A nonlinear second derivative method with a variable step-size based on continued fractions for singular initial value problems

S.N. Jator1* and N. Coleman2

Abstract: We propose a nonlinear one-step second derivative method (NSDM) with a variable step-size implementation based on continued fractions for the numerical solution of singular initial value problems (IVPs). The singular IVPs typically originate from models in Mathematical Biology which are represented by reaction diffusion partial differential equations (PDEs). These PDEs are then discretized in space through finite difference methods to yield a system of ordinary differential equations with singular behavior associated with finite time blow-up. We are motivated by the fact that numerical methods based on polynomial approximation perform poorly in the proximity of a singular point when applied to singular IVPs. Therefore, a continuous method based on a finite continued fraction approximation is derived and used to obtain the NSDM. The NSDM is implemented in a self-starting mode and in a predictor-corrector mode, equipped with a variable step-size implementation. The superiority of the NSDM over standard methods in the literature is established numerically.

Subjects: Advanced Mathematics; Applied Mathematics; Applied Physics; Computational Physics

Keywords: singular initial value problem; nonlinear second derivative; finite continued fraction; variable step-size implementation

AMS subject classifications: 65L05; 65L06

ABOUT THE AUTHORS

S.N. Jator is currently a Professor of Mathematics and the Chair of the Department of Mathematics and Statistics, Austin Peay State University, Clarksville, Tennessee, USA. His research interests include Numerical Analysis, Mathematical Biology, Ordinary and Partial Differential Equations, Financial Mathematics, and Predictive Analytics.

N.A. Coleman is currently an Assistant Professor of Computer Science at Austin Peay State University, Clarksville, Tennessee, USA. His research interests include Programming Language Semantics, Automata Theory, Formal Methods, Type Theory, and Mathematical Logic.

PUBLIC INTEREST STATEMENT

Standard methods such as Runge-Kutta and linear multistep methods perform poorly in the proximity of a singular point when applied to singular initial value problems. Hence, we have proposed an accurate nonlinear one-step second derivative method with a variable step-size implementation based on continued fractions for the numerical solution of singular initial value problems. The singular initial value problems typically originate from models in Mathematical Biology which are represented by reaction diffusion partial differential equations discretized in space through finite difference or finite element methods to yield a system of ordinary differential equations with singular behavior associated with finite time blow-up.
1. Introduction
We consider the IVP
\[ y' = f(x, y), \quad y(a) = y_0, \]
\[ y, f \in \mathbb{R}^m, \quad x \in [a, b], \quad a, b \in \mathbb{R}. \]  
(1)

Several physical processes in science and engineering can be modeled in the form (1) which could be non-stiff, stiff or singular in nature. Specifically, we consider singular IVPs that typically originate from models in Mathematical Biology triggered by a random walk of organisms, such as bacterial in the presence of a chemical resulting in interesting spatial patterns (see Tyson, Lubkin, & Murray, 1999; Tyson, Stern, & LeVeque, 2000). This kind of motion is characterized as chemotaxis (see Tyson et al., 2000) and are traditionally modeled through a reaction diffusion system of partial differential equations (PDEs). Moreover, physical processes such as flows in porous media, heat transfer, and biochemical kinetics are modeled through PDEs (see Abia, López-Marcos, & Martínez, 1998). These PDEs are then discretized in space through finite difference or finite element methods to yield a system of ODEs with singular behavior associated with finite time blow-up (see Abia et al., 1998). We note that the solutions of the singular ODEs may become unbounded in finite time and such a phenomenon is often called blow-up, while the finite time is called the blow-up time (see Cho, 2013; Groisman, 2006; Guo & Yang, 2015; Takayasu, Matsue, Sasaki, Tanaka, Mizuguchi, & Oishi, 2017). It is also reported in Takayasu et al. (2017) that blow-up solutions describe the combustion of solid fuels via the nonlinear heat equation.

In applications, most equations in the form (1) do not possess exact solutions, hence numerical methods are imperative for solving (1). Several numerical methods based on Runge-Kutta methods, linear multistep methods, Boundary value methods, and block methods have been extensively discussed in the literature for solving non-stiff and stiff IVPs (see Akinfenwa & Jator, 2013; Brugnano & Trigiante, 1998, Jator, 2010; Cash, 1981; Hairer & Wanner, 1996; Lambert, 1991; Onumanyi, Awoyemi, Jator, & Sirisena, 1994; Onumanyi, Sirisena, & Jator, 1999). It turns out that these methods do not perform well when applied to solve singular problems of the form (1) since they are based on polynomial approximation (see Hairer & Wanner, 1996; Lambert, 1991). In particular, methods constructed using polynomial basis functions perform poorly in the proximity of a singular point when applied to singular IVPs. Hence, rational interpolants have been proposed for solving singular IVPs and were first introduced by Lambert and Shaw (1965). Other techniques based on rational functions have been proposed to cope with singular IVPs (see Fatunla, 1982; Hasan, Ahamed, Alam, & Hossain, 2013; Ikhile, 2004; Odekunle, Oye, Adee, & Ademiluyi, 2004; Van Niekerk, 1988). Some authors have proposed more efficient techniques based on a special rational interpolation function expressed in the form of a finite continued fraction (see Van Niekerk, 1988). Nonstandard methods for solving singular IVPs have also been proposed (see Ramos & Vigo-Aguiar, 2008), which a one-step method based on rational logarithmic basis functions has also been developed (see Garwood & Jator, 2016).

