# Abstract

The Kuramoto-Shivashinsky equation provides a chaotic equation that considers dissipation and elasticity of a function. It is thus of great interest to study to understand the behavior. A solver was created for this study that provides flexibility in generating a solution. In this study, the solver uses a 6th order in space method with the Runge-Kutta 4, 4th order time integration. The boundary conditions were set up using a 5th order in space method. Overall, the solver shows good order of accuracy to the expected values.

The Kuramoto-Shivashinsky equation solution offers an interesting look into the interactions between different waves in an advection behavior that is influenced by diffusion and stiffness effects. The results in this study are nondimensionalized by the traditional Reynolds number, but also by a novel stiffness quantity to better compare these results against those of other studies. The addition of stiffness behavior appears to affect the coalescence of waves to form discontinuities in that the properties of the advection and diffusion change. Additionally, the time integration method seems to affect the transport of various smaller waves with higher order methods dissipating these smaller motions much more rapidly.

# Introduction

The Kuramoto-Shivashinsky (KS) equation was developed concurrently by two researchers focusing on development in laminar flames in [1] and [2]. The premise is that for a laminar flame, the combustion of various species is a chaotic behavior that results in certain instabilities within the flame front. To save the derivation, the KS equation is shown in Equation 1.

Equation . Kuramoto-Shivashinsky Equation with Coefficients. Adapted from [3].

There are five terms in this form of the KS equation. The first is the time-derivative, which has an obvious effect. It is accompanied by the second term, the convection term, which provides non-linearity to the equation. This takes the form like the Navier-Stokes equation. The third term is the diffusion term, which is the anti-diffusion, or coalescence, term in this form due to the lack of a negative sign. The fourth term is what is called a biharmonic operator [1], the most obvious example being an elastic deflection on a beam. On the right-hand side (RHS) of the equation, there is a forcing function that is not present during the rest of the study.

This study looks at solving the KS-equation to better understand the stability and underlying behavior.

# Methodology

This project uses the methods outlined in Table 1. A minimum of 4th order is required for adequate assessment of the 4th derivative, and this study will go two steps further using a 6th order in space solver. The time method uses the Runge-Kutta 4th order time integration, which seems to be more common in the research world. During some experimentation, the 5th order helped to avoid generation of random wave-spikes and instabilities along the boundaries.

Table . Solver Methods.

|  |  |
| --- | --- |
| Solver | Methodology |
| Spatial | 6th Order |
| Time | RK-4 |
| BC Spatial | 5th Order |

## Spatial Gradients

For the fourth order gradient, the stencil requires four points. The stencils for the interior are shown in Figure 1, Figure 2, and Figure 3.

A black circle with a black line

Description automatically generated

Figure . Interior Stencil.

A black circle with a black dot

Description automatically generated

Figure . LHS Boundary Stencil.

A black dot with a black line

Description automatically generated

Figure . RHS Boundary Stencil.

These stencils result in the numerical gradients in the following sections. These numerical gradients are in the matrices that solve the KS equation.

### 1st Derivative

Equation . Interior 1st Derivative, 6th Order.

Equation . LHS Boundary 1st Derivative, 6th Order.

Equation . RHS Boundary 1st Derivative, 6th Order.

### 2nd Derivative

Equation . Interior 2nd Derivative, 6th Order.

Equation . LHS Boundary 2nd Derivative, 6th Order.

Equation . RHS Boundary 2nd Derivative, 6th Order.

### 3rd Derivative

Equation . Interior 3rd Derivative, 6th Order.

Equation . LHS Boundary 3rd Derivative, 6th Order.

Equation . RHS Boundary 3rd Derivative, 6th Order.

### 4th Derivative

Equation . Interior 4th Derivative, 6th Order.

Equation . LHS Boundary 4th Derivative, 6th Order.

Equation . RHS Boundary 4th Derivative, 6th Order.

## Time Integration Method

As mentioned before, the RK-4 time integration method is used for explicit time stepping. The central equation for integration is Equation 2. There are four functions for one real and three virtual time steps, denoted by the values [6].

