t=0.641s) thus N=28 was an optimal value of parameter N for Example B.

Tests of the Dubner and Abate method show that accuracy of computations is lower than in previous tests. Similar (but still lower) accuracy as the Gaver-Stehfest method was obtained for N=10000 for which time of computations was approximately 4500 times longer (about 50 minutes). Obtained results show that for case B better method of solution finding is the Gaver-Stehfest method.

The tests performed showed that in investigated cases faster and more accurate procedure is the Gaver-Stehfest algorithm and this method will be used for a real measurement system.

4 Results

In the previous section, we have chosen the Gaver-Stehfest algorithm as the best method. Now, we will test, whether the Gaver-Stehfest method can be used for solving of the model of real gas flow process. The measurement unit, its application and the model of the unit were described in details in [1]. Founded there The final form of the model in Laplace domain is eq. (8). The proper n1...n4 values were determined by trial and error method in [1].

$$c_{4,n4} = \frac{1}{s} \cdot \left[\left(\frac{q}{n4 \cdot V_{c4} \cdot s + q} \right)^{n4} \left(\frac{q}{n3 \cdot V_{c3} \cdot s + q} \right)^{n3} \left(\frac{q}{n2 \cdot V_{c2} \cdot s + q} \right)^{n2} \cdot c_{in} + \left(\frac{q}{n4 \cdot V_{c4} \cdot s + q} \right)^{n4} \cdot c_{in} \right] \cdot e^{-t_d s} \qquad (8)$$

where:

$c_{4,n4}$ – outlet concentration of gas, measured by TCD detector
q – gas flow [dm^3/min]
c_{in} – gas concentration in inlet to zone 0 [mol/dm^3]
V_{ck} – volume of a single cell in *k*-th cell [dm^3]
$c_{j,k}$ – gas concentration in *j*-th zone, in cell *k* [mol/dm^3]
t – time [min]
t_d – time delay
s – complex variable.

Eq. (8) was solved using the Gaver-Stehfest method. Computations were done for following parameters: gas flow – 0.01 dm^3/min. Numbers of cells in each of zones: n1=1, n2=12, n3=1, n4=1. Calculations were carried out using Maple® with precision up to 48 decimal digits. Results are presented on Figure 1.

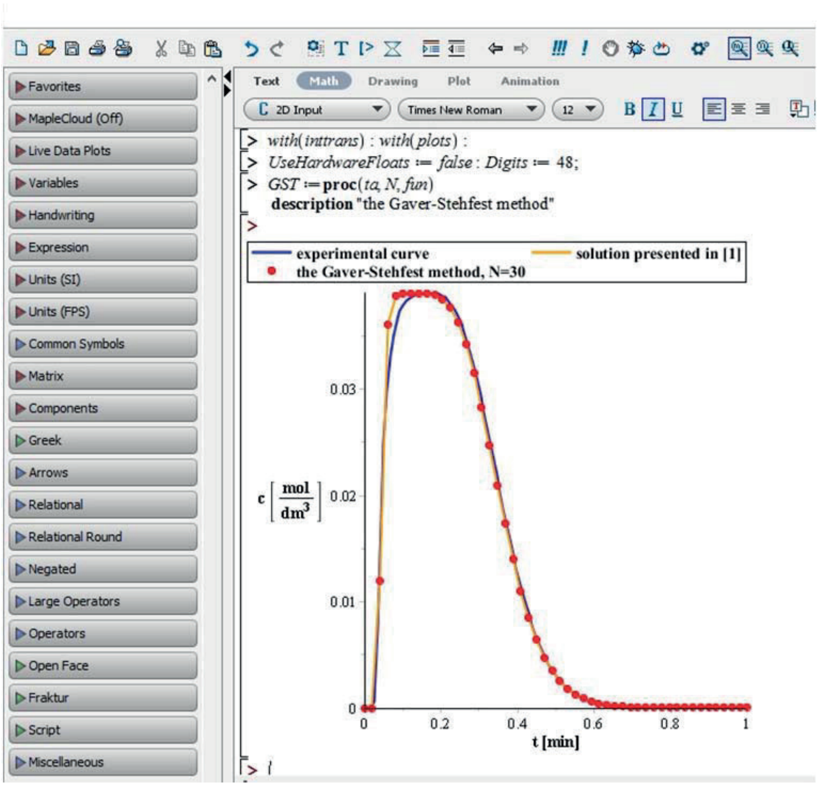


Figure 1. Screenshot of program Maple®. Numerical, theoretical and experimental profiles gas concentrations for 0.01 dm^3/min. The Gaver-Stehfest method

The results will be compared with those presented in [1]. The computations performed showed, that the Gaver-Stehfest method can solve the gas flow problem with high accuracy (minimal standard deviation is equal to 1.7E-07, obtained for N=30) and fast (time of calculations t=1.18s for N=30). The tests presented in section 3 enable to carry out calculations for the gas flow model without numerical problems. Even determining number of terms for Gaver functional seems to be correct, although in investigated case a bit of more accurate results were obtained for N=34. In our opinion it means that value of N should be fitted to the final model (if possible), but results for N=30 (the optimal value of N for test functions) are also satisfactory.

5 Conclusions

On the basis of performed calculations, the following conclusions can be drawn:

1. The Gaver-Stehfest method is very effective algorithm for solution of gas flow problem. It works fast and with high accuracy.
2. The preliminary tests of methods of numerical inverse Laplace transform is purposeful. They are contradictory reports in literature and the tests may allow the elimination of wrong or time-consuming procedures.

Test functions should contain the terms, the presence of which is expected in the problem solution.

Appendices

This work was supported by The National Centre for Research and Development (Poland) under a grant PBS1/A1/6/2012.

References

1. Wójcik M., Szukiewicz M., Kowalik P., 2014, *The application of the Laplace transform for modeling of gas flow using Maple®,* The Journal of Applied Computer Science Methods, vol. 6, pp. 43-53
2. Gaul L., Schanz M., 1999, *A comparative study of three boundary element approaches to calculate the transient response of viscoelastic solids with unbounded domains*, Computational Methods in Applied Mathematics, vol. 179, pp. 111-123
3. Mathur R., Poonia S., 2013, *Application of Laplace transform to Newtonian fluid problems*, Journal of Science and Research, vol. 2, pp. 358-361
4. Coronado M., Ramirez-Sabag J., Valdiviezio-Mijangos O., 2007, *On the boundary conditions in tracer transport model for fractured porous underground formations*, Revista Mexicana De Fisica, vol. 53, pp. 260-269
5. Chiang Li-Wei., 1989, *The application of numerical Laplace inversion methods to groundwater flow and solute transport problems*, New Mexico Institute of Mining and Technology, New Mexico
6. Kocabas I., 2011, *Application of iterated Laplace transformation to tracer transients in heterogeneous porous media*, Journal of the Franklin Institute, vol. 348, pp. 1339-1362
7. Hassanzadeh H., Poolad-Darvish M., 2007, *Comparison of different numerical Laplace inversion methods for engineering application*, Applied Mathematics and Computation, vol. 189, pp. 1966-1981
8. Abate J., Valkó P., 2004, *Multi-precision Laplace transform inversion*, International Journal for Numerical Methods in Engineering, vol. 60, pp. 979-993

9. Wang Q., Zhan H., 2015, *On different numerical inverse Laplace methods for solute transport problems*, Advances in Water Resources, vol. 75, pp. 80-92
10. Cheng A., Sidauruk P., 1994, *Approximate inversion of the Laplace transform*, The Mathematical Journal, vol. 4, pp.76-82
11. Knight J.H., Raiche A.P., 1982, *Transient electromagnetic calculations using the Gaver-Stehfest inverse Laplace transform method*, Geophysics, vol. 47, pp. 47-50

THE BIG DATA MINING APPROACH FOR FINDING TOP RATED URL

J.S.Shyam Mohan[1], P.Shanmugapriya[2], Bhamidipati Vinay Pawan Kumar[3]

[1] Assistant Professor, Dept. Of CSE, SCSVMV University
jsshyammohan@kanchiuniv.ac.in

[2] Associate Professor, Dept. Of IT, SCSVMV University
priya_prakasam@yahoo.co.in

[3] IV-B.E.(CSE), SCSVMV University
vinay90005@gmail.com

Abstract

Finding out the widely used URL's from online shopping sites for any particular category is a difficult task as there are many heterogeneous and multi-dimensional data set which depends on various factors. Traditional data mining methods are limited to homogenous data source, so they fail to sufficiently consider the characteristics of heterogeneous data. This paper presents a consistent Big Data mining search which performs analytics on text data to find the top rated URL's. Though many heuristic search methods are available, our proposed method solves the problem of searching compared with traditional methods in data mining. The sample results are obtained in optimal time and are compared with other methods which is effective and efficient.

Keywords: Big Data, Data Mining

1 Introduction

Information extraction (IE) is the task of automatically extracting structured information from unstructured or semi-structured documents [1]. Unsupervised information extraction is based on training data as they do not require human intervention, they can recursively discover new relations, attributes, and instances in a fully automated, scalable manner [2]. Collecting huge body of information by searching the Web can be a tedious, manual process. Unless you find the "right" document or database, you are reduced to an error-prone and ineffective search. Clustering of Web search results is the basis of information retrieval community since long time in the web [3].

Currently, Big Data processing mainly depends on parallel programming models like MapReduce, as well as providing a cloud computing platform of Big Data services for the public. MapReduce is a batch-oriented parallel computing model. There is still a certain gap in performance with relational databases. Improving the performance of MapReduce and enhancing the real-time nature of large-scale data processing have received a significant amount of attention, with MapReduce parallel programming being applied to many machine learning and data mining algorithms. Data mining algorithms usually need to scan through the training data for obtaining the statistics to solve or optimize model parameters [4].

Efficiency of learning algorithms can be increased by general-purpose parallel programming method which is applicable to a large number of machines based on the simple MapReduce programming model on multicore processors. Ex: weighted linear regression, k-Means, logistic regression, naive Bayes, etc. [5].

As an example, suppose that a customer wants to buy 40 inch LED TV online. Majority of the users shopping online search the product from many websites say like Flipkart, Amazon, Snapdeal, E-Bay,etc. where they can buy the product with affordable price. The search stops until the user gets the product within their budget. Doing this is a difficult process as to find out the right product and with the right price. To overcome this problem, we have focused on making the search easier using Sentimental Key word search by applying data mining techniques over big data in which we have considered the product reviews and comments. Finally Text Data Analytics is performed over the huge data and as final step we will get the top rated and mostly used URL from the entire online shopping websites for the given category.

2 Background and Literature Survey

Thematic locality of the web graph is a directed graph in which nodes are Web pages and edges are hyperlinks in which if hypothetically if page A hyperlinks page B, then the creator of page A intentionally raised the topic of page B in the context of page A which indicates that pages A and B are semantically close [6]. In an average case, if two static Web pages are located in a short proximity to each other in the web graph, then they stand in a semantic relation.

Multi-agent heuristic search algorithms for topologically clustering n pages where collection of n web agents are created and each of which is assigned one source page and each agent maintains a search frontier which consists of list of nodes (URLs) to be expanded. At any search iteration each agent obtains URLs hyperlinked from the nodes of its search frontier and then applies heuristics to select potentially good URLs to become its new search frontier [7]. In the final step the intersects of URLs obtained by all agents. If a common URL is found for two source pages the clusters are merged.

KNOWITALL is an automated system which creates a set of extraction rules for each predicate, consisting of an extraction pattern, constraints, bindings, and keywords using two modules: Extractor and the Assessor. The Extractor creates a query from keywords in each rule, sends the query to a Web search engine, and applies the rule to extract information from the resulting Web pages [8]. The Assessor computes a probability that each extraction is correct before adding the extraction to KNOWITALL's knowledge base.

The Assessor based probability computation on search engine hit is done by an extension of Turney's PMI-IR algorithm [9]. The predicate gives the relation name and class name of each predicate argument.

Harmony Search algorithm (HS) explores optimal word size (Ws) and alphabet size (a) for SAX time series. Harmony search algorithm is an optimization algorithm that generates randomly solutions (Ws, a) and select two best solutions. H SAX algorithm is developed to maximize information, rather than improve classification accuracy [10].

Keyword extraction which is used in on-line text documents which rapidly increase in size with the growth of WWW, has become a basis of several text mining applications such as search engine, text categorization, summarization, and topic detection [11].

A Big Data mining system has to model mining and correlations which are the key steps to ensure that models or patterns discovered from multiple information sources can be consolidated to meet the global mining objective. Global mining can be of two steps viz., local mining and global correlation. By exchanging patterns between multiple sources, new global patterns can be synthetized by aggregating patterns across all sites.

Multi-source data mining algorithms provides a solution for the problem of full search and for finding global models that traditional mining methods cannot find [12].

3 Existing Method For Parsing Reviews

We resort reviews to exterior resources. A large proportion of reviews have false opinion targets viz., "feelings", "time" and "mood", etc., are not domain-specific words and occur frequently in common texts. Therefore, generate a small domain-independent general noun (GN) corpus from web to cover some of the most frequently occurring phrases. Some of the frequently used nouns(say 1000) are extracted in Google's ngram corpus[13] and we add all the nouns in the top three levels of hyponyms in four WordNet (Miller, 1995) synsets "object", "person", "group" and "measure" into the GN corpus.

In total, 3,071 English words are selected for the GN corpus. Employing the GN corpus merely covers a portion of phrases. To increase the coverage, we use a machine learning based technique. For Chinese language we can generate general nouns using HowNet [14]. In different domains, users may use the same words to express their reviews, such as “good”, “worst” and “average”. In addition, some opinion words are domain dependent, such as “spicy” in restaurant reviews or “powerful” in auto reviews. It is difficult to generate a corpus to flag false opinion words by exterior resources and hence negative labeled instances for training a regression model is missing. Thus, make use of exterior manual labeled resources such as SentiWordNet [15] and Hownet Sentiment Word Collection (HSWC)[16] to flag a portion of correct reviews. For English reviews, if a candidate is in SentiWordNet, its prior confidence value in Io is the subjective score (PosScore þ NegScore) annotated in SentiWordNet; otherwise, it is 0. For Chinese reviews, if a candidate is in HSWC, its prior confidence value in Io is 1; otherwise, it is 0.We select one English dataset containing the Customer Review Datasets (CRD), which includes English reviews of five products. CRD was also used in [17], [18]. Reviews are first segmented into sentences or words according to punctuation. Next, sentences are tokenized, with part-of-speech tagged using the Stanford NLP tool. Minipar toolkit is used to parse English sentences [19].

The rest of the paper is organized as follows: Section 4 covers data mining process using Hadoop in detail. Section 5 about the methodology followed, section 6 deals with the experiments and results obtained.

4 Data Mining Process using Hadoop

The flow of events in data mining process is shown in Figure 1.

Flow of Events in Data Mining Process

The process of categorizing the unstructured reviews from the repository can be separated into three parts:

(1) Input unstructured data reviews.(2) Data Mining process using Hadoop (3) the top rated URL under each category is obtained. The flow can be illustrated in Figure 1:

Step 1: Load XML files containing URL’s, reviews, comments (negative, positive) as input.

Step 2: Dump the entire data to MySQL using Sqoop which provides the top rated URL.

Step 3: Top rated URL is obtained after the analytical process done in step 2.

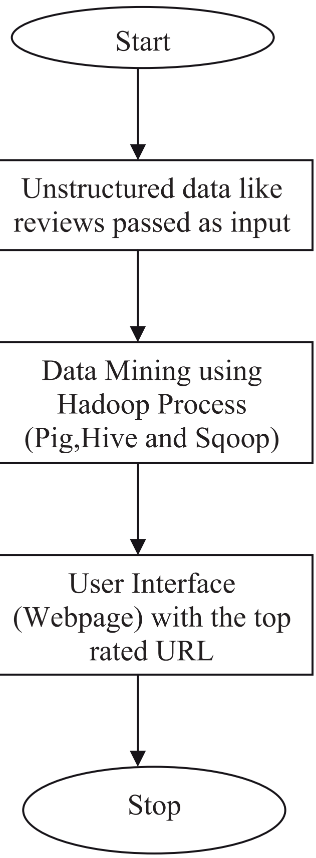


Figure 1. Data Mining process using Hadoop

5 Methodology Of Big Data Mining

5.1 The Map Phase

As discussed in section 4, the input passed is probably unstructured data like reviews. After that Map reduce job reads reviews and decides whether the reviews are good or bad. Using some basic commands, do map reduce job.

Structured Output data with URL, category and ratings is obtained. When the data transforms from MapReduce to HDFS, XML file separates them into four 128 MB files. MapReduce works by breaking the processing into two functions: the map function and the reduce function. Map phase has key-values pairs as input and output. Choose text input format which gives each line in the dataset as a text value, the key is the offset of the beginning of the line from the beginning of the file. Map function is simple where we pull out the category, hash code, URL's, number of positive and negative reviews and number of users. The output from the Map function is processed by MapReduce framework before being sent to reduce function. This processing sorts and groups the key-value pairs by key. The methodology of the entire process is shown in Figure 2.

5.2 The Reduce Phase

After the Map Phase is done successfully, the combiner function is defined using the reducer class which is same to that of the implementation as the reducer function. The files splitted can be seen in Hadoop Name node. Status of MapReduce can be seen in link which will be generated by job tracker. The splitted files goes into the Pig usage.

The MapReduce process shown in Figure 3 illustrates splitting of files and how sorting is done and finally combined using the Reduce method.

5.3 The Pig Phase

LOAD the input file into variable 'a' by parsing. Generate URL category data as 'hash, category'. Store the data and generate URL review data and thereby remove duplicate records. For simplicity, the program assumes that the input is tab-delimiter text ,with each line having just category ,hash code and other as hash code, URL and positive & negative review, total number of reviews.

The load function produces a set of(category ,hash code and other as hash code ,URL and positive & negative review, total number of reviews) tuples that are present in input file and even (hash code, category).Write the tuples which are separated. To explicitly set the number of reducers for each job ,use parallel clause for operation which runs in reduce phase .The data is divided into parts and are separated using pig commands. In the next stage we will get the top URL by positive sentiment using Hive.

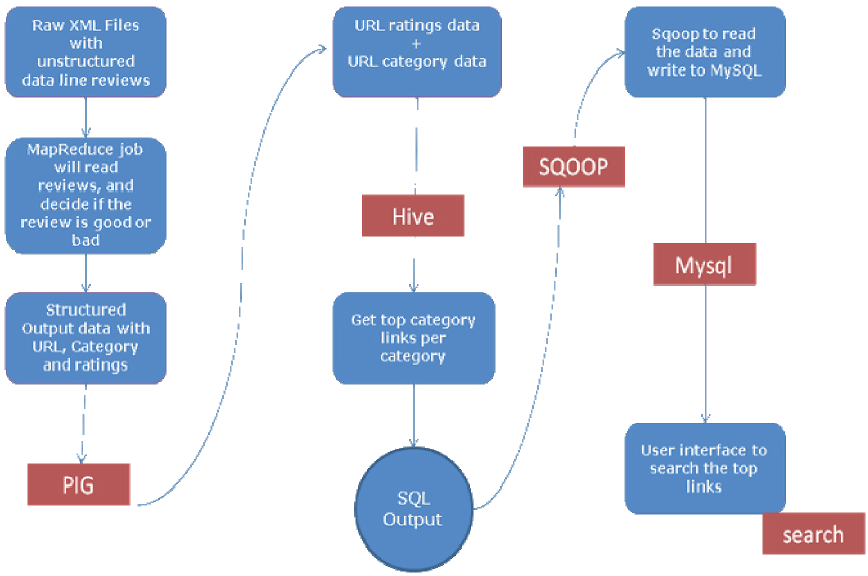


Figure 2. Methodology of Data Mining using Hadoop

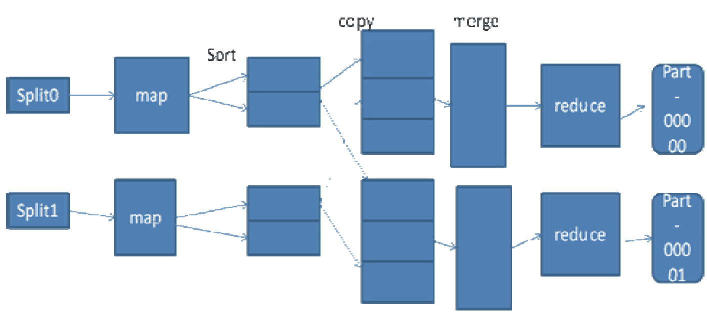


Figure 3. MapReduce Phase

5.4 Hive Phase

Hive is a framework for Data Warehousing which manages huge volumes of data stored in HDFS. Load data into Hive's managed storage using local file system for storage. The sample queries used in performing the entire process is shown in table 1.