In terms of implementation, is generally implemented in a step-by-step fashion in which on the partition \( H_N \), an approximation \( y_n \) is obtained at \( x_n \) only after an approximation at \( x_{n-1} \) has been computed, where for some constant step-size \( h = \frac{x_n - x_{n-1}}{N} \) and integer \( N > 0 \),

\[ H_N := \{ x_0 < x_1 < \cdots < x_N \}, \quad x_n = x_{n-1} + h, \quad n = 1, \ldots, N. \]

In this paper, this approach is used as a predictor-corrector method with a fixed step-size and variable step-size. We have also used another approach that is self-starting and implemented without predictors (Akinfenwa & Jator, 2013; Jator, 2010). We propose a new method based on a continuous finite continued fraction approximation (see Van Niekerk, 1988), which is used to obtained the NSDM. The NSDM is implemented in a SSM and PCM equipped with a variable step-size implementation. The superiority of the NSDM over standard methods, such as the Trapezoidal Method (TM) and the fourth order Runge-Kutta Method (RKM) is established numerically. We also show that the method can be
used to solve evolution equations and parabolic problems with blow-up solutions (see Cho, 2013; Groisman, 2006; Guo & Yang, 2015; Takayasu et al., 2017).

The paper is organized as follows. In Section 2, we derive the NSDM and examine its implementation in Section 3. In Section 4, we perform numerical experiments and give the conclusion of the paper in Section 5.

2. Derivation of method

Although (1) is a system of ODEs, in our derivation, we initially assume the scalar form of (1) for simplicity, since the corresponding system can also be solved by obvious notational modifications. We are motivated by the fact that numerical methods based on polynomial approximation perform poorly in the proximity of a singular point when applied to singular IVPs. Therefore, a continuous method based on a rational function is proposed since rational are considered better approximants for singular IVPs (see Fatunla, 1982; Ikhole, 2004; Odeku et al., 2004). The derivation of the method is based on a special kind of rational interpolation function expressed as a continued fraction (see Van Niekerk, 1988). In order to solve the IVP (1) on the sequence of points \( \{x_n, x_{n+1}\} \) the exact solution \( y(x) \) is approximated by the finite continued fraction \( u(x) \), since it is known that; that is, \( y(x) \approx u(x) \), thus,

\[
u(x) = a_0 + \frac{a_1x}{1 + \frac{a_2x}{1 + \frac{a_3x}{1 + \cdots}}} = \frac{a_0x + a_1x^2 + a_2x^3 + \cdots}{1 + \frac{\cdots}{1 + \frac{a_3x^3}{1 + \frac{a_2x^2}{1 + \frac{a_1x}{1 + \frac{a_0}{1}}}}}}.
\]

where \( a_0, a_1, \ldots, a_n \) are coefficients to be uniquely determined. We further assume that \( y_n + hj \) and \( y(x_n) \approx y_{n+1} \). \( f_{n+1} = U'(x_n + jh) \) is an approximation to \( y'(x_{n+1}) \), \( g_{n+1} = U''(x_n + jh) \) is an approximation to \( y''(x_{n+1}) \), \( j = 0, 1 \). In order to determine the coefficients \( a_0, a_1, a_2, a_3 \) in (2), we demand that the following conditions must be satisfied

\[
u(x_n) = y_n, \quad u'(x_n) = f_n, \\
\]

and lead to a system of four equations and four unknowns, which is solved to obtain the coefficients \( a_0, a_1, a_2, a_3 \). These coefficients are now known and substituted into in (2) to yield the continuous form \( u(x) \) which is then evaluated at \( x_{n+1} \) to yield the NSDM given by

\[
y_{n+1} = -\frac{2h^2f_{n+1} + 4hf_{n+1}y_n + (h^2g_{n+1} + 2y_n)(y_{n+1} + f_n(h^2g_{n+1} - 2hy_{n+1}))}{2(hf_n - 2h_{n+1} - y_n + y_{n+1})}.
\]

We note that \( y_{n+1} \) is the numerical approximation to the analytical solution \( y(x_{n+1}) \) \( f_{n+1} = f(x_n,y_n) \) and \( g_{n+1} = \frac{d^2f(x_n,y_n)}{dx^2} \), \( j = 0, 1 \).