Equation . RK-4 Time Stepping Integration.

Equation . First Virtual Time Step.

Equation . Second Virtual Step.

Equation . Third Virtual Step.

### Alternative Time Integration Schemes

The RK-4 scheme is something of a hybrid scheme between implicit and explicit time integration. is the implicit time function, & are forward and backward Euler steps with a predictor-corrector combination, and is the explicit step. It is essentially a method to find “the best of both worlds” between explicit and implicit methods.

An alternative method is to back away from the RK-4 method and move schemes that are simply implicit or explicit. The solver comes with a simple Euler forward, an explicit method, time integration. This method is only first order accurate in time, and thus much data will be lost. Additionally, since RK-4 is a hybrid explicit scheme, there is reportedly a wider stability region than Euler forward. RK-4 can use a CFL up to a little under 3, whereas Euler forward can only go up to 1 [7].

Equation . Euler Forward Time Integration.

An alternative method is an implicit time stepping method, which should provide more stability. However, more time is needed to implement this in the code.

## Algorithm

### Initialization

The first step in the solve is to initialize the object that contains the data. i.e.:

\_\_init\_\_():

Initialization

### Linear System of Equations Setup

The next step is to set up a linear system of equations in the form Equation 6. Lower case variables represent arrays iterating, and . The upper case variables represent matrices to solve the system. In this case, is the sum of the various matrices representing the higher derivatives, thus .

Equation . The Linear Equation for KS Solve.

i.e:

A=numericalGradient(1,<stencil>)(sparse)

B\_diffusion=numericalGradient(2,<stencil>)(sparse)

B\_third=numericalGradient(3,<stencil>)(sparse)

B\_fourth=numericalGradient(4,<stencil>)(sparse)

B=B\_diffusion+B\_third+B\_fourth

D=identity(Nx x Nx)(sparse)

e=zeros(Nx)

### Boundary Condition Setup

In this study, there is a fourth derivative, thus four boundary conditions are required. The code then becomes:

Pull 0th Derivative BCs

Condition A, B, D, and e for these BCs

Pull 1st Derivative BCs

Condition A, B, D, and e for these BCs

Pull 2nd Derivative BCs

Condition A, B, D, and e for these BCs

Pull 3rd Derivative BCs

Condition A, B, D, and e for these BCs

Pull 4th Derivative BCs

Condition A, B, D, and e for these BCs

Note that in this study, the boundary conditions are zero for the value of at the edges, and zero gradient at the edges, thus the 2nd, 3rd, and 4th orders

### Time Stepping

Finally, to complete the solve, the solver will go through the time stepping process, thus the algorithm is:

for time in time domain:

v = ½\*u^2

f = A\*v + B\*u + e

if time step index < time integration order or time integration order==1:

phi = dt \* f + u

next u = D\phi

elif time integration order==4:

Rn = f

# First virtual time step

phi1 = (dt/2)\*Rn + u

u1 = D\phi1

v1 = ½\*u1^2

R1 = A\*v1 + B\*u1 + e

# Second virtual time step

phi2 = (dt/2)\*R1 + u

u2 = D\phi2

v2 = ½\*u2^2

R2 = A\*v2 + B\*u2 + e

# Third virtual time step

phi3 = dt \* R2 + u

u3 = D\phi3

v3 = ½\*u3^2

R3 = A\*v3 + B\*u3 + e

# RK4 time step

phi = u + (dt/6)\*(Rn + 2R1 + 2R2 + R3)

next u = D\phi

# Analysis

## Algorithm FLOPs Intensity

### Numerical Gradients

Each of the numerical gradient calculations contain the same number of floating point operations in the numerator, which comes to be 11. Then each order of derivative adds on floating point operation to the calculation, thus:

Equation . FLOP Count for a Spatial Gradient.

These spatial gradients feed into the linear algebra equation in Equation 18. The flux formulation for advection as 2 FLOPs per spatial point. This means that the advection is 14 FLOPs per spatial point. Then the diffusion and stiffness gradients form 13 and 14 plus 1 FLOPs per spatial point. Then adding the vector adds an additional FLOP per spatial point. Combining these to formulate the FLOP count for each time step:

Equation . FLOP Count for Each Time Spatial Point.