Table 1. Sample queries for performing the Hadoop Process

```
1. CREATE TABLE STATEMENT.

2. CREATE EXTERNAL TABLE (any name)
(hash string, category string) ROW
FORMAT DELIMITED FIELDS TERMINATED
BY'/t'

3. CREATE EXTERNAL TABLE (any
name)(hash string, URL string,
positive int, negative int, total
review int) ROW FORMAT DELIMITED
FIELDS TERMINATED BY '\t'
```

5.4.1 Explanation of Table 1

The first line declares a record table with two columns: hash and category. The type of each column must be specified here hash and category are strings.

The second line declares a record table with five columns: hash, URL, positive, negative and total reviews. Column type must be specified along with hash and category as string ,positive, negative and total reviews as integer.

With the EXTERNAL keyword, hive knows that it is not managing the data ,so it doesn't move it to its ware house directory .It doesn't even check whether the external location exists at the time is defined which is a useful feature because it means you can create the data lazily after creating the table. Store Hive tables on local file system directory under Hive's Warehouse and populate the table using Hive buckets.

Each bucket is just a file in the table directory. Infact, buckets correspond to MapReduce output file partition; a job will produce as many as many buckets (output files) as reduce tasks. By typing some logical commands we can find top rated URL as per category.

5.5 Process Of Sqoop

Sqoop is an open source tool that allows to extract the data from a structured data store into Hadoop for further processing. The processing can be done with MapReduce programs or other higher –level tools such as Hive and therefore using Sqoop, data is exported from Hive Meta store to MySQL database. Finally category is given and the top rated URL is obtained under positive sentiment.

6 Empirical Experiments and Analysis

Experimental platform is setup with a Hadoop cluster which considers the following configuration:

– Hadoop cluster, developed on machines with Ubuntu, a 64 bits architecture with maven, libssl-dev, cmake and protobuf libraries;
– HDFS configuration, with dfs.replication set to support fault tolerance;
– Scheduling Load Simulator (SLS) setup (the entire simulator memory or the one for each container, the number of virtual cores, etc.);

It is difficult to collect reviews from online shopping sites for performing analytics and is hard to deploy them in real clusters. What if the developers will have away to just gather the output data of the cluster jobs and then analyze them in a simulator? That is the way Hadoop cluster was designed and implemented which is easier to deploy various and different types of reviews. Note that one can use only one machine to deploy them. In this way, the Hadoop cluster saves a lot of time and, of course a lot of money, because the cost for deploying the same is very less and it is less time consuming.

As shown in Figure 1 and as we previously mentioned, everything runs on a single machine. This means that all Hadoop components should run on the same machine, in the same java virtual machine (JVM), without network component. When jobs are running to perform map or reduce tasks, the all history is kept by the JobHistory daemon. However, this history is not complete, so the test system will utilize additional information.

6.1 Experimental Results

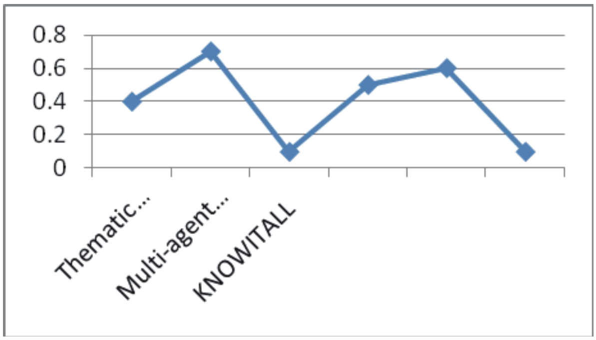
In order to test the accuracy of the proposed methodology, we use LFR[20] dataset which consists of many tasks 50000(S), 50000(B), etc. We took 12 nodes and the first data set say 50000(S) tasks in LFR are used .Some basic Hadoop tools to obtain the workload in the format needed by the simulator using some basic commands which takes as input the XML raw files along with reviews(negative, positive)etc., and generates the jobs traces.

Figure 4 shows the comparisons mentioned in section 2 with existing data mining search techniques. Figure 4a) shows the performance of existing search techniques on LFR 50000(S) dataset. The time shown in the graph is assumed to be in seconds.

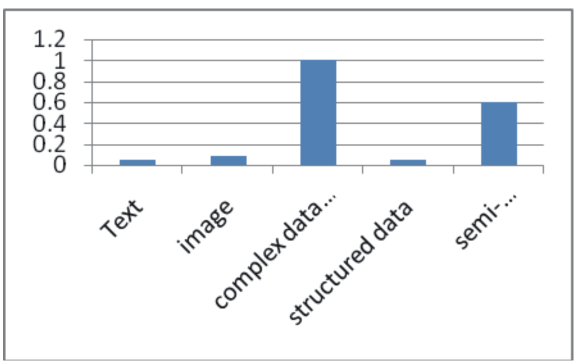
Figure 4b) shows the performance measures of existing data mining techniques on the same dataset.

Figure 5 shows the allocated memory for each node in the cluster. The results correlated with job execution time (Figure 7 b) show that this method is the best one.When a job uses more memory for map or reduce task, the simulator memory will increase. We can clearly see (Figure 5) that the lines marked in Red follow a common strategy regardless of the tasks because they run the job with extra big input.

Figure 6 presents the performance of resource provisioning in Hadoop Cluster.



a) 50000(S) dataset



b) 50000(S) dataset

Figure 4. Comparisons on existing search techniques mentioned in section 2 and performance of Existing Data Mining techniques

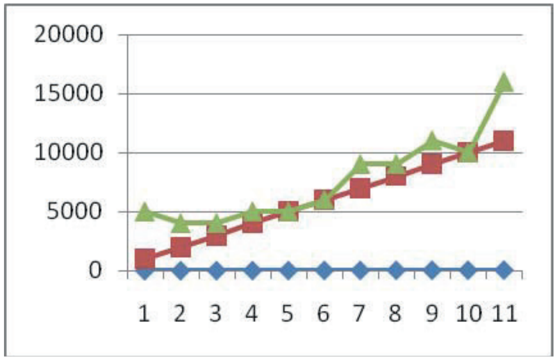


Figure 5. Allocated memory for a node in Hadoop Cluster

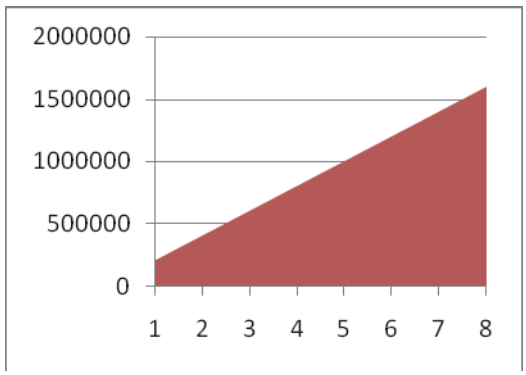
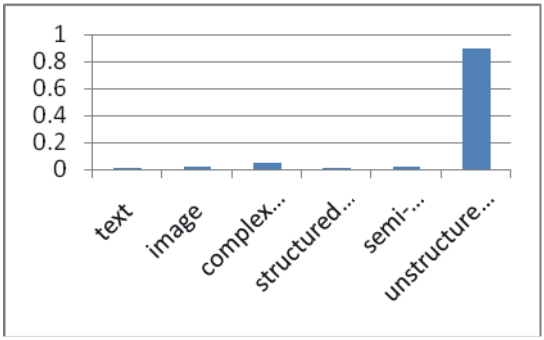
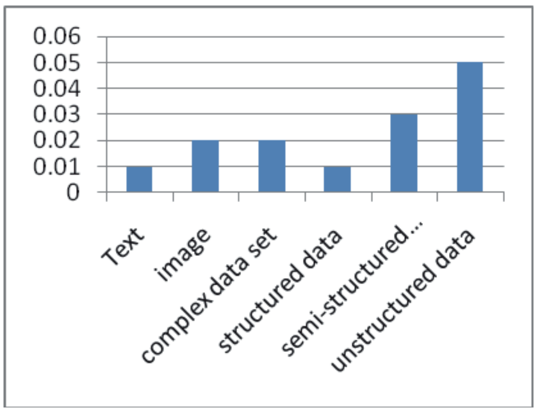


Figure 6. Performance of resource provisioning in Hadoop Cluster



a) 50000(S) dataset



b) Proposed Methodology 50000(S) dataset

Figure 7. Performance comparison of Big data analytics with data mining using Hadoop

In figure 6, the jobs are submitted in the according to a small strategy i.e., those with small input first, then medium input, big and extra big. The jobs are executed in the same order as they are submitted. The scheduler rearranges the jobs because all of them should have access to the resource, even if one has more input.

Figure 7 a) shows the performance comparison of big data analytics on 50000(S)dataset and b) shows the performance of data mining on big data which yields the results in optimal time which increases the search strategy for finding out the top rated and mostly used URL. Note that the time shown in figure 7 is in seconds.

Though many techniques available for searching online, many of them fail to get the results of the search exactly. Therefore our proposed method if implemented in real-time may achieve the optimal results which largely reduces the complexity of searching from unstructured dataset. Figure 8 shows the time costs for our method and the same can be implemented for searching videos online more effectively.

Figure 8 shows the costs comparison between existing data mining techniques with big data mining techniques. The Red colored bar in the graph is the costs estimation for existing search techniques and the green colored bar shows the costs estimation for Big data analytics techniques.

7 Conclusion

Big Data analytics request new methods to meet the demands of applications.To support Big Data mining, high-Performance computing platforms are required, which impose systematic designs to unleash the full power of the Big Data. At the data level, the autonomous information sources and the variety of the data collection environments, often result in data with complicated conditions, such as missing/uncertain values [4]. MapReduce is widely used in big data processing scenarios [21]. Our considerations for the model: all MapReduce jobs are independent and there is no nodes failure before/during the process.

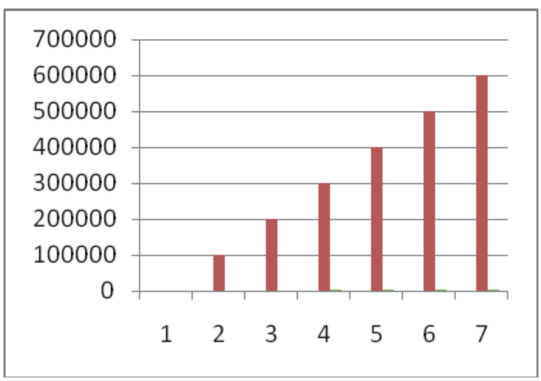


Figure 8. Time costs for data mining using Hadoop

During our experiments, we found some interesting measures such as video analytics which we will consider for our future work.

Appendix

A.1: Pseudo code For Input File

1. Load the XML Input File using Hadoop File Loader

2. Pass the xml data between each of the start/end tags to mapper
3. initiated with default values. //These get overrided by driver
4. Set START and END tags
5. XML Input File created
6. Function override
7. Split Files using XML Record Reader
8. Declare XMLRecordReader to read through a given xml document to output xml
9. hadoop file system data input stream

   // By default the Text input format uses line record reader, which reads the input line by line. But as are defining a new XML input format, in which we have the data in the XML format. So we need to define a new record reader to read the input XML data properly.
10. open the file and get the start/end positions of the split
11. open input file into the file system data input stream
12. if the current position is lesser than end position Then
13. return True
14. End If
15. End Function
16. Function override
17. Return new text
18. Return Long Writable
19. Return Position
20. End Function
21. read the file byte by byte; b becomes -1 at the end of the file;
22. In other cases, b gets a values between 0 to 255

    //save to buffer: This works when we call read Until Match with 'within Block' as true
23. If input read b equals match Then
24. Return True

A.2: Pseudo code For Reading Reviews from URL's

1. Load Hadoop MapReduce Files
2. set the start and end tag to identify and read reviews about each URLs
3. number of reducers is zero; only Mappers are generated
4. Specify the input format type as XML
5. Set Input and Output Paths.

A.3: Pseudo code For XML Parsing Data

1. Load XML Parsing files
2. Input XML File
3. Use the builder parse for XML input source
4. calculate total number of users that have given reviews for a url

/*Returns a Node List of all the Elements in document order with a given tag name and are contained in the document*/

5. Initialize String Tokeniser to get Category tag names
6. Node list generated with Hash Code
7. Return String List

A.4: Pseudo code For Bad words

1. For each String of bad word check if String Bad words Matches with any of the declared bad words
2. If review contains Bad Words
3. Return True
4. End For

References

1. InformationExtraction-http://en.wikipedia.org/wiki/Information_extraction.
2. Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates, 2005, *Unsupervised Named-Entity Extraction from the Web: An Experimental Study*, University of Washington Seattle, WA 98195-2350
3. Hearst and Pedersen, 1996, M. A. Hearst and J. O. Pedersen. *Reexamining the cluster hypothesis: Scatter/gather on retrieval results. In Proceedings of SIGIR*, pages 76–84
4. Xindong Wu, Xingquan Zhu, Gong-Qing Wu, and Wei Ding, *Data Mining with Big Data*, Ieee Transactions On Knowledge And Data Engineering, 2014, Vol. 26, No. 1, pp. 97-107
5. C.T. Chu, S.K. Kim, Y.A. Lin, Y. Yu, G.R. Bradski, A.Y. Ng, and K. Olukotun, 2006, *Map-Reduce for Machine Learning on Multicore,* Proc. 20th Ann. Conf. Neural Information Processing Systems (NIPS ’06), pp. 281-288
6. Ron Bekkerman, Shlomo Zilberstein, James Allan-Web Page Clustering using Heuristic Search in the Web Graph, IJCAI'07, *Proceedings of the 20th international joint conference on Artifical intelligence*, pp. 2280-2285.
7. J. Zhu, H. Wang, B. K. Tsou, and M. Zhu, 2009, *Multi-aspect opinion polling from textual reviews*, in *Proc. 18th ACM Conf. Inf. Knowl. Manage.*, Hong Kong, pp. 1799–1802
8. Kang Liu, Liheng Xu, and Jun Zhao, 2015, *Co-Extracting Opinion Targets and Opinion Words from Online Reviews Based on the Word Alignment Model*, IEEE Transactions On Knowledge And Data Engineering, Vol. 27, No. 3, pp.636-650
9. P. D. Turney, 2001, *Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL*, In Proceedings of the Twelfth European Conference on Machine Learning, pages 491–502, Freiburg, Germany
10. Ahmed, A.M., Bakar, A.A.; Hamdan, A.R., 2011, *Harmony Search algorithm for optimal word size in symbolic time series representation*, DMO, pp. 57 – 62.

11. Sungjick Lee, Han-joon Kim, 2008, *News Keyword Extraction for Topic Tracking, NCM*, Vol. 2, pp. 554 – 559.
12. X. Wu and S. Zhang, 2003, *Synthesizing High-Frequency Rules from Different Data Sources,* IEEE Trans. Knowledge and Data Eng.,vol. 15, no. 2, pp. 353-367
13. http://books.google.com/ngrams/datasets
14. K. W. Gan and P. W. Wong, 2000, *Annotating information structures in chinese texts using hownet*, in Proc. 2nd Workshop Chin. Lang. Process.: Held Conjunction 38th Annu. Meeting Assoc. Comput. Linguistics, Hong Kong, pp. 85–92
15. http://sentiwordnet.isti.cnr.it/
16. http://www.keenage.com/html/c_index.html
17. M. Hu and B. Liu, 2004, *Mining opinion features in customer reviews*, in Proc. 19th Nat. Conf. Artif. Intell., San Jose, CA, USA, pp. 755–760.
18. G. Qiu, L. Bing, J. Bu, and C. Chen, 2011, *Opinion word expansion and target extraction through double propagation*, *Comput. Linguistics,* vol. 37, no. 1, pp. 9–27
19. http://nlp.stanford.edu/software/tagger.shtml
20. Lancichinetti, A., Fortunato, S., Radicchi, F., 2008, *Benchmark graphs for testing community detection algorithms.* Phys. Rev. E 78(4), 046110
21. Songchang Jin, Wangqun Lin, Hong Yin, Shuqiang Yang, Aiping Li,Bo Deng, 2015, *Community structure mining in big data social media networks with MapReduce*, Cluster Comput DOI 10.1007/s10586-015-0452-x,Springer

Performance Analysis Of Vertical Handoff Metrics Using Variance Based Algorithms

Abhijit Bijwe[1], C.G. Dethe[2]

[1] Department of Electronics & Communication. Engineering., Priyadarshini Institute of Engineering & Technology, Nagpur
bijweabhijit@rediffmail.com

[2] UGC-Academic staff College, RTMNU, Nagpur
cgdethe@rediffmail.com

Abstract

Vertical handover is the evolving concept in 4G for seamless communication between heterogeneous networks. In this paper, our main objective is to analyze handover between two WLAN, two Wimax, two UMTS networks. The vertical handover decision is taken based on the variance based algorithm, which calculates the variance of parameters such as delay, jitter, bandwidth and packet loss for various above mentioned networks and the network with most of the parameters with minimum variance is selected. This algorithm is also compared with other algorithms such as MEW (Multiplicative experiment weighting), SAW (Simple Additive Weighting), TOPOSIS (Technique for order preference by similarity to ideal solution) and GRA (Grey Relational Analysis). These algorithms are appropriate for different traffic classes. Simulation results for proposed variance based algorithm in Matlab is discussed and compared with other Multiple attribute decision making algorithm basis of bandwidth; jitter, delay etc. are discussed in the paper. It can be seen that the proposed variance algorithm gives less packet delay than all the algorithms, Jitter is also is least than all the other algorithms.

Key words: SAW, TOPOSIS, GRA, MEW

1 Introduction

In today's scenario, we are having access to various wireless networks such as WLAN, UMTS, Wimax etc. so there is need for switching from one wireless network to other for better services. Vertical handoff can be used for load balancing to get high data rate etc. mechanism can support such switching and maintain the connectivity from one network to another. For this mobile devices with multiple interface card which may help in connecting to different access networks. This is called to vertical handoff which is different from horizontal handoff where the MN moves from one BS to another BS in the same network i.e. homogeneous network. The multiple interface card in mobile node can choose the access links such as wireless local area network (WLAN) IEEE 802.11, worldwide inter-operability for microwave access (Wimax) IEEE 802.16, UMTS network. There should be seamless transfer to the link which is appropriate as per the requirement e.g. for video session high data rate required. Seamless vertical handoff between heterogeneous network is expected.

This can be done by using IEEE 802.21 standard (Media Independent handover) which enables vertical handoff. IEEE 802.21 creates frame work for the protocol needed for vertical handoff. Actual algorithms need to be implemented by the designer. The IEEE 802.21 standard gives link layer intelligence and other network information to higher layers in order to execute handover between different networks such as wi-fi and wi-max. The paper is organized as follows:

- Section 2 explains the importance IEEE 802.21 and how it can be incorporated in ns-2
- Section 3 explains about the multiple attribute decision making MADM algorithm such as TOPOSIS, MEW, SAW, GRA and how it is included at MAC layer so that the vertical handoff decision is taken depending on the traffic classes.
- Section 4 gives the result and conclusion for the above algorithm on the basis of various parameters such as throughput jitter, dropping ratio, delay etc.

2 Importance of media independent handover IEEE 802.21

IEEE 802.21 provides framework for the protocol required for vertical handoff. It facilitates the seamless handover between heterogeneous network supporting different technologies. This standard provides link layer intelligence and related network information to upper layer to optimize vertical handoff[6].

IEEE 802.21 defines three types of services: Information services, Command services, Event services Information services provide multiple types of networks that exist in a given geographical area, command services are used by higher layer to control physical, data link and logical link layers. Event services gives changes in state and transmission behavior of physical, data link, logical link layers.

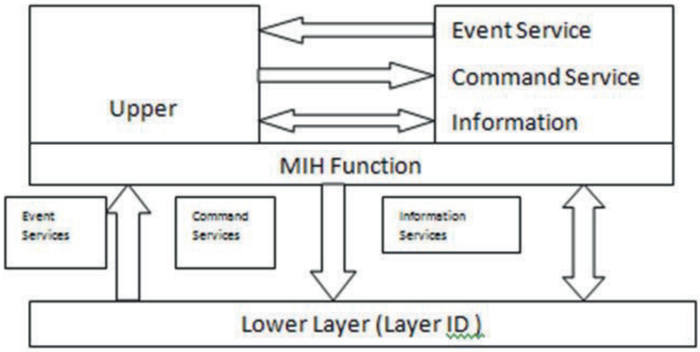


Figure 1. Media Independent handover

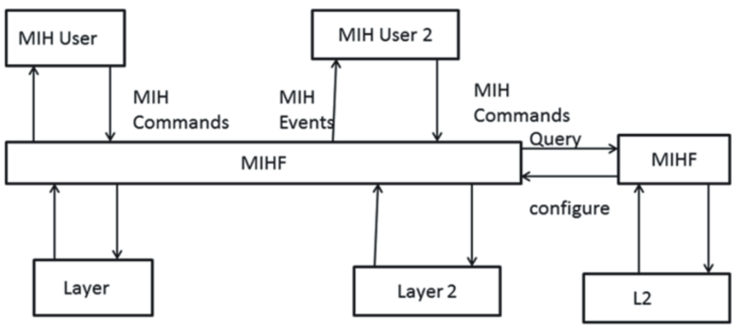


Figure 2. Media Independent handover frame work

3 Proposed variance based algorithm

Proposed algorithm is variance based algorithm which calculates the variance of parameters such as delay, jitter, bandwidth and packet loss for various above mentioned networks and the network with most of the parameters with minimum variance is selected. In our proposed algorithm, handoff metrics such as delay, bandwidth, jitter, packet loss etc. are included.

