Appl. Math. J. Chinese Univ.
2021, 36(3): 370-389

A multiplicative Gauss-Newton minimization algorithm: Theory and application to exponential functions

Anmol Gupta Sanjay Kumar*

Abstract. Multiplicative calculus (MUC) measures the rate of change of function in terms of ratios, which makes the exponential functions significantly linear in the framework of MUC. Therefore, a generally non-linear optimization problem containing exponential functions becomes a linear problem in MUC. Taking this as motivation, this paper lays mathematical foundation of well-known classical Gauss-Newton minimization (CGNM) algorithm in the framework of MUC. This paper formulates the mathematical derivation of proposed method named as multiplicative Gauss-Newton minimization (MGNM) method along with its convergence properties. The proposed method is generalized for n number of variables, and all its theoretical concepts are authenticated by simulation results. Two case studies have been conducted incorporating multiplicatively-linear and non-linear exponential functions. From simulation results, it has been observed that proposed MGNM method converges for 12972 points, out of 19600 points considered while optimizing multiplicatively-linear exponential function, whereas CGNM and multiplicative Newton minimization methods converge for only 2111 and 9922 points, respectively. Furthermore, for a given set of initial value, the proposed MGNM converges only after 2 iterations as compared to 5 iterations taken by other methods. A similar pattern is observed for multiplicatively-non-linear exponential function. Therefore, it can be said that proposed method converges faster and for large range of initial values as compared to conventional methods.

§1 Introduction

Ever since the introduction of multiplicative calculus (*a.k.a.* non-Newtonian calculus) in the last quarter of the nineteenth century [9], tremendous applications have been found in numerous research fields. Two operations, differentiation, and integration, are basic operations in any calculus and analysis. The preliminary study of Newtonian calculus employs the measurement of the *instantaneous* rate of change of a certain quantity of interest over small intervals of time. The instantaneous rate of change of a function, say $f(t)$, as defined by Newton and Leibnitz,

Received: 2019-05-22. Revised: 2020-02-27.
MR Subject Classification: 00A71, 65Y20, 68Q25.
Keywords: multiplicative calculus, multiplicative least square method, multiplicative Newton minimization, multiplicative Gauss-Newton minimization, non-linear exponential functions.
Digital Object Identifier(DOI): https://doi.org/10.1007/s11766-021-3814-6.
*Corresponding author.

is given as

$$f'(t) = \lim_{h\to 0} \frac{1}{h}\left[f(t+h) - f(t)\right] \tag{1}$$

However, the notion of *multiplicative calculus* (MUC) states the change of paradigm i.e. it was anticipated that the variation between any two function values may be more naturally estimated if the deviations are measured by ratios; instead of differences as in Newtonian calculus [3]. Hence, the rate of change of a function in the sense of MUC is defined as

$$f^*(t) = \lim_{h\to 0} \left(\frac{f(t+h)}{f(t)}\right)^{\frac{1}{h}} \tag{2}$$

Arithmetically, the difference $f(t+h) - f(t)$ in (1) is replaced by the ratio $f(t+h)/f(t)$ in (2) and the division by h is replaced by raising the reciprocal power $1/h$. The limit in (1) is called the additive or classical derivative (denoted by $f'(t)$), whereas the limit in (2) is termed as multiplicative derivative (denoted by $f^*(t)$), because of the algebraic subtraction and division utilized in the definition of change rate, respectively.

Both of the expressions in (1) and (2) are meaningful to measure rate of change of quantities with respect to certain parameters in physical processes. However, depending on the physical process under consideration, one of these interpretations may be more appropriate than the other. For example, if $f(t)$ represents the mass of a radioactive substance and t stands for the time variable, then $f(t)$ will be an exponentially varying function of t. In that case, the interpretation in the sense of multiplicative calculus as in (2) will be more appropriate [6].

There are many important works in the literature considering the advantages and applications of MUC in numerous fields. Initially, Bashirov *et al.* [3] introduced the basic concepts of MUC along with its various applications, which put forward the theoretical investigation of MUC among researchers from various fields. In early stages, most of the reported applications of MUC were restricted to the fields of economics and finance, which involves study of growth and decay phenomenon (e.g. in radioactive decay, economic growth and bacterial growth) [6].

Recently, MUC has also made inroad to innumerable applications such as, numerical analysis [12], biomedical image analysis [7], complex analysis [16], image time-series analysis [2], exponential signal processing [5, 11], multiplicative differential equations [1, 14, 19], high-frequency wave scattering problems [15], contour detection in images [10, 20], to name a few. Thus, the potential applications of MUC have been increased rapidly in various fields.

The importance of multiplicative calculus has recently gained more insight of the research community as it was demonstrated that many useful theorems of additive calculus such as Cauchy's integral theorem, Cauchy's residue theorem, and Taylor series expansion formula can be reformulated in the framework of multiplicative calculus [16]. The applicability of these theorems concluded that analysing some of the engineering problems in the framework of multiplicative calculus could be more appropriate than the additive calculus [15]. Furthermore, an analogous to classical least square method (LSM); named as multiplicative least square method (MLSM) has been introduced and is applied to integrals for the finite product representation of the positive functions [11]. Some real applications have also been provided adequately to demonstrate that the product representation of non-linear exponential signals provides better accuracy and less computational complexity, as compared to the classical LSM.

On the other hand, multiplicative minimization methods, introduced in [12] have found astounding advantages as compared to classical minimization methods (such as Newton and

Gauss-Newton minimization methods) in solving non-linear least squares problems. Bilgehan [5] used the concept of MLSM to obtain an optimum representation for both linear and non-linear exponential signals. Furthermore, the challenging case of more complicated non-linear exponential signals has been resolved exceptionally by utilizing an additional analytical and numerical minimization technique, named as multiplicative Newton minimization (MNM) method [5].

Since many non-linear exponential functions are considerably linear in the framework of MUC, the nonlinear regression problems involving exponential functions can be solved more effectively, if the algorithm in classical calculus is reformulated in the framework of multiplicative calculus. The most efficient Newton minimization method which has been reformulated in MUC framework [12], is proved to be beneficial in providing an accurate representation for non-linear exponential signals [5]. However, the formulation of MNM method often requires the task of estimating the second-order multiplicative derivatives of an objective function, which can sometimes be challenging to compute [17]. The formulation of the minimization methods often demands the Hessian matrix to be invertible, which is not always possible when second-order derivatives are considered. Furthermore, the range of initial values for which the multiplicative Newton minimization method converges is limited.

Therefore, in this paper, a well-known Gauss-Newton minimization method is derived in the framework of MUC for n number of variables, as an alternative to classical Gauss-Newton minimization (CGNM) method. The proposed method is termed as *multiplicative Gauss-Newton minimization* (MGNM) method, which has advantages especially for representing non-linear exponential functions. To the best of author's knowledge, the formulation of this method in MUC framework has not been reported in the literature so far. Basically, the implication is that the proposed method is likely to be beneficial for applications where non-linear exponential functions are involved in parameter estimation; or where, convergence to the optimal solution is not obtained by the classical Gauss-Newton minimization method.

The rest of the paper proceeds as follows: Section 2 formulates a parameter estimation problem in the form of a regression equation. In Section 3, the analytical formulation of multiplicative Gauss-Newton minimization (MGNM) method is proposed for n number of variables and the essential mathematical background of multiplicative calculus is discussed. Section 4 discusses the properties of the proposed minimization method along with its convergence analysis. Simulation results and the comparison among various minimization methods in terms of rate of convergence and the maximum number of iterations are presented in Section 5. Finally, conclusions and future scope of the proposed work is summarized in Section 6.

§2 Problem Formulation

Consider the following parameter estimation problem:

Let $\mathbf{y}$ denotes the vector whose elements are n output samples of any single input single output system; i.e. $\mathbf{y} = [y_1, \ldots, y_n]$ corresponding to n input samples denoted by the vector $\mathbf{t}$ as $\mathbf{t} = [t_1, \ldots, t_n]$. Many parameter estimation problems often encountered in statistical signal processing require the task of establishing the relationship between the output samples y_i $(i = 1, 2, \ldots, n)$ and the independent input samples t_i of the system by means of a given collection of data. The analytical formulation of this relationship is usually expected to be known and is represented in the form of a *regression equation* as

$$\widehat{\mathbf{y}} = f(\boldsymbol{\beta}, \mathbf{t}) \tag{3}$$

where f is an analytically known variable function of unknown adjustable parameter values denoted in vector form as $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_m]^T$ where $m < n$. The major objective is to estimate the optimal values of these parameters from the set of observed outputs and its corresponding inputs. $\widehat{\mathbf{y}}$ is the estimated output vector computed using (3) as compared to the desired output vector denoted by $\mathbf{y}$.

