

A Laplace transform finite difference scheme for the Fisher-KPP equation

Journal of Algorithms & Computational
Technology
Volume 15: 1–11
© The Author(s) 2021
DOI: 10.1177/1748302621999582
journals.sagepub.com/home/act
SAGE

Colin L Defreitas  and Steve J Kane

Abstract

This paper proposes a numerical approach to the solution of the Fisher-KPP reaction-diffusion equation in which the space variable is developed using a purely finite difference scheme and the time development is obtained using a hybrid Laplace Transform Finite Difference Method (LTFDM). The travelling wave solutions usually associated with the Fisher-KPP equation are, in general, not deemed suitable for treatment using Fourier or Laplace transform numerical methods. However, we were able to obtain accurate results when some degree of time discretisation is inbuilt into the process. While this means that the advantage of using the Laplace transform to obtain solutions for any time t is not fully exploited, the method does allow for considerably larger time steps than is otherwise possible for finite-difference methods.

Keywords

Laplace transform, Fisher-KPP equation, travelling wave solutions, Talbot inversion, Stehfest inversion, finite difference schemes

Received 11 December 2020; revised 11 December 2020; accepted 12 February 2021

Fisher's equation

Fisher¹ suggested the equation,

$$\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2} + Ku(1 - u), \quad (1)$$

to describe the propagation of a favourable gene in an infinitely long domain. **The equation models the diffusion of an advantageous gene in a 1D habitat. A very informative discussion of the Fisher equation as it relates to propagation is given by.^{2,3} The expression in equation (1) combines the logistic and diffusion equations to simulate the respective processes of population growth and random dispersal of the advantageous gene under consideration. Thus D and K in (1) are the usual positive parameters associated with the diffusion and logistic equations.**

Since its original development the Fisher-KPP equation has been used extensively to describe a wide variety of processes including biology, chemical kinetics, auto-catalytic chemical reactions, branching Brownian motion, flame propagation, neurophysiology, the evolution of a neutron population in a nuclear reactor and chemical wave propagation.⁴

Seen in terms of gene propagation the solution $u(x, t)$ of (1) represents the *proportion* of the mutant gene at a point x in its domain at some time t . Hence we must have that,

$$0 \leq u(x, t) \leq 1 \quad (2)$$

Fisher showed that (1) together with the additional boundary conditions,

$$u(-\infty, t) = 1 \text{ and } u(+\infty, t) = 0 \quad (3)$$

Exhibit travelling wave solutions of the form,

$$u(x, t) = u(x - ct) \quad (4)$$

School of Physics Astronomy and Mathematics, University of Hertfordshire, Hatfield, UK

Corresponding author:

Colin L Defreitas, University of Hertfordshire, College Lane, Hatfield AL10 9AB, UK.
Email: colindefreitas@hotmail.com



moving at constant speed c in the positive x direction provided

$$c \geq C_{min} \quad (5)$$

when $C_{min} = 2\sqrt{KD}$.⁵

Thus the Fisher-KPP equation has an infinite number of travelling wave solutions, each moving with a wave speed $c \geq 2$. The solutions of $u(x, t)$ then connects the steady-state solution at $u=1$ and the steady-state solution at $u=0$. In keeping with the analysis of these steady-state conditions, $u=1$ is stable, while $u=0$, unstable.

It is worth noting that analytical solutions of the Fisher-KPP equation exist for only a small class of problems and hence the importance of developing efficient numerical schemes to obtain solutions to (1).

Although Fisher proposed his model for the wave advancement of an advantageous gene in 1937, it was not until 1974 that numerical solutions to the equation began to appear. The first of which was the seminal paper by Canosa⁶ who used the Accurate Space Derivative method (ASD), sometimes referred to as the pseudo-spectral approach. Since then, many researchers have investigated numerical solutions to equation (1) for which Anjal et al. give a comprehensive summary of the main contributions.⁴ However, these methods all incorporate some small time discretisation process, which requires iterations of the algorithm at each time step. As we discuss in the next section, our proposed solution to (1) allows us to obtain accurate results with considerably larger discretisation in the time domain.

In developing a numerical approach to solve the Fisher-KPP equation, we needed to keep two important points in mind. First, Canosa⁶ showed that all waves are stable against *small* local perturbations but linearly unstable against general perturbations of *infinite* extent. This sensitivity to perturbations of infinite extent is essential for us because, as we explain in section 'Numerical examples and discussion', the LTFDM involves inversion procedures which can introduce perturbations into the solution.

The second point is that Canosa was able to demonstrate by a simple stability analysis that computation is unstable against round-off errors building up at the leading tail of the waves.⁶ We were able to overcome this difficulty by a particular application of the inversion process for the LTFDM.

The Laplace transform finite difference method

We consider an approach to the numerical solution of the Fisher-KPP equation (1) in which the space

variable is developed using a purely finite difference approach and the time development is obtained using a hybrid LTFDM.

The significant advantage of this method is that it eliminates the time dependency parameter and the associated discretisations which are necessary to obtain solutions at a particular time t .

When using finite difference and other time discretisation methods to solve differential equations, the size of the time step is limited by the stability conditions required for convergence of the scheme. In linear cases, this usually involves hundreds and sometimes thousands of time steps to arrive at the solution for some desired time. Iterations are then required at each time step which involves using a variety of matrix methods to solve the vast systems of linear equations generated by the scheme.

For non-linear cases, this is compounded by the fact that a further iterative process is usually required at each time step. Since each of these iterations introduces a certain amount of round-off and truncation error, careful consideration must be given to their control and management when implementing these schemes.

The Laplace transform has the potential of doing away with time discretisation, and its associated error management by transforming the time domain into the Laplace space, s , via the integral transform,

$$\int_0^{\infty} f(t)e^{-st} dt = F(s) \quad (6)$$

Then computations done in the Laplace space, s , can be inverted back into the time domain at any desired time t . Hence the LTFDM can lead to the required solution with virtually one-time step. By employing this method, we can potentially obtain substantial increases in speed and accuracy over traditional finite difference and time discretisation methods. With the additional benefit of reducing by one the dimensions of the governing equation, simplifying the resulting finite difference scheme needed to discretise the remaining space variable.