The Local Truncation Error (LTE) of the method (3) is obtained by carry out a Taylor Series expansion about the point \( x_n \) to get

\[
\text{LTE}(y(x_n), h) = \left( \frac{y''''(x_n)^2}{54y''''(x_n)} + \frac{1}{72}y''''(x_n) \right) h^4 + O(h^5),
\]

hence the method is of order 3.

**Derivation of the predictor:** Since the method is also implemented in the PCM, the predictor is derived similarly. Thus, the predictor is constructed by assuming that on the interval \( [x_{n-1}, x_{n+1}] \) the exact solution \( y(x) \) is approximated by the finite continued fraction \( u_p(x) \); i.e. \( y(x) \approx u_p(x) \), thus,
\[ u_p(x) = b_0 + \frac{b_1 x}{1 + b_2 x^2} \] \hspace{1cm} (5)\]

where \( b_0, \ldots, b_2 \) are coefficients that are uniquely determined. We further assume that \( y_n = u(x_n) \) and \( y(x_n) \approx y_n \), \( f_n = U'(x_n) \) is an approximation to \( y'(x_n) \), \( g_n = U''(x_n) \) is an approximation to \( y''(x_n) \), and \( z_{n+1} = U'''(x_n) \) is an approximation to \( y'''(x_n) \). In order to determine the coefficients \((b_0, b_1, b_2, b_3)\) in (5), we demand that the following conditions must be satisfied

\[ u_p(x_n) = y_n, \quad u_p'(x_n) = f_n, \quad u_p''(x_n) = g_n, \quad u_p'''(x_n) = z_n, \]

and lead to a system of four equations and four unknowns, which is solved to obtain the coefficients \((b_0, b_1, b_2, b_3)\). These coefficients are now known and substituted into in (5) to yield the continuous form of the predictor \( u_p(x) \) which is then evaluated at \( x_{n+1} \) to yield

\[ y_{n+1}^p = \frac{-3h^2g_n^2 - 6g_n y_n + 2h y_n z_n + 2h f_n (-3g_n + h z_n)}{-6g_n + 2h z_n} \]

\[ g_n = \frac{df(x,y(x))}{dx}\bigg|_{y_n} \text{ and } z_n = \frac{df(x,y(x))}{dx}\bigg|_{y_n} \]

The LTE for the predictor is given by

\[ \text{LTE}^p(y(x_n), h) = \frac{1}{72} \left( -4y'''(x_n)^2/y''(x_n) + 3y'''(x_n) \right) h^4 + O(h^5), \] \hspace{1cm} (7)\]

hence the method is also of order 3.

Remark 2.1 While the conditions imposed to construct (2) and (5) are based on interpolation and collocation, Hermite interpolation generally involves the use of a table of divided differences. We also note that \( g(x, y(x)) = \frac{df(x,y(x))}{dx}\bigg|_{y_n} \) and \( z(x, y(x)) = \frac{df(x,y(x))}{dx}\bigg|_{y_n} \). In particular, \( g_{n+1} = \frac{df(x,y(x))}{dx}\bigg|_{y_{n+1}}, g_n = \frac{df(x,y(x))}{dx}\bigg|_{y_n} \), and \( z_n = \frac{df(x,y(x))}{dx}\bigg|_{y_n} \).

3. Implementation of method

**PCM-fixed step size:** The method is implemented as a predictor-corrector using (3) and (6) in a step-by-step fashion on the partition \( H_N \). The approximation at \( x_n \) is used to obtain the approximation at \( x_{n+1} \), such that

\[ H_N: a = x_0 < x_1 < \cdots < x_N = b, \quad x_n = x_{n-1} + h, \quad n = 1, \ldots, N \]

where \( h = \frac{b-a}{N} \) is the constant step-size, \( N \) is a positive integer, and \( n \) is the grid index. The PCM-fixed step-size implementation is given in Algorithm 1.

---

**Algorithm 1** PCM-fixed step size Algorithm

1: procedure Enter Partitions(a, b)
2: \[ x_n = x_{n-1} + h, \quad n = 1, \ldots, N, \quad h = \frac{b-a}{N} \] > This is the predictor
3: Compute \( y_{n+1}^p \)
4: Compute \( f_{n+1} = f(x_{n+1}, y_{n+1}^p) \) > This is to be used in the corrector
5: Compute \( y_{n+1} \) > This is the corrector
6: end procedure
SSM: The method is implemented in the SSM as follows: Using method (3), \( n = 0 \), solve for the values of \( y_1 \) with the aid of Newton’s Method on the sub-interval \([x_0, x_1]\) as \( y_0 \) is known from the IVP (1). Next, for \( n = 1 \), the values of \( y_2 \) is obtained over the sub-interval \([x_1, x_2]\), as \( y_1 \) is known from the previous sub-interval. The process is continued for \( n = 2, \ldots, N-1 \) to obtain the numerical solution to (1) on the sub-intervals \([x_2, x_3], [x_3, x_4], \ldots, [x_{N-1}, x_N]\). The details of the implementation is given in Algorithm 2.