The widely accepted method of estimation which is usually employed for estimating the unknown parameter values β_j $(j = 1, 2, \ldots, m)$ is the method of least squares. In the least square method (LSM), the difference between the desired output samples y_i and the estimated output samples $\widehat{y_i}$ i.e. $(y_i - \widehat{y_i})$ is evaluated based upon some initial values. Finally, the sum of squares of these differences is utilized as an objective function that is minimized as a function of these trial parameters i.e. the optimal parameters are estimated by minimizing the objective function expressed as

$$\mathrm{S}(\boldsymbol{\beta}) = (\mathbf{y}-\widehat{\mathbf{y}})\,(\mathbf{y}-\widehat{\mathbf{y}})^{\boldsymbol{T}} = \sum_{i=0}^{n} (y_i - f(\boldsymbol{\beta}, t_i))^2 \tag{4}$$

The classical error function between the actual value of the dependent variable and the value estimated by the model can be defined as $e_i(\boldsymbol{\beta}) = y_i - f(\boldsymbol{\beta}, t_i)$. LSM is a method which is most frequently utilized in the literature to estimate the unknown parameters of the system, and thus establishes a relationship between the input and the output variables [8]. However, when the regression function is non-linear in parameters, the logarithmic conversion is usually employed to calculate the parameters in classical LSM [18]. Furthermore, this conversion process requires more execution time and often turns out to be lossy. Therefore, this estimation procedure becomes considerably difficult in the classical framework.

Since many least square problems involve exponential fitting and the non-linear exponential functions are linear in MUC, multiplicative least square method (MLSM) is established [11] as an alternative to classical LSM. In addition, it has been inferred that if a quantity in a physical phenomenon exhibits an exponential variation, then its formulation in terms of the multiplicative calculus is expected to be more convenient. As a consequence, this feature of MUC is exploited here to derive the formulation of the proposed MGNM method for efficient approximation of any non-linear exponential model.

2.1 Preliminaries of Multiplicative Calculus

Before delving into the mathematics of multiplicative Gauss-Newton minimization method, the necessary mathematical tools such as exponential arithmetic and multiplicative matrix algebra needs to be discussed briefly. The detailed description of exponential arithmetic and multiplicative matrix algebra is present in [4]. Some of the major basic operations of exponential arithmetic are defined as [11]:

1. $a \oplus_{exp} b = e^{\ln a + \ln b} = ab$
2. $a \ominus_{exp} b = e^{\ln a - \ln b} = a/b$
3. $a \otimes_{exp} n = a \oplus_{exp} a \oplus_{exp} \cdots \oplus_{exp} a = a.a \ldots a = a^n$

4. $a \oslash_{exp} n = a^{1/n}$
where $\oplus_{exp}$, $\ominus_{exp}$, $\otimes_{exp}$, and $\oslash_{exp}$ represent multiplicative addition, subtraction, multiplication and division operators [11]. Thus, these points express the four basic operations of exponential arithmetic. Using exponential arithmetic as discussed in [11], the multiplicative least square method (MLSM) can explicitly be stated as

$$\overset{m}{\mathrm{S}}(\boldsymbol{\beta}) = exp\left\{\sum_{i=0}^{n} \ln\left(\frac{y_i}{f(\boldsymbol{\beta}, t_i)}\right)^2\right\} = \prod_{i=1}^{n} exp\left\{\ln\left(\frac{y_i}{f(\boldsymbol{\beta}, t_i)}\right)^2\right\} = \prod_{i=0}^{n}\left[\left(\frac{y_i}{f(\boldsymbol{\beta}, t_i)}\right)^{\ln\left(\frac{y_i}{f(\boldsymbol{\beta}, t_i)}\right)}\right] \tag{5}$$

where $\overset{m}{\mathrm{S}}(\boldsymbol{\beta})$ represents the multiplicative objective function in MLSM sense and the superscript m illustrates that the framework of MUC is considered. Therefore, (5) represents the formulation of multiplicative least squares method as compared to the classical LSM, defined in (4). Based on the exponential arithmetic defined above, let $\mathbf{A} = (a_{ij})$ be a matrix with elements a_{ij}, where $i, j \in \mathbb{N}$ and $\mathbf{x} = (x_j)$ be the vector sequence, then the matrix product in terms of multiplicative calculus is defined as:

$$\mathbf{A} \otimes \mathbf{x} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1j} \\ a_{21} & a_{22} & \dots & a_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{ij} \end{bmatrix} \otimes \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_j \end{bmatrix} = \begin{bmatrix} a_{11}^{\ln x_1} \cdot a_{12}^{\ln x_2} \dots a_{1j}^{\ln x_j} \\ a_{21}^{\ln x_1} \cdot a_{22}^{\ln x_2} \dots a_{2j}^{\ln x_j} \\ \vdots \\ a_{i1}^{\ln x_1} \cdot a_{i2}^{\ln x_2} \dots a_{ij}^{\ln x_j} \end{bmatrix} \tag{6}$$

where $\otimes$ represents multiplicative multiplication operator as defined in [4]. Therefore, this formulation of multiplicative matrix algebra and exponential arithmetic discussed above acts as the necessary mathematical tools to formulate the analytical derivation of the proposed MGNM method discussed in the next section.

§3 Multiplicative Gauss-Newton Minimization (MGNM) Method

The classical Gauss-Newton minimization method [17] is an iterative method which is usually employed for finding the minimum of an objective function defined in (4). The optimal parameter vector $\boldsymbol{\beta}$ that minimizes the objective error function $\mathrm{S}(\boldsymbol{\beta})$ should satisfy the condition that its gradient vector must be equal to zero, i.e.,

$$\frac{\partial}{\partial \beta_j}\mathrm{S}(\boldsymbol{\beta}) = \frac{\partial}{\partial \beta_j}\sum_{i=0}^{n}[y_i - f(\boldsymbol{\beta}, t_i)]^2 \tag{7}$$

Now, substituting $e_i(\boldsymbol{\beta}) = y_i - f(\boldsymbol{\beta}, t_i)$ in (7), implies

$$\frac{\partial}{\partial \beta_j}\mathrm{S}(\boldsymbol{\beta}) = \frac{\partial}{\partial \beta_j}\sum_{i=0}^{n} e_i(\boldsymbol{\beta})^2 = 2\sum_{i=0}^{n} e_i(\boldsymbol{\beta})\frac{\partial}{\partial \beta_j}e_i(\boldsymbol{\beta}) \tag{8}$$

where $\frac{\partial}{\partial \beta_j}e_i(\boldsymbol{\beta})$ represents the elements of the classical Jacobian matrix $\mathbf{J}$ defined as:

$$J_{ij} = \begin{bmatrix} \frac{\partial e_1(\boldsymbol{\beta})}{\partial \beta_1} & \cdots & \frac{\partial e_n(\boldsymbol{\beta})}{\partial \beta_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial e_n(\boldsymbol{\beta})}{\partial \beta_1} & \cdots & \frac{\partial e_n(\boldsymbol{\beta})}{\partial \beta_m} \end{bmatrix} \tag{9}$$

Therefore, the gradient $\boldsymbol{\nabla}$ of the objective function presented in (8) can be written in compact vector form as:

$$\boldsymbol{\nabla} = \frac{d}{d\boldsymbol{\beta}}\mathrm{S}(\boldsymbol{\beta}) = 2\mathbf{J}^T\mathbf{e}(\boldsymbol{\beta}) \tag{10}$$

However, these equations may not have a closed-form solutions for $\boldsymbol{\beta}$. Therefore, to find the optimal paramater values β_j, the following iteration is used:

$$\boldsymbol{\beta}^{(p+1)} = \boldsymbol{\beta}^{(p)} + \boldsymbol{\delta} \tag{11}$$

Starting with the initial guess $\boldsymbol{\beta}^{(0)}$, the value of the unknown parameters β_j $(j = 1, 2, \ldots, m)$ are updated at each iteration value p. Therefore, the main objective is to find the value of $\boldsymbol{\delta}$ such that the optimal parameters can be obtained using the recurrence relation presented in (11).

The recurrence relation for classical Newton minimization method, that is widely employed to find the optimal parameters of the objective function is given as [17]:

$$\boldsymbol{\beta}^{(p+1)} = \boldsymbol{\beta}^{(p)} - \mathbf{H^{-1}}\boldsymbol{\nabla} \tag{12}$$

where $\boldsymbol{\nabla}$ denotes the gradient vector defined in (10) and $\mathbf{H}$ represents the Hessian matrix of $\mathrm{S}(\boldsymbol{\beta})$. The elements of the Hessian matrix are calculated by differentiating the elements of the gradient vector $\boldsymbol{\nabla}$ defined in (8), with respect to parameters β_j, i.e.,

$$H_{ij} = 2\sum_{i=0}^{n}\left[\frac{\partial e_i(\boldsymbol{\beta})}{\partial \beta_j}\frac{\partial e_i(\boldsymbol{\beta})}{\partial \beta_k} + e_i(\boldsymbol{\beta})\frac{\partial^2 e_i(\boldsymbol{\beta})}{\partial \beta_j \partial \beta_k}\right] \tag{13}$$

Since the Gauss-Newton minimization method is obtained by eliminating the second-order derivative terms presented in the Hessian matrix of Newton's minimization method, the above equation (13) can be written as:

$$H_{ij} \cong 2\sum_{i=0}^{n}\frac{\partial e_i(\boldsymbol{\beta})}{\partial \beta_j}\frac{\partial e_i(\boldsymbol{\beta})}{\partial \beta_k} = 2\sum_{i=0}^{n} J_{ij}J_{ik} \tag{14}$$