Inverting the data

The recovery of the function $f(t)$ is via the inverse Laplace transform which is most commonly defined by the Bromwich integral formula

$$\mathcal{L}^{-1}\{F(s)\} = f(t) = \frac{1}{2\pi i} \int_{u-i\infty}^{u+i\infty} F(s) e^{st} ds \quad (7)$$

for some $u \in \mathbb{R}$.⁷ The choice of s in (6) and so in (7) is not an arbitrary one. If we choose s so that it lies on

the positive real axis, we are treating the solution of (6) as a positive real integral equation. The problem here is that the inverse problem is known to be ill-posed, meaning that small changes in the values of $F(s)$ can lead to large errors in the values for $f(t)$.³

Hence when Laplace transform methods are employed for finding numerical solutions to partial differential equations, we must take account of the fact that the corresponding inversion methods can be highly sensitive to the inevitable noisy data that arises in their computation via truncation and round-off error, a process which is exacerbated for non-linear schemes. Our method attempts to mitigate these factors by employing the Fixed Talbot inversion algorithm. In our earlier work,⁸ we have shown that this inversion scheme reduces the effects that noisy data can have in adversely perturbing the finite difference scheme. In this sense it can produce better results than the widely used Stehfest inversion method.

Method

We first non-dimensionalise equation (1) by letting

$$u = U\bar{u} \quad x = L\bar{x} \quad \text{and} \quad t = T\bar{t} \quad (8)$$

Employing the chain rule,

$$\frac{\partial}{\partial x} = \frac{1}{L} \frac{\partial}{\partial \bar{x}} \quad \text{and} \quad \frac{\partial}{\partial t} = \frac{1}{T} \frac{\partial}{\partial \bar{t}} \quad (9)$$

By letting $U = a$, $T = \frac{1}{ka}$ and $L = \sqrt{\frac{D}{ka}}$, and dropping the overbars (1) becomes,

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + u(1 - u) \quad (10)$$

with boundary conditions

$$u(-\infty, t) = 1 \quad \text{and} \quad u(+\infty, t) = 0 \quad (11)$$

The Laplace transform of the time derivative in (10) is

$$\mathcal{L}\left\{\frac{\partial u}{\partial t}\right\} = s\bar{u}(x, s) - u(x, 0) \quad (12)$$

where

$$\bar{u}(x, s) = \mathcal{L}\{u(x, t)\} \quad (13)$$

And the Laplace transform of the space derivative in (10) is

$$\mathcal{L}\left\{\frac{\partial^2 u}{\partial x^2}\right\} = \frac{d^2}{dx^2} \bar{u}(x, s) \quad (14)$$

However, it is well known that the Laplace transform cannot be successfully performed on non-linear governing equations and so some linearization process is necessary before the LTFDM can be implemented.⁹ To overcome this, we follow Zhu et al.¹⁰ who successfully applied the Laplace Transform dual reciprocity method to diffusion equations of the form,

$$\nabla^2 u = \frac{\partial u}{\partial t} - \beta f(u) \quad (15)$$

where β is a given constant and $f(u)$ is a non-linear function. **Zhu first decomposed the function $f(u)$ in equation (15) into $\tilde{f}(\bar{u})$** then in order to find the solution of the unknown function at particular time t_1 equation (15) is linearised as

$$\nabla^2 u = \frac{\partial u}{\partial t} - \beta \tilde{f}(\bar{u})u \quad (16)$$

in which \bar{u} is the previously iterated solution at time t_1 . Thus for equation (1) we would have,

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + u - u\bar{u} \quad (17)$$

Then the Laplace transform of (17) is,

$$\bar{u}(x, s) - u(x, 0) = \frac{d^2}{dx^2} \bar{u}(x, s) + \bar{u}(1 - \bar{u}) \quad (18)$$

with transformed boundary conditions,

$$\bar{u}(-\infty, t) = \frac{1}{s} \quad \text{and} \quad \bar{u}(+\infty, t) = 0 \quad (19)$$

Using a central-difference scheme on the space derivative, the finite difference scheme for (18) is,

$$\bar{u}_{i-1} - \bar{u}_i(2 + \delta x^2 s_j + \delta x^2 - \delta x^2 \bar{u}_i) + \bar{u}_{i+1} = -\delta x^2 u(0)_i \quad (20)$$

where δx is the size of the space step in the x -direction, s_j is the j th Laplace parameter and $u(0)_i = u(x_i, 0)$. Then (20) can be expressed as the tridiagonal system,

$$\begin{pmatrix} a_{1j} & 1 & & 0 \\ 1 & a_{2j} & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 1 & a_{n-1j} \end{pmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \dots \\ u_{n-1} \end{bmatrix} = - \begin{bmatrix} \delta x^2 u(x_1, 0) + \frac{1}{s_j} \\ \delta x^2 u(x_2, 0) \\ \dots \\ \delta x^2 u(x_{n-1}, 0) \mathfrak{R} \end{bmatrix} \quad (21)$$

where

$$a_{ij} = 2 + \delta x^2 s_j + \delta x^2 - \delta x^2 \tilde{u}_i \quad (22)$$

in (20). After solving this tridiagonal system the data is then inverted to transition from the Laplace space, s , back into the time domain.

The Stehfest algorithm for numerically inverting the Laplace transform. In their wide-ranging study of algorithms for inverting the Laplace transform Davies and Martin¹⁰ cite the Stehfest algorithm [15] as providing accurate results on a wide variety of test functions. Since then this algorithm has become widely used for inverting the Laplace Transform and being favoured due to its reported accuracy and ease of implementation.