### Time Stepping

Each time step for the RK-4 method has two steps, calculate the RHS of the time integration, which is 3 FLOPs per spatial point, and then solve the LHS of the time integration, which we will assume is some LU decomposition method. It is an algorithm called COLAND [4], which is some modification of the LU method, actually a QR decomposition [5], for improved efficiency. For this analysis, we will assume the efficiency of a LU decomposition. One can note that Gaussian is technically twice as efficient but does not fit ALU parallelization well. Thus, each time step solve requires FLOPs to solve, totaling:

Equation . FLOP Count for Each Time Integration.

For the RK-4 method, there are four time steps, which get combined by an equation that uses 6+2+1 FLOPs per spatial point. Thus:

Equation . FLOP Count Total at Each Time Step.

Bringing all these values together, the total FLOP count becomes:

Equation . FLOP Count Total.

One can notice that for high spatial point counts, it is the matrix solver that drives computational intensity. Alternatively, for lower spatial point counts, the spatial gradients then drive the computational intensity.

## Convergence Analysis & Order of Accuracy

Overall, the KS solver shows acceptable order of accuracy for the analysis that is in this study. There are two methods that are used. The first is to take the exact gradients based on the initialized values, since there is an exact solution. The second is to make an extreme-fine mesh to find a quasi-exact solution.

### Exact Gradients

The exact solution for the gradients is known for a sine wave, which provides an easy reference point for diagnosing each individual term in the equation. The first and second derivatives show good agreement with the expected value, with order shown in Table 2. The third and fourth derivatives show some issues with order of accuracy, coming in at about 1.5 orders less than the expected value. If reduced to a theoretically fourth order solver, this remains consistent. This anomaly led to switching from a fourth to sixth order theoretically accurate solver. However, since they are still over fourth order accurate, this may be considered acceptable.

Table . Orders of Accuracy for Exact Derivative Comparison.

|  |  |
| --- | --- |
| Derivative | Order of Accuracy |
| 1st | 5.86 |
| 2nd | 5.93 |
| 3rd | 4.57 |
| 4th | 4.64 |

A graph of a line

Description automatically generatedA graph of a line

Description automatically generatedA graph of a line

Description automatically generatedA graph of a line

Description automatically generated

Figure . Convergence Plot for Initial Derivatives.

### Quasi-Exact Solution

A graph of a graph

Description automatically generated

Figure .

Shown in Figure 4, there is a clear difference between the numerical solutions. Taking the RMS error yields Figure 5. There is clearly a serious improvement after the medium case as the error difference between medium-fine is much less than coarse-medium. This results in a convergence value 4.25, which is acceptable, although not perfect, for this study. At some point, the author will likely have to iterate on the solver code to improve this.

A graph with a blue line

Description automatically generated

Figure .

# Results

## Regime Definition

It is apparent when studying the KS equation that different values for the coefficients induce different behaviors from the solution. Thus, defining the difference between various conditions is helpful to compare different cases between studies.

### Length Scale Definition

There is not an inherent length scale if the behavior in the domain is periodic. A periodic behavior implies potential across an infinite domain. Additionally, the period or wavenumber of the most prominent, or say coherent, motion in the domain will inevitably change with diffusion and stiffness effects. Thus, for a length scale, this study suggests two options. The first is a cell size, which is the size of the steps in the spatial domain. The second is a quantity called an integral length scale, described in Equation 24, which measures the statistical length that a motion will remain coherent. The coherency is measured by two-point correlation, which is the normalized convolution of a flow parameter.

Equation . Integral Length Scale.

Equation . Two-Point Correlation.

### Non-Dimensionalized Diffusion

The first way that the solution can vary is by the diffusion behavior present. For this, we use the classic Reynolds number with some modifications since this non-dimensional number is still considering the diffusion of momentum.

For the first Reynolds number is the cell Reynolds number, which essentially determines the how much the momentum behavior determines the solution versus the diffusion at a specific cell, thus in the localized mathematics.