Algorithm 2 Block Algorithm
1: procedure Enter Partitions(a, h, N, b, variables)
2: For \( x_n = x_{n-1} + h, n = 1, \ldots, N, \) \( h = \frac{b - a}{N} \)
3: Generate block System \( \triangleright \) A scalar problem will have only one equation
4: Solve[System, variables] \( \triangleright \) Block-by-block procedure
5: Obtain \( y_n \) \( \triangleright \) No need for starting values and predictors
6: end procedure

Variable-step implementation of method: The method is also implemented in a PCM in a variable-step fashion on the partition \( H_N \), in which case, we need an error estimate \((\varepsilon_{n+1})\) for \( y_{n+1} \). In order to obtain the error estimate, we use (7) to estimate the error in the predictor \( (y_{n+1}^p) \), which is then used to estimate the error in the corrector \( (y_{n+1}) \) given by (4). Thus, using

\[
y(x_{n+1}) - y_{n+1} = \left( -\frac{y''''(x_n)^2}{54y''(x_n)} + \frac{1}{72}y^4(x_n) \right) h^4,
\]

and

\[
y(x_{n+1}) - y_{n+1}^p = \frac{1}{72} \left( -\frac{4y''''(x_n)^2}{y''(x_n)} + 3y''''(x_n) \right) h^4,
\]

\[
\varepsilon_{n+1} = \frac{1}{2} \left\| y_{n+1} - y_{n+1}^p \right\|.
\]

In order to proceed with the variable step-size implementation, we use \( \varepsilon_{n+1} \) as an estimate for the LTE in \( y_{n+1}^p \) and control the step-size of the integration on the basis of this estimate (see Cash, 1981). If a local tolerance (Tol) is imposed at each step, then, the new step-size \( h_{\text{new}} \) and the old step-size \( h_{\text{old}} \) are related as described in Algorithm 3.

Algorithm 3 Variable-Step Algorithm
1: procedure Enter Partitions(a, h, N, Tol, b)
2: For \( x_n = x_{n-1} + h, n = 1, \ldots, N, \) \( h = \frac{b - a}{N} \)
3: Compute \( y_{n+1}^p \) \( \triangleright \) This is the predictor
4: Compute \( f_{n+1} = f(x_{n+1}, y_{n+1}^p) \) \( \triangleright \) This is to be used in the corrector
5: Compute \( y_{n+1} \) \( \triangleright \) This is the corrector
6: If \( \varepsilon_{n+1} > \text{Tol} \), reject \( y_{n+1} \) and set \( h_{\text{new}} = \frac{1}{2}h_{\text{old}} \)
7: If \( \text{Tol} > \varepsilon_{n+1} > \frac{2 \varepsilon_{n+1}}{9} \), where \( \varepsilon = 2 \times 2^{p-1} \), accept \( y_{n+1} \) and set \( h_{\text{new}} = h_{\text{old}} \)
8: If \( \varepsilon_{n+1} < \frac{2 \varepsilon_{n+1}}{9} \), reject \( y_{n+1} \) and set \( h_{\text{new}} = 2h_{\text{old}} \)
9: end procedure

4. Numerical examples
In this section, we give numerical examples to illustrate the accuracy of the method in the neighborhood of a singularity. Let \( y(x_n) \) be the exact solution and \( y_n \) the approximate solution on the partition \( H_N \), we find the error of the approximate solution as \( |y(x_n) - y_n| \). The NSDM is implemented in a
Jator & Coleman, Cogent Mathematics (2017), 4: 1335498
https://doi.org/10.1080/23311835.2017.1335498

SSM and PCM equipped with a variable step-size implementation. The superiority of the NSDM over the TM and the fourth order RKM is established numerically.