The algorithm takes the transformed data in the Laplace space $F(s)$ and produces $f(t_1)$ for a specific time value $t = t_1$. Choosing

$$s_j = j \frac{\ln 2}{t_1}, \quad j = 1, 2, \dots, M, \text{ for } M \text{ even} \quad (23)$$

The numerical inversion is given by

$$f(t) \approx \frac{\ln 2}{t} \sum_{j=1}^M A_j F(s_j) \quad (24)$$

with

$$A_j = (-1)^{\frac{M}{2}+j} = \sum_{k=\lfloor \frac{j+1}{2} \rfloor}^{\min(j, \frac{M}{2})} \frac{k^{\frac{M}{2}} (2k)!}{(\frac{M}{2} - k)! k! (k-1)! (j-k)! (2k-j)!} \quad (25)$$

Theoretically, $f(t)$ becomes more accurate for larger M , but the reality is that rounding errors worsen the results if M becomes too large. According to Stehfest, “The optimum M is approximately proportional to the number of digits the machine is working with”.¹¹

Also in our earlier work we found that the Stehfest algorithm does not handle noisy data well.⁸ As we show in section ‘Numerical examples and discussion’, this can have the effect of introducing perturbations into the travelling wave solutions of the Fisher-KPP equation.

The fixed Talbot algorithm for numerically inverting the Laplace transform

Here we use the function.

$$S(z) = \frac{z}{1 - e^{-z}} \quad (26)$$

which maps the closed interval $M = [-2\pi i, 2\pi i]$ on the imaginary z -plane onto the curve L in the s -plane giving the integral,

$$f(t) = \frac{1}{2\pi i} \int_L F(s) e^{st} ds \quad (27)$$

(See Logan¹² for the details of this transformation).

Next we follow the procedure as adopted by Logan for numerically integrating (27).

With $s = S(z)$ this becomes

$$f(t) = \frac{1}{2\pi i} \int_M [F(S(z)) e^{S(z)t} S'(z)] dz \quad (28)$$

where

$$S'(z) = \frac{1 - (1+z)e^{-z}}{(1 - e^{-z})^2} \quad (29)$$

For convenience we write,

$$f(t) = \frac{1}{2\pi i} \int_M I(z, t) dz \quad (30)$$

where

$$I(z, t) = [F(S(z)) e^{S(z)t} S'(z)] \quad (31)$$

The integral in (30) is then rotated by $\frac{\pi}{2}$ so the interval of integration is now real and becomes $[-2\pi, 2\pi]$, then we use the trapezoid rule with n odd and $w = -iz$ to obtain

$$f(t) \approx \frac{1}{n} \left\{ (I(2\pi i) + I(-2\pi i) + 2 \sum_{j=1}^{n-1} I(iw_j)) \right\} \quad (32)$$

where

$$w_j = 2\pi \left\{ \frac{2j}{n} - 1 \right\} \quad (33)$$

and we note that $I(2\pi i) = I(-2\pi i) = 0$.¹²

Numerical examples and discussion

Example 1

For our first example we use (10)

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + u(1 - u)$$

and its associated boundary conditions,

$$u(-\infty, t) = 1 \text{ and } u(+\infty, t) = 0.$$

Ablowitz and Zeppetella¹³ give an exact solution for a particular wave speed $c = \pm \frac{5}{\sqrt{6}}$ as

$$u(x, t) = \frac{1}{\left[1 + \exp\left(\sqrt{\frac{1}{6}}x - \left(\frac{5}{6}\right)t\right)\right]^2} \quad (34)$$

which we use to compare our numerical results with.

When we first implemented the LTFDM it produced distortions in the upper tail of the travelling wave for larger values of t . This is shown in Figure 1. We eventually surmised that these distortions were due to the existence of perturbations of infinite extent. In other words, the approximation of the initial condition on a finite domain. A stability analysis carried out by Gazdang et al.¹⁴ showed that super speed waves or waves with speed greater than C_{min} could be maintained if subject only to infinitesimally small positive perturbations.

As is well known, the numerical inversion of the Laplace transform is a perturbed problem. Thus perturbations generated by the numerical scheme itself can then introduce noise into the inversion algorithms which cannot be completely filtered out. However, we found that these perturbations can be reduced if some time discretisation, together with a reinitialisation of the initial condition, is introduced into the numerical method. While the full benefit of using the Laplace transform, i.e., to solve for any time t is partially diminished, introducing some measure of time discretisation meant we were able to use larger time steps than would be the case for other finite difference methods.¹⁵

As we have shown in our previous work,⁸ the Fixed Talbot inversion method is more efficient at filtering out this noise than the more widely used Stehfest algorithm. This is shown in Figures 2 and 3 where oscillations in the right-hand tail are present when using the Stehfest inversion method for the time step $t=0.8$ but are absent in the Talbot inversion method. Thus smaller time steps are required for comparable accuracy for the Stehfest inversion than for the Talbot. Because of its inability to deal adequately with noisy data, the Stehfest algorithm is also sensitive to the spatial step size δx as smaller space discretisations can also introduce round-off error into our computations. Hence our choice of using the Talbot algorithm for carrying out the LTFDM inversion procedure. **Details of this method on the field of study can be found in.**⁸

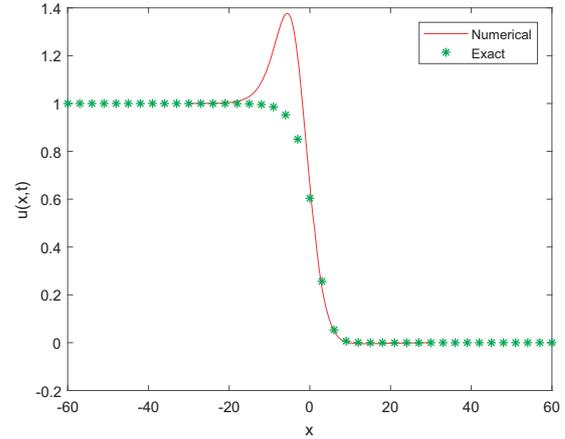


Figure 1. Profile without time discretisation. $t = 1.5$.