Equation . Cell Reynolds Number.

The second Reynolds number is the coherency Reynolds number, which simply measures the strength of momentum against its diffusion for a given motion.

Equation . Coherency Reynolds Number.

These Reynolds numbers can be used for maximum values or some other measure. For example, a convective velocity can define the speed of travel of a certain waveform. However, the use of these in this context is not clear, and thus not discussed in the study.

### Non-Dimensionalized Stiffness

Non-dimensionalizing stiffness is unusual for fluid mechanics outside of the classic Mach number, which measures the velocity of some behavior against the velocity that pressure waves can propagate. Thus, for this application an alternative non-dimensional number will be used, formulated in Equation 28.

Equation . Non-Dimensional Stiffness.

is the dialation of the flow, which represents half the normal stress on a convecting fluid element. is the stiffness coefficient for the KS equation. is the integral length as described previously and is squared to give an effective area of coherency. Other lengths can be used, but the integral length seems the best one to use since the stiffness drives the response to a motion and the integral length describes the length the motion will be coherent over. Since the study is 1D in space, the formulation is:

Equation . Non-Dimensional Stiffness for 1D.

## First Case

To look at the KS equation solution, the first case is set up as Case 1 in Table 1. The initialized function is plotted in Figure 6.

Table . Data to Define Case 1.

|  |  |
| --- | --- |
| Parameter | Value |
| x-Domain |  |
| Time Domain |  |
| CFL | *0.1* |
| Initialized function |  |
| Signal to Noise Ratio | *0.05* |
|  | *-100e-6* |
|  | *1e-6* |
| BC, | *1,-1* |

A graph of a function

Description automatically generated

Figure . Initializated Function for Case 1.

A chart of a contour plot

Description automatically generated

Figure . Contour Plot Case 1.

The KS solution is in Figure 8. The solution inherently makes sense since the Burger’s equation behavior should dominate given the relatively low coefficients. After some experimentation, there is clearly a band for the different coefficients where the solution is stable, more on that in a bit. There is clearly some issue with the boundaries, but it seems not affect the main spatial domain.

Another way to look at the data is to understand the energy spectra, as in Figure 9, which breaks down the response relative to the size of its motion. This is, quite frankly, a more informative approach to this data. The sawtooth-wave behavior is apparent with the jagged descending ramp in the spectral space. There does appear to be some high frequency square wave shape, as illustrated by the sinc-function-like behavior at the tail end of the spectra. It is unclear if this is the noise added to the initialized signal, or an error the solution is building. After some experimentation, the magnitude of this shape is unaffected by the SNR. Overall, the decreasing energy content indicates that the diffusion is working properly as smaller motions become more dissipated.

Finally, we can look at the integral length scale, as illustrated in Figure 10. The integral length will increase with time, which makes sense that as the motions become larger with time as the behavior tends towards a steady state. This is more helpful as the Case is defined by the regime it measures.

A graph of energy spectrum

Description automatically generated

Figure . Energy Spectra for Case 1. Spectra averaged by indicated time interval.

A graph of a line

Description automatically generated

Figure . Integral Length Scale Along Time.

The first parameter that defines the regime is the coherent Reynolds number, the bounds shown in Figure 11. This value varies over about one order of magnitude throughout time. As with integral length, this appears to reach a steady state with the maximum at about 2e4 and minimum at about 3.

A graph with numbers and lines

Description automatically generated

Figure . Comparison of ReLu Throughout Time.

To understand the stiffness in the KS equation, the nondimensional stiffness helps describe the resistance to stress against the stress applied to the fluid being simulated. Seen in Figure 12, there is clearly a decreasing resistance relative to the actual stress on the fluid.

Finally, to look at changing the conditions, the stability of the solve needs to be characterized by the ability to represent physics in the discretized domain. Since stiffness can simply transport through the cells, the diffusion is the most interesting aspect of stability. The Reynolds number of a cell, shown in Figure 13, shows that for the fastest motions, the momentum dominates the behavior in the cell. Alternatively, the slowest motions have dissipation dominate the physics of the cell. It appears that there is a good balance between the two physics.