Example 4.1 We consider the singular IVP that is also solved in Garwood and Jator (2016),

\[ y' = 1 + y^2, \ y(0) = 1, \ x \in [0, 1], \ N = 20, \]

whose exact solution is

\[ y(x) = \tan\left(x + \frac{\pi}{4}\right). \]

This example is solved using the NSDM in the PCM and SSM as well as solved using the TM and RKM and the absolute errors in the neighborhood of the singularity \( x = \pi/4 \approx 0.7853981633974483 \) are compared. The results displayed in Table 1 show the superiority of the NSDM over the TM and RKM. Figure 1 also shows that the NSDM is in good agreement with the exact solution in the neighborhood of the singularity.

| \( x \) | PCM | SSM | TM | RKM |
|---|---|---|---|---|
| 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 0.10 | 1.06 \times 10^{-5} | 1.03 \times 10^{-6} | 2.73 \times 10^{-4} | 2.15 \times 10^{-8} |
| 0.20 | 2.71 \times 10^{-6} | 2.45 \times 10^{-6} | 9.51 \times 10^{-4} | 2.80 \times 10^{-8} |
| 0.30 | 5.37 \times 10^{-6} | 4.68 \times 10^{-6} | 2.68 \times 10^{-4} | 5.21 \times 10^{-7} |
| 0.40 | 1.03 \times 10^{-5} | 8.71 \times 10^{-6} | 7.57 \times 10^{-4} | 3.63 \times 10^{-6} |
| 0.50 | 2.13 \times 10^{-5} | 1.75 \times 10^{-5} | 2.42 \times 10^{-3} | 2.60 \times 10^{-5} |
| 0.60 | 5.54 \times 10^{-6} | 4.41 \times 10^{-6} | 1.05 \times 10^{-4} | 2.91 \times 10^{-8} |
| 0.65 | 1.08 \times 10^{-6} | 8.42 \times 10^{-6} | 2.78 \times 10^{-4} | 1.46 \times 10^{-3} |
| 0.70 | 2.81 \times 10^{-6} | 2.14 \times 10^{-4} | 1.01 \times 10^{-4} | 1.34 \times 10^{-2} |
| 0.75 | 1.69 \times 10^{-5} | 1.25 \times 10^{-4} | 8.00 \times 10^{-4} | 5.44 \times 10^{-1} |
| 0.80 | 1.02 \times 10^{-2} | 7.37 \times 10^{-3} | 1.41 \times 10^{2} | 1.39 \times 10^{3} |
| 0.85 | 4.87 \times 10^{-5} | 3.75 \times 10^{-4} | 3.03 \times 10^{3} | \infty |
| 0.90 | 1.58 \times 10^{-6} | 1.18 \times 10^{-4} | \infty | \infty |
| 0.95 | 7.88 \times 10^{-6} | 5.61 \times 10^{-5} | \infty | \infty |
| 1.00 | 4.43 \times 10^{-5} | 3.22 \times 10^{-5} | \infty | \infty |

Note: The NSDM implemented as SSM and PCM are superiority to the TM and the fourth order RKM in the proximity of a singular point \( x = \pi/4 \approx 0.7853981633974483 \).

Figure 1. The results are plotted in the \( x-y \)-plane and the results provided by the SSM (red) are in good agreement with those of the exact solution (blue) in the neighborhood of the singularity \( x = \pi/4 \approx 0.7853981633974483 \).
Example 4.2 We consider the following singular IVP that was also solved in Hasan et al. (2013).

\[ y' = \frac{y^2}{\sqrt{1-x}}, \quad y(0) = 1/3, \quad x \in [0, 1], \quad N = 100, \]

whose exact solution is

\[ y(x) = \frac{1}{1 + 2 \sqrt{1-x}}. \]

This example is solved using the NSDM in the PCM and SSM as well as solved using the TM and RKM and the absolute errors in the neighborhood of the singularity (\(x = 1\)) are compared. The results displayed in Table 2 show the superiority of the SSM over TM and RKM. We note that the PCM does not perform well in the proximity of the singularity. Nevertheless, it performs excellently when applied adaptively as the results in Tables 3 and 4 indicate.

| \(x\)     | PCM | SSM | TM      | RKM     |
|-----------|-----|-----|---------|---------|
| 0.00      | 0.00| 0.00| 0.00    | 0.00    |
| 0.10      | \(1.17 \times 10^{-10}\) | \(1.17 \times 10^{-10}\) | \(1.06 \times 10^{-8}\) | \(4.66 \times 10^{-13}\) |
| 0.20      | \(3.14 \times 10^{-10}\) | \(3.14 \times 10^{-10}\) | \(3.05 \times 10^{-8}\) | \(1.33 \times 10^{-12}\) |
| 0.30      | \(6.56 \times 10^{-10}\) | \(6.56 \times 10^{-10}\) | \(6.82 \times 10^{-8}\) | \(3.01 \times 10^{-12}\) |
| 0.40      | \(1.29 \times 10^{-9}\) | \(1.29 \times 10^{-9}\) | \(1.42 \times 10^{-7}\) | \(6.57 \times 10^{-12}\) |
| 0.50      | \(2.59 \times 10^{-9}\) | \(2.59 \times 10^{-9}\) | \(2.96 \times 10^{-7}\) | \(1.50 \times 10^{-11}\) |
| 0.60      | \(5.57 \times 10^{-9}\) | \(5.57 \times 10^{-9}\) | \(6.48 \times 10^{-7}\) | \(3.84 \times 10^{-11}\) |
| 0.70      | \(1.40 \times 10^{-8}\) | \(1.41 \times 10^{-8}\) | \(1.59 \times 10^{-6}\) | \(1.23 \times 10^{-10}\) |
| 0.80      | \(4.80 \times 10^{-8}\) | \(4.83 \times 10^{-8}\) | \(4.90 \times 10^{-6}\) | \(6.06 \times 10^{-10}\) |
| 0.90      | \(3.58 \times 10^{-7}\) | \(3.64 \times 10^{-7}\) | \(2.66 \times 10^{-5}\) | \(8.67 \times 10^{-9}\) |
| 1.00      | \(9.06 \times 10^{-10}\) | \(9.74 \times 10^{-2}\) | \(\infty\) | \(\infty\) |