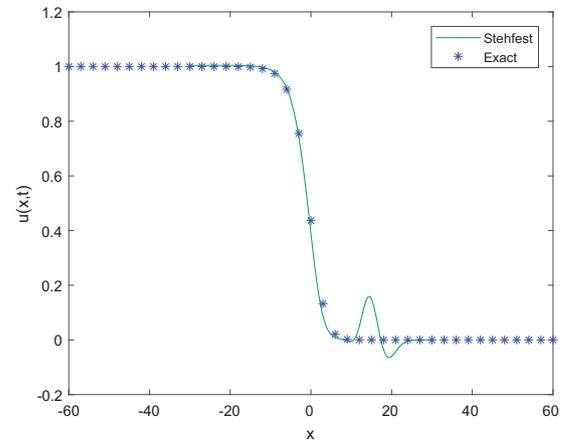


Figure 2. Profile using Stehfest. $t = 0.8$.

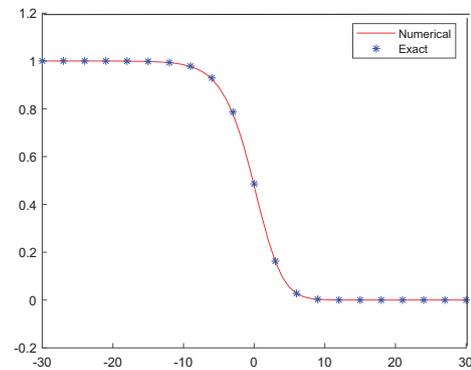


Figure 3. Profile using Talbot. $t = 0.8$.

The Talbot algorithm is also very effective in dealing with the build-up of round-off error in the right tail of the waves. As Canosa points out “This does not seem due to the numerical method used but to the physical nature of the problem described by the equation which

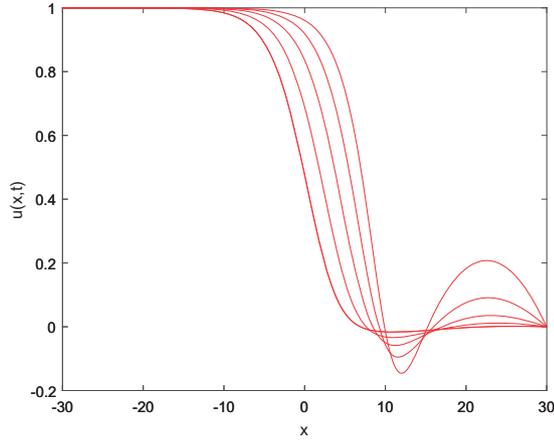


Figure 4. Talbot: $n = 55$ for $t = 1$ to 5.

gives rise to an exponential growth of the solutions when this is exponentially small. This basic difficulty makes it difficult to do a rigorous simulation of the solutions of Fisher's equation".

This effect is shown in Figure 4. However, we overcome this problem by merely increasing n in (32) from $n = 55$ to 555, which completely removes the instability and restores the travelling wave profile Figure 5. The restoration of the travelling profile is due to the Talbot algorithm's ability to filter out noise with increasing n .⁸

While no exact solutions exist for (1) for wave speeds other than $c = \pm \frac{5}{\sqrt{6}}$ we can use the approximate asymptotic solution,

$$U(z) = (1 + e^{\frac{z}{c}})^{-1} + \frac{1}{c^2} e^{\frac{z}{c}} (1 + e^{\frac{z}{c}})^{-2} \log \left[\frac{4e^{\frac{z}{c}}}{(1 + e^{\frac{z}{c}})^2} \right] + O\left(\frac{1}{c^4}\right) \quad (35)$$

With $c \geq C_{min} = 2$ and $z = x - ct$, to test our numerical scheme for a variety of wave speeds. The accuracy of the asymptotic solution increases for large c .¹⁶

Example 2

Cattani, Carlo et al.¹⁷ give an exact solution for the Fisher type equation,

$$\frac{\partial u}{\partial t} = v \frac{\partial^2 u}{\partial x^2} - bu^2 + au \quad (36)$$

where

$$t \in [0, t], \quad 0 < t \leq \infty, \quad -\infty < x < \infty \quad (37)$$

with the boundary condition

$$u(-\infty, t) = 0.5, \quad u(\infty, t) = 0 \quad (38)$$

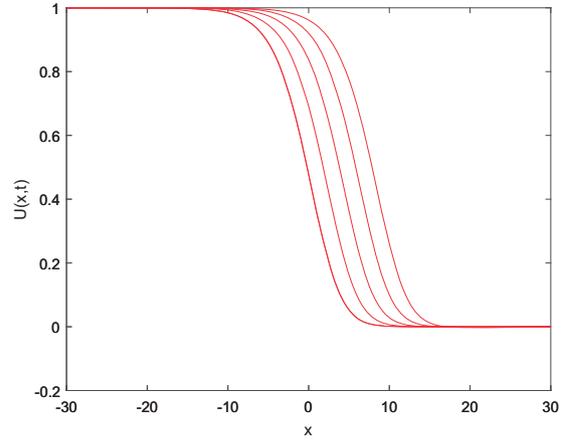


Figure 5. Talbot: $n = 555$ for $t = 1$ to 5.

and initial condition,

$$u(x, 0) = -\frac{1}{4b} \left[\operatorname{sech}^2 \left(-\sqrt{\frac{a}{24c}} x \right) - 2 \tanh \left(-\sqrt{\frac{a}{24c}} x \right) - 2 \right] \quad (39)$$

The exact solution is,

$$u(x, t) = -\frac{1}{4b} \left[\operatorname{sech}^2 \left(-\sqrt{\frac{a}{24c}} x + \frac{5a}{12} t \right) - 2 \tanh \left(-\sqrt{\frac{a}{24c}} x + \frac{5a}{12} t \right) - 2 \right] \quad (40)$$

Since no exact solution exists for all wave speeds for (36) we derived a perturbation solution to test the numerical scheme at a variety of wave speeds. The perturbation solution for this case is given as,

$$U(z) = \frac{1}{2} (1 + e^{\frac{z/2}{c}})^{-1} + \frac{1}{c^2} e^{\frac{z/2}{c}} (1 + e^{\frac{z/2}{c}})^{-2} \log \left[\frac{\sqrt{2} e^{\frac{z/8}{c}}}{(1 + e^{\frac{z/2}{c}})^2} \right] + O\left(\frac{1}{c^4}\right) \quad (41)$$

Results

In our investigations we found our algorithm performs with equal accuracy for space steps $0 < \delta x \leq 1$ and with the Laplace transform used within the time steps $\Delta t = 0.1, 0.2$ and 0.4. This shows that it is stable across a wide variety of parameters.