Hence, the approximated Hessian matrix can further be represented in compact vector form as:

$$\mathbf{H} \cong 2\mathbf{J^T J} = \begin{bmatrix} \frac{\partial^2 e_1(\boldsymbol{\beta})}{\partial \beta_1^2} & \cdots & \frac{\partial^2 e_n(\boldsymbol{\beta})}{\partial \beta_1 \partial \beta_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 e_n(\boldsymbol{\beta})}{\partial \beta_m \partial \beta_1} & \cdots & \frac{\partial^2 e_n(\boldsymbol{\beta})}{\partial \beta_m^2} \end{bmatrix} \tag{15}$$

Substituting (10) and (15) into (11), gives the recurrence relation for classical Gauss-Newton minimization method as denoted by:

$$\boldsymbol{\beta}^{(p+1)} = \boldsymbol{\beta}^{(p)} - \left\{\left[\mathbf{J^T J}\right]^{-1}\left[\mathbf{J}^T\mathbf{e}\left(\boldsymbol{\beta}^{(\mathrm{p})}\right)\right]\right\} \tag{16}$$

where $\mathbf{e}$ and $\boldsymbol{\beta}$ represents the column vectors of error function and parameters to be estimated, respectively and the symbol ^{T} denotes the matrix transpose. Thus, (16) represents the recurrence relation for CGNM method. The similar approach is utilized further to derive the recurrence relation for multiplicative Gauss-Newton minimization method.

Since many non-linear exponential functions are linear in the MUC framework, a more robust version of Gauss-Newton minimization method can be formulated by employing the concepts of MUC. Thus, the *iterative formulation of the proposed MGNM method* is achieved by minimizing

$$\overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta}) = \prod_{i=1}^{n}\left[\left(\frac{y_i}{f(\boldsymbol{\beta}, t_i)}\right)^{\ln\left(\frac{y_i}{f(\boldsymbol{\beta}, t_i)}\right)}\right] \tag{17}$$

where $\overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})$ represents the multiplicative objective function in the multiplicative least square sense, as defined in (5) (*here superscript indicates that the framework of MUC is considered*). To derive the formulation for the proposed MGNM method, the error function $\overset{m}{e}_i(\boldsymbol{\beta}) = y_i/f(\boldsymbol{\beta}, t_i)$ is considered; which is an alternative to the error function $e_i(\boldsymbol{\beta}) = y_i - f(\boldsymbol{\beta}, t_i)$ defined in the classical sense. Therefore, the multiplicative objective function, $\overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})$ as defined in (17) is minimized using proposed MGNM method and can be described in compact form as:

$$\overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta}) = \prod_{i=1}^{n} \left[\left(\overset{m}{e}_i(\boldsymbol{\beta}) \right)^{\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)} \right] \tag{18}$$

To begin with, the multiplicative gradient vector (denoted by $\overset{\mathbf{m}}{\boldsymbol{\nabla}}$) corresponding to the multiplicative objective function $\overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})$ with respect to parameter values β_j can be defined as:

$$\overset{\mathbf{m}}{\boldsymbol{\nabla}} = \left[\frac{\partial^* \overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})}{\partial \beta_1}, \frac{\partial^* \overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})}{\partial \beta_2}, \ldots, \frac{\partial^* \overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})}{\partial \beta_m} \right] \tag{19}$$

The elements of multiplicative gradient vector ($\overset{\mathbf{m}}{\boldsymbol{\nabla}}$) are calculated by using the definition of multiplicative derivatives [3] as

$$\begin{aligned} \left(\frac{\partial^* \overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})}{\partial \beta_j} \right) = \exp\left\{ \frac{\partial}{\partial \beta_j} \left[\ln \left(\prod_{i=1}^{n} \left(\overset{m}{e}_i(\boldsymbol{\beta}) \right)^{\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)} \right) \right] \right\} &= \exp\left\{ \frac{\partial}{\partial \beta_j} \left[\sum_{i=1}^{n} \left(\ln \left(\overset{m}{e}_i(\boldsymbol{\beta}) \right) \right)^2 \right] \right\} \\ &= \exp\left\{ 2 \sum_{i=1}^{n} \left[\tfrac{\partial}{\partial \beta_j} \left(\ln(\overset{m}{e}_i(\boldsymbol{\beta}) \right) \right] \ln\left(\overset{m}{e}_i(\boldsymbol{\beta}) \right) \right\} \end{aligned} \tag{20}$$

Therefore, by using (20), the multiplicative gradient vector $\overset{\mathbf{m}}{\boldsymbol{\nabla}}$ can be written in compact form:

$$\overset{\mathbf{m}}{\boldsymbol{\nabla}} = \exp\left[2 \overset{\mathbf{m}}{\mathbf{J}}^T (\ln \overset{\mathbf{m}}{\mathbf{e}}) \right] \tag{21}$$

where $\overset{\mathbf{m}}{\mathbf{e}}$ represents the column vector of the multiplicative error function and $\overset{\mathbf{m}}{\mathbf{J}}$ represents the multiplicative Jacobian matrix with entries

$$\overset{\mathbf{m}}{\mathbf{J}}_{ij} = \frac{\partial}{\partial \beta_j} \left[\ln \left(\overset{m}{e}_i(\boldsymbol{\beta}) \right) \right] \tag{22}$$

and hence the multiplicative Jacobian matrix can be represented as:

$$\overset{\mathbf{m}}{\mathbf{J}} = \begin{bmatrix} \frac{\partial}{\partial \beta_1} \left[\ln \left(\overset{m}{e}_1(\boldsymbol{\beta}) \right) \right] & \cdots & \frac{\partial}{\partial \beta_m} \left[\ln \left(\overset{m}{e}_1(\boldsymbol{\beta}) \right) \right] \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial \beta_1} \left[\ln \left(\overset{m}{e}_n(\boldsymbol{\beta}) \right) \right] & \cdots & \frac{\partial}{\partial \beta_m} \left[\ln \left(\overset{m}{e}_n(\boldsymbol{\beta}) \right) \right] \end{bmatrix} \tag{23}$$

Now, the elements of multiplicative gradient vector $\overset{\mathbf{m}}{\boldsymbol{\nabla}}$ are again multiplicatively differentiated with respect to the parameter value β_j and a matrix named as multiplicative Hessian matrix $\overset{\mathbf{m}}{\mathbf{H}}$ is computed which can be described as:

$$\overset{\mathbf{m}}{\mathbf{H}} = \begin{bmatrix} \frac{\partial^{**} \overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})}{\partial \beta_1^2} & \cdots & \frac{\partial^{**} \overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})}{\partial \beta_1 \partial \beta_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial^{**} \overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})}{\partial \beta_m \partial \beta_1} & \cdots & \frac{\partial^{**} \overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})}{\partial \beta_m^2} \end{bmatrix} \tag{24}$$

The elements of multiplicative Hessian matrix $\overset{\mathbf{m}}{\mathbf{H}}$ can be computed as:

$$\begin{aligned}\left(\frac{\partial^{**}\overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})}{\partial\beta_i\partial\beta_j}\right) &= \frac{\partial^*}{\partial\beta_i}\left\{\exp\left[2\sum_{i=1}^{n}\ \ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)\ \left(\tfrac{\partial}{\partial\beta_j}\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)\ \right)\ \right]\right\} \\ &= \exp\left\{\frac{\partial}{\partial\beta_i}\left[2\sum_{i=1}^{n}\ \ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)\ \left(\tfrac{\partial}{\partial\beta_j}\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)\ \right)\ \right]\right\}\end{aligned} \tag{25}$$

$$= \exp\left\{2\left[\sum_{i=1}^{n}\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)\ \left(\frac{\partial^2}{\partial\beta_i\partial\beta_j}\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)\ \right)+\left(\ \tfrac{\partial}{\partial\beta_i}\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)\ \ \frac{\partial}{\partial\beta_j}\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)\ \right)\right]\right\} \tag{26}$$

The benefit of Gauss-Newton minimization method lies in the fact that it avoids the evaluation of second-order derivatives of the objective function, which could be both analytically and computationally expensive to compute [17]. Therefore, the proposed MGNM method is obtained by ignoring the multiplicative second-order derivative term i.e., the elements of multiplicative Hessian matrix $\overset{\mathbf{m}}{\mathbf{H}}$ are approximated as

$$\left(\frac{\partial^{**}\overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta})}{\partial\beta_i\partial\beta_j}\right)\approx\exp\left\{2\sum_{i=1}^{n}\left[\ \tfrac{\partial}{\partial\beta_i}\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)\ \ \ \frac{\partial}{\partial\beta_j}\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)\ \right]\right\} \tag{27}$$

Therefore, the multiplicative Hessian matrix $\overset{\mathbf{m}}{\mathbf{H}}$ can be written in compact form as:

$$\overset{\mathbf{m}}{\mathbf{H}} = \exp\left[2\overset{\mathbf{m}}{\mathbf{J}}^T\overset{\mathbf{m}}{\mathbf{J}}\right] \tag{28}$$

where $\overset{\mathbf{m}}{\mathbf{J}}$ and $\overset{\mathbf{m}}{\mathbf{J}}^T$ represent the multiplicative Jacobian matrix and its transpose as illustrated in (23).