Note: The NSDM implemented as SSM and PCM are superiority to the TM and the fourth order RKM in the proximity of a singular point \(x = 1\).

| \(x\)     | PCM | SSM | TM      | RKM     |
|-----------|-----|-----|---------|---------|
| 0.00      | 0.00| 0.00| 0.00    | 0.00    |
| 0.10      | \(1.17 \times 10^{-10}\) | \(1.17 \times 10^{-10}\) | \(1.06 \times 10^{-8}\) | \(4.66 \times 10^{-13}\) |
| 0.20      | \(3.14 \times 10^{-10}\) | \(3.14 \times 10^{-10}\) | \(3.05 \times 10^{-8}\) | \(1.33 \times 10^{-12}\) |
| 0.30      | \(6.56 \times 10^{-10}\) | \(6.56 \times 10^{-10}\) | \(6.82 \times 10^{-8}\) | \(3.01 \times 10^{-12}\) |
| 0.40      | \(1.29 \times 10^{-9}\) | \(1.29 \times 10^{-9}\) | \(1.42 \times 10^{-7}\) | \(6.57 \times 10^{-12}\) |
| 0.50      | \(2.59 \times 10^{-9}\) | \(2.59 \times 10^{-9}\) | \(2.96 \times 10^{-7}\) | \(1.50 \times 10^{-11}\) |
| 0.60      | \(5.57 \times 10^{-9}\) | \(5.57 \times 10^{-9}\) | \(6.48 \times 10^{-7}\) | \(3.84 \times 10^{-11}\) |
| 0.70      | \(1.40 \times 10^{-8}\) | \(1.41 \times 10^{-8}\) | \(1.59 \times 10^{-6}\) | \(1.23 \times 10^{-10}\) |
| 0.80      | \(4.80 \times 10^{-8}\) | \(4.83 \times 10^{-8}\) | \(4.90 \times 10^{-6}\) | \(6.06 \times 10^{-10}\) |
| 0.90      | \(3.58 \times 10^{-7}\) | \(3.64 \times 10^{-7}\) | \(2.66 \times 10^{-5}\) | \(8.67 \times 10^{-9}\) |
| 1.00      | \(9.06 \times 10^{-10}\) | \(9.74 \times 10^{-2}\) | \(\infty\) | \(\infty\) |

Example 4.3 We consider the following nonlinear ODE whose blow-up time \(T = \sqrt{5}\) that was also solved in Cho (2013).

\[ y'' = y^2, \quad y(0) = 1, \quad y'(0) = \sqrt{2/3}, \]

whose exact solution is

\[ y(x) = -\frac{6}{-6 + 2 \sqrt{6x - x^2}}. \]

This example is solved using the NSDM in the PCM and SSM and the results given in Figure 2 show that the NSDM is in good agreement with the exact solution in the neighborhood of the singularity. In particular, we can see from Figure 2 that at the at blow-up time \(T = \sqrt{6}\) the blow-up solutions provided by PCM and SSM are consistent with those given by the exact blow-up solution (Table 5).
4.1. Parabolic equations with blow-up solutions

In this subsection, we SSM to solve parabolic equations with blow-up solutions, since the SSM has the advantage of being self-starting and used without predictors (Figure 3).

Example 4.4

We consider the PDE given (Guo & Yang, 2015).

\[ u_t = u_{xx} + (x^2 - 1)u, \quad x \in [0, 1] \]
\[ u(0, t) = 0, \quad u(1, t) = 0, \quad u(x, 0) = \sin(\pi x), \quad 0 \leq x \leq 1, \]

and the exact solution is given by \( u(x, t) = e^{-t} \sin(\pi x) \).

We obtain the following first order system after discretizing the spatial variable \( x \) using the second order finite difference scheme...
which is written in the form

\[
\begin{aligned}
\frac{du}{dt} &= f(t, u), \\
   u(x_m, 0) &= \sin(\pi x_m),
\end{aligned}
\]  

(8)

where \( f(t, u) = Au + q \), \( q \) is the matrix arising from the central difference approximations and \( q \) is a vector of constants. We then define \( u_m(t) \approx u(x_m, t), u = [u_1(t), u_2(t), \ldots, u_{M-1}(t)]^T \), \( T \) is the transpose.