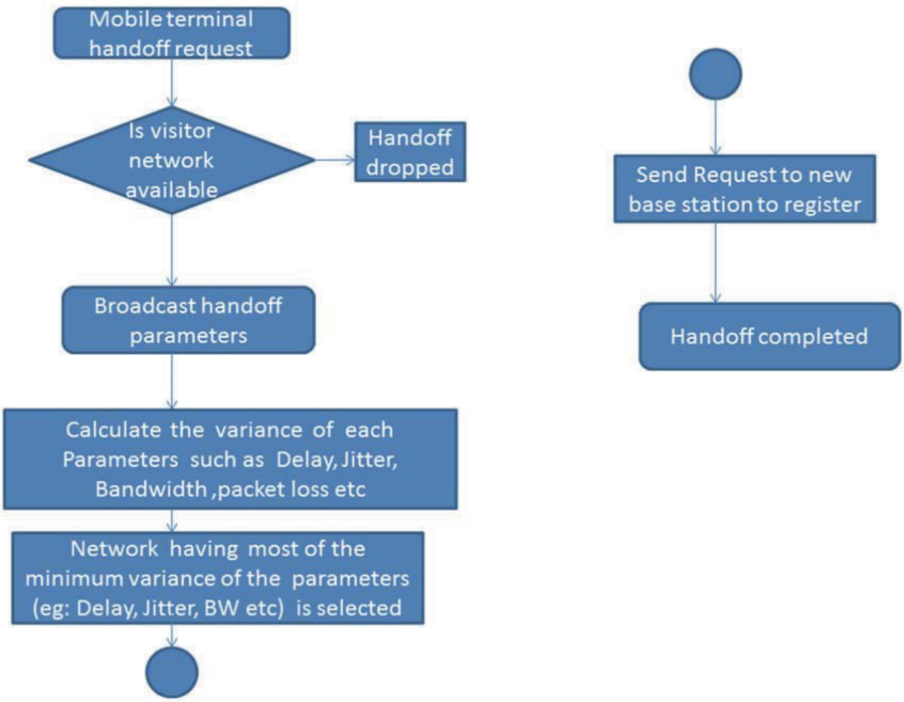


Figure 3. Flow chart of Proposed variance based algorithm

$Variance = \frac{\sum(x-\mu)^2}{N}$ where x is any metrics such as delay, bandwidth, jitter etc and μ is its mean of a set of samples of the particular parameters. N is set of samples.

4 Other madm algorithms used for vertical handover

SAW – simple additive weighting [4]

Simple additive weighting(SAW) is the weighted sum method, where Score of each candidate network is calculated as below:

$$Score_{SAW} = arg\ MAX \sum_{j=1}^{N} [W_j * N_{ij}] \quad (1)$$

where $N_{ij} = P_{ij} / max(P_{ij})$ for benefit parameters (eg: Bandwidth) $N_{ij} = min(P_{ij}) / P_{ij}$ for cost parameters (eg: delay, Jitter). P_{ij} can be any parameter such as Bandwidth, delay, jitter, packet loss etc., with i networks and j attributes.

The Score of each candidate network is found and the score with maximum value is selected.

TOPOSIS – Technique for order preference by similarity to ideal solution [4]

In this algorithm ideal solution is calculated. The selected candidate network will have shortest distance to ideal solution and will have longest distance from worst case solution.

To find the rank of network we this technique follows the step as below:

1. To construct normalized decision matrix consisting of various attributes such as Bandwidth, delay, Jitter, Packet loss etc., $N_{ij} = P_{ij} / \sqrt{\sum_{m}^{i=1} P_{ij}}^2$
2. To Compute weighted normalized matrix by $V_{ij} = N_{ij} * W_j$
3. To find positive and negative ideal solution
   $A+ = \{(maxV_{ij}/j€J), (minV_{ij}/j€J')\}$,
   $A- = \{(minV_{ij}/j€J), (maxV_{ij}/j€J')\}$ where J is a set of benefit parameters and J' is a set of cost parameters
4. Calculate the separation distance between available networks and positive and negative ideal solutions.
   $S^{+} = \sqrt{\sum_{j \epsilon N}(V_{ij} - V_j^{+})^2}$, $S^{-} = \sqrt{\sum_{j \epsilon N}(V_{ij} - V_j^{-})^2}$
5. Compute the closeness to ideal solution $C_i = S^{-}/(S^{\mp} + S^{-})$

$$Score_{TOPSIS} = arg\ max_{i \in M} C_i \tag{2}$$

GRA(Grey relational Analysis)-

It is a technique of grey systems theory (GST**)** invented by Professor ju long Deng. Grey lies between black (no information) and white (full information) which means partial information. It is suitable for unascertained problems with poor information. GRA gives the similarity between the candidate network and best ideal network [4]. Here, grey relational coefficient (GRC) is calculated, before that normalization of data and defining ideal sequence. Normalization of data is done for three situations larger the better, smaller the better, nominal the best.

$$N_{ij} = P_{ij} - min(P_{ij})/(max(P_{ij}) - min(P_{ij}))$$
$$N_{ij} = max(P_{ij}) - P_{ij} /(max(P_{ij}) - min(P_{ij}))$$
$$N_{ij} = 1 - |\ P_{ij} - m_j|/max\{\ max(P_{ij}) - m_j,\ m_j - min(P_{ij})\}$$

m_j is the largest value in the nominal the best situations.

$$\Delta_i = |P_{0j} - N_{ij}|, \Delta_{max} = max_{i \in M, j \in N}\ \Delta_i,\ \Delta_{min} = min_{i \in M, j \in N}\ \Delta_i$$

$$GRC_{\ i} = 1/m \sum_{j=1}^{m} ((\Delta_{max} + \Delta_{min}/(\Delta_i + \Delta_{max}))$$

$$Score_{GRA} = arg\ max_{i \in M} GRC_i \tag{3}$$

MEW –Multiplicative Exponent Weighting [4]
is the weighted product of attributes method, where $SCORE_{MEW}$ is calculated by the best values of attributes in networks.

$$SCORE_{MEW} = arg\ max_{i \in M} \prod_{j=1}^{N} \left[N_{ij}^{W_j} \right] \tag{4}$$

The best algorithm with maximum score is selected for taking vertical handoff decision.

5 Results & Discussion

Performance evaluation of algorithms are done using Matlab simulation, which considers three types of network with two each of the type. The networks considered are UMTS, WLAN, Wimax. The Range of value for various parameters are as follows: Bandwidth for UMTS network 0.1-2Mbps, Packet delay for UMTS network 25-50ms, Jitter for UMTS network 5-10ms. Bandwidth for WLAN network 1-54Mbps, Packet delay for WLAN network 100-150ms, Jitter for WLAN network 10-20ms. Bandwidth for Wimax network 1-60Mbps, Packet delay for Wimax network 60-100ms, Jitter for Wimax network 3-10ms.

Here, we consider three cases as follows: 1) All Parameters are given equal weights 2) Delay, jitter are given 70% weights which suits voice connections 3) Bandwidth is given 70% weights which suits data connections. These cases are applicable to the algorithms where weights are used.

In variance based algorithm, which calculates variance of each parameter and the network having at least two or more than two parameters with minimum variance is selected. In this algorithm weights are not assigned to the parameters.

In Proposed algorithm 10 iterations are taken i.e. handoffs are taken and the network selected in each iteration is also shown in the figure below. Here, 1, 2, 3, 4, 5, 6 represents UMTS1, UMTS2, WLAN1, WLAN2, Wimax1, Wimax2 respectively

In the Simulation, Comparative analysis of various algorithms such as SAW, TOPSIS, MEW, GRA, Variance based algorithms. Various graphs have been plotted on 1. Packet Delay Vs Number of Handoffs, 2. Jitter Vs Number of Handoffs.

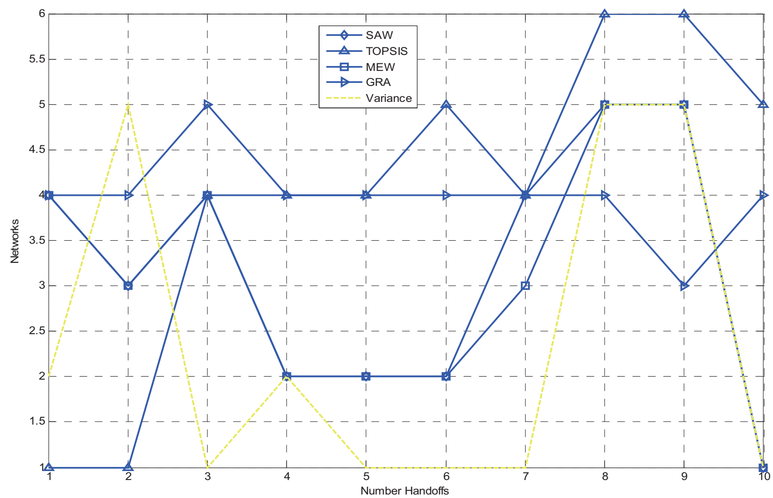


Figure 4. Network selection by the various algorithms

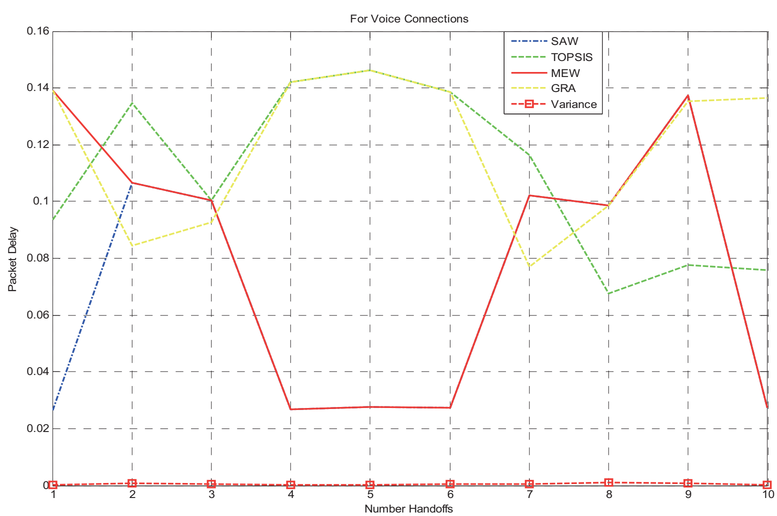


Figure 5. Packet delay vs number of handoffs for various algorithms

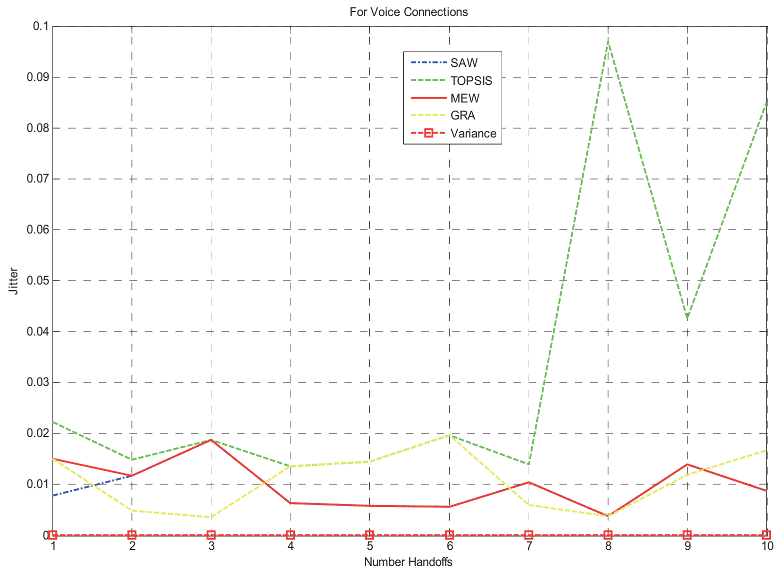


Figure 6. Jitter vs number of handoffs for various algorithms

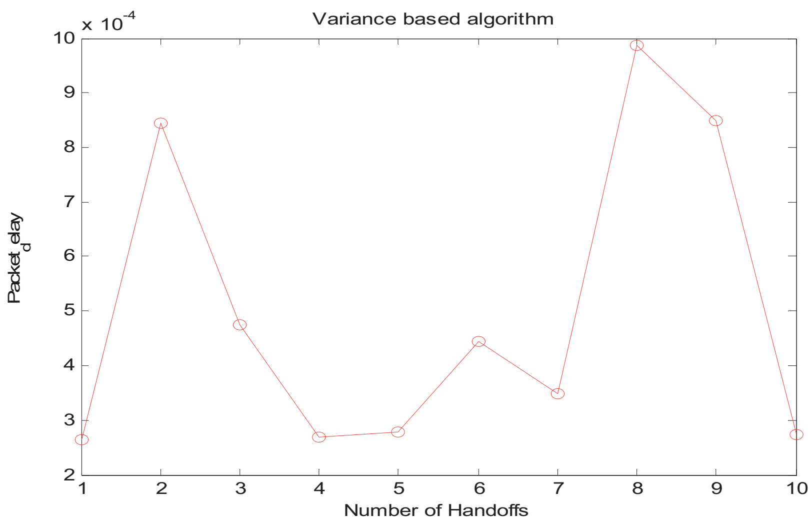


Figure 7. Packet delay vs number of handoffs for variance based algorithms

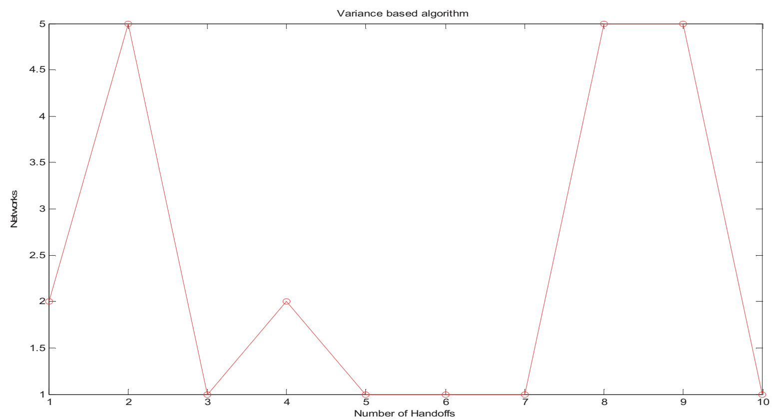


Figure 8. Network selection for variance based algorithms

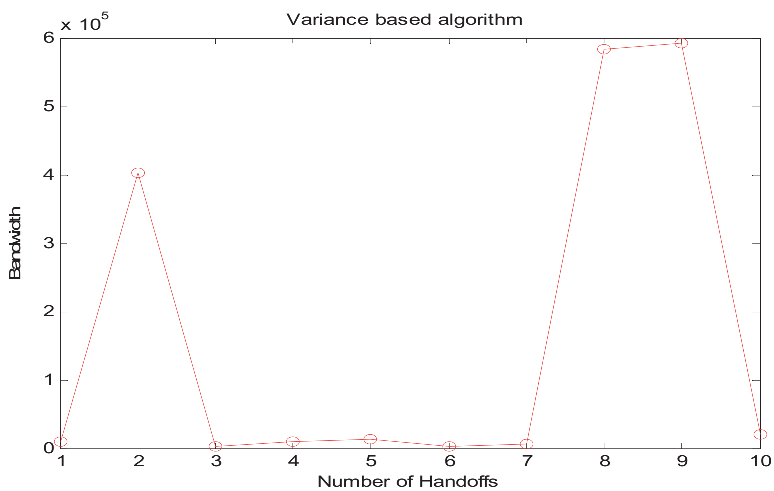


Figure 9. Bandwidth vs Number of handoffs for variance based algorithms

6 Conclusion

It is quite evident from above discussion that, the proposed algorithm has the lowest packet delay than MEW, SAW, GRA, TOPSIS algorithms. Also, it is observed that the proposed algorithm offers least jitter than any other above mentioned algorithms. These two observations makes the proposed algorithm best suited for voice connections. However, the observations obtained are from the simulation results. Hence, it is recommended that for practical scenarios the proposed algorithm can be utilized to have better voice communication.

References

1. T. S. Rappport, 1999, *Wireless Communications: Principles and Practice*, Prentice Hall
2. S. Mohanty and I. F. Akyildiz, 2006, *A cross-layer(Layer 2+3) handover management protocol for next-generation wireless systems*, IEEE Trans. Mobile Computing, vol. 5, pp. 1347– 1360
3. Yu-Chang Chen*, Ja-Hsing Hsia, Yi-Ju Liao, 2009, *Advanced seamless vertical handoff architecture for WiMAX and WiFi heterogeneous networks with QoS guarantees*, Elsevier, pp. 281-293
4. Enrique Stevens-Navarro and Vincent W.S. Wong, 2006, *Comparison between Vertical Handoff Decision Algorithms for Heterogeneous Wireless Networks*, in: IEEE 0-7803-9392-9/06, pp. 947-951
5. Xiaohuan Yan, Ahmet S¸ekercioˇglu, Sathyanarayan, 2010, *A Survey of vertical Decision algorithms in fourth generation Heterogeneous networks,* Elsevier, pp. 1848-1863
6. Hugo Marques, Jos´e Ribeiro, Paulo Marques, and Jonathan Rodriguez, 2010, *Simulation of 802.21 Handovers Using ns-2*, Hindawi Publishing Corporation Journal of Computer Systems, Networks, and Communications, Article ID 794749, 11 pages
7. N. Nasser, S. Guizani, and E. Al-Masri, 2007, *Middleware vertical handover manager: A neural network-based solution.* In Proceedings of the 2007 IEEE International Conference on Communications (ICC'07), pages 5671-5676, Glasgow, Scotland
8. K. Pahlavan, P. Krishnamurthy, A. Hatami, M. Ylianttila, J. P. Makela, R. Pichna, and J. Vallstron, 2000, *Handover in hybrid mobile data networks*, IEEE Personal Communications, 7(2): 34-47

DESIGN AND ANALYSIS OF EDF SCHEDULING ALGORITHM FOR A CHOSEN NETWORK TRAFFIC PROBLEM: IMPLEMENTATION AND SIMULATION ANALYSIS

Jude Ighodaro

Department Of Computer Science
Wroclaw University of Technology, Poland
ighodaro@consultant.com, jude.ighodaro@pwr.edu.pl

Abstract

Workstations, CPU etc are in a period of transition, moving from a relatively slow network links and data oriented services to a high set fibre optic services and more diverse services. As such many of the hard and soft real time will not only demand high bandwidth but also a predictable and guarantee Quality of Service (QoS) along side ,which is not offered in a best effort network (BE). The provision and terms of (QoS) to real-time applications is a key issue in promising broadband packet switched networks and network of other sorts. Applications such as voice and video typically require QoS guarantees in terms of end-to-end data transfer delays. Supporting the diverse and of course, heterogeneous delay requirements of applications with widely varying characteristics requires packet scheduling schemes more sophisticated than First-In-First-Out (FIFO) at each switch in the network(2) are one of most accepted examples of such scheduling schemes. This points to the convincing need for a QoS, which apportion a high network utilization to be realized.In this paper, we focus on the EDF scheduling scheme, since it is known to provide the optimal delay performance (2), in the deterministic environment, and can for this reason be expected to perform well in the real time setting as well. The large amount of bandwidth as promised by the future network also offer the integration of real time applications, hard or soft.

Key words: EDF (Earliest Deadline First), FIFO (First In Fist Out), QoS (Quality Of Service)

1 Introduction

1.1 Overview

Invariably, non of the early computerized network applications were concerned or sensitive to any of the QoS assurances such as the delay that might arise or the data losses as a result of any unprecedented event. Yes, we had internet a number of years ago ,but this type of network was not intended to provide services for those types of data traffic mentioned above, but rather a services that is a called a best effort services. Today the internet offers different types of guarantees or warranties to the different classes of applications using it, these guarantees are called the (QoS) D. Anick, D. Mitra, and M. M. Sondhi.(1982)[1]. The QoS guarantees can be in diverse forms, such as bandwidth, miss ratio in terms of packet loss, total packet delays e.t.c G. de Veciana, C. Courcoubetis, and J. Walrand. CA, (1994) [3]. The applications that convey this type of network, which offers QoS guarantees on the data traffic are of diverse types, some of them are real-time applications in which a rigid QoS guarantees are required from the network, while other applications does not require any guarantees (best-effort traffic),in this regard the network should be capable of providing such guarantees to the various types of traffic that made use of this network. Different technique can perhaps be use to provide this QoS guarantees to the traffic link users.