The final iterative criterion for the proposed MGNM method is then derived by utilizing the formulation of multiplicative Newton minimization method [12], expressed as

$$\beta^{(p+1)} = \beta^{(p)} - \frac{\ln f^*(\beta^{(p)})}{\ln f^{**}(\beta^{(p)})} \tag{29}$$

where $f^*(\beta)$ and $f^{**}(\beta)$ denotes the multiplicative first-order and second-order derivatives, respectively of any function f with respect to any parameter β. Replacing the multiplicative first-order (i.e. $f^*(\boldsymbol{\beta})$) and second-order (i.e. $f^{**}(\boldsymbol{\beta})$) derivative terms presented in (29) by the elements of multiplicative gradient vector (i.e. $\overset{\mathbf{m}}{\boldsymbol{\nabla}}$) and multiplicative Hessian (i.e. $\overset{\mathbf{m}}{\mathbf{H}}$) matrix presented in (21) and (24) respectively, yields:

$$\boldsymbol{\beta}^{(p+1)} = \boldsymbol{\beta}^{(p)} - \left\{\left[2\overset{\mathbf{m}}{\mathbf{J}}^T\overset{\mathbf{m}}{\mathbf{J}}\right]^{-1}\left[2\ \overset{\mathbf{m}}{\mathbf{J}}^T(\ln\overset{\mathbf{m}}{\mathbf{e}}\ )\right]\right\} \tag{30}$$

Thus, (30) represents an iterative criterion for the proposed MGNM method which is utilized further to find the optimal parameter values of any non-linear exponential model.

The proposed MGNM method differs from the CGNM method in the sense that how the objective functions are defined and how the multiplicative derivatives are employed instead of the classical derivatives. The main implication is that whenever non-linear exponential functions are encountered, the variation between the desired output and the estimated output is calculated more effectively in terms of ratios; instead of differences as in Newtonian calculus. The theoretical and analytical foundation of proposed MGNM method is represented in an algorithm form in Algorithm 1.

Stopping Criteria (STOP-CRIT): The stopping criterion for the proposed MGNM method can be formalized analogously to that of CGNM method. The algorithm stops as soon as one of the following conditions is met i.e. when

1. The maximum number of iterations 'p' (*as specified by the user*) is reached: $p > p_{max}$.
2. The current objective function value reaches the user-specified threshold value σ_{user} ($\overset{m}{\sigma}_{user}$ in MUC framework) i.e.

In the classical framework, when the difference between the desired output samples and the estimated output samples is approximately equal to zero or any other user-specified value (say 10^{-3}).

$$\mathrm{S}\left(\boldsymbol{\beta}\right) \leq \sigma_{user}; \sigma_{user} \approx 0 \tag{31}$$

Similarly, in MUC framework, when the ratio between the desired output samples and the estimated output samples is approximately equal to one.

$$\overset{\mathrm{m}}{\mathrm{S}}\left(\boldsymbol{\beta}\right) \leq \overset{m}{\sigma}_{user}; \overset{m}{\sigma}_{user} \approx 1$$

Algorithm 1: Proposed Multiplicative Gauss-Newton minimization (MGNM) method

Input: $\overset{\mathrm{m}}{\mathrm{S}}$: An objective function to be minimized such that
$\overset{\mathrm{m}}{\mathrm{S}}(\boldsymbol{\beta}) = \prod_{i=1}^{n}\left[\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)^{\ln\left(\overset{m}{e}_i(\boldsymbol{\beta})\right)}\right]$
$\boldsymbol{\beta}^{(0)}$: Initial value of the parameters to be estimated.
$\overset{m}{\sigma}_{user}$: a user-specified threshold value for convergence i.e. $\overset{m}{\sigma}_{user} = 1$

Output: $\boldsymbol{\beta}_{min}$, local minima of the cost function $\overset{\mathrm{m}}{\mathrm{S}}$; i.e. the optimal parameters.

begin
- $p \leftarrow 0$;
- **while** STOP $-$ CRIT **do**
  - $\boldsymbol{\beta}^{(p+1)} = \boldsymbol{\beta}^{(p)} + \delta^{(p)}$;
  - with $\delta^{(p)} = -\left\{\left(2\overset{\mathbf{m}}{\mathbf{J}}^{T}\overset{\mathbf{m}}{\mathbf{J}}\right)^{-1}\left[2\,\overset{\mathbf{m}}{\mathbf{J}}^{T}(\ln\overset{\mathbf{m}}{\mathbf{e}}\ )\right]\right\}$;
  - $p \leftarrow p + 1$;
- **return** $\boldsymbol{\beta}^{(p)}$

end

§4 Convergence Properties of Multiplicative Minimization

The selection of initial values plays an important role in determining the convergence of minimization methods. Before considering the convergence properties of multiplicative minimization methods, multiplicative tests for monotonicity, local extremum, multiplicative mean value theorems, discussed in [3] should be considered.

Properties of the Proposed MGNM Method: The convergence of the proposed multiplicative minimization method in terms of first-order and second-order necessary conditions can be illustrated as:

1. Let $f \in \mathtt{C}^2[a,b]$ and f is a positive function $\forall\ t\ \in (a,b)$. There exists a strict local minimum $\beta \in (a,b)$ of the function f if $f^{*}\left(\beta\right) = 1$ and $f^{**}\left(\beta\right) > 1$.
2. In the proposed multiplicative Gauss-Newton minimization method, the approximation error is small if either the residual term $\ln\left(\overset{m}{e}_i(\beta)\right)$ or the second-order derivative term $\left(\frac{\partial^2}{\partial^2\beta}\ln\left(\overset{m}{e}_i(\beta)\right)\right)$ is small.
3. Moreover, the convergence of the proposed MGNM method requires that the approximated multiplicative Hessian matrix must be positive definite i.e. the matrix $\overset{\mathbf{m}}{\mathbf{J}}$ must have a full column rank.

Range of Convergence: The convergence range analysis of multiplicative minimization methods, provided in [12] and [13] is considered and hence, the range of convergence of the proposed MGNM method is stated analogously as follows:

Theorem 1. Let the multiplicative objective function $\overset{m}{S}$ (as given in (8)), is a positive function $\forall\ t \in [a, b]$ and $\overset{m}{S} \in \mathbb{C}^2[a, b]$. Assume that there exists a number $\beta \in (a, b)$ such that $\overset{m}{S}(\beta) = 1$. If $\overset{m*}{S}(\beta) \neq 1$ and $g(x) = x - \left\{ \left[2\overset{m}{\mathbf{J}}^T \overset{m}{\mathbf{J}} \right]^{-1} \left[2\overset{m}{\mathbf{J}}^T (\ln \overset{m}{\mathbf{e}}) \right] \right\}$ then there exist a number $\delta > 0$ such that the sequence $\{\beta^p\}_{p=1}^{\infty}$ defined by the iteration

$$\beta^{\mathrm{p}} = \mathrm{g}\left(\beta^{\mathrm{p}-1}\right) = \beta^{\mathrm{p}-1} - \left\{ \left[2\overset{m}{\mathbf{J}}^T \overset{m}{\mathbf{J}} \right]^{-1} \left[2\overset{m}{\mathbf{J}}^T (\ln \overset{m}{\mathbf{e}}) \right] \right\};\ \forall\ p = 1, 2, 3, \ldots$$

will converge to β for any initial point $\beta^{(0)} \in [\beta - \delta, \beta + \delta]$ and δ can be selected such that $\left(e^{g(\beta)}\right)^* < e$; for all $\beta \in [\beta - \delta, \beta + \delta]$. The proof of this theorem can be conducted analogously as given in [13]. The validation of convergence for a large range of initial values of the proposed MGNM method has been provided in Section 5.

§5 Results and Discussion

To demonstrate the effectiveness of the proposed method, some simulation examples are presented, which are identical to the ones utilized in [5]. The results of the proposed method are tested against classical Newton, classical Gauss-Newton and multiplicative Newton minimization methods for the function of two variables. In all the simulations performed, the estimation accuracy of the proposed method is verified for different initial values.

Example 1: Example 2, presented in [5] is utilized here to illustrate the effectiveness of the proposed method. In this example, the system is said to have an exponentially increasing response denoted by $f(\beta_1,\ \beta_2) = \beta_1 e^{\beta_2 t}$. The major objective is to estimate unknown parameters $\beta_1,\ \beta_2$ for test input signal.

The optimal parameters as indicated by the simulation results are $\beta_1 = 2.541$ and $\beta_2 = 0.2595$. Table 1 and Table 2 demonstrate the iteration values of the proposed MGNM method for different initial values. The initial values $(2.3, 0.2)$ and $(1.9, 1)$ are used for the function $f(\beta_1,\ \beta_2) = \beta_1 e^{\beta_2 t}$. Furthermore, Table 1 and Table 2 presents the comparison among various minimization methods such as classical Gauss-Newton minimization method (CGNM), multiplicative Newton minimization (MNM) method, and the proposed multiplicative Gauss-Newton minimization (MGNM) method for different initial values.