The problem is now a system of ODEs which is solved by the SSM. In Figure 4, we show that the numerical method (SSM) is in good agreement with the exact solution at \( T = 0.1 \). We note that this consistent is also observed at \( x = 1 \) where a numerical method could perform poorly.

Figure 5. Graphical evidence for 4.1, \( M = 10, N = 80 \). The exact solution and the results provided by the SSM are in good agreement and the errors provided by the numerical solution are also small validating the fact the SSM can perform well when applied to parabolic equations with blow-up solutions.
and exact solutions are in agreement for all \( t \). It is also shown that the global error given by
\[
\text{Error} = \max |u(x_m, t_n) - u_m(t_n)|, \quad m = 0, 1, 2, \ldots, M, \quad n = 0, 1, 2, \ldots, N \quad \text{is small.}
\]
Hence, the SSM is expected to yield good numerical blow-up time for parabolic equations with blow-up solutions.

Example 4.5  We consider the nonlinear PDE given (Guo & Yang, 2015).

\[
\begin{align*}
  u_t &= u_{xx} + u^2, \quad x \in [0, 1] \\
  u(0, t) &= 0, \quad u(1, t) = 0, \quad u(x, 0) = 20 \sin(\pi x), \quad 0 \leq x \leq 1,
\end{align*}
\]

we obtain the following first order system after discretizing the spatial variable \( x \) using second order finite difference schemes

\[
\begin{align*}
  \frac{u_m}{\Delta t} &= \left(\frac{u_{m+1} - 2u_m + u_{m-1}}{(\Delta x)^2}\right) + u_m^2, \quad m = 1, \ldots, M - 1, \\
  u(x_m, 0) &= 20 \sin(\pi x_m),
\end{align*}
\]

which is written in the form \( \dot{u} = f(t, u) \), subject to the initial conditions \( u(t_0) = u_0 \) where \( f(t, u) = Au + q, \ A \) is the matrix arising from the central difference approximations and \( q \) is a vector of constants. We then define \( u_m(t) \approx u(x_m, t) \), \( u = [u_1(t), u_2(t), \ldots, u_{M-1}(t)]^T \), \( T \) is the transpose.

The problem is now a system of ordinary differential equations which is solved by the SSM. In Figure 6, we show that the SSM provides results for the reference blow-up time of \( T = 8.24371 \times 10^{-2} \) for \( N = 80 \) and \( N = 160 \) that are consistent with the results given in Guo and Yang (2015). It observed that the blow-up is located at only one point \( x = 0.5 \) (see Guo & Yang, 2015).
Remark 4.6 Specifically, the PDE in 4.4 is computed by letting

\[
\begin{align*}
\begin{cases}
    f_{m,n} &= \frac{w_{m+1,n} - 2w_{m,n} + w_{m-1,n}}{(\Delta x)^2} + (\alpha^2 - 1)w_{m,n}, \\
    g_{m,n} &= \frac{w_{m+1,n} - w_{m-1,n}}{2\Delta x}, \\ 
    m &= 1, 2, \ldots, M - 1, \quad n = 0, 1, 2, \ldots, N.
\end{cases}
\end{align*}
\]  

(10)

Remark 4.7 Specifically, the PDE in 4.5 is computed by letting

\[
\begin{align*}
\begin{cases}
    f_{m,n} &= \frac{w_{m+1,n} - 2w_{m,n} + w_{m-1,n}}{(\Delta x)^2} + \alpha^2 w_{m,n}, \\
    g_{m,n} &= \frac{w_{m+1,n} - w_{m-1,n}}{2\Delta x}, \\ 
    m &= 1, 2, \ldots, M - 1, \quad n = 0, 1, 2, \ldots, N.
\end{cases}
\end{align*}
\]  

(11)


4.2. Variable step implementation
In this subsection, we used the variable step version of the PCM described in Algorithm 3.

5. Conclusion
We have proposed a NSDM with a variable step-size implementation based on continued fractions for the numerical solution of singular IVPs. It is shown that the NSDM implemented in either the PCM or SSM performs well in the neighborhood of the singularity. Moreover, it is shown that the performance of the NSDM is further enhanced by its variable step-size implementation. Parabolic equations with blow-up solutions are solved using the SSM by discretizing the space variable through finite difference methods to yield a system of ODEs with singular behavior associated with finite time blow-up. Our future research will be to extend the procedure in this paper to deriving and implementing methods with special basis functions (see Garwood & Jator, 2016).

Funding
The authors received no direct funding for this research.