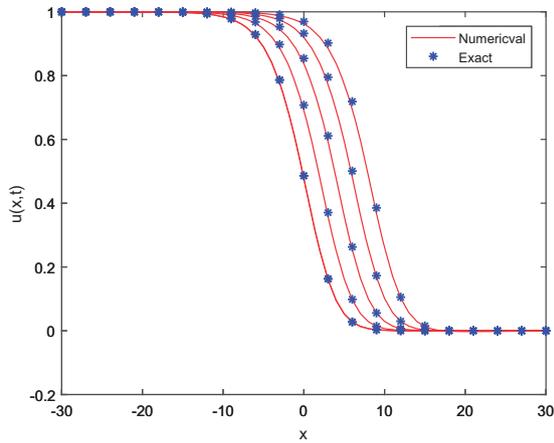


Figure 6. Profile $\Delta t = 0.2, t = 1$ to 5.

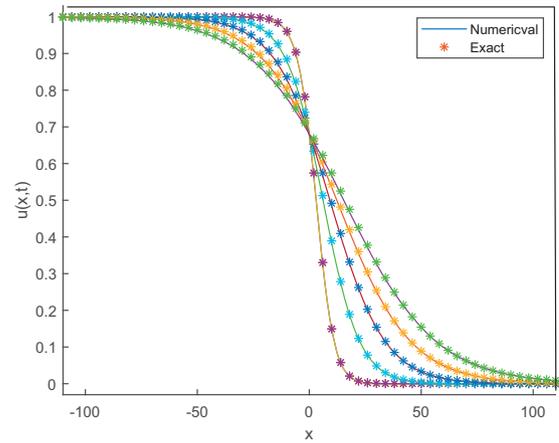


Figure 8. Profile Example 1, $\Delta t = 0.2, t = 1, C = 4 : 4 : 20$.

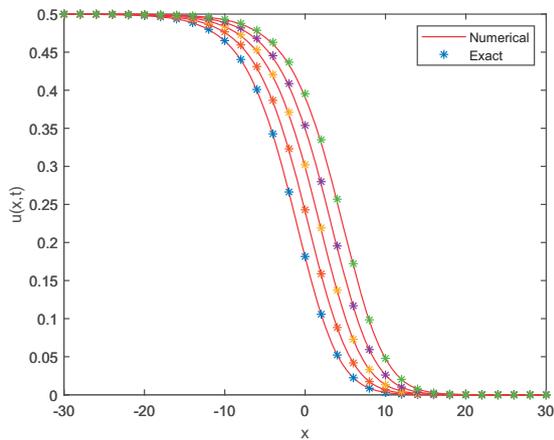


Figure 7. Profile $\Delta t = 0.2, t = 1$ to 5.

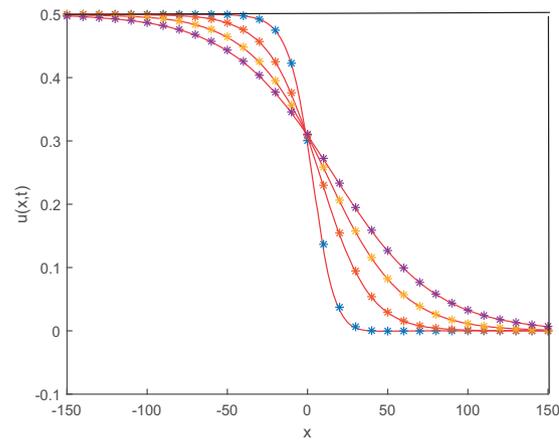


Figure 9. Profile Examole 2, $\Delta t = 0.2, t = 1, C = 4 : 4 : 20$.

Figure 6 shows the travelling wave profile for (11) example 1, compared with exact solution.¹³ The time discretisation used in the LTFDM is $\Delta t = 0.2$ with a space step of $\delta x = 0.1$. The numerical results show good agreement with the exact solution.

Figure 7 shows the travelling wave profile for (32) example 2 with exact solution.¹⁷ The time descritisation used in the LTFDM is $\Delta t = 0.2$ with a space step of $\delta x = 0.1$. The numerical results show good agreement with the exact solution.

Figure 8 shows the travelling wave profile for example 1 compared with the perturbation solution for wave speeds $C = 4, 8, 12, 16, 20$. The time descritisation used in the LTFDM is $\Delta t = 0.2$ with a space step of $\delta x = 0.1$. The numerical results show good agreement with the exact solution.

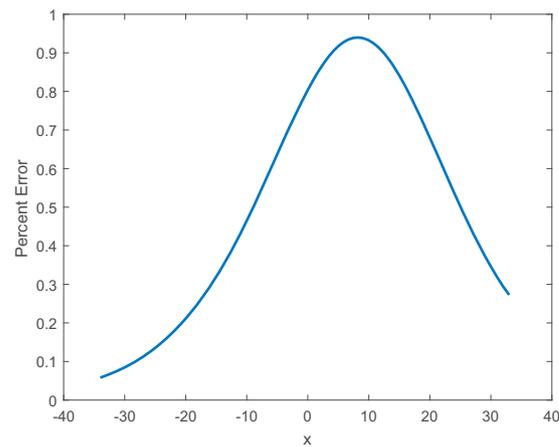


Figure 10. Example 1 Error Profile $\Delta t = 0.1, t = 1, C = 10$.