Case 1: For initial values $\boldsymbol{\beta_1^{(0)} = 2.3}$, $\boldsymbol{\beta_2^{(0)} = 0.2}$: From the results presented in Table 1, it can be observed that the proposed MGNM method converges after 2 iterations for the given initial values whereas CGNM and MNM method take at most 5 iterations to reach to the optimal solution. However, the classical Newton minimization (CNM) method diverges for the selected initial values as given in Table 2, presented in [5]. The results of CNM method are omitted here for brevity.

Furthermore, the performance of the proposed method is evaluated in terms of relative error for each parameter separately. The formula utilized for computing the relative error is given as $re_{\beta_i} = |1 - \beta_{i_{approx}}/\beta_{i_{exact}}|$, where $\beta_{i_{approx}}$ represents the estimated value of the parameter β_i using minimization methods and $\beta_{i_{exact}}$ represents the optimal value of the parameter β_i. The simulation results verify that the relative errors for the proposed MGNM method are comparatively less than the CNM, CGNM, and MNM methods.

Table 1. Iteration results of CGNM, MNM, and the proposed MGNM method for function $f(\beta_1, \beta_2) = \beta_1 e^{\beta_2 t}$, where β_1, β_2 denotes the results of the p^{th} iteration for initial values $\beta_1^{(0)} = 2.3, \beta_2^{(0)} = 0.2$ with tabulated relative errors represented by re_{β_1} and re_{β_2}.

	CGNM [5]				**MNM** [5]				**Proposed MGNM**			
p	β_1	re_{β_1}	β_2	re_{β_2}	β_1	re_{β_1}	β_2	re_{β_2}	β_1	re_{β_1}	β_2	re_{β_2}
0	*2.3*		*0.2*		*2.3*		*0.2*		*2.3*		*0.2*	
1	0.420	0.8347	0.290	0.1175	2.402	0.0547	0.269	0.0366	2.529	0.0047	**0.2595**	0
2	2.511	0.0118	0.263	0.0134	2.529	0.0047	0.260	0.0019	**2.541**	0	0.2595	0
3	2.540	0.0003	**0.259**	0.0019	2.540	0.0003	**0.259**	0.0019	2.541	0	0.2595	0
4	**2.541**	0	0.259	0.0019	**2.541**	0	0.259	0.0019	2.541	0	0.2595	0
5	2.541	0	0.259	0.0019	2.541	0	0.259	0.0019	2.541	0	0.2595	0

Fig. 1 represents the behavior of the objective function on the contour plot for CGNM, MNM, and MGNM methods. (*The contour plot representation of classical Newton minimization (CNM) method is omitted here because the CNM method diverges for the selected initial values*). The initial value of the iteration and the optimal point of convergence are depicted in Fig. 1. From Fig. 1, it can be observed that the solution of the proposed MGNM method immediately heads-off in the direction of the optimal point, following the direct path as compared to CGNM and MNM methods. This clearly depicts that *the convergence rate of the proposed MGNM method is faster than the other methods.*
In addition, Fig. 2 presents the comparison among various minimization methods in terms of the

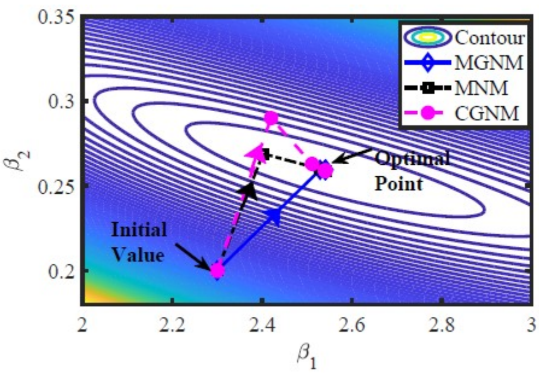


Figure 1. Contour plot representation of the objective function given by (17) for CGNM, MNM, and the proposed MGNM method, for initial values $\beta_1^{(0)} = 2.3, \beta_2^{(0)} = 0.2$.

iteration number versus the relative error plots for each parameter separately. From Fig. 2, it can be interpreted that the convergence to the optimal solution in the case of proposed MGNM method is reached in the minimum number of iterations with relative error far less than the CNM method and comparatively less than the CGNM and MNM methods.

Case 2: For initial values $\boldsymbol{\beta_1^{(0)} = 1.9}$, $\boldsymbol{\beta_2^{(0)} = 1}$: The results presented in Table 2 demonstrates the comparison among various minimization methods for initial values $\beta_1^{(0)} = 1.9$, $\beta_2^{(0)} = 1$. From the results, it can be observed that the proposed MGNM method converges only after 3 iterations whereas CGNM method diverges for the selected initial values and MNM method shows convergence after at most 5 iterations.

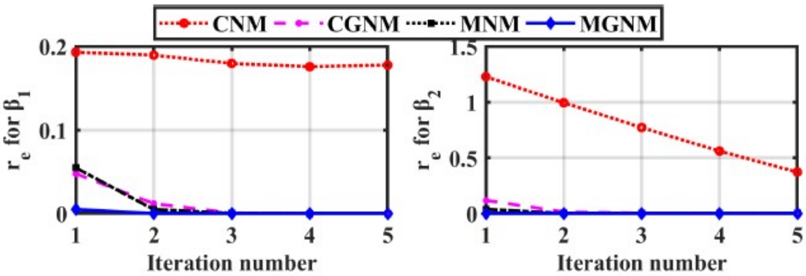


Figure 2. Iteration number versus relative error plots for parameters **(i)** β_1 and **(ii)** β_2 separately for various minimization methods using the function $f(\beta_1,\ \beta_2) = \beta_1 e^{\beta_2 t}$ for initial values $\beta_1^{(0)} = 2.3,\ \beta_2^{(0)} = 0.2$.

Table 2. Iteration results of CGNM, MNM, and the proposed MGNM method for function $\mathrm{f}(\beta_1,\ \beta_2) = \beta_1 \mathrm{e}^{\beta_2 \mathrm{t}}$, where β_1, β_2 denotes the results of the p^{th} iteration for initial values $\beta_1^{(0)} = 1.9, \beta_2^{(0)} = 1$ with tabulated relative errors represented by re_{β_1} and re_{β_2}.

	CGNM [5]				**MNM** [5]				**Proposed MGNM**			
p	β_1	re_{β_1}	β_2	re_{β_2}	β_1	re_{β_1}	β_2	re_{β_2}	β_1	re_{β_1}	β_2	re_{β_2}
0	*1.9*		*1*		*1.9*		*1*		*1.9*		*1*	
1	0.181	0.9287	0.988	2.8073	1.837	2.7708	0.318	0.22543	2.452	0.0349	**0.2595**	0
2	0.191	0.2483	0.862	2.3217	2.286	0.1003	0.274	0.05587	2.539	0.0007	0.2595	0
3	0.347	0.8634	0.649	1.5009	2.492	0.0193	0.262	0.00963	**2.541**	0	0.2595	0
4	0.891	0.6493	0.369	0.4219	2.539	0.0007	**0.259**	0.0019	2.541	0	0.2595	0
5	2.244	0.1169	0.202	0.2215	**2.541**	0	0.259	0.0019	2.541	0	0.2595	0

Similarly, the comparison among various minimization methods can be observed from the contour plot, presented in Fig. 3. From the figure, it can be clearly seen that for initial values $\beta_1^{(0)} = 1.9, \beta_2^{(0)} = 1$, the CGNM method fails to reach the optimal point even after 5 iterations. On the other hand, both MNM and MGNM methods converge immediately to the optimal point but the convergence rate of the proposed MGNM method is comparatively greater than the MNM method.

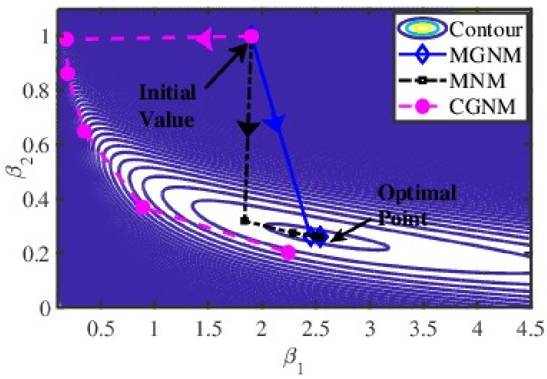


Figure 3. Contour plot representation of the objective function given by (17) for CGNM, MNM, and the proposed MGNM method, for initial values $\beta_1^{(0)} = 1.9,\ \beta_2^{(0)} = 1$.

Furthermore, the comparison among various minimization methods is presented in Fig. 4 in terms of iteration number versus relative error plots for each parameter separately. From Fig. 4, it can be observed that the relative error for the proposed MGNM method for each parameter β_1 and β_2 is considerably less than the conventional methods discussed so far.