2 Simulation Results and Performance Analysis

2.1 EDF/FCFS

To verify the performance of EDF and FCFS, we have simulated EDF and FCFS features, In each simulation, logical channels are randomly generated , in our work we assume poison model packet arrival and other assumptions were also made (chapter 2 and 3). The traffic intensity is increased by increasing the number of logical channels. Such simulations are run 100 times to get the average utilization at different traffic loads. Simulations were performed to analyze delay, throughput, acceptance ratio and deadline miss ratio of the HRT and SRT traffic(EDF, FCFS) by varying the intensity of the two real-time traffic classes respectively. Sporadic traffic channels in the simulator are treated as follows. In order to represent and simulate incoming traffic over the whole period, a arbitrary offset is computed that decides in which time slot in each period a certain channel will be requested for transmission. Our channel comprises of increasing packet load and as such shows the behaviour of the system as represented below

2.2 Acceptance ratio

Figure 1 shows how the values of acceptance ratio decreases as the number accepted packets in the requested/available channels increases, it can be seen that at a value of about 40 packets per channels, the acceptance ratio remain about 100%. Which shows that all packets where eventually accepted for transmission. But these ratio decreases as the number of packet requesting transmission increases. Our Queues consist of Soft and hard real time queues, FCFS and EDF. Since the utilization, deadline miss ratio and delay are affected by the accepted ratio, G. de Veciana and G. Kesidis (1996). As a result, a reduction in acceptance ratio is to ensure a more feasible schedule and prioritize hard real time task so that they all meet deadlines.

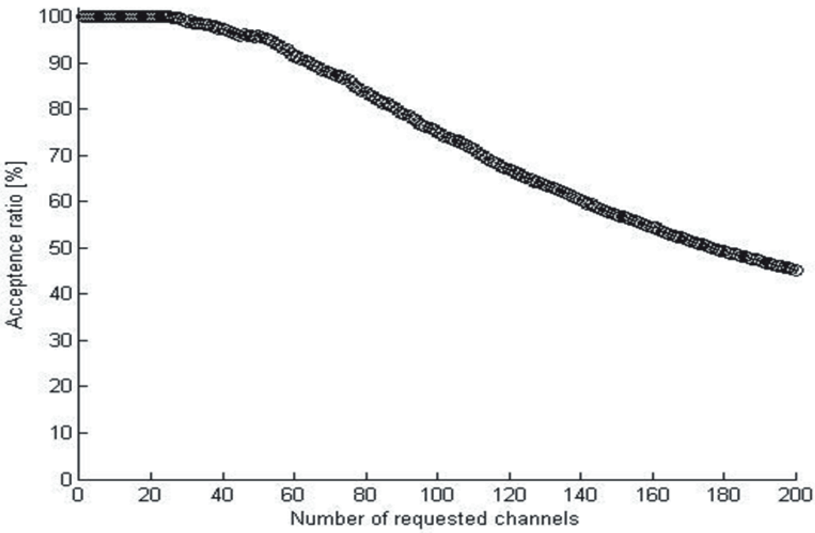


Figure 1. Acceptance ratio

2.3 Utilization (throughput) of the link analysis

By utilization (throughput) we mean an average rate of successful message delivery over a communication channel. This data may be delivered over a physical or logical link, or pass through a certain network node. The throughput is usually measured in bits per second (bit/s or bps), and sometimes in data packets per second or data packets per time slot. Figure 1 shows the throughput of the traffic classes, plotted against the logical channel. As the intensity of the traffic increases, its throughput reaches a maximum of about 0.98 and 15 packets per time slot in the channel at its highest traffic intensity (15 packets per period). Due to its high priority, the throughput of the HRT traffic is not affected and remains feasible and schedulable so that the utilization those not exceed 1. Comparing the throughput of both algorithm, according to our analysis, this figure appears to be the same in that the utilization of the link for both classes must not exceed 100%.

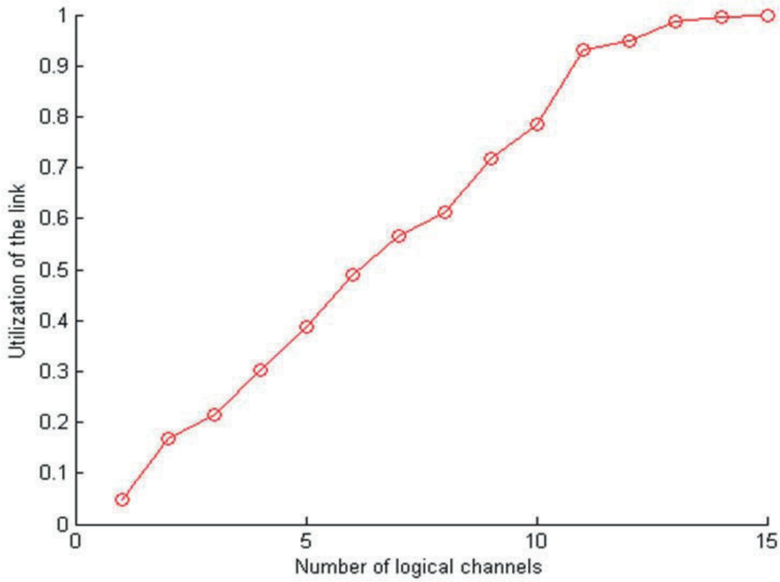


Figure 2. Utilization of the link for DPQ (Dynamic Priority Queue) FCFS

Figures 1 and 2 shows how the values of the deadlines influence the utilization.

In the test result, when the test task workload is less than or equal to 100%, both FCFS and the EDF scheduling algorithms can complete tasks; but the CPU efficacious utilization of the EDF algorithm is higher than FCFS , which can save resources, especially when the task workload closes to 100%. Figure 1 also reveals that our system can reach rather high utilization (close to the theoretical utilization upper bound), i.e., utilization reaches 93% when deadlines are twice the periods (the theoretical utilization bound is 100% in this case).

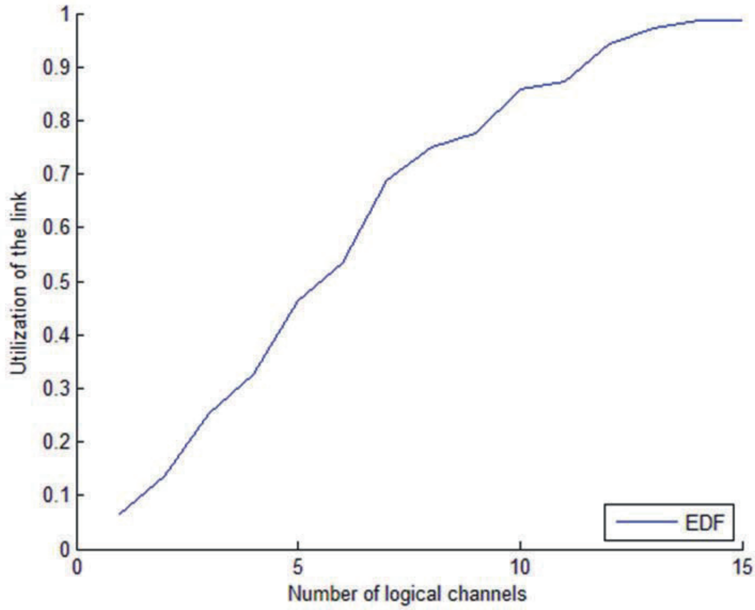


Figure 4. Utilization of the link for EDF DPQ (Dynamic Priority Queue)

The major differences are as follows. Both Figures 3 and 4 visualises the throughput for messages with different deadlines in a network with 4 nodes for a traffic intensity up to 15 packets per node and time slot, i.e. it shows the number of generated packets, which are actually accepted for transmission in each time slot. Independent of the deadline, the throughput of the traffic rises linearly up to a traffic density of 11 and 14 packets per node and time slot for Figures 3 and 4 respectively. The saturation of the network at that point becomes obvious. We can see here that for the static nature of FCFS, the saturation occurred earlier as compared to Dynamic EDF. In that the curves flattens out at a throughput of slightly more than 11 packets per time slot for FCFS. But EDF was able to handle individual packet up to about 14 packets per node before it flatten out and reaches saturation

As expected, throughput varies according to network size. The throughput is proportional to the number of in the system, i.e. proportional to the maximum number of generated packets. Increasing the network size from 8 to 16 nodes for example, results in a doubled throughput; an increase from 8 to 32, leads to a throughput of four times more packets per time slot.

2.4 Average Delay of The link Analysis

By average delay we mean the amount of individual packet delay over the total delay of the overall systems. Figure 4 shows the average delay for EDF queues plotted against the number of channels. In our model as represented below, each channels are made up of 5, 10, 15 packets per slots of the traffic classes. As the traffic load in the system increases, the traffic experiences a slight increase in delay, while the HRT (EDF) traffic remains at a constant average delay of 10 slots per packet throughout the duration of the simulation. Because of EDF queues , its low priority traffic has to wait for the real-time or HRT traffic classes or higher priority to be sent, which increases its delay considerably.

The behaviour of the delay in Figure 4 can be explained accordingly. Basically this is to understanding of mainly EDF queues or HRT queues. Since we just have an EDF queue , With an increased HRT traffic intensity, the traffic class with the highest priority, HRT traffic, is affected the least, while the remaining other traffic types experience a more significant increase in delay. The average delay for the NRT traffic/FCFS queues is not plot in the figure, as the network already is saturated with NRT traffic which would lead to irregularities in the curve due to insufficient statistical data.

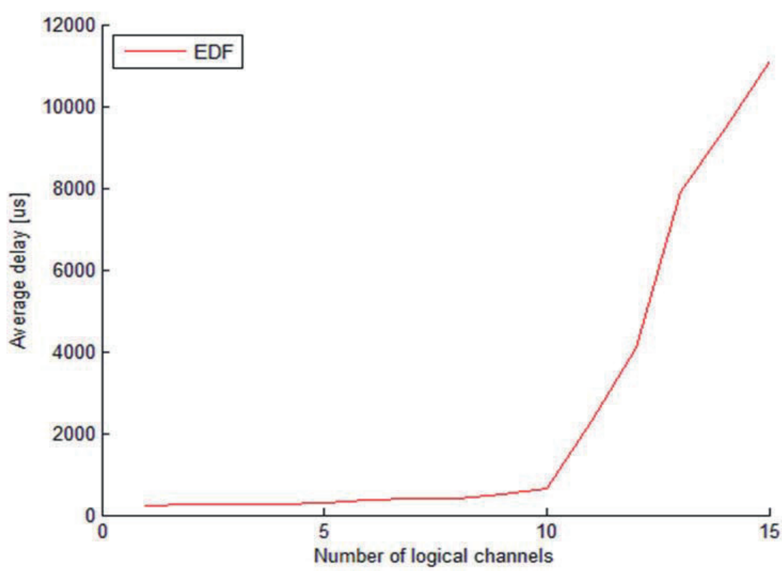


Figure 5. Average delay For DPQ (HRT) EDF

The diagram below contains EDF and FCFS queues. With the average delay plotted against the channels/slots containing increasing traffic load. The behaviour of the delay in Figure 6 below can be explained accordingly. With an increased HRT (EDF queues) traffic intensity, the traffic class with the highest priority, HRT traffic, is affected the least, while the remaining traffic types experience a more significant increase in delay. The average delay for the NRT traffic is now plotted in the figure. As the traffic load in the system increases, the traffic experiences a slight increase in delay, while the HRT (EDF) traffic remains at a constant average delay of 2 packets per slot throughout duration of the simulation. Because of its low priority, the NRT (FCFS) traffic has to wait for the real-time traffic classes to be sent, which increases its delay considerably of the FCFS queues as more packet were enqueued. For high HRT (EDF) traffic intensity, SRT traffic gets starved and hence increases the delay and too few packets are sent to provide statistically reliable results for Figure 5

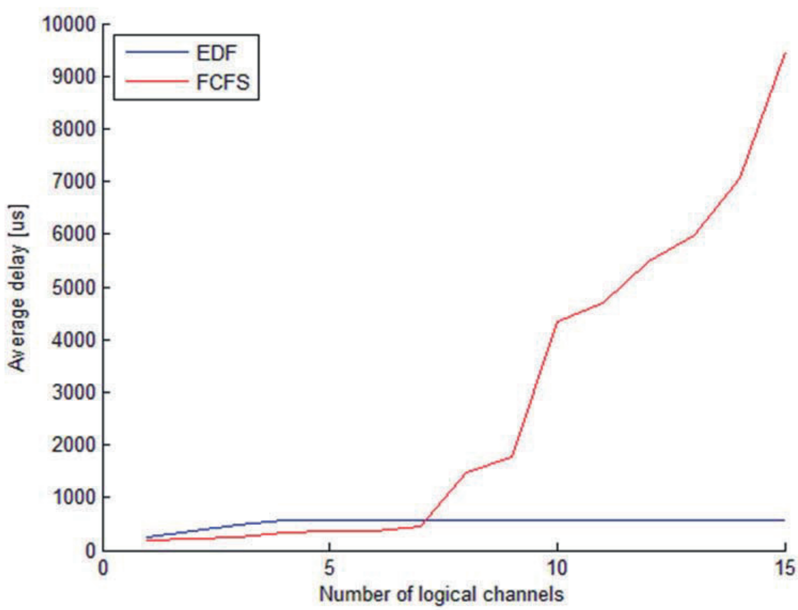


Figure 6. Average delay for FPQ(SRT) and DPQ(HRT) FCFS AND EDF respectively

With an increased HRT (hard real time traffic) intensity (Figure 6), the prioritized HRT traffic is affected the least, while the SRT traffic type experiences a higher increase in average delay. In Figure 6, the delay for the simulated FCFS traffic is plotted against a traffic probability of up to 15%. The diagram contains values for packets with different deadlines: As expected, the delay times rise as the traffic in the networks gets denser, but a change of deadline does not influence the delay of a packet, which results in almost identical curves for the different assumed deadlines. Up to an intensity of 7, the delay is very low for FCFS, but at intensities greater than that, the delay rises rapidly, which indicates that the network becomes saturated around a traffic density of 7 packets per node and time slot. This finding shows EDF remains in a constant average delay because of its priority. It can be concluded that EDF scheduling algorithm outperforms FCFS.

2.5 Deadline miss ratio analysis

First by deadline miss ratio we mean the slight different in the execution time and the known deadline and is termed Deadline miss ratio. Deadline miss ratio, M is the ratio of the number of missed deadline and the total number of deadline in an observable window. This can be controlled usually in EDF by changing their service level, as earlier described above.

In Figure 7 we plot the deadline miss ratio against the channels. As seen below each channels consist of 5, 10 and 15 packets per channels. Our queue consist of diverse priority queue and are been scheduled by their deadline. Below verifies that at low traffic load the deadline miss remain at Zero. But it got to a point at 8 packets per node, which the system could not actually cope and hence results in dead line miss for lower priority packets.

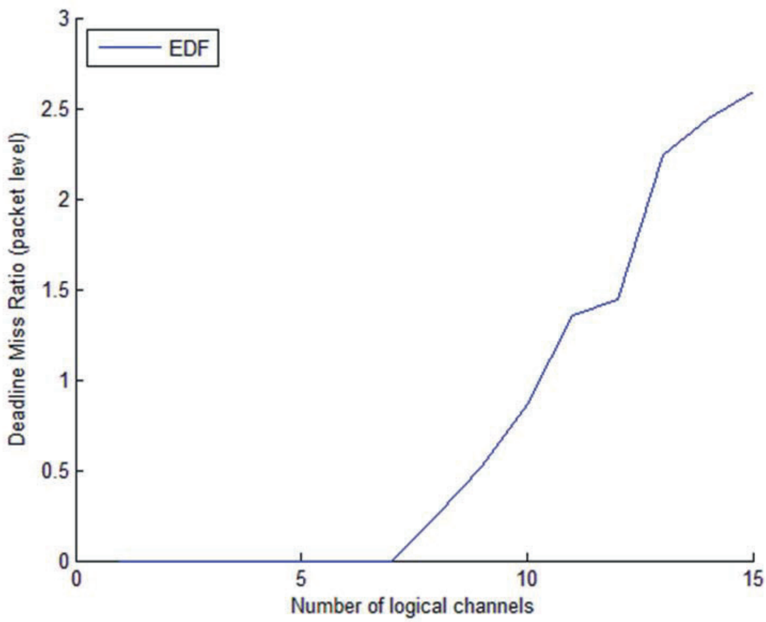


Figure 7. Deadline miss ratio for EDF

As can be seen in Figure 8, is a graph that shows EDF and FCFS comparison in terms of deadline miss ratio. Figure 8 verifies that the HRT traffic meets all its deadline requirements. In spite of the increasing SRT traffic load, the deadline miss ratio of the HRT traffic remains zero until a traffic intensity9, while the SRT/FCFS deadline miss ratio increases considerably to 0.25% even at traffic load of 6 pockets per node, due to its low priority. It also verifies how slight increase in average delay at high HRT traffic intensities does not affect the HRT deadline miss ratio. This shows that the HRT capacity of the network is not challenged by the imposed traffic intensities. The sudden increase in average delay for SRT packets, as seen in Figure 8, leads to a deadline miss ratio of about 0.25 at the maximum HRT traffic intensity of 6 packets per nodes, while EDF remains at 0.1. even at 10 packet per node, before the overload that results in a deadline miss ratio of 2.5 at a traffic load of 15 packets per node.

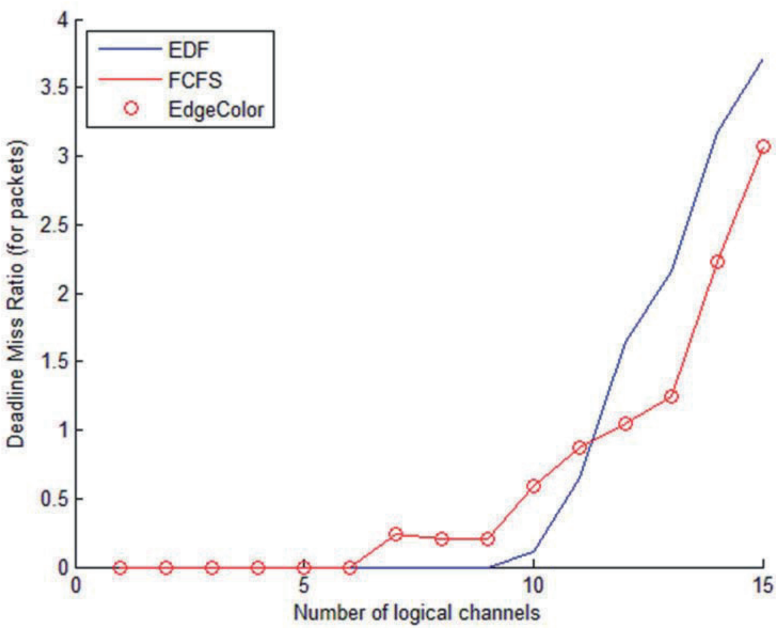


Figure 8. Deadline Miss ratio for EDF and FCFS

In conclusion, it can be said that EDF outperform FCFS in most of our simulations. The deadline miss ratio is for both policy is plotted in Figure 8, as the intensity approaches the network's saturation point, the deadline-miss-ratio rises rapidly to almost 4% immediately. The deadline-miss-ratio is not completely independent of the deadline, as a shorter deadline results in a higher deadline-miss-ratio at lower traffic intensities. The deadline-miss-ratio is independent of the number of nodes connected to the network. The diagram is an additional proof of how both policy struggle for task scheduling and priority.

3 Conclusions

In this paper we have reviewed form and methods for real time communication in packet switched networks. We illustrate both architectural, mathematical and algorithmic aspect of real time communication traffic and non real time traffic. Model such as this will play a significant role in an era of high speed integrated network. We have revised a packet scheduling algorithm from its original proof. The importance of this algorithm lies in the fact that we can use it to deploy real-time packet switching network that supports real-time distributed applications. In our original analysis, different from the original algorithm, the EDF algorithm is considered to reason about packet orders.

REFERENCES

1. Xiaojie Li, Xianbo He, 2010, *The improved EDF scheduling algorithm for embedded real-time system in the uncertain environment*. Advanced Computer Theory and Engineering (ICACTE), 3rd International Conference V4-563 - V4-566
2. Pachchigar P., Eswaran P.; Boke A.K., 2013, *Design and implementation of deadline based EDF algorithm on ARM LPC2148*, Information & Communication Technologies (ICT), IEEE Conference. pages: 994 - 997
3. Abugchem, F.; Short, M.; Donglai Xu, 2012, *A hybrid EDF algorithm for implementing resource-constrained real-time control applications*, Emerging Technologies & Factory Automation (ETFA), IEEE 17th Conference, pages: 1-4
4. Gupta A., Kohli S., *Jha S., 2013, Performance of EDF-BF algorithm under QoS constraint in grid heterogeneous environment*. Information Systems and Computer Networks (ISCON), International Conference, pages: 170-172.