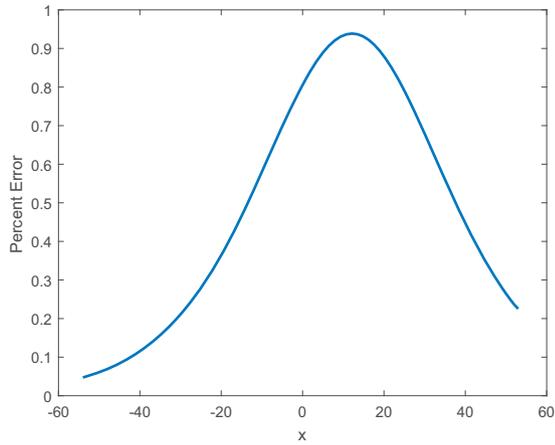


Figure 11. Example 1 Error Profile $\Delta t = 0.1, t = 1, C = 15$.

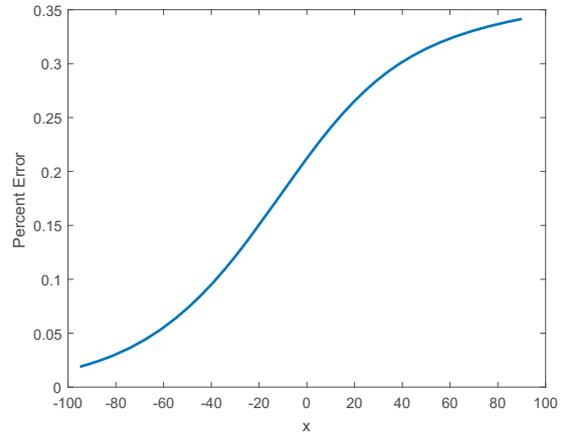


Figure 14. Example 2 Error Profile $\Delta t = 0.1, t = 1, C = 15$.

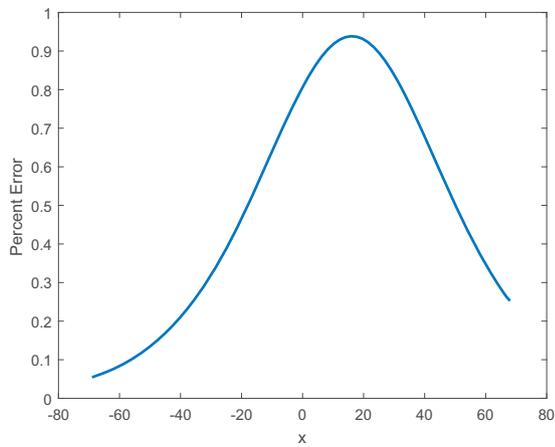


Figure 12. Example 1 Error Profile $\Delta t = 0.1, t = 1, C = 20$.

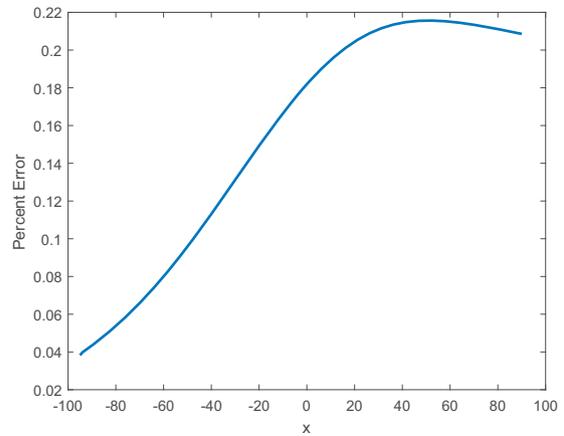


Figure 15. Example 2 Error Profile $\Delta t = 0.1, t = 1, C = 20$.

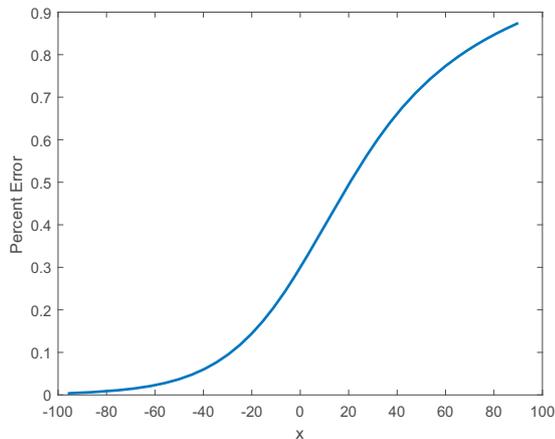


Figure 13. Example 2 Error Profile $\Delta t = 0.1, t = 1, C = 10$.

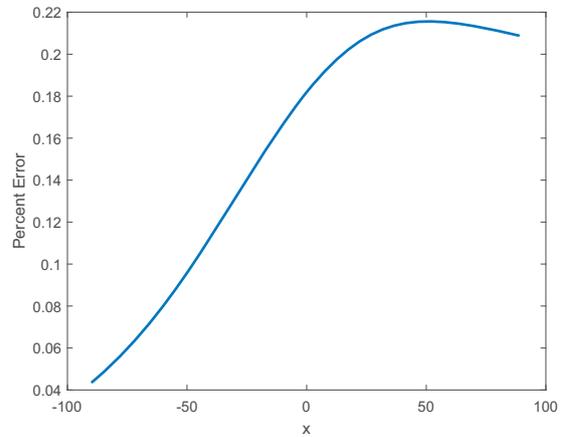


Figure 16. Example 2 Error Profile $\delta x = 1$.

Figure 9 shows the travelling wave profile for example 2 compared with perturbation solution (41) for wave speeds $C = 4, 8, 12, 16, 20$. The time discretisation used in the LTFDM is $\Delta t = 0.2$ with a space step of $\delta x = 0.1$. The numerical results show good agreement

with the exact solution. Figures 10-12 show the error profile for example 1 for the wave speed shown.