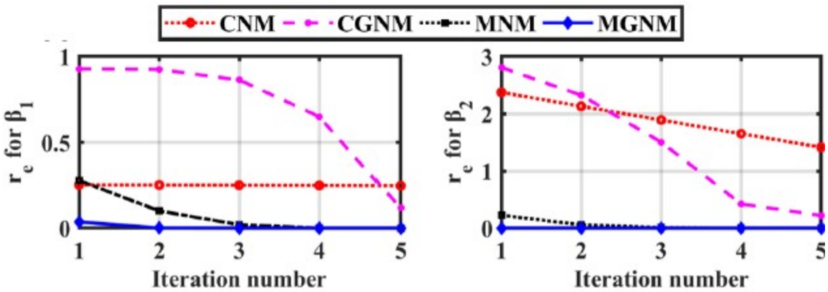
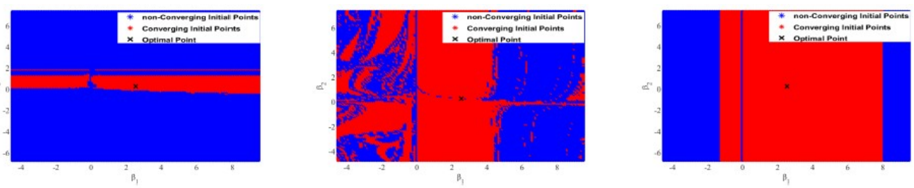


Figure 4. Iteration number versus relative error plots for parameters **(i)** β_1 and **(ii)** β_2 separately for various minimization methods using the function $f(\beta_1,\ \beta_2) = \beta_1 e^{\beta_2 t}$ for initial values $\beta_1^{(0)} = 1.9,\ \beta_2^{(0)} = 1$.

Therefore, it has been shown that the proposed MGNM method produces optimal parameter estimation for non-linear exponential functions in a fewer number of iterations as compared to CGNM method. In addition, the main *advantage of the proposed MGNM method* lies in the fact that the range of the initial values for which it converges is considerably larger than that of the conventional methods.

To account for the convergence of the proposed method for larger range of initial values than the existing methods, the comparison is made by considering a varied set of initial points in the vicinity of the optimal point. As shown in Fig. 5, a rectangular grid of length 7 is considered around the optimal point with a step size of 0.1 i.e., the algorithm is tested against all the points present in the grid with the maximum number of iterations $p_{max} = 20$. The black, red and blue points in Fig. 5 represent the optimal point, the initial values for which the algorithm converges and diverges, respectively. The optimal point for the function $f(\beta_1,\ \beta_2) = \beta_1 e^{\beta_2 t}$ is $\beta_1 = 2.541,\ \beta_2 = 0.2595$.



(a) (b) (c)

Figure 5. The region of convergence for various minimization methods the function $f(\beta_1,\ \beta_2) = \beta_1 e^{\beta_2 t}$ **(a)** classical Gauss-Newton minimization (CGNM) method **(b)** multiplicative Newton minimization (MNM) method **(c)** proposed multiplicative Gauss-Newton minimization (MGNM) method.

Fig. 5 (a)-(c) demonstrates the range of initial values for which the CGNM, MNM and MGNM methods converge, respectively. From the results, it can be observed that the region of convergence for the proposed MGNM method is greater than the established MNM and CGNM methods. However, the proposed MGNM method shows divergence in the case when the initial value is chosen such that β_1

is zero. This is because in this case, the function $f(\beta_1,\ \beta_2) = \beta_1 e^{\beta_2 t}$ itself becomes equal to zero and hence the error (defined as $\overset{m}{e}_i(\beta) = y_i/f(\beta, t_i)$) becomes infinite. Also, the convergence to the optimal point is achieved in a smaller number of iterations by the proposed MGNM method as compared to the established MNM and CGNM methods.
These results are further illustrated in Table 3, in which it is demonstrated that out of 19600 initial points chosen around the optimal point, the CGNM method converges only for 2111 points, MNM method converges for 9922 points and the proposed MGNM method shows convergence for 12972 points and thus, outperforms the classical and existing multiplicative minimization methods.

However, this argument of convergence is based upon the selection of initial points in the vicinity of the optimal point. Therefore, the implication is that if the initial points are chosen randomly in the vicinity of the optimal point, the proposed method outperforms the conventional minimization methods. However, the convergence and the numerical results may vary based upon different selected initial values.

Table 3. Number of initial points for which the algorithm converges for the function $\mathrm{f}(\beta_1,\ \beta_2) = \beta_1 \mathrm{e}^{\beta_2 \mathrm{t}}$.

	CGNM [5]	**MNM** [5]	**Proposed MGNM**
Converging Initial Points (*Max. 19600*)	2111	9922	**12972**

Example 2: The effectiveness of the proposed method is further investigated for more complicated signal representation, using a function that shows non-linear behavior in both ordinary and multiplicative calculi. In Example 3 of [5], the system model is interpreted using the function $f(\beta_1, \beta_2) = e^{-\beta_1 t} + te^{\beta_2 t}$. This function is non-linear for both classical and multiplicative least square methods. The proposed MGNM is applied to estimate the optimal parameters for the function $f(\beta_1, \beta_2) = e^{-\beta_1 t} + te^{\beta_2 t}$.

According to the obtained simulation results, the optimal parameters are $\beta_1 = 2$ and $\beta_2 = 0.3$. Table 4 and Table 5 present the comparison among the iteration values of the proposed MGNM method with other minimization methods, for different initial values.

Table 4. Iteration results of CGNM, MNM, and the proposed MGNM method for function $\mathrm{f}(\beta_1, \beta_2) = \mathrm{e}^{-\beta_1 \mathrm{t}} + \mathrm{te}^{\beta_2 \mathrm{t}}$, where β_1, β_2 denotes the results of the p^{th} iteration for initial values $\beta_1^{(0)} = 2.5, \beta_2^{(0)} = 0.1$ with tabulated relative errors represented by re_{β_1} and re_{β_2}.

	CGNM [5]				**MNM** [5]				**Proposed MGNM**			
p	β_1	re_{β_1}	β_2	re_{β_2}	β_1	re_{β_1}	β_2	re_{β_2}	β_1	re_{β_1}	β_2	re_{β_2}
0	*2.5*		*0.1*		*2.5*		*0.1*		*2.5*		*0.1*	
1	3.507	0.75358	0.195	0.35	2.415	0.2075	0.300	0	1.953	0.0234	**0.300**	0
2	3.323	0.66158	0.227	0.2433	2.440	0.22006	0.299	0.003	**1.9990**	0.0004	0.300	0
3	2.970	0.48507	0.247	0.1766	1.755	0.12245	**0.300**	0	1.9999	0	0.300	0
4	2.720	0.36006	0.260	0.1333	1.938	0.03095	0.300	0	1.9999	0	0.300	0
5	2.545	0.27256	0.269	0.1033	**1.999**	0.00045	0.300	0	1.9999	0	0.300	0

Case 1: For initial values $\boldsymbol{\beta_1^{(0)} = 2.5}$, $\boldsymbol{\beta_2^{(0)} = 0.1}$: From the results presented in Table 4, it can be observed that the proposed MGNM method converges after only 2 iterations for the selected initial values. However, the MNM method provides the appropriate solution after 5 iterations and CGNM method fails to provide an optimal solution within 5 iteration steps.

Similarly, Fig. 6 and Fig. 7 shows the behavior of the objective function on the contour plot for CGNM, MNM, MGNM methods and the comparison among them in terms of iteration number versus relative error plots, respectively. From Fig. 6, it can be observed that the CGNM method fails to reach the optimal point whereas MGNM method immediately sets out in the direction of optimal point and reaches it directly as compared to the MNM method. Furthermore, it can be interpreted from Fig. 7 that the relative error for the proposed MGNM method reaches its minimum value in the lesser number of iterations as compared to the CGNM and MNM methods.

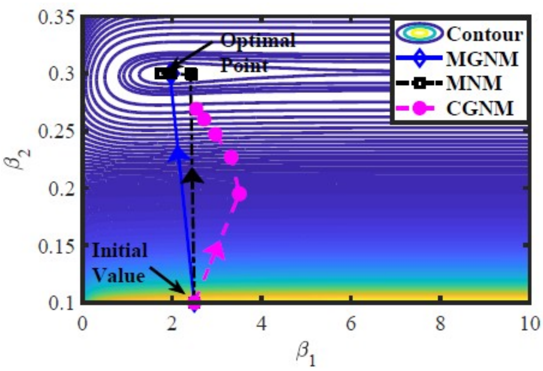


Figure 6. Contour plot representation of the objective function given by (17) for CGNM, MNM, and the proposed MGNM method, for initial values $\beta_1^{(0)} = 2.5,\ \beta_2^{(0)} = 0.1$.