NUMERICAL SIMULATIONS FOR SOLVING FUZZY FRACTIONAL DIFFERENTIAL EQUATIONS BY MAX-MIN IMPROVED EULER METHODS

Najeeb Alam Khan, Oyoon Abdul Razzaq, Fatima Riaz

Department of Mathematical Sciences
University of Karachi, Karachi 75270, Pakistan
njbalam@yahoo.com

Abstract

In this paper, an extension is introduced into Max-Min Improved Euler methods for solving initial value problems of fuzzy fractional differential equations (FFDEs). Two modified fractional Euler type methods have been proposed and investigated to obtain numerical solutions of linear and nonlinear FFDEs. The proposed algorithms are tested on various illustrative examples. Exact values are also simulated to compare and discuss the closeness and accuracy of approximations so obtained. Comparatively, tables and graphs results reveal the complete reliability, efficiency and accuracy of the proposed methods.

Key words: fuzzy, improved Euler method, numerical solution, Mathematica 10

1 Introduction

Generalization of ordinary calculus and an important branch in mathematical analysis, "fractional calculus" has become a subject of interest among mathematicians, physicists, and engineers in last few decades [1-5]. Its extensive development and influence in many areas occurred after the invention of differential calculus by Leibniz and Newton. Modeling of physical phenomenon in fractional differential equations (FDEs), although, a difficult task nonetheless has demonstrated applications in diverse fields of science. Recently, problems in many areas like diffusion process, rheology, electrochemistry, viscoelasticity, etc., have been developed and formulated in terms of fractional derivatives and fractional integrals, for instance, time-space fractional diffusion equation models, structural damping models, acoustical wave equations for complex media, fractional Schrodinger equation in quantum theory, etc. [6-10].

Whenever a real-world problem is converted to an ordinary differential equation, sometimes it cannot readily or rapidly be solved by a traditional mathematical method and a numerical method is usually sought and carried out. Correspondingly to ordinary differential equations (ODEs), the exact analytical solutions of fractional differential equations are often difficult, and sometimes impossible to obtain; thus numerical–analytical methods for solving fractional differential equations are of particular importance [11-20].

Fractional differential equations of physical phenomenon are said to be modeled perfectly if every uncertainty is coped out. Therefore, it is necessary to have other theories along with, which would handle issues of uncertainty while modeling. Various theories exist for describing such situations and the most popular among them is fuzzy set theory [21-22]. On contribution of this theory more realistic models have been obtained. Over the last few years, the theoretical framework of fractional differential equations with fuzzy theory has been an on go research field. Fuzzy fractional differential equation is generalization of fuzzy differential equation, which is initially introduced by Agarwal et al. [23] in which he considered differential equation of fractional order with uncertainty and presented the concept of solution. This initiative motivated several researchers to establish some results on the existence and uniqueness of the solutions, for instance, fuzzy Laplace transform, fractional Euler method, iterative techniques etc. are being employed for this purpose [24-28].

In the present paper, an interpretation for fuzzy fractional differential equations has been proposed. Essentially, this work is generalization and extension of Method 1 and Method 2 Improved Euler Method in paper of Smita and Chakraverty [29] to fractional order. Before Smita and Chakraverty, many other authors have made considerable efforts to improve an Euler method, and using it as the stepping-stone of numerical methods for solving initial value problems in differential equations. Basically, this method was introduced by Euler in 1728, which was moreover improved and improved by Heun. Further, Abraham et al. [30] proposed a new improvement on Euler method. Ma et al. [31] used standard Euler approximation method for linear and nonlinear first order fuzzy differential equation. Improved fractional Euler method for fuzzy fractional initial value problem is illustrated by Mazandarani et al. [32]. Ahmad et al. [33] proposed a new fuzzification of the classical Euler method for solution of linear and nonlinear fuzzy differential equations. Allahviranaloo [34] and Shokri [35] have employed iterative solution of improved Euler method for numerical solution of fuzzy differential equations. Duraisamy and Usha[36] used improved Euler method on fuzzy ordinary differential equations. Odibat [37] achieved his goal of numerical solution of linear and nonlinear equations of fractional order by using algorithm based on improved trapezoidal rule and the fractional Euler's method. At this note, two methods of improved fractional Euler methods, Max-Min IFEM and Average IFEM, have been illustrated. The approaches are demonstrated by applying them on various linear and nonlinear examples. Obtained numerical values are tabulated for different step size and fuzziness. Further, Graphical representation and their discussions are also presented. And lastly, conclusion is drawn from numerical investigation and comparisons.

2 Importance of media independent handover IEEE 802.21

In this section, several basic concepts of Riemann -Liouville integral and Caputo fractional derivative are recalled, which are used throughout this article.

Definition 2.1
The Riemann-Liouville integral of order $\alpha > 0$ for a function f is defined by [38]

$$J^{\alpha} f(x) = \int_0^x \frac{(x-\psi)^{\alpha-1}}{\Gamma(\alpha)} f(\psi)d\psi, \quad x > 0 \tag{1}$$

Lemma 2.1. Let $f(x)$ be a crisp continuous function and $\lceil \beta \rceil$ - times differentiable in the independent variable x over the interval of differentiation (integration) $[0, x]$. Then the following relation is hold [3].

$${}^{c}D^{\beta} f(x) = {}^{RL}D^{\beta}\left(f(x) - \sum_{k=0}^{\lfloor \beta \rfloor} \frac{x^k}{k!} f_0^{(k)} \right) \qquad \beta \in (n-1, n), \quad n \in N \tag{2}$$

where, $f_0^{(k)} = \left.\frac{d^k f(x)}{dx^k}\right|_{x=0}$ and ${}^{c}D^{\beta}$ are the Caputo derivative operators, $\lfloor \beta \rfloor$ and $\lceil \beta \rceil$ are just the values of β rounded up and down to the nearest integer number respectively. ${}^{RL}D^{\beta}$ is the more common Riemann–Liouville fractional derivative operator which can be defined as follows:

$${}^{RL}D^{\beta} f(x) = \frac{1}{\Gamma(\lceil \beta \rceil - \beta)} \frac{d^{\lceil \beta \rceil}}{dx^{\lceil \beta \rceil}} \int_0^x \frac{f(\psi)}{(x-\psi)^{1-\lceil \beta \rceil + \beta}} d\psi \tag{3}$$

3 Fuzzy Number

Fuzzy number can be defined as a fuzzy set on R^n which is upper semi-continuous, convex, normal and compactly supported in a metric space denoted by E^1. Also an arbitrary fuzzy number can be represented by an ordered pair of lower and upper bound in which lower bound is left-continuous non-decreasing and upper bound is non-increasing functions over the interval $[0,1]$.

3.1 Trapezoidal Fuzzy Number (TrFN)

A trapezoidal fuzzy number is defined as $A=(b_1,b_2,b_3,b_4)$ such that $b_1<b_2<b_3<b_4$ are four elements of a fuzzy set on $R=(-\infty,\infty)$. Its membership function $\mu A(x)$ is defined as follows:

$$\mu A(x)=\begin{cases}\dfrac{x-b_1}{b_2-b_1}, \text{if } b_1\le x\le b_2\\ 1, \quad \text{if } b_2\le x\le b_3\\ \dfrac{b_4-x}{b_4-b_3}, \text{if } b_3\le x\le b_4\\ 0, \quad \text{otherwise}\end{cases} \tag{4}$$

Though an r-cut approach trapezoidal fuzzy number can be represented as an ordered pair, i.e., $[(b_2-b_1)r+b_1, -(b_4-b_3)r+b_4]$, where $r\in[0,1]$.

3.2 Triangular Fuzzy Number (TFN)

A triangular fuzzy number B is defined as $B=(b_1,b_2,b_3)$ such that $b_1<b_2<b_3$ are three elements of a fuzzy set on $R=(-\infty,\infty)$. Its membership function $\mu B(x)$ is defined as follows:

$$\mu B(x)=\begin{cases}\dfrac{x-b_1}{b_2-b_1}, \text{ if } b_1\le x\le b_2\\ \dfrac{b_3-x}{b_3-b_2}, \text{ if } b_2\le x\le b_3\\ 0, \quad \text{otherwise}\end{cases} \tag{5}$$

Definition of fuzzy initial value problem can be found in [29].

4 Fuzzy fractional differential equations

Consider the following nth order FFDE by

$$D_x^{n\beta}y(x)+g\left(x,y(x),D_x^{\beta}y(x),...\right)=f\left(x,y(x),D_x^{\beta}y(x),...\right) \tag{6}$$

$$y(x_0)=y_0\,,\ y'(x_0)=y'_0 \tag{7}$$

where $y(x)=\left(\underline{y}(x,r),\overline{y}(x,r)\right)$ is a fuzzy function of x, $f\left(x,y(x),D_x^{\beta}y(x),...\right)$ and $g\left(x,y(x),D_x^{\beta}y(x),...\right)$ are linear fuzzy-valued functions.

The generalized fractional Euler's method that has been formulated for the numerical solution of initial value problems with Caputo derivatives is presented by Odibat and Momani [37].

Suppose that $D_x^{k\beta}f(x)\in C(0,a]$, for $k=0,1,2,...,n+1$, where $0<\beta\le 1$ Then we have

$$f(x)=\sum_{i=o}^{n}\frac{x^{i\beta}}{\Gamma(i\beta+1)}\left(D^{i\beta}\right)(0+)+\frac{\left(D^{(n+1)\beta}f\right)(\zeta)}{\Gamma((n+1)\beta+1)}x^{(n+1)\beta}, \tag{8}$$

With $0\le\zeta\le x,\forall x\in(0,a]$.

In case $\beta=1$, the generalized Taylor's formula in Eq. (8) reduces to the classical Taylor's formula. With the aim to find the high accuracy for solving fuzzy fractional differential equations, the initial value problem is considered as:

$$ {}^{c}D_{x}^{\beta}\tilde{y}_{n}(x)=f(x_{n},\tilde{y}_{n}(x_{n})),\quad \tilde{y}(0)=\tilde{y}_{0},\quad 0<\beta\le 1,\quad x_{n}>0 \tag{9}$$

The fuzzy fractional initial value problem (FFIVP) can be considered equivalently by the following initial value problems

$$\begin{aligned} {}^{c}D_{x}^{\beta}\underline{y}_{n}^{r}(x)&=\left[f(x_{n},\underline{y}_{n}(x_{n}))\right]^{r}=F(x_{n},\overline{y}_{n}^{r}(x_{n}),\underline{y}_{n}^{r}(x_{n}))\\ {}^{c}D_{x}^{\beta}\overline{y}_{n}^{r}(x)&=\left[f(x_{n},\overline{y}_{n}(x_{n}))\right]^{r}=G(x_{n},\overline{y}_{n}^{r}(x_{n}),\underline{y}_{n}^{r}(x_{n}))\end{aligned} \tag{10}$$

$$\overline{y}^{r}(0)=\overline{y}_{0}^{r},\quad \underline{y}^{r}(0)=\underline{y}_{0}^{r} \tag{11}$$

Let $[0,a]$ be the interval over which the solution of the problem is needed. The focus is not to acquire a function $y(x)$ that satisfies the initial value problem. Alternately, an approximation has been made with the help of a set of points $\{(x_{j},y(x_{j}))\}$. Using Taylor expansion:

$$\underline{y}_{n+1}(x_{n+1})=\underline{y}_{n}(x_{n})+\frac{h^{\beta}}{\Gamma(\beta+1)}\left({}^{c}D_{x}^{\beta}\underline{y}_{n}(x_{n})\right)+\frac{h^{2\beta}}{\Gamma(2\beta+1)}\left({}^{c}D_{x}^{2\beta}\underline{y}_{n}(x_{n})\right)+\frac{h^{3\beta}}{\Gamma(3\beta+1)}\left({}^{c}D_{x}^{3\beta}\underline{y}_{n}(x_{n})\right) \tag{12}$$

and

$$\overline{y}_{n+1}(x_{n+1})=\overline{y}_{n}(x_{n})+\frac{h^{\beta}}{\Gamma(\beta+1)}\left({}^{c}D_{x}^{\beta}\overline{y}_{n}(x_{n})\right)+\frac{h^{2\beta}}{\Gamma(2\beta+1)}\left({}^{c}D_{x}^{2\beta}\overline{y}_{n}(x_{n})\right)+\frac{h^{3\beta}}{\Gamma(3\beta+1)}\left({}^{c}D_{x}^{3\beta}\overline{y}_{n}(x_{n})\right) \tag{13}$$

Since

$${}^{c}D_{x}^{2\beta}\underline{y}_{n}(x_{n})=\Gamma(\beta+1)\left(\frac{{}^{c}D_{x}^{\beta}\underline{y}_{n+1}(x_{n+1})-{}^{c}D_{x}^{\beta}\underline{y}_{n}(x_{n})}{h^{\beta}}\right)$$

and

$${}^{c}D_{x}^{2\beta}\overline{y}_{n}(x_{n})=\Gamma(\beta+1)\left(\frac{{}^{c}D_{x}^{\beta}\overline{y}_{n+1}(x_{n+1})-{}^{c}D_{x}^{\beta}\overline{y}_{n}(x_{n})}{h^{\beta}}\right)$$

Therefore, Eqs. (12)-(13) becomes,

$$\underline{y}_{n+1}(x_{n+1})=\underline{y}_{n}(x_{n})+\frac{h^{\beta}}{\Gamma(\beta+1)}{}^{c}D_{x}^{\beta}\underline{y}_{n}(x_{n})+\frac{h^{2\beta}}{\Gamma(2\beta+1)}\left(\Gamma(\beta+1)\left(\frac{{}^{c}D_{x}^{\beta}\underline{y}_{n+1}(x_{n+1})-{}^{c}D_{x}^{\beta}\underline{y}_{n}(x_{n})}{h^{\beta}}\right)\right) \tag{14}$$

and

$$\overline{y}_{n+1}(x_{n+1}) = \overline{y}_n(x_n) + \frac{h^{\beta}}{\Gamma(\beta+1)} {}^{c}D_x^{\beta}\overline{y}_n(x_n) + \frac{h^{2\beta}}{\Gamma(2\beta+1)}\left(\Gamma(\beta+1)\left(\frac{{}^{c}D_x^{\beta}\overline{y}_{n+1}(x_{n+1}) - {}^{c}D_x^{\beta}\overline{y}_n(x_n)}{h^{\beta}}\right)\right) \tag{15}$$

Implies

$$\underline{y}_{n+1}(x_{n+1}) = \underline{y}_n(x_n) + h^{\beta}\left(\frac{\Gamma(2\beta+1) - (\Gamma(\beta+1))^2}{(\Gamma(\beta+1))(\Gamma(2\beta+1))}\right) {}^{c}D_x^{\beta}\underline{y}_n(x_n) + \frac{h^{\beta}(\Gamma(\beta+1))}{\Gamma(2\beta+1)} {}^{c}D_x^{\beta}\underline{y}_{n+1}(x_{n+1}) \tag{16}$$

and

$$\overline{y}_{n+1}(x_{n+1}) = \overline{y}_n(x_n) + h^{\beta}\left(\frac{\Gamma(2\beta+1) - (\Gamma(\beta+1))^2}{(\Gamma(\beta+1))(\Gamma(2\beta+1))}\right) {}^{c}D_x^{\beta}\overline{y}_n(x_n) + \frac{h^{\beta}(\Gamma(\beta+1))}{\Gamma(2\beta+1)} {}^{c}D_x^{\beta}\overline{y}_{n+1}(x_{n+1}) \tag{17}$$

when

$${}^{c}D_x^{\beta}\overline{y}_n(x_n) = \overline{f}(x_n, y_n(x_n)),\ {}^{c}D_x^{\beta}\underline{y}_n(x_n) = \underline{f}(x_n, y_n(x_n)),$$
$${}^{c}D_x^{\beta}\overline{y}_{n+1}(x_{n+1}) = \overline{f}(x_{n+1}, y_{n+1}(x_{n+1}))$$

and

$${}^{c}D_x^{\beta}\underline{y}_{n+1}(x_{n+1}) = \underline{f}(x_{n+1}, y_{n+1}(x_{n+1}))$$

are substituted into equations (16) and (17), consequently an expression for $\underline{y}_{n+1}, \overline{y}_{n+1}$ is obtained:

$$\underline{y}_{n+1}(x_{n+1}) = \underline{y}_n(x_n) + h^{\beta}\left(\frac{\Gamma(2\beta+1) - (\Gamma(\beta+1))^2}{(\Gamma(\beta+1))(\Gamma(2\beta+1))}\right)\underline{f}(x_n, y_n(x_n)) + \frac{h^{\beta}(\Gamma(\beta+1))}{\Gamma(2\beta+1)}\underline{f}(x_{n+1}, y_{n+1}(x_{n+1})) \tag{18}$$

and

$$\overline{y}_{n+1}(x_{n+1}) = \overline{y}_n(x_n) + h^{\beta}\left(\frac{\Gamma(2\beta+1) - (\Gamma(\beta+1))^2}{(\Gamma(\beta+1))(\Gamma(2\beta+1))}\right)\overline{f}(x_n, y_n(x_n)) + \frac{h^{\beta}(\Gamma(\beta+1))}{\Gamma(2\beta+1)}\overline{f}(x_{n+1}, y_{n+1}(x_{n+1})) \tag{19}$$

A system of points that approximate the solution of $y(x)$ is produced by above recursive process and at each step the fractional Euler's method is used as a prediction.

5 Proposed methods for FFDE

In this section, two methods to solve FFDE are proposed and illustrated.

5.1 Max-Min Modified Fractional Euler Method (Method 1)

Considering all the possible combination of lower and upper values of the variable and by using modified fractional Euler method we obtain

$$\underline{y}_{n+1}^{(1)}(x_{n+1};r)=\underline{y}_n+h^{\beta}\left(\frac{\Gamma(2\beta+1)-(\Gamma(\beta+1))^2}{(\Gamma(\beta+1))(\Gamma(2\beta+1))}\right)f(x_n,\underline{y}_n(x_n;r))+\frac{h^{\beta}(\Gamma(\beta+1))}{\Gamma(2\beta+1)}f(x_{n+1},\underline{y}_{n+1}(x_{n+1};r)),\tag{20}$$

$$\overline{y}_{n+1}^{(1)}(x_{n+1};r)=\overline{y}_n+h^{\beta}\left(\frac{\Gamma(2\beta+1)-(\Gamma(\beta+1))^2}{(\Gamma(\beta+1))(\Gamma(2\beta+1))}\right)f(x_n,\overline{y}_n(x_n;r))+\frac{h^{\beta}(\Gamma(\beta+1))}{\Gamma(2\beta+1)}f(x_{n+1},\overline{y}_{n+1}(x_{n+1};r)),\tag{21}$$

$$\underline{y}_{n+1}^{(2)}(x_{n+1};r)=\underline{y}_n+h^{\beta}\left(\frac{\Gamma(2\beta+1)-(\Gamma(\beta+1))^2}{(\Gamma(\beta+1))(\Gamma(2\beta+1))}\right)f(x_n,\overline{y}_n(x_n;r))+\frac{h^{\beta}(\Gamma(\beta+1))}{\Gamma(2\beta+1)}f(x_{n+1},\overline{y}_{n+1}(x_{n+1};r)),\tag{22}$$

$$\overline{y}_{n+1}^{(2)}(x_{n+1};r)=\overline{y}_n+h^{\beta}\left(\frac{\Gamma(2\beta+1)-(\Gamma(\beta+1))^2}{(\Gamma(\beta+1))(\Gamma(2\beta+1))}\right)f(x_n,\underline{y}_n(x_n;r))+\frac{h^{\beta}(\Gamma(\beta+1))}{\Gamma(2\beta+1)}f(x_{n+1},\underline{y}_{n+1}(x_{n+1};r)).\tag{23}$$

The exact and approximate solution to $x_n;\quad 0\le n\le N$ are denoted by $[Y_n(x_n;r)]=[(\underline{Y}(x_n;r),\overline{Y}(x_n;r))]$ and $[y_n(x_n;r)]=[(\underline{y}(x_n;r),\overline{y}(x_n;r))]$, respectively. However, minimum and maximum are taken from Eqs. (18)-(21) for lower and upper values of the independent variable x_n.

$$\underline{y}_{n+1}(x_{n+1};r)=\min\{\underline{y}_{n+1}^{(1)}(x_{n+1};r),\underline{y}_{n+1}^{(2)}(x_{n+1};r)\}\tag{24}$$

$$\overline{y}_{n+1}(x_{n+1};r)=\max\{\overline{y}_{n+1}^{(1)}(x_{n+1};r),\overline{y}_{n+1}^{(2)}(x_{n+1};r)\}\tag{25}$$

The above procedure leads to the better approximations to the exact solutions. The efficiency and powerfulness of the methodology are demonstrated by variety of examples.