Tables 1-3 present the results for Example 1 for times $t = 1, t = 2$, and $t = 4$. For all cases shown we set $\Delta t = 0.1, n = 555, \delta x = 0.25$, and $L = 60$.

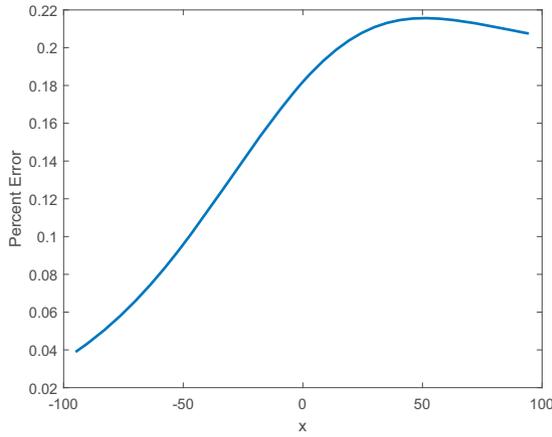


Figure 17. Example 2 Error Profile $\delta x = 0.5$.

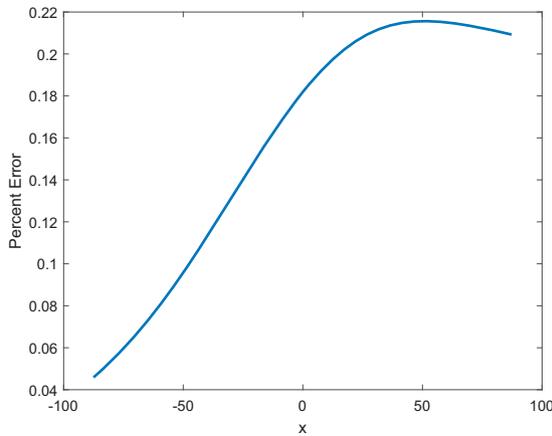


Figure 18. Example 2 Error Profile $\delta x = 0.25$.

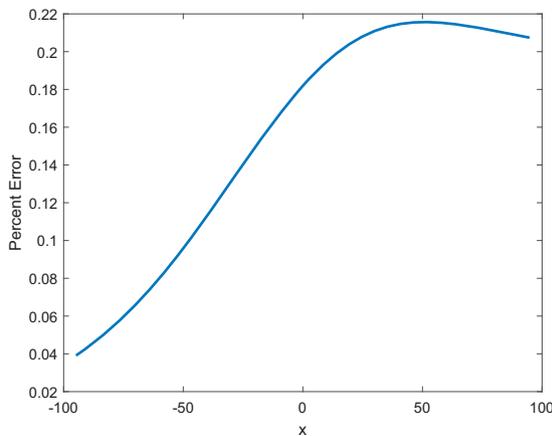


Figure 19. Example 2 Error Profile $\delta x = 0.05$.

Tables 4-6 present the results for problem 2 for times $t = 1$, $t = 2$, and $t = 4$. For all cases shown we set $\Delta t = 0.4$, $n = 555$, $\delta x = 0.25$, and $L = 60$.

For brevity we give a sample of the results in Tables 7-9 of the comparison of our method with the

Table 1. Example 1, $t = 1$.

x	Numerical	Exact	Error
-20	0.999742	0.999753	1.0 (-5)
-12	0.993285	0.995552	2.6 (-4)
-4	0.845643	0.849618	1.2 (-3)
2	0.252043	0.254227	8.6 (-3)
4	0.0956577	0.096161	5.2 (-3)
8	0.0006466	0.006515	7.5 (-3)
12	0.000178	0.000284	3.7 (-4)

Table 2. Example 1, $t = 2$.

x	Numerical	Exact	Error
-20	0.999884	0.999893	8.8 (-6)
-12	0.996963	9.97,190	2.3 (-4)
-8	0.984656	0.995740	1.1 (-3)
0	0.698929	0.707501	1.2 (-2)
4	0.255059	0.490844	1.5 (-2)
8	0.027968	0.028250	2.0 (-2)
10	0.006447	0.006719	2.1 (-2)

Table 3. Example 1, $t = 4$.

x	Numerical	Exact	Error
-20	0.999976	99,99,808	3.4 (-6)
-12	0.999379	0.999468	8.9 (-5)
-6	0.992958	0.993864	9.2 (-4)
2	0.846588	0.856205	1.1 (-2)
4	0.88938265	0.90291741	1.5 (-2)
6	0.490436	0.500723	2.1 (-2)
10	0.100190	0.103045	2.8 (-2)

Table 4. Example 2. $t = 1$.

x	Numerical	Exact	Error
-20	0.497713	0.497780	1.3 (-4)
-12	0.477704	0.478304	1.3 (-3)
-8	0.434286	0.435769	3.4 (-3)
-1	0.220496	0.98636	8.5 (-3)
4	0.0512810	0.051519	4.5 (-3)
15	0.000163	0.000162	6.2 (-3)
20	9.33(-6)	9.34(-6)	4.5 (-4)

approximate perturbation solution for example 2 with $t = 1$. The length L is increased for higher wave speeds to ensure full propagation of the wave as it moves to the right with increasing speed. In all cases $n = 555\delta x = 1$, $\Delta t = 0.1$. Figures 13-16 below show the error profile for example 2 comparing our scheme with the approximate perturbation solution at $t = 1$, for

Table 5. Example 2. $t = 2$.

x	Numerical	Exact	Error
-20	0.498548	0.497780	2.2 (-4)
-8	0.456965	0.459476	5.8 (-3)
-8	0.434286	0.435769	3.4 (-3)
01	0.238570	0.242958	1.8 (-2)
4	0.087159	0.088343	1.3 (-2)
15	0.000432	0.000432	7.9 (-2)
20	2.53(-5)	2.52(-5)	1.1 (-4)

Table 6. Example 2. $t = 4$.