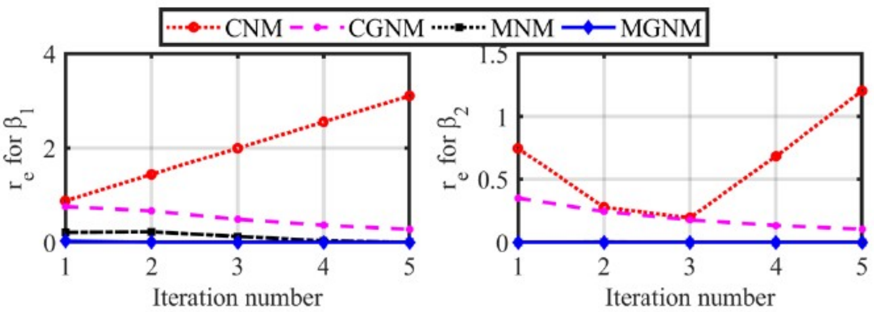


Figure 7. Iteration number versus relative error plots for parameters **(i)** β_1 and **(ii)** β_2 separately for various minimization methods using the function $f(\beta_1, \beta_2) = e^{-\beta_1 t} + te^{\beta_2 t}$ for initial values $\beta_1^{(0)} = 2.5,\ \beta_2^{(0)} = 0.1$.

Case 2: For initial values $\boldsymbol{\beta_1^{(0)} = 1.8, \beta_2^{(0)} = 0.8}$: Table 5 presents the comparison among various minimization methods for initial values $\beta_1^{(0)} = 1.8,\ \beta_2^{(0)} = 0.8$. From the results, it can be depicted that the proposed MGNM method converges after only 2 iterations whereas the CGNM method diverges for the selected set of initial values and MNM method reaches the optimal solution after 4 iterations. Therefore, it can be inferred that the proposed MGNM method offer superior representation of non-linear exponential functions than the existing minimization methods in classical calculus as well as in multiplicative calculus.

Furthermore, for initial values $\beta_1^{(0)} = 1.8,\ \beta_2^{(0)} = 0.8$, the convergence to the optimal point is not achieved by CGNM method as can be observed from the contour plot presented in Fig. 8. However, both MNM and MGNM methods reach the optimal point but the convergence rate of proposed MGNM method is significantly greater than the MNM method as MGNM method follows the direct path and reaches the optimal solution immediately.

Table 5. Iteration results of CGNM, MNM, and the proposed MGNM method for function f $(\beta_1, \beta_2) = \mathrm{e}^{-\beta_1 \mathrm{t}} + \mathrm{te}^{\beta_2 \mathrm{t}}$, where β_1, β_2 denotes the results of the p^{th} iteration for initial values $\beta_1^{(0)} = 1.8, \beta_2^{(0)} = 0.8$ with tabulated relative errors represented by re_{β_1} and re_{β_2}.

	CGNM [5]				**MNM** [5]				**Proposed MGNM**			
p	β_1	re_{β_1}	β_2	re_{β_2}	β_1	re_{β_1}	β_2	re_{β_2}	β_1	re_{β_1}	β_2	re_{β_2}
0	*1.8*		*0.8*		*1.8*		*0.8*		*1.8*		*0.8*	
1	3.368	0.6848	0.787	1.62333	2.264	0.13205	**0.300**	0	1.953	0.0234	**0.300**	0
2	5.497	1.7486	0.775	1.58333	1.847	0.07645	0.300	0	**1.9990**	0.0005	0.300	0
3	16.60	7.3004	0.763	1.5433	1.973	0.01345	0.300	0	1.9999	0	0.300	0
4	-		-		**1.999**	0.00045	0.300	0	1.9999	0	0.300	0
5	-		-		1.999	0.00045	0.300	0	1.9999	0	0.300	0

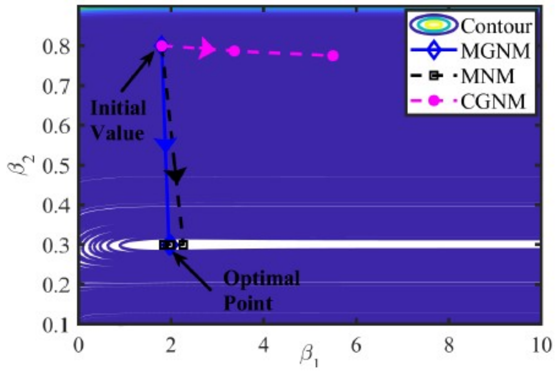


Figure 8. Contour plot representation of the objective function given by (17) for CGNM, MNM, and the proposed MGNM method, for initial values $\beta_1^{(0)} = 1.8,\ \beta_2^{(0)} = 0.8$.

Similarly, the comparison among various minimization methods can be done in terms of relative error, as shown in Fig. 9. From Fig. 9, it can be seen that the relative error increases as the number of iterations increases for CGNM method, whereas the relative error for the proposed MGNM method is significantly less than the MNM method.

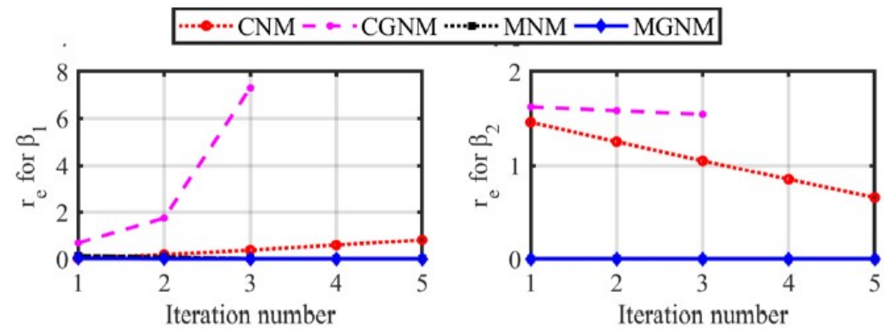


Figure 9. Iteration number versus relative error plots for parameters **(i)** β_1 and **(ii)** β_2 separately for various minimization methods using the function $f(\beta_1, \beta_2) = e^{-\beta_1 t} + te^{\beta_2 t}$ for initial values $\beta_1^{(0)} = 1.8,\ \beta_2^{(0)} = 0.8$.

Likewise, the region of convergence for the function $f(\beta_1, \beta_2) = e^{-\beta_1 t} + te^{\beta_2 t}$ is tested by considering a rectangular grid of length 5 with a step size of 0.1 around the optimal point $\beta_1 = 2,\ \beta_2 = 0.3$, as shown in the Fig. 10 with the maximum number of iterations $p_{max} = 20$. Since the function is more complicated in terms of non-linearity, the CGNM method converges only for few initial points as compared to multiplicative minimization methods, as depicted in the Fig. 10(a). However, the region of convergence for the proposed MGNM method is larger as compared to both MNM and CGNM methods, as shown in Fig. 10(b) and 10(c).

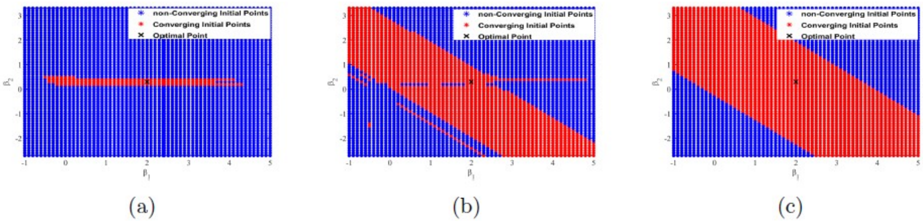


Figure 10. The region of convergence for various minimization methods for the function $f(\beta_1, \beta_2) = e^{-\beta_1 t} + te^{\beta_2 t}$ **(a)** classical Gauss-Newton minimization (CGNM) method **(b)** multiplicative Newton minimization (MNM) method **(c)** proposed multiplicative Gauss-Newton minimization (MGNM) method.

This can be further illustrated in terms of the number of initial points for which the corresponding algorithm converges as presented in Table 6. Table 6 states that out of 3600 initial points chosen around the optimal point, the CGNM method converges only for 141 points, the MNM method converges for 1500 points and the proposed MGNM method converges for 2101 initial points, which clearly shows the effectiveness of the proposed method.

Furthermore, the algorithm is tested on various examples that exhibit non-linear exponential behavior. The results obtained are satisfactory and illustrate the effectiveness of the proposed method. From the obtained results, it can be said that the algorithm also works fine for a large number of data samples.

Table 6. Number of initial points for which the algorithm converges for function $f(\beta_1, \beta_2) = e^{-\beta_1 t} + te^{\beta_2 t}$.

	CGNM [5]	**MNM** [5]	**Proposed MGNM**
Converging Initial Points (*Max. 3600*)	141	1500	**2101**

From all the theoretical and simulation studies, it can be inferred that the choice of the optimal method among CNM, CGNM, MNM, and the proposed MGNM method relies upon the nature of the function to be minimized and the initial values of the iteration.