5.2 Average Modified Fractional Euler Method (Method 2)

In the second method, similarly averages of lower and upper values are computed respectively. Then the Eqs. (18)-(21) reduces to

$$\underline{y}_{n+1} = \frac{1}{2}\left[\underline{y}^{(1)}\left(x_{n+1};r\right) + \underline{y}^{(2)}\left(x_{n+1};r\right)\right], \tag{26}$$

$$\overline{y}_{n+1} = \frac{1}{2}\left[\overline{y}^{(1)}\left(x_{n+1};r\right) + \overline{y}^{(2)}\left(x_{n+1};r\right)\right] \tag{27}$$

6 Numerical problems

Problem 1
Let us consider a linear triangular FIVP given in [29] for fractional order.

$$D_x^{\beta} y(x) = y(x),\ \ y(0) = (0.75, 1, 1.25). \tag{28}$$

Then using the r-cut approach, the triangular fuzzy initial condition can be represented as

$$y(0) = [0.75 + 0.25r, 1.125 - 0.125r], 0 \le r \le 1. \tag{29}$$

The results of problem 1, obtained by the above Method 1 and Method 2 are tabulated in Table 1and 2 for different values of.

Table 1. Comparison between Exact and proposed methods of Problem 1 for $x = 1$, $h = 0.001$, $\beta = 1$

r	Exact values	Modified Fractional Euler Methods	
		Method 1 $\beta = 1$	Method 2 $\beta = 1$
0	[2.03871,3.05807]	[2.03871,3.05807]	[2.03922,3.05756]
0.2	[2.17463,2.99011]	[2.17463,2.99011]	[2.17503,2.9897]
0.4	[2.31054,2.92215]	[2.31054,2.92215]	[2.31084,2.92185]
0.6	[2.44645,2.8542]	[2.44645,2.8542]	[2.44666,2.85399]
0.8	[2.58237,2.78624]	[2.58237,2.78624]	[2.58247,2.78614]
1	[2.71828,2.71828]	[2.71828,2.71828]	[2.71828,2.71828]

Table 2. Results obtained for Problem 1 by Method 1 and Method 2 for $x = 1$, $h = 0.1$, $\beta = 0.75$ and $\beta = 0.85$

Modified Fractional Euler Methods				
r	Method 1 $\beta = 0.75$	Method 2 $\beta = 0.75$	Method 1 $\beta = 0.85$	Method 2 $\beta = 0.85$
0	[5.35740,8.03611]	[5.59647,7.79704]	[3.37463,5.06194]	[3.49243,4.94414]
0.2	[5.71456,7.85753]	[5.90582,7.66627]	[3.59960,4.94945]	[3.69384,4.85521]
0.4	[6.07173,7.67895]	[6.21516,7.53551]	[3.82458,4.83696]	[3.89526,4.76628]
0.6	[6.42889,7.50037]	[6.52451,7.40474]	[4.04955,4.72448]	[4.09667,4.67736]
0.8	[6.78605,7.32179]	[6.83386,7.27397]	[4.27453,4.61199]	[4.29809,4.58843]
1	[7.14321,7.14321]	[7.14321,7.14321]	[4.27453,4.4995]	[4.49950,4.49950]

Problem 2

Next consider a linear trapezoidal FIVP given in [29] for fractional order.

$$D_x^{\beta} y(x) = y(x), \quad y(0) = (0.75, 0.85, 1.1, 1.25) \tag{30}$$

Then using the r-cut approach, the trapezoidal fuzzy initial condition can be represented as

$$y(0) = [0.75 + 0.1r, 1.125 - 0.025r], \; 0 \le r \le 1 \tag{31}$$

The results of problem 2, obtained by the above Method 1 and Method 2 are tabulated in Table 3 and 4 for different values of r.

Table 3. Comparison between Exact and proposed methods of Problem 2 for $x = 1$, $h = 0.001$, $\beta = 1$

r	Exact Values	Modified Fractional Euler Methods	
		Method 1 $\beta = 1$	Method 2 $\beta = 1$
0	[2.03871,3.05807]	[2.03871,3.05807]	[2.03922,3.05756]
0.2	[2.09308,3.04448]	[2.09308,3.04448]	[2.09355,3.04400]
0.4	[2.14744,3.03088]	[2.14744,3.03088]	[2.14788,3.03044]
0.6	[2.20181,3.01729]	[2.20181,3.01729]	[2.20222,3.01688]
0.8	[2.25617,3.0037]	[2.25617,3.0037]	[2.25655,3.00333]
1	[2.31054,2.99011]	[2.31054,2.99011]	[2.31088,2.98977]

Table 4. Results obtained for Problem 2 by Method 1 and Method 2 for $x=1$, $h=0.1$, $\beta=0.75$ and $\beta=0.85$

Modified Fractional Euler Methods				
R	Method 1 $\beta=0.75$	Method 2 $\beta=0.75$	Method 1 $\beta=0.85$	Method 2 $\beta=0.85$
0	[5.35740,8.03611]	[5.59647,7.79704]	[3.37463,5.06194]	[3.49243,4.94414]
0.2	[5.50027,8.00039]	[5.72340,7.77726]	[3.46462,5.03944]	[3.57457,4.92949]
0.4	[5.64313,7.96467]	[5.85032,7.75748]	[3.55461,5.01695]	[3.65670,4.91485]
0.6	[5.78600,7.92896]	[5.97725,7.73771]	[3.64460,4.99445]	[3.73884,4.90021]
0.8	[5.92886,7.89324]	[6.10418,7.71793]	[3.73459,4.97195]	[3.82098,4.88556]
1	[6.07173,7.85753]	[6.23110,7.69815]	[3.82458,4.94945]	[3.90311,4.87092]

Problem 3

Consider a nonlinear triangular FIVP for fractional order.

$$D_x^\beta y(x) = -y(x) + y(x)^2, \quad y(0) = (0.75, 1, 1.25), \quad 0 \le r \le 1 \tag{32}$$

Using the r-cut approach, the triangular fuzzy initial condition can be represented as

$$y(0) = [0.75 + 0.25r, 1.125 - 0.125r], \quad 0 \le r \le 1 \tag{33}$$

Obtained results by Method 1 Method 2 are tabulated in Table 5 and 6 for different values of r.

Table 5. Comparison between Exact and proposed methods of Problem 3for $x=1$, $h=0.001$, $\beta=1$

r	Exact Values	Modified Fractional Euler Methods	
		Method 1 $\beta=1$	Method 2 $\beta=1$
0	[0.524633,1.43273]	[0.524633,1.43273]	[0.525068,1.43229]
0.2	[0.59539,1.32823]	[0.59539,1.32823]	[0.595729,1.32789]
0.4	[0.675814,1.23403]	[0.675814,1.23403]	[0.676068,1.23378]
0.6	[0.768031,1.14869]	[0.768031,1.14869]	[0.768205,1.14851]
0.8	[0.874839,1.07101]	[0.874839,1.07101]	[0.874932,1.07091]
1	[1.000000,1.00000]	[1.000000,1.00000]	[1.000000,1.00000]

Table 6. Results obtained for Problem 3 by Method 1 and Method 2 for $x = 1$, $h = 0.1$, $\beta = 0.75$ and $\beta = 0.85$

Modified Fractional Euler Methods				
R	Method 1 $\beta = 0.75$	Method 2 $\beta = 0.75$	Method 1 $\beta = 0.85$	Method 2 $\beta = 0.85$
0	[0.34438,2.87449]	[0.72317,2.49570]	[0.40220,2.00084]	[0.54076,1.65376]
0.2	[0.36505,2.85525]	[0.73933,2.48096]	[0.47238,1.69349]	[0.56576,1.60011]
0.4	[0.44727,2.00193]	[0.62307,1.82613]	[0.55867,1.45838]	[0.62081,1.39625]
0.6	[0.56075,1.51888]	[0.64884,1.43079]	[0.66742,1.27305]	[0.70700,1.23346]
0.8	[0.72802,1.21165]	[0.76858,1.17109]	[0.80876,1.12335]	[0.82935,1.10276]
1	[1.00000,1.00000]	[1.00000,1.00000]	[1.00000,1.00000]	[1.00000,1.00000]

Problem 4

Now consider a nonlinear trapezoidal FIVP for fractional order.

$$D_x^{\beta} y(x) = -y(x) + y(x)^2, \quad y(0) = (0.75, 0.85, 1.1, 1.25r), \quad 0 \le r \le 1 \tag{34}$$

Using r- cut approach, the trapezoidal fuzzy initial condition can be represented as

$$y(0) = [0.75 + 0.1r, 1.125 - 0.025r], \quad 0 \le r \le 1 \tag{35}$$

Results of this problem, obtained by Method 1 Method 2 are tabulated in Table 7 and 8 for different values of r.

Table 7. Comparison between Exact and proposed methods of Problem 4for $x = 1$, $h = 0.001$ and $\beta = 1$

r	Exact Values	Modified Fractional Euler Methods	
		Method 1 $\beta = 1$	Method 2 $\beta = 1$
0	[0.52463,1.43273]	[0.52463,1.43273]	[0.52507,1.43229]
0.2	[0.55189,1.41092]	[0.55189,1.41092]	[0.55230,1.41051]
0.4	[0.58052,1.38959]	[0.58052,1.38959]	[0.58092,1.38919]
0.6	[0.61064,1.36870]	[0.61064,1.36870]	[0.61101,1.36833]
0.8	[0.64236,1.34825]	[0.64236,1.34825]	[0.64271,1.34790]
1	[0.67581,1.32823]	[0.67581,1.32823]	[0.67614,1.32790]

Table 8. Results obtained for Problem 4 by Method 1 and Method 2 for $x=1$, $h=0.1$, $\beta=0.75$ and $\beta=0.85$

Modified Fractional Euler Methods				
r	Method 1 $\beta=0.75$	Method 2 $\beta=0.75$	Method 1 $\beta=0.85$	Method 2 $\beta=0.85$
0	[0.34438,2.87449]	[0.72317,2.49570]	[0.43847,1.75604]	[0.54076,1.65376]
0.2	[0.32569,4.16151]	[1.07889,3.40834]	[0.42864,1.93209]	[0.55812,1.80261]
0.4	[0.35124,3.74846]	[0.97758,3.12212]	[0.45721,1.86728]	[0.57654,1.74796]
0.6	[0.37964,3.40168]	[0.90444,2.87688]	[0.48819,1.80610]	[0.59812,1.69617]
0.8	[0.41143,3.10741]	[0.85409,2.66475]	[0.52188,1.74826]	[0.62307,1.64707]
1	[0.44727,2.85525]	[0.82280,2.47971]	[0.55867,1.69349]	[0.65165,1.60051]

Problem 5

Let us consider another nonlinear triangular FIVP, also studied in [29], for fractional order.

$$D_x^\beta y(x)=e^{-y(x)^2}, \quad y(0)=(0.75,1,1.5), \quad 0\le r\le 1 \tag{36}$$

Usingthe r-cut approach, the triangular fuzzy initial condition can be represented as

$$y(0)=[0.75+0.25r,1.5-0.5r] \quad ,0\le r\le 1. \tag{37}$$

Outcomes so obtained by Method 1 Method 2 are tabulated in Table 9 and 10 for different values of r.

Table 9. Comparison between Exact and proposed methods of Problem 5 for $x=1$, $h=0.001$, $\beta=1$

r	Exact Values	Modified Fractional Euler Methods	
		Method 1 $\beta=1$	Method 2 $\beta=1$
0	[1.14353,1.59134]	[1.14334,1.59154]	[1.14343,1.59144]
0.2	[1.16752,1.51823]	[1.16736,1.51838]	[1.16744,1.51831]
0.4	[1.19208,1.4497]	[1.19196,1.44982]	[1.19202,1.44976]
0.6	[1.21727,1.38565]	[1.21719,1.38573]	[1.21723,1.38569]
0.8	[1.24318,1.32584]	[1.24314,1.32588]	[1.24316,1.32586]
1	[1.26987,1.26987]	[1.26987,1.26987]	[1.26987,1.26987]

Table 10. Results obtained for Problem 5 by Method 1 and Method 2 for $x=1$, $h=0.1$, $\beta=0.75$ and $\beta=0.85$

Modified Fractional Euler Methods				
r	Method 1 $\beta=0.75$	Method 2 $\beta=0.75$	Method 1 $\beta=0.85$	Method 2 $\beta=0.85$
0	[1.31283,1.67807]	[1.32335,1.66755]	[1.23527,1.64905]	[1.24582,1.63849]
0.2	[1.33222,1.61581]	[1.34072,1.60731]	[1.25774,1.58069]	[1.26631,1.57212]
0.4	[1.35233,1.55901]	[1.35874,1.55261]	[1.28098,1.51748]	[1.28748,1.51097]
0.6	[1.37317,1.5073]	[1.37744,1.50302]	[1.30502,1.45916]	[1.30939,1.45479]
0.8	[1.39474,1.46018]	[1.39688,1.45804]	[1.32989,1.40536]	[1.33208,1.40316]
1	[1.41710,1.41710]	[1.41710,1.41710]	[1.35560,1.35560]	[1.35560,1.35560]

Problem 6

Now consider nonlinear trapezoidal FIVP for fractional order.

$$D_x^{\beta} y(x) = e^{-x^2} \quad y(0) = (0.75, 0.85, 1.3, 1.5) \quad 0 \le r \le 1 \tag{38}$$

Using r- cut approach, the trapezoidal fuzzy initial condition can be represented as

$$y(0) = [0.75 + 0.1r, 1.5 - 0.2r], \quad 0 \le r \le 1 \tag{39}$$

Results so obtained by Method 1 Method 2 are tabulated in Table 11 and 12 for different values of r.

Table 11. Comparison between Exact and proposed methods of Problem 6for $x=1$, $h=0.001$ and $\beta=1$.

r	Exact Values	Modified Fractional Euler Methods	
		Method 1 $\beta=1$	Method 2 $\beta=1$
0	[1.14353,1.59134]	[1.14334,1.59154]	[1.14343,1.59144]
0.2	[1.15306,1.56155]	[1.15288,1.56173]	[1.15297,1.56164]
0.4	[1.16268,1.53248]	[1.16252,1.53265]	[1.16260,1.53257]
0.6	[1.17238,1.50416]	[1.17223,1.50430]	[1.17231,1.50423]
0.8	[1.18218,1.47656]	[1.18205,1.47670]	[1.18211,1.47663]
1	[1.19208,1.44970]	[1.19196,1.44982]	[1.19202,1.44976]

Table 12. Results obtained for Problem 6 by Method 1 and Method 2 for $x = 1$, $h = 0.1$, $\beta = 0.75$ and $\beta = 0.85$

Modified Fractional Euler Methods				
r	Method 1 $\beta = 0.75$	Method 2 $\beta = 0.75$	Method 1 $\beta = 0.85$	Method 2 $\beta = 0.85$
0	[1.31283,1.67807]	[1.32335,1.66755]	[1.23527,1.64905]	[1.24582,1.63849]
0.2	[1.3205,1.65249]	[1.33022,1.64277]	[1.24416,1.62108]	[1.25394,1.61131]
0.4	[1.32829,1.62782]	[1.33719,1.61891]	[1.25318,1.59395]	[1.26216,1.58497]
0.6	[1.33619,1.60402]	[1.34427,1.59594]	[1.26232,1.56765]	[1.27049,1.55948]
0.8	[1.3442,1.5811]	[1.35145,1.57385]	[1.27159,1.54216]	[1.27893,1.53482]
1	[1.35233,1.55901]	[1.35874,1.55261]	[1.28098,1.51748]	[1.28748,1.51097]

7 Results and discussion

In this section, corresponding plots of problems 1-6 are given for different values of r. All the figures represent the comparison between exact and approximated values obtained by Method 1 and Method 2. Triangular curves in figures 1-2, 5-6 and 9-10 verifies the conditions of triangular fuzzy number, i.e. the lower and upper values converge to one value at $r = 1$. On the other hand figures 3-4, 7-8, 11-12, having trapezoidal curves indicate the verification of trapezoidal fuzzy number, i.e. the lower and upper values converge to two different values $r = 1$. The overlapping of dotes and lines indicates closeness of the approximated and exact values which further verifies the accuracy of the proposed methods.

Figures 3, 11, 19 represent triangular curves of problems 1, 3 and 5, respectively, for 0.75 order of derivative i.e. $\beta = 0.75$, calculated by Method 1and Method 2. And figures 4, 12, 20 show triangular plots for 0.85 order of derivative i.e. $\beta = 0.85$, of example 1,3 and 5, respectively, calculated by Method 1 and Method 2. The triangular curves indicate that the fractional order calculated by considered methods also satisfy conditions of triangular fuzzy numbers. Conversely, figures 7, 15, 23 represent trapezoidal curves for $\beta = 0.75$ and figures 8, 16, 24 trapezoidal curves for $\beta = 0.85$, of examples 2, 4, and 6, respectively. Observably, fractional orders of these examples, calculated using Method 1 and Method 2, also converge to two different points, making a trapezoidal curve. From these figures it can be clearly resolved that the proposed methods are also applicable and efficient for fractional order derivatives.

8 Conclusions

In this paper, advancement in Method 1 and Method 2 of Max-Min Improved Euler methods [29] for the fractional order has been made. The proposed methods are demonstrated through linear and nonlinear fractional differential equations with triangular and trapezoidal fuzzy initial values. Relatively, followings results are concluded:

- The proposed method converges to the exact solution more rapidly.
- Numerical results show that for smaller step size smaller error and hence better and accurate results are obtained.
- Table values and graphical plots for trapezoidal fuzzy initial values satisfy the definition of trapezoidal fuzzy number in section 3.1, i.e. results obtained from proposed methods approach to two different solutions for the upper and the lower solutions of the problems.
- Similarly, Table values and graphical plots for triangular fuzzy initial values satisfy the definition of triangular fuzzy number in section 3.2, i.e.results obtained from proposed method approach to one solution for the upper and the lower solutions of the problems.
- It is shown that the proposed method yielded accurate approximations.

The computations corresponding to the problems have been performed using Mathematica 9. Accordingly, the proposed methods are efficiently applicable for the fuzzy fractional differential equations of both linear and nonlinear problems. It can be noticed that presented methods show an easy and efficient way of acquiring accurate solution.

Appendix

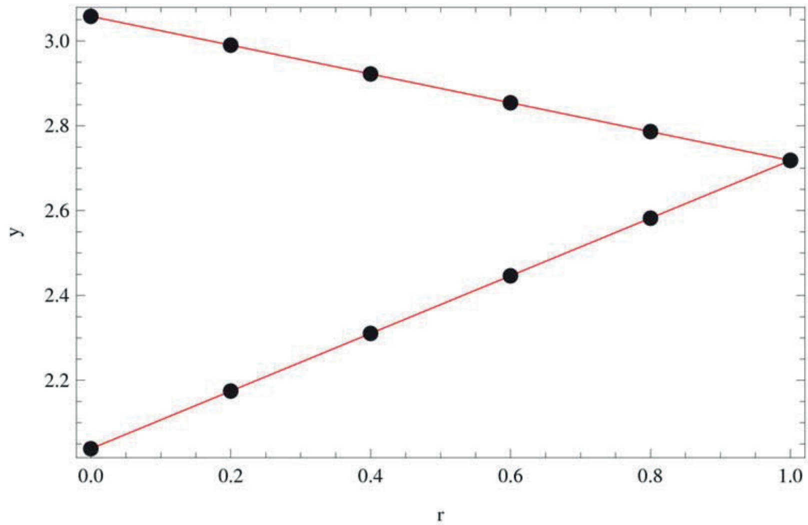


Figure 1. Comparison between exact and calculated values of problem 1 by Method 1 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

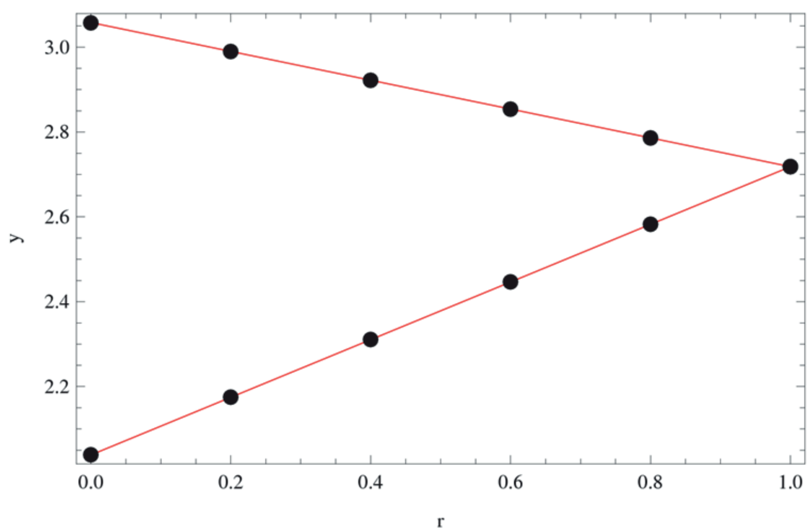


Figure 2. Comparison between Exact and calculated values of problem 1 by Method 2 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