Author details
S.N. Jator
E-mail: Jators@apsu.edu
ORCID ID: http://orcid.org/0000-0001-8167-731X
N. Coleman
E-mail: colemann@apsu.edu

1 Department of Mathematics and Statistics, Austin Peay State University, Clarksville, TN 37044, USA.
2 Department of Computer Science and Information Technology, Austin Peay State University, Clarksville, TN 37044, USA.

Citation information
Cite this article as: A nonlinear second derivative method with a variable step-size based on continued fractions for singular initial value problems, S.N. Jator & N. Coleman, Cogent Mathematics (2017), 4: 1335498.

References
Akinfenwa, O. A., & Jator, S. N. (2013). Continuous block backward differentiation formula for solving stiff ordinary differential equations. Computers and Mathematics with Applications, 65, 998–1005.
Abia, U. M., Lopez-Marcos, J. C., & Martinez, J. (1998). On the blow-up time convergence of semidiscretizations of reaction-diffusion equations. Applied Numerical Mathematics, 26, 389–414.
Brugnano, L., & Trigiante, D. (1998). Solving differential problems by multistep initial and boundary value methods. Amsterdam: Gordon and Breach Science Publishers.
Cash, J. R. (1981). Second derivative extended backward differentiation formulas for the numerical integration of stiff systems. SIAM Journal on Numerical Analysis, 18, 21–36.
Cho, C. (2013). On the computation of the numerical blow-up time. Japan Journal of Industrial and Applied Mathematics, 30, 331–349.
Fatunla, S. O. (1982). Nonlinear multistep methods for initial value problems. Computers & Mathematics with Applications, 8, 231–239.
Garwood, J. J., & Jator, S. N. (2018). Using rational logarithmic basis functions to solve singular differential equations. Electronic Journal of Differential Equations, Conference, 23, 1–7.
Groisman, P. (2006). Totally discrete explicit and semi-implicit Euler methods for a blow-up problem in several space dimensions. Computing, 76, 325–352.
Guo, L., & Yang, Y. (2015). Positivity preserving high-order local discontinuous Galerkin method for parabolic equations with blow-up solutions. Journal of Computational Physics, 289, 181–195.
Hairer, E., & Wanner, G. (1996). Solving ordinary differential equations II. New York, NY: Springer.
Hasan, M. K., Ahamed, M. S., Alam, M. S., & Hossain, M. B. (2013). An Implicit method for numerical solution of singular and stiff initial value problems. Journal of Computational Engineering. doi:10.1155/2013/720812
Ikhile, M. N. O. (2004). Coefficients for studying one-step rational schemes for IVPs in ODEs: II. Extrapolation methods. Computers & Mathematics with Applications, 47, 1463–1475.
Jator, S. N. (2010). On the hybrid method with three-off step points for initial value problems. International Journal of Mathematical Education in Science and Technology, 41, 110–118.
Lambert, J. D. (1991). Numerical methods for ordinary differential systems. England: JohnWiley.
Lambert, D., & Shaw, B. (1965). On the numerical solution of \( y' = f(x, y) \) by a class of formulae based on rational approximation. Mathematics of Computation, 19, 456–462.
Odekunle, M. R., Oye, N. D., Adee, S. O., & Ademiluyi, R. A. (2006). A class of inverse Runge-Kutta schemes for the numerical integration of singular problems. Applied Mathematics and Computation, 158, 149–158.
Onumanyi, P., Awoyemi, D. O., Jator, S. N., & Sirisena, U. W. (1996). New linear multistep methods with continuous coefficients for first order initial value problems. Journal of the Nigerian Mathematical Society, 13, 37–51.
Onumanyi, P., Sirisena, U. W., & Jator, S. N. (1997). Continuous finite difference approximations for solving differential equations. International Journal of Computer Mathematics, 72, 15–27.
Ramos, H., & Vigo-Aguiar, J. (2008). A new algorithm appropriate for solving singular and singularly perturbed autonomous initial-value problems. International Journal of Computer Mathematics, 85, 603–611.
Takayasu, A., Matsue, K., Sasaki, T., Tanaka, K., Mizuguchi, M., & Oishi, S. (2017). Numerical validation of blow-up solutions of ordinary differential equations. Journal of Computational and Applied Mathematics, 314, 10–29.
Tyson, R., Lubkin, S. R., & Murray, J. D. (1999). Model and analysis of chemotactic bacterial patterns in a liquid medium. Journal of Mathematical Biology, 38, 359–375.
Tyson, R., Stern, L.G., & LeVeque, R.J. (2000). Fractional step methods applied to a chemotaxis model. Journal of Mathematical Biology, 41, 455–475.
van Niekerk, F. D. (1998). Rational one-step methods for initial value problems. Computers & Mathematics with Applications, 16, 1035–1039.