x	Numerical	Exact	Error
-20	0.499320	0.499413	1.9 (-4)
-8	0.479300	0.481756	5.1 (-3)
-1	0.375701	0.383712	2.1 (-2)
01	0.238570	0.242958	1.8 (-2)
2	0.272176	0.279941	2.8 (-2)
15	0.002122	0.002123	2.3 (-2)
20	0.00013143	0.0001311	1.9 (-3)

Table 7. Example 2 $t = 1$ $C = 10$.

x	Numerical	Exact	Error
-124	0.499186	0.499190	1.0 (-5)
-100	0.497323	0.497399	1.5 (-4)
-40	0.451696	0.451966	6.0 (-4)
-30	0.425057	0.425457	9.4 (-4)
10	0.214997	0.215853	4.0 (-3)
20	0.155856	0.156620	5.0 (-3)
50	0.009998	0.010083	8.5 (-3)

Table 8. Example 2 $t = 1$ $C = 15$.

x	Numerical	Exact	Error
-140	0.496314	0.496336	4.4 (-5)
-100	0.486330	0.486413	1.6 (-4)
-40	0.414178	0.414573	6.0 (-4)
-20	0.425057	0.425457	9.5 (-4)
0	0.279545	0.280139	2.1 (-3)
40	0.1242715	0.124647	3.0 (-3)
80	0.039467	0.039600	8.5 (-3)

various wave speeds. The error profile and the corresponding range of errors remain unchanged for varying wave speeds, for example, 1, but the error decreases with increasing wave speed, for example 2.

The Figures 17 to 19 demonstrate the stability of the results for varying mesh size δx . For brevity we show this for problem 2 at wave speed $C = 20$ and for time $t = 1$.

Table 9. Example 2, $t = 1$ $C = 20$.

x	Numerical	Exact	Error
-182	0.495876	0.495901	4.9 (-5)
-100	0.469727	0.469891	3.5 (-4)
-10	0.310285	0.310802	1.7 (-3)
-20	0.425057	0.425457	9.5 (-4)
50	0.133185	0.133472	2.2 (-3)
70	0.090021	0.090213	2.1 (-3)
176	0.007480	0.007496	8.5 (-3)

We were able to achieve the same error profile with these mesh sizes for all the problems investigated in this paper.

Conclusion

This paper proposes a numerical approach to the solution of the Fisher-KPP reaction-diffusion equation in which the space variable is developed using a purely finite difference scheme and the time development is obtained using a hybrid Laplace Transform Finite Difference Method (LTFDM). This method to our knowledge has not previously been applied to the Fisher-KPP equation, and Laplace transform methods are generally not deemed suitable for equations which have travelling wave solutions.

However, by introducing some time discretisation into our LTFDM we were able to obtain results which have less than one per cent error over a range of times, space and time discretisation, for variety of wave speeds. The time discretisation was necessary to reduce perturbations of infinite extent which occur in numerical schemes for the Fisher-KPP equation. These perturbations can have a detrimental effect on the LTFDM since all the numerical schemes for inverting the Laplace transform are highly perturbed problems.

Thus crucial to the success of the method outlined in this paper is the choice of the Fixed Talbot inversion algorithm which as we have shown in our earlier work is best at dealing with the inherent noise generated in finite difference schemes. This algorithm also had the effect of ironing out the build-up of round-off error in the right-hand tail of the travelling wave a consequence of the physical nature of the problem.

Authors' note

The code used in this paper can be accessed at: <https://github.com/SJKane/LTFDM—Fisher>

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

ORCID iD

Colin L Defreitas  <https://orcid.org/0000-0002-5215-7670>

References

1. Fisher RA. The wave of advance of advantageous genes. *Ann Eugen* 1937; 7: 355–369.
2. Bakhoun EG and Toma C. Mathematical transform of traveling-wave equations and phase aspects of quantum interaction. *Math Prob Eng* 2010; 2010: 1–15.
3. Moran PAP, et al. The statistical processes of evolutionary theory. *Am J Hum Genet* 1962; 14: 438–439.
4. Verma A, Jiwari R and Koksai ME. Analytic and numerical solutions of nonlinear diffusion equations via symmetry reductions. *Adv Diff Equ* 2014; 2014: 229.
5. Murray JD. *Mathematical biology: I. An introduction*. Vol. 17. Berlin: Springer Science & Business Media, 2007.
6. Canosa J. On a nonlinear diffusion equation describing population growth. *IBM J Res Dev* 1973; 17: 307–313.
7. Spiegel MR. *Laplace transforms*. New York: McGraw-Hill, 1965.
8. Defreitas CL and Kane SJ. The noise handling properties of the Talbot algorithm for numerically inverting the Laplace transform. *J Algorith Comput Technol* 2019; 13: 1748301818797069.
9. Zhu S and Satravaha P. Solving nonlinear time-dependent diffusion equations with the dual reciprocity method in Laplace space. *Eng Anal Boundary Elem* 1996; 18: 19–27.
10. Davies B and Martin B. Numerical inversion of the Laplace transform: a survey and comparison of methods. *J Comput Phys* 1979; 33: 1–32.
11. Stehfest H. Algorithm 368: numerical inversion of Laplace transforms [d5]. *Commun Acm* 1970; 13: 47–49.
12. Logan JDT. *Modeling in hydrogeochemical systems*. Vol. 15. Berlin: Springer Science & Business Media, 2013.
13. Ablowitz MJ and Zeppetella A. Explicit solutions of fisher's equation for a special wave speed. *Bull Math Biol* 1979; 41: 835–840.
14. Gazdag J and Canosa J. Numerical solution of fisher's equation. *J Appl Probab* 1974; 11: 445–457.
15. Simmonds JG and Mann JE. *A first look at perturbation theory*. North Chelmsford: Courier Corporation, 2013.
16. Murray J. *Mathematical biology I: an introduction, ser. interdisciplinary applied sciences*. NY, USA: Springer-Verlag, 2002.
17. Cattani C and Kudreyko A. Multiscale analysis of the fisher equation. In *International conference on computational science and its applications*, Perugia, Italy, 30 June–3 July 2008, pp. 1171–1180. Berlin: Springer.