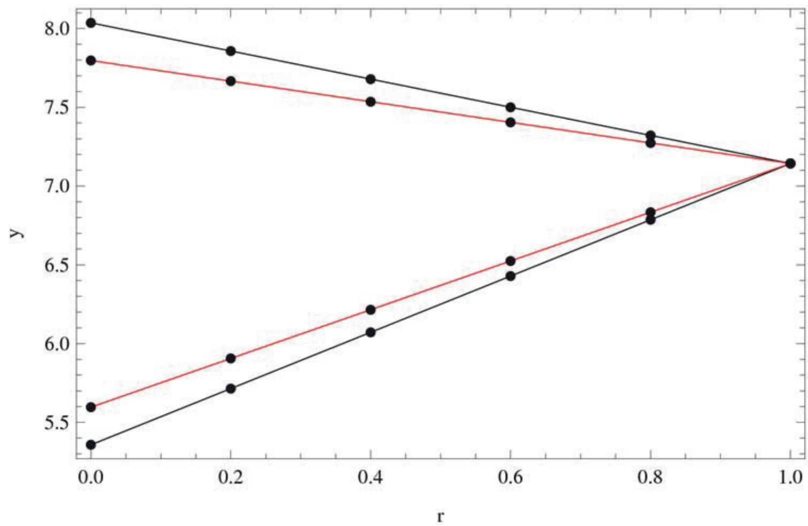


Figure 3. Comparison between Method 1and Method 2 of problem 1 with $\beta = 0.75$ and $h = 0.1$ at $x = 1$. (Black for Method 1, Red for Method 2)

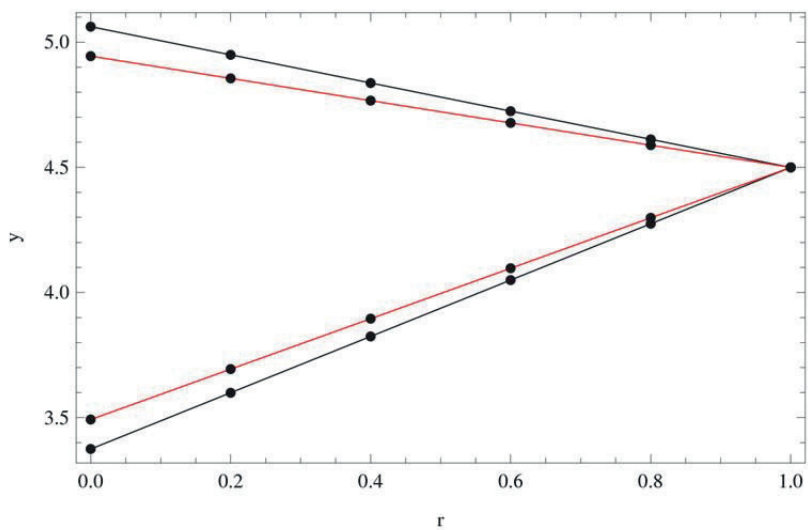


Figure 4. Comparison between Method 1and Method 2 of problem 1 with $\beta = 0.85$ and $h = 0.1$ at $x = 1$. (Black for Method 1, Red for Method 2)

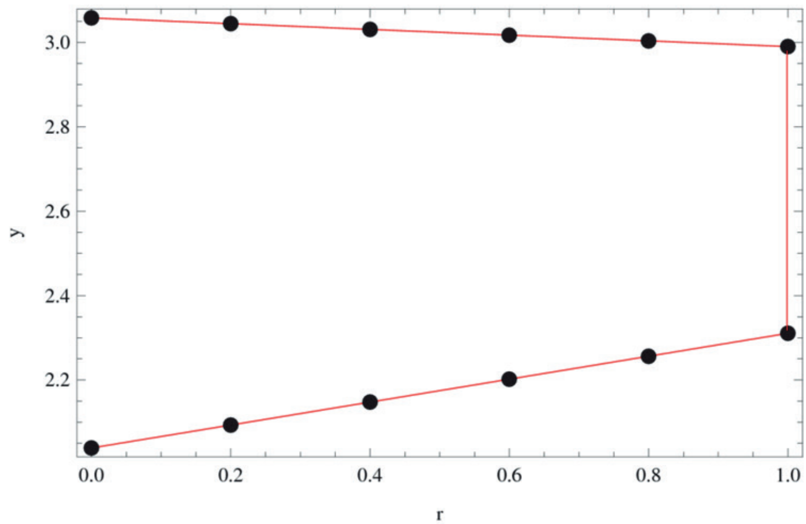


Figure 5. Comparison between exact and calculated values of problem 2 by Method 1 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

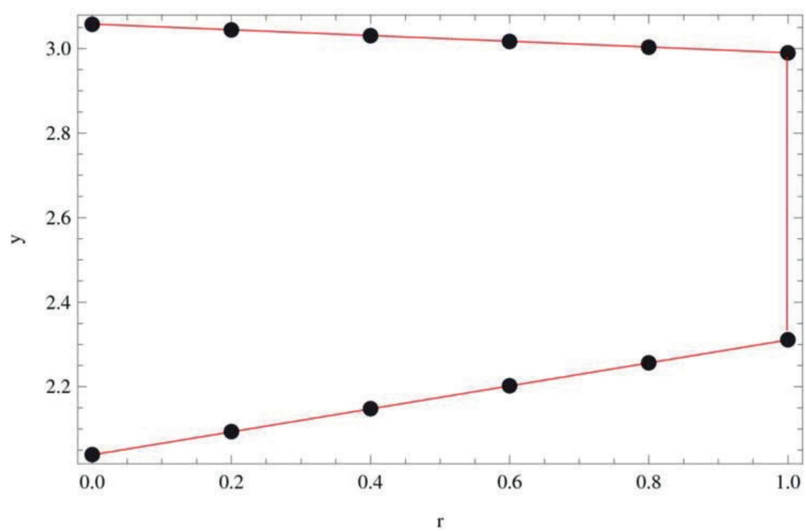


Figure 6. Comparison between Exact and calculated values of problem 2 by Method 2 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

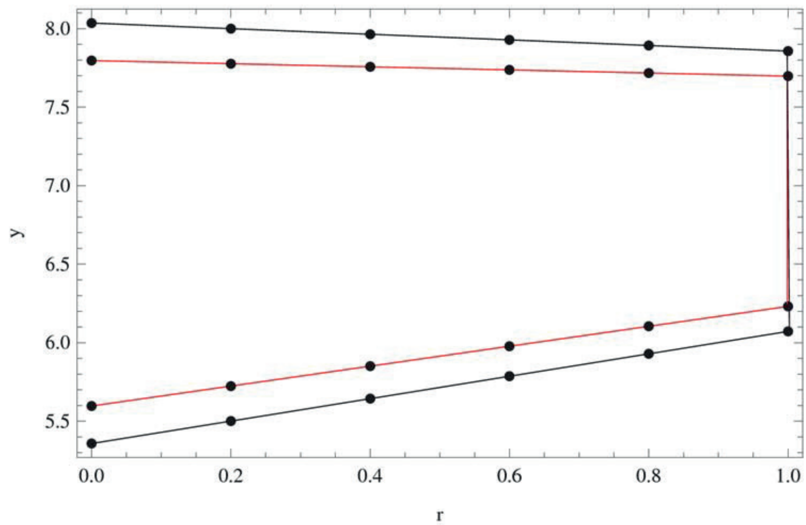


Figure 7. Comparison between Method 1and Method 2 of problem 2 with $\beta = 0.75$ and $h = 0.1$ at $x = 1$. (Black for Method 1, Red for Method 2)

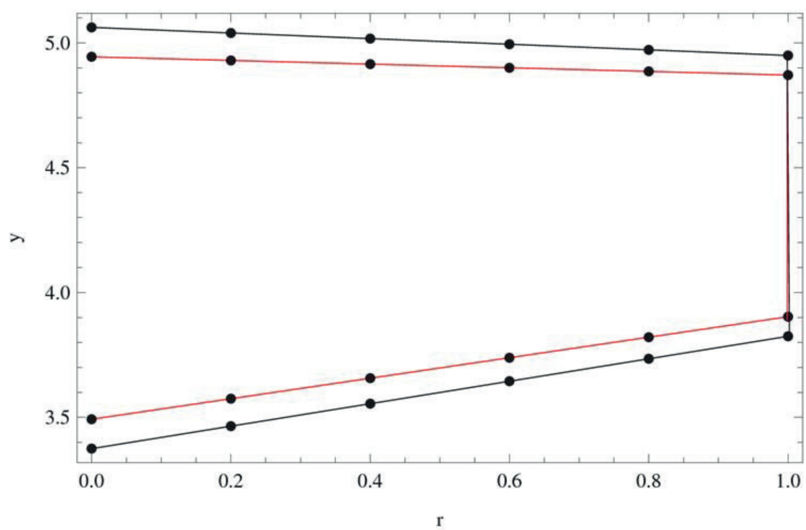


Figure 8. Comparison between Method 1and Method 2 of problem 2 with $\beta = 0.85$ and $h = 0.1$ at $x = 1$. (Black for Method 1, Red for Method 2)

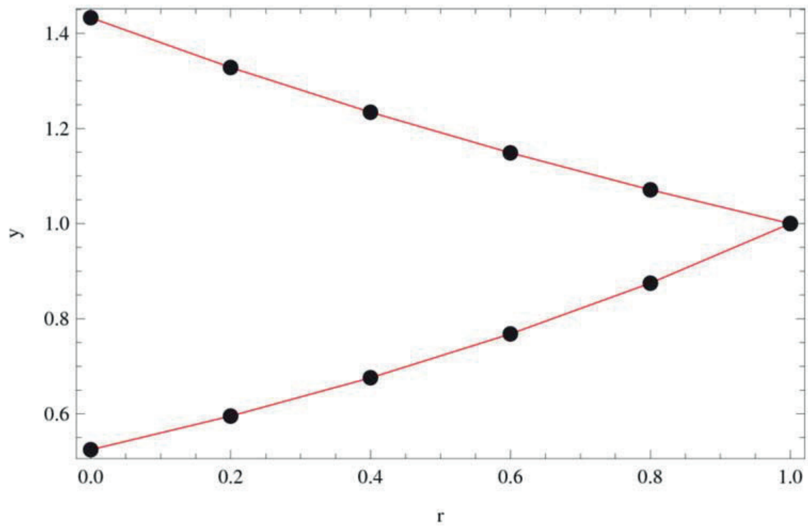


Figure 9. Comparison between Exact and calculated values of problem 3 by Method 1 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

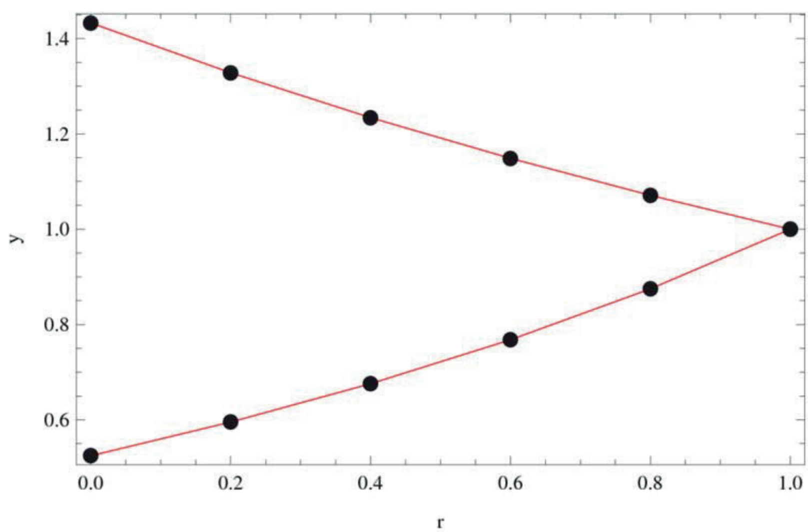


Figure 10. Comparison between Exact and calculated values of problem 3 by Method 2 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

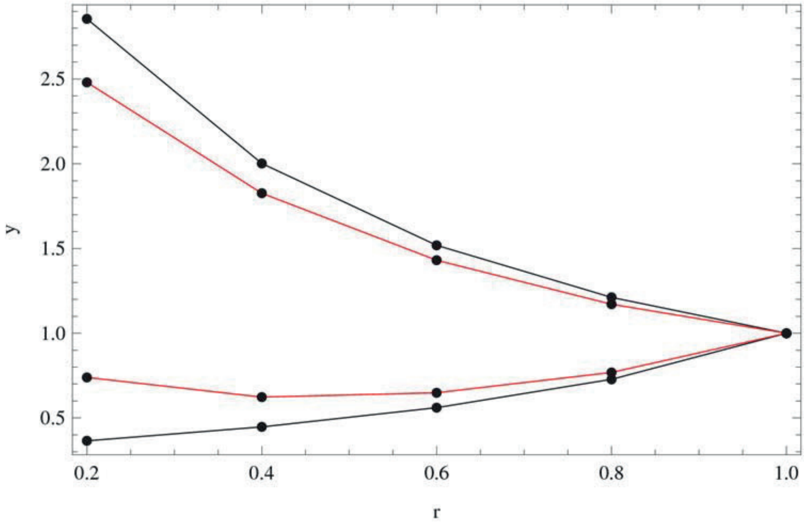


Figure 11. Comparison between Method 1and Method 2 of problem 3 with $\beta = 0.75$ and $h = 0.1$ at $x = 1$. (Black for Method 1, Red for Method 2)

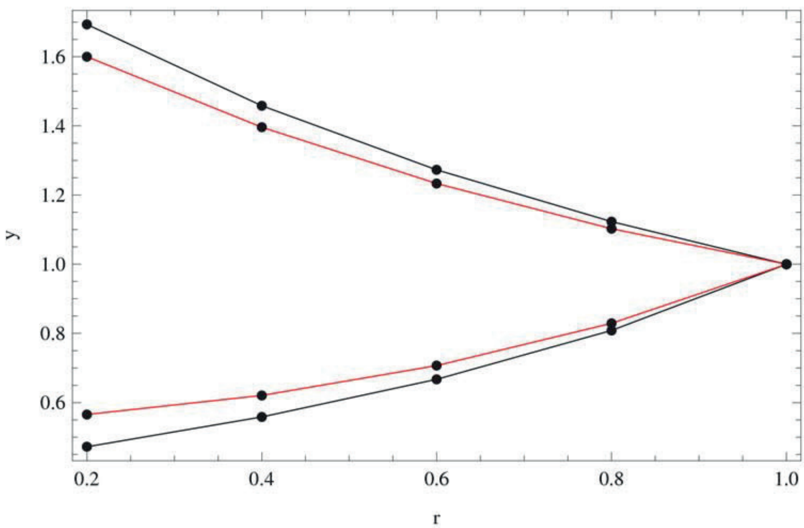


Figure 12. Comparison between Method 1and Method 2 of problem 3 with $\beta = 0.85$ and $h = 0.1$ at $x = 1$. (Black for Method 1, Red for Method 2)

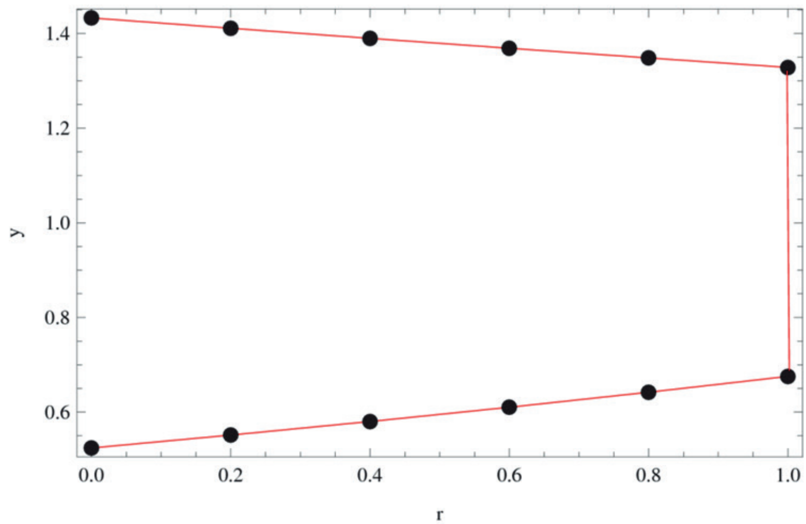


Figure 13. Comparison between Exact and calculated values of problem 4 by Method 1 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

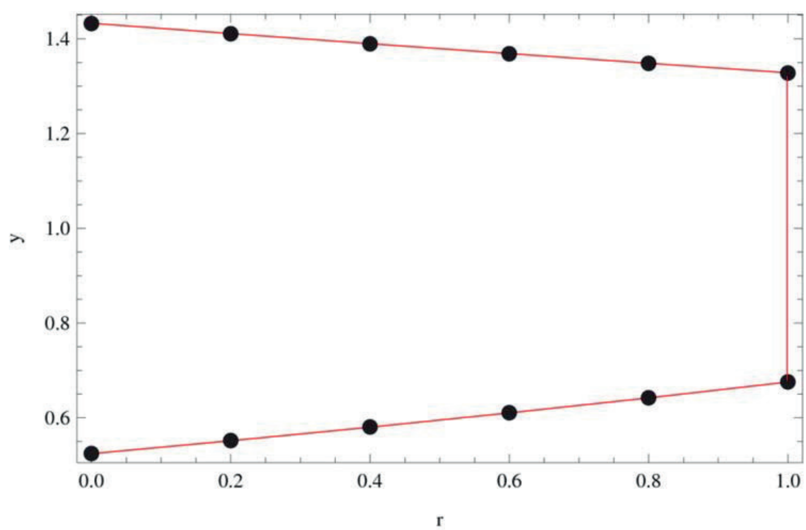


Figure 14. Comparison between Exact and calculated values of problem 4 by Method 2 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

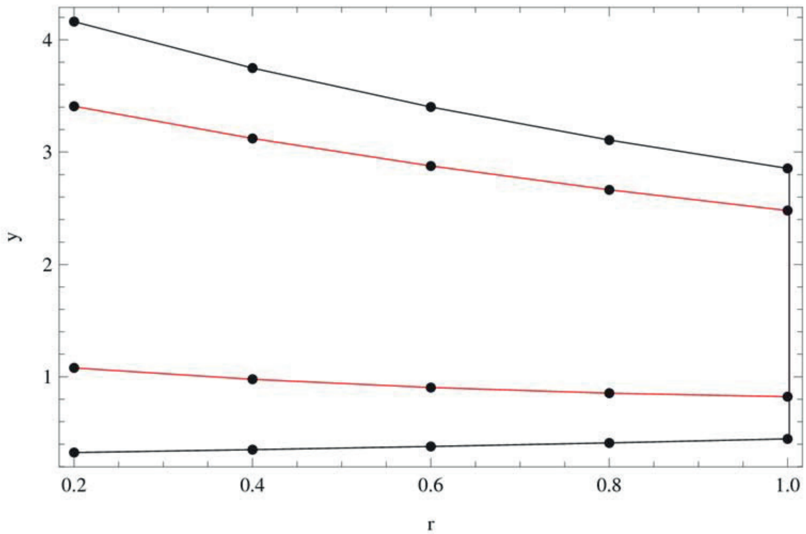


Figure 15. Comparison between Method 1and Method 2 of problem 4 with $\beta = 0.75$ and $h = 0.1$ at $x = 1$. (Black for Method 1, Red for Method 2)

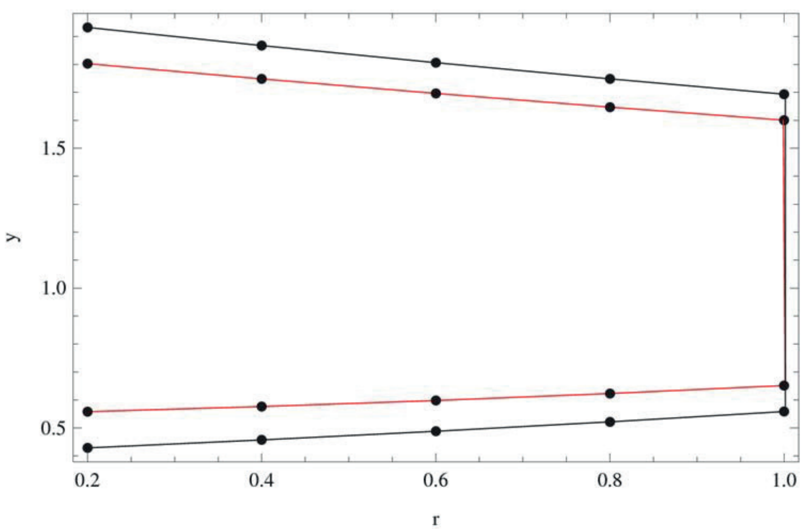


Figure 16. Comparison between Method 1and Method 2 of problem 4 with $\beta = 0.85$ and $h = 0.1$ at $x = 1$. (Black for Method 1, Red for Method 2)