A graph of a blue line

Description automatically generated

Figure . Comparison of Nondimensional Stiffness Througout Time.

A graph of a cell

Description automatically generated

Figure . Comparison of Cell Reynolds Number Throughout Time.

## Second Case

This project has a requested set of conditions to test the KS solver with, shown in Table 4. The coefficients had to be normalized with the cell size, otherwise the time integration would be so dominated by the diffusion and stiffness that the solve becomes quickly unstable. It is potentially possible to simulate flow dominated by these behaviors, but a different time integration method should be used. The explicit Eulerian time stepping seems unable to initialize the solution at a reasonable point. In either case, the CFL is ½, which should push the stability of the time integration. Finally, the boundary conditions are changed to be more representative of the expected simulation. The boundary has floating velocity with a first derivative set to zero, which will force a reflection of waves that approach the wall. Finally, the fourth derivative is set to a very high value, which will help enforce this simulated condition. Case 2 is initialized as in Figure 14, which is set up to simulate two opposing waves traveling towards each other.

Table . Data to Define Case 2.

|  |  |
| --- | --- |
| Parameter | Value |
| x-Domain |  |
| Time Domain |  |
| CFL | *0.5* |
| Initialized function |  |
| Signal to Noise Ratio | *0.05* |
|  |  |
|  |  |
| BC, | *0,0* |
| BC, | *1,-1* |

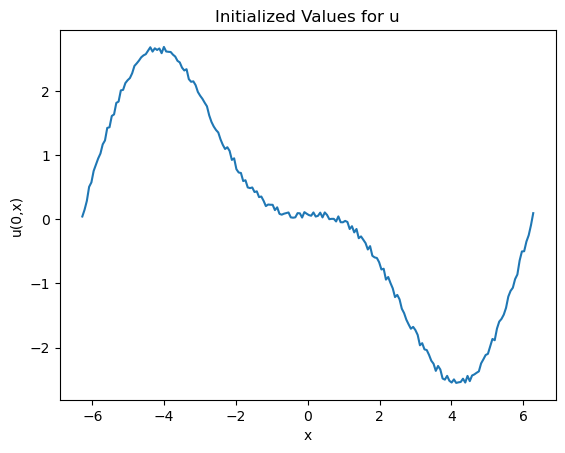


Figure . Initialized Function for Case 2.

The results of this case are in Figure 15 and are interesting as there are multiple regimes. Early on is the traveling wave phase, as the waves are coalescing. Shortly before , the waves form a shock discontinuity on both ends. Around the shocks are Gibbs phenomenon waves that are amplified by the negative dissipation. At some point, the shocks meet in the middle and keep traveling slightly towards the positive x-direction.

Looking at the data in the spectral data in Figure 16 reveals much about what is occurring in the simulation. The initial wave dominates the early spectra until the shocks coalesce. Once the shocks are formed, there is a dissipation slope that corresponds to a 1D dissipation. When the shocks reach steady state, then a steep drop-off forms at the end of the spectra.

The integral length scale throughout time, shown in Figure 17, also contains some interesting behavior. As the shocks are forming, the integral length goes to infinity and then comes back down asymptotically in what could be considered the supersonic regime. This behavior is a departure from Case 1 where the integral length scale continued to increase with time.

A chart of a contour plot

Description automatically generated

Figure . Contour Plot of Case 2.

A graph of energy spectrum

Description automatically generated

Figure . Energy Spectra for Case 2.

A graph of a function

Description automatically generated

Figure . Integral Length Scale Along Time for Case 2.

A graph of a cell

Description automatically generated

Figure . Cell Reynolds Number Throughout Time for Case 2.

A graph of a graph

Description automatically generated

Figure . Comparison of ReLu Throughout Time for Case 2.

A graph of a graph showing a blue and orange line

Description automatically generated

Figure . Comparison of Nondimensional Stiffness Throughout Time.

Looking at cell Reynolds number in Figure 18, there is an interesting behavior in that the early stages of the