Analogy between classical calculus and multiplicative calculus: As an illustration, consider an example of a non-linear exponential function $f(\beta_1, \beta_2) = \exp(\beta_1 t^{\beta_2})$, in multiplicative calculus whose parameters are to be estimated using the proposed MGNM method. The data has been generated randomly with optimal parameters $\beta_1 = 1.5$ and $\beta_2 = 2$. Fig. 11(a) demonstrates the range of initial values for which the MGNM method converges. The corresponding/analogous case in the classical calculus is the function $f(\beta_1, \beta_2) = \beta_1 t^{\beta_2}$ whose parameters are estimated using CGNM method in the same number of iterations as that of MGNM method for the function $f(\beta_1,\ \beta_2) = \exp(\beta_1 t^{\beta_2})$. Also, the region of convergence is same for both cases. This is not really an astonishing result because the non-linear exponential functions are linear in multiplicative calculus. The proof of this statement is validated by testing both the methods against a large number of initial values around the optimal point, as demonstrated in Fig. 11(a) and 11(b). By comparing Fig. 11(a) and 11(b), it can be clearly observed that the region of convergence is same in both the cases.

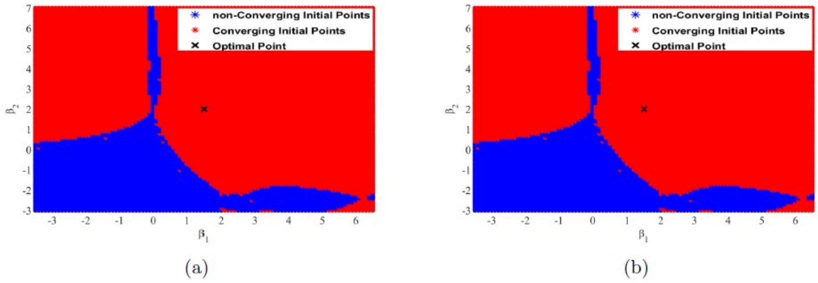


Figure 11. The analogy between the range of initial values for **(a)** classical and **(b)** multiplicative Gauss-Newton minimization methods.

This is because the nature of the underlying calculus plays a crucial role in finding the minimum of the function. Since non-linear exponential functions are linear in multiplicative calculus framework, they can be approximated with the same ease and efficiency as linear functions are approximated in the classical calculus framework. Also, in many cases, the classical minimization methods fail to converge for non-linear exponential functions or converge slower than the multiplicative minimization methods.

Thus, the minimization method must be chosen based upon the nature of the function to be minimized. An assessment of the performance of each method is generally measured in terms of the maximum number of iterations taken by each method to converge to the optimal solution and the range of initial values for which the method converges. As a comment, it cannot be said that multiplicative approximations are always better than the additive approximations. It highly depends upon the nature of the function to be approximated. For example, if a function involves a polynomial or a ratio of two polynomials, it can be classified as an additive type function. In such a case, an approximation in the framework of additive calculus will be more appropriate than a multiplicative approximation.

As a result, it has been verified that in certain situations where non-linear exponential functions are involved, the multiplicative minimization methods perform better than the classical minimization methods. Also, it can be said that the proposed MGNM method converges, where classical minimization methods fail to converge; or it converges faster than the classical and multiplicative Newton minimization methods. The proposed method can be extremely beneficial for scientists and engineers to fit experimental data using exponentially varying functions. Therefore, the formulations in multiplicative calculus are expected to be well-suited for functions which exhibit exponential behavior.

§6 Conclusion and Future Scope

In this article, a novel theory of numerical minimization method in the framework of multiplicative calculus has been proposed. The detailed mathematical formulation of the proposed theory is presented along with its convergence properties and analysis. The primary focus of the proposed work is to efficiently approximate non-linear exponential functions. Furthermore, the proposed method is validated by simulation results for representing different exponential functions with various initial values. Two case studies have been conducted incorporating multiplicatively linear and non-linear exponential functions and are compared to classical Gauss-Newton minimization (CGNM) and multiplicative Newton minimization (MNM) methods for different set of initial values. For multiplicative linear function, the CGNM, MNM and the proposed MGNM methods converge after 5, 5, and 2 iterations, respectively, for a particular set of initial value. However, for multiplicative non-linear function, the CGNM method diverges and MNM and MGNM methods converges after 5 and 2 iterations, respectively.

Furthermore, for multiplicative linear function, out of 19600 initial points considered in the vicinity of optimal point, the CGNM, MNM and the proposed MGNM converges for 2111, 9922, and 12972 points, respectively. On the other hand, for multiplicative non-linear function, out of 3600 initial points considered, CGNM, MNM, and the proposed MGNM method converge for 141, 1500, and 2101 points, respectively. Therefore, it can be said that the proposed MGNM method converges faster and for a large range of initial values as compared to the existing methods in the literature such as Newton, Gauss-Newton and multiplicative Newton minimization methods. However, the convergence may vary significantly if different initial values are selected.

Gauss-Newton method is an efficient algorithm for solving non-linear regression problems. But the implication is that if the same algorithm is employed in the multiplicative framework, the realization of any physical phenomenon exhibiting non-linear exponential behavior is expected to be more efficient. Hence, the proposed method has found effective and robust applications in solving non-linear least square problems for efficient representation of non-linear exponential functions.

Based on the encouraging results obtained, it is possible to solve many other engineering problems in the framework of multiplicative calculus using the proposed MGNM method with better accuracy. This work can further be extended to derive other algorithms (such as gradient descent, Levenberg-Marquardt algorithm, and various optimization algorithms, etc.) in the framework of multiplicative calculus for various control, science, engineering and machine learning applications.

Acknowledgement

We are deeply indebted to the Editor and the anonymous referees whose insightful comments and suggestions helped us a lot to improve the quality of this paper.

References

[1] D Aniszewska. *Multiplicative Runge–Kutta methods*, Nonlinear Dyn, 2007, 50(1): 265-272.

[2] A Atto, E Trouve, J Nicolas, T Le. *Wavelet Operators and Multiplicative Observation Models—Application to SAR Image Time-Series Analysis*, IEEE Trans Geosci Remote Sens, 2016, 54(19): 6606-6624.

[3] A Bashirov, E Kurpınar, A Özyapıcı. *Multiplicative calculus and its applications*, J Math Anal Appl, 2008, 337(1): 36-48.

[4] A Bashirov, E Misirli, Y Tandogdu, A Ozapici. *On modelling with multiplicative differential equations*, Appl Math J Chinese Univ, 2011, 26(4): 425-438.

[5] B Bilgehan. *Efficient approximation for linear and non-linear signal representation*, IET Signal Process, 2015, 9(3): 260-266.

[6] D Filip, C Piatecki. *An overview on the non-Newtonian calculus and its potential applications to economics*, 2014, arXiv:00945788v1, https://hal.archives-ouvertes.fr/hal-00945788/.

[7] L Florack, H V Assen. *Multiplicative calculus in biomedical image analysis*, J Math Imaging Vis, 2011, 42(1): 64-75.

[8] A Giordano, F Hsu. *Least square estimation with applications to Digital Signal Processing*, New York: John Wiley & Sons, 1985.

[9] M Grossman, R Katz. *Non-Newtonian Calculus: A Self-contained, Elementary Exposition of the Authors' Investigations*, MA: Pigeon Cove, 1972.

[10] M Mora, F Córdova-Lepe, R Del-Valle. *A non-Newtonian gradient for contour detection in images with multiplicative noise*, Pattern Recognit Lett, 2012, 33(18): 1245-1256.

[11] A Özyapıcı, B Bilgehan. *Finite product representation via multiplicative calculus and its applications to exponential signal processing*, Numer Algorithms, 2016, 71(2): 475-489.

[12] A Özyapıcı, M Riza, B Bilgehan, A Bashirov. *On multiplicative and Volterra minimization methods*, Numer Algorithms, 2013, 67(3): 623-636.

[13] A Özyapıcı, Z Sensoy, T Karanfiller. *Effective Root-Finding Methods for Nonlinear Equations Based on Multiplicative Calculi*, J Math, 2016, 2016: 1-7.

[14] H Özyapıcı, İ Dalcı, A Özyapıcı. *Integrating accounting and multiplicative calculus: An effective estimation of learning curve*, Comput Math Organ Theory, 2016, 23(2): 258-270.

[15] A Uzer. *Evaluations of a class of integrals that arise in wave scattering problems by using the Taylor product theorem*, IEEE Trans Antennas Propag, 2018, 66(2): 1031-1034.

[16] A Uzer. *Multiplicative type complex calculus as an alternative to the classical calculus*, Comput Math with Appl, 2010, 60(18): 2725-2737.

[17] Y Wang. *Gauss-Newton method*, Wiley Interdisciplinary Reviews: Computational Statistics, 2012, 4(4): 415-420.

[18] J Wolberg. *Data Analysis Using the Method of Least Squares: Extracting the most information from experiments*, Springer-Verlag Berlin Heidelberg, 2006.

[19] N Yalcin, E Celik, A Gokdogan. *Multiplicative Laplace transform and its applications*, Optik, 2016, 127(29): 9984-9995.

[20] Y Zheng, Y Zhou, H Zhou, X Gong. *Ultrasound image edge detection based on a novel multiplicative gradient and canny operator*, Ultrason Imaging, 2014, 37(3): 238-250.

Department of Electronics and Communication Engineering, Thapar Institute of Engineering and Technology, Patiala 147004, India.

Email: sanjay.kumar@thapar.edu