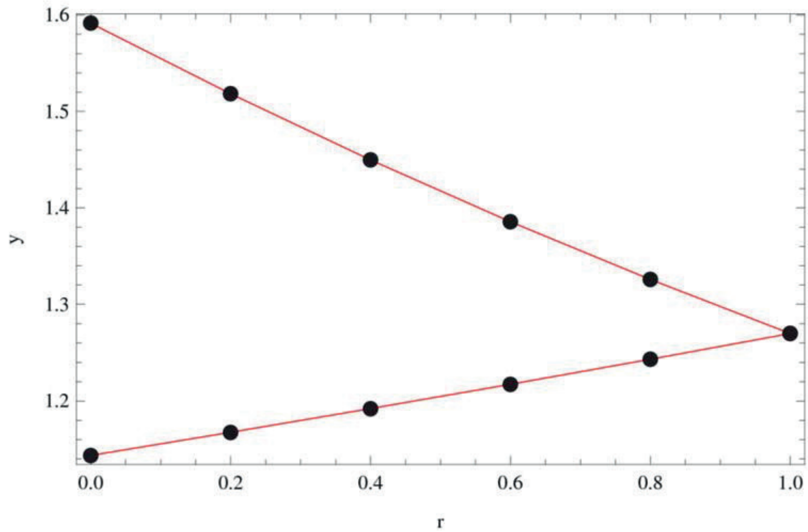


Figure 17. Comparison between Exact and calculated values of problem 5 by Method 1 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

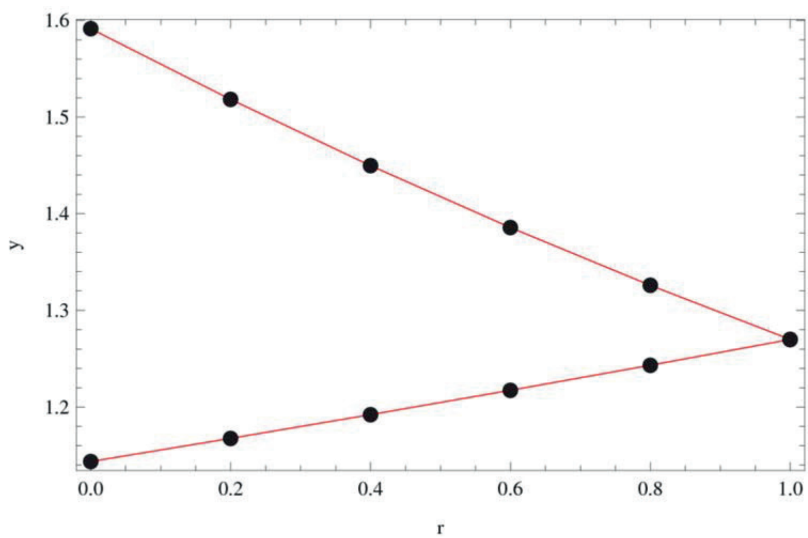


Figure 18. Comparison between exact and calculated values of problem 5 by Method 2 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

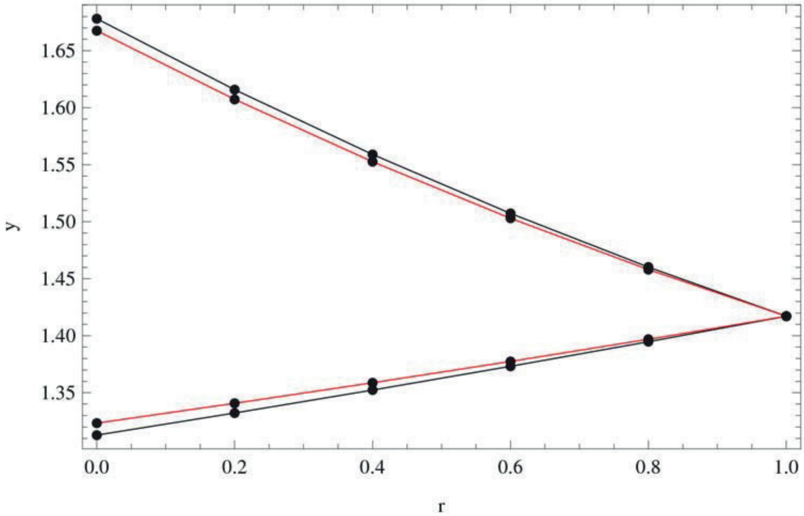


Figure 19. Comparison between Method 1and Method 2 of problem 5 with $\beta = 0.75$ and $h = 0.1$ at $x = 1$. (Black for Method 1, Red for Method 2)

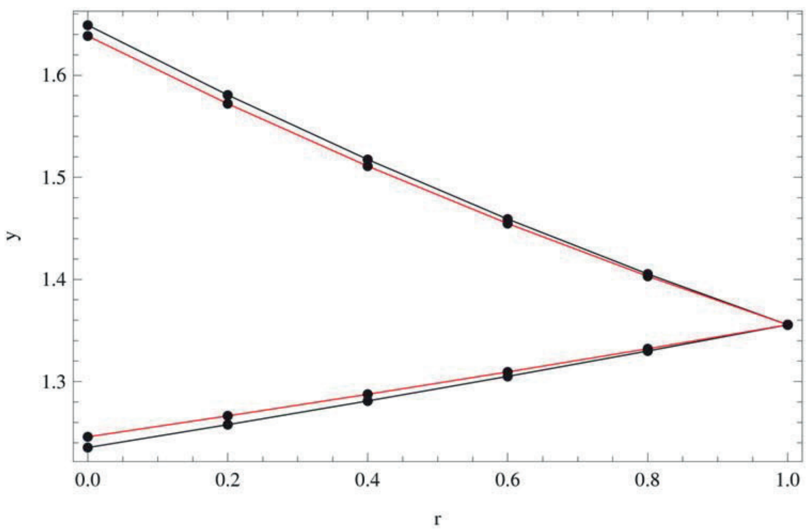


Figure 20. Comparison between Method 1and Method 2 of problem 5 with $\beta = 0.85$ and $h = 0.1$ at $x = 1$. (Black for Method 1, Red for Method 2)

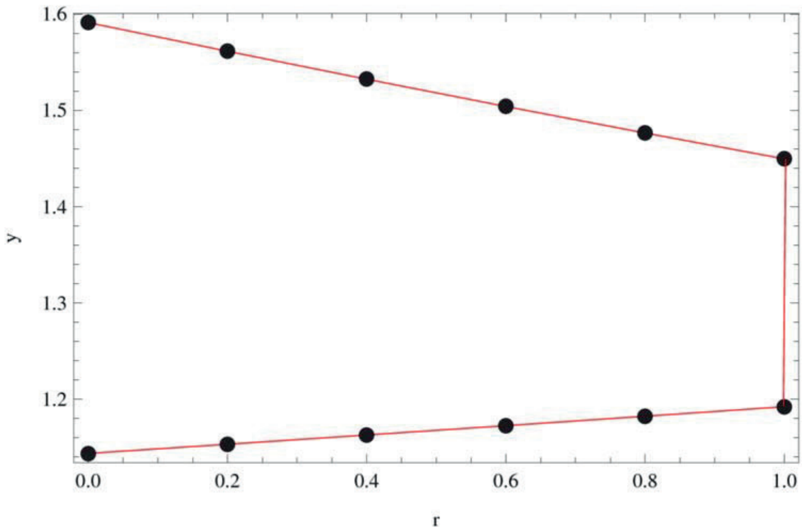


Figure 21. Comparison between Exact and calculated values of problem 6 by Method 1 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

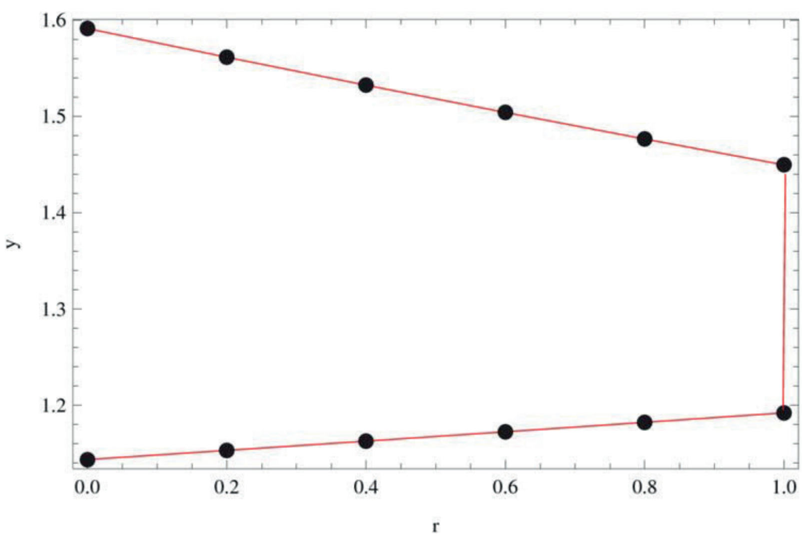


Figure 22. Comparison between Exact and calculated values of problem 6 by Method 2 with $\beta = 1$ and $h = 0.001$ at $x = 1$. (Exact represented by line and calculated represented by dots)

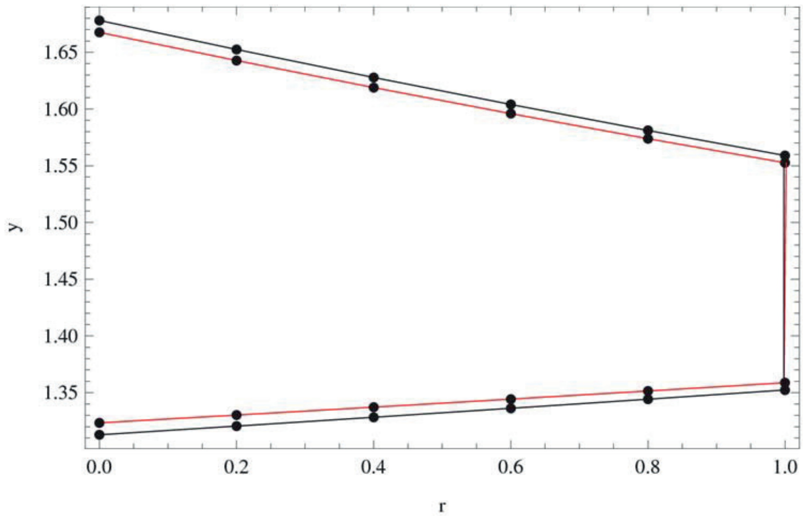


Figure 23. Comparison between Method 1and Method 2 of problem 6 with $\beta = 0.75$ and $h = 0.1$ at $x = 1$. (Black for Method 1, red for Method 2)

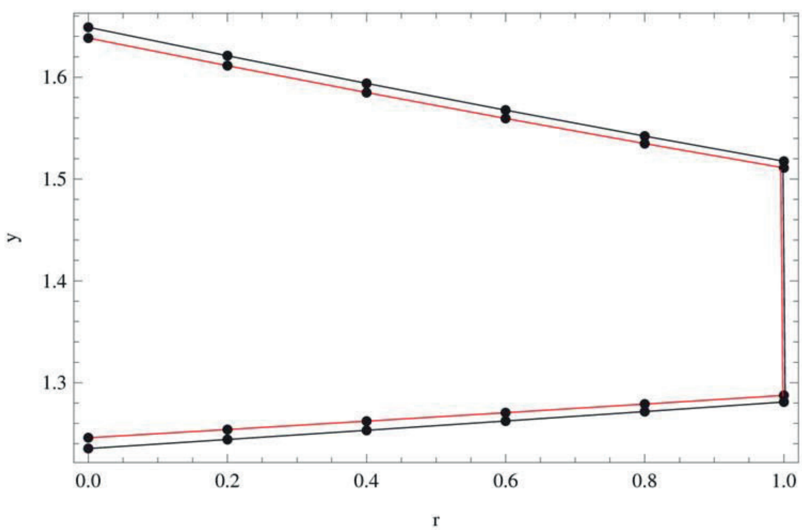


Figure 24. Comparison between Method 1and Method 2 of problem 6 with $\beta = 0.85$ and $h = 0.1$ at $x = 1$. (Black for Method 1, red for Method 2)

References

1. Kilbas AA, Srivastava HM, Trujillo JJ, 2006, *Theory and applications of fractional differential equations*, San Diego, Elsevier.
2. Bagley RL, 1990, *On the fractional order initial value problem and its engineering applications*, in Fractional Calculus and Its Applications, K. Nishimoto, Ed., College of Engineering, Nihon University, Tokyo, Japan 12–20.
3. Podlubny I, 1990, *Fractional differential equations*, Academic Press, San Diego.
4. Machado JT, Kiryakova V, Mainardi F, 2011, *Recent history of fractional calculus*. Communications in Nonlinear Science and Numerical Simulation 16: 1140–1153.
5. Diethelm K, Ford NJ, 2002, *Analysis of fractional differential equations*. Journal of Mathematical Analysis and Applications 265: 265-322.
6. Das S, 2008, *Functional fractional calculus for system identification and controls*. New York: Springer.
7. Adda FB, Cresson J, 2005, *Fractional differential equations and the Schrodinger equation*. Applied Mathematics and Computation 161: 323–345.
8. Fellah ZEA, Depollier C, Fellah M, 2002, *Application of fractional calculus to the sound waves propagation in rigid porous materials: validation via ultrasonic measurements*. Acta Acustica united with Acustica, 88: 34–39.
9. Rahimy M, 2012, *Applications of fractional differential equations*. Applied Mathematical Sciences 4: 2453–2461.
10. Khan NA, Mahmood A, Ara A, Khan NA, 2012, *Analytical study of nonlinear fractional order Integro-differential equation: Revisit Volterra's population model*. International Journal of Differential Equations: Article ID 845945.
11. Ahmad B, Nieto JJ, 2009, *Existence results for a coupled system of nonlinear fractional differential equations with three-point boundary conditions.* Computer and Mathematics with Applications 58: 1838-1843.
12. Jumarie G, 2009, *Table of some basic fractional calculus formulae derived from a improved Riemann-Liouville derivative for non-differentiable functions.* Applied Mathematics Letters 22: 378–385.
13. Khan NA, Khan NU, Ara A, Jamil M, 2012, *Approximate analytical solutions of fractional reaction-diffusion equations.* Journal of King Saud University – Science 24: 111–118.
14. Khan NA, Ara A, Mahmood A, 2012, *Numerical solutions of time fractional Burger equations: a comparison between generalized transformation technique with homotopy perturbation method.* International Journal of Numerical Method and Heat Fluid Flow 22: 175–93.
15. Jafari H, Tajadodi H, Matikolai SAH, 2011, *Homotopy perturbation pade technique for solving fractional Riccati differential equations.* The International Journal of Nonlinear Sciences and Numerical Simulation 11: 271–276.
16. Jafari H, Tajadodi H, Nazari H, Khalique CM, 2011, *Numerical solution of nonlinear Riccati differential equations with fractional order.* The International Journal of Nonlinear Sciencesand Numerical Simulation 11: 179–182.
17. Araya D, Lizama C, 2008, *Almost automorphic mild solutions to fractional differential equations.* Nonlinear Analysis 69: 3692-3705.

18. Khan NA, Ara A, Jamil M, 2011, *An efficient approach for solving the Riccati equation with fractional orders.* Computer and Mathematics with Applications 61: 2683-2689.
19. Abbasbandy S, Shirzadi A, 2010, *Homotopy analysis method for multiple solutions of the fractional Sturm–Liouville problems.* Numerical Algorithms 54: 521–532.
20. Khan NA, Jamil M, Ara A, Khan NU, 2011, *On efficient method for system of fractional differential equations.* Advances in Difference Equations: Article ADE/303472
21. Zadeh LA, 1965, *Fuzzy sets.* Information and Control 8: 338-353.
22. Zadeh LA, 1978*, Fuzzy sets as a basis for a theory of possibility.* Fuzzy Sets and Systems 1: 3-28.
23. Agarwal RP, Lakshmikantham V, Nieto JJ, 2010, *On the concept of solution for fractional differential equations with uncertainty.* Nonlinear Analysis 72: 2859-2862.
24. Ahmad MZ, Hasan MK, Abbasbandy S, 2013, *Solving fuzzy fractional differential equations using Zadeh's extension principle.* The ScientificWorld Journal 2013: Article ID 454969.
25. Arshad S, Lupulescu V, 2011, *On the fractional differential equations with uncertainty.* Nonlinear Analysis: Theory, Methods &Applications 74: 3685–3693.
26. Allahviranloo T, Salahshour S, Abbasbandy S, 2012, *Explicit solutions of fractional differential equations with uncertainty.* Soft Computing 16: 297–302.
27. Salahshour S, Allahviranloo T, Abbasbandy S, 2012, *Solving fuzzy fractional differential equations by fuzzy Laplace transforms.* Communications in Nonlinear Science and Numerical Simulation 17: 1372–1381.
28. Alikhani R, Bahrami F, 2013*, Global solutions for nonlinear fuzzy fractional integral and integro differential equations.* Communications in Nonlinear Science and Numerical Simulation 18: 2007–2017.
29. Tapaswini S, Chakraverty S, 2012, *A new approach to fuzzy initial value problem by improved Euler method.* Fuzzy Information and Engineering 3: 293-312.
30. Ochoche A, 2008, *Improving the improved Euler method for better performance on autonomous initial value problems.* Leonardo Journal of Sciences 12: 57 – 66.
31. Ma M, Friedman M, Kandel A, 1999*, Numerical solutions of fuzzy differential equations.* Fuzzy Sets and Systems 105: 133–138.
32. Mazandarani M, Kamyad AV, 2013, *Improved fractional Euler method for solving fuzzy fractional initial value problem.* Communications in Nonlinear Science and Numerical Simulation 18: 12–21.
33. Ahmad MZ, Hasan MK, Baets BD, 2013, *Analytical and numerical solutions of fuzzy differential equations.* Information Sciences 236: 156–167.
34. Allahviranloo T, Abbasbandy S, Salahshour S, Hakimzadeh A, 2011, *A new method for solving fuzzy linear differential equations.* Computing 92: 181–197.
35. Shokri J, 2007, Numerical solution of fuzzy differential equations. Applied Mathematical Sciences 1: 2231-2246.
36. Duraisamy C, Usha B, 2010, *Another approach to solution of fuzzy differential equations by improved Euler's method.* Proceedings of the International conference on Communication and Computational Intelligence-kongu Engineering College, Perundurai, Erode, TN, India: 52-55.

37. Odibat Z, Momani S, 2008, *An algorithm for the numerical solution of differential equations of fractional order.* Journal of Applied Mathematics & Informatics 26: 15--27.
38. Aminikhah H, Hemmatnezhad M, 2010, *An efficient method for quadratic Riccati differential equation.* Communication in Nonlinear Science and Numerical Simulation 15: 835--839.
39. Bede B, Gal SG, 2005, *Generalizations of the differentiability of fuzzy-number-valued functions with applications to fuzzy differential equations.* Fuzzy Sets System 151: 581--599.

Information for Authors

Authors are invited to submit original papers, within the scope of the JACSM, with the understanding that their contents are unpublished and are not being actively under consideration for publication elsewhere.

For any previously published and copyrighted material, a special permission from the copyright owner is required. This concerns, for instance, figures or tables for which copyright exists. In such a case, it is necessary to mention by the author(s), in the paper, that this material is reprinted with the permission.

Manuscripts, in English, with an abstract and the key words, are to be submitted to the Editorial Office in electronic version (both: source and pdf formats) via e-mail. A decision to accept/revise/reject the manuscript will be sent to the Author along with the recommendations made by at least two referees. Suitability for publications will be assessed on the basis of the relevance of the paper contents, its originality, technical quality, accuracy and language correctness.

Upon acceptance, Authors will transfer copyright of the paper to the publisher, i.e. the Academy of Management, in Lodz, Poland, by sending the paper to the JACSM Editorial Office.

The papers accepted for publication in the JACSM must be typeset by their Authors, according to the requirements concerning final version of the manuscripts. However, minor corrections may have been carried out by the publisher, if necessary.

The printing area is 122 mm × 193 mm. The text should be justified to occupy the full line width, so that the right margin is not ragged, with words hyphenated as appropriate. Please fill pages so that the length of the text is no less than 180 mm.

The first page of each paper should contain its title, the name of the author(s) with the affiliation(s) and e-mail address(es), and then the abstract and keywords, as show above. Capital letters must be applied in the title of a paper, and 14 pt. boldface font should be used, as in the example at the first page. Use: 11-point type for the name(s) of the author(s), 10-point type for the address(es) and the title of the abstract, 9-points type for the abstract and the key words.

For the main text, please use 11-point type and single-line spacing. We recommend using Times New Roman (TNR) fonts in its normal format. Italic type may be used to emphasize words in running text. Bold type and underlining should be avoided. With these sizes, the interline distance should be set so that some 43 lines occur on a full-text page.

The papers should show no printed page numbers; these are allocated by the Editorial Office.

Authors have the right to post their Academy of Management-copyrighted material from JACSM on their own servers without permission, provided that the server displays a prominent notice alerting readers to their obligations with respect to copyrighted material. Posted work has to be the final version as printed, include the copyright notice and all necessary citation details (vol, pp, no, …).

An example of an acceptable notice is:
"This material is presented to ensure timely dissemination of scholarly work, and its personal or educational use is permitted. Copyright and all rights therein are retained by the Copyright holder. Reprinting or re-using it in any form requires permission of the Copyright holder.

The submission format for the final version of the manuscript
and the MS Word template are available on our web site at:
http://acsm.spoleczna.pl

Journal of Applied Computer Science Methods

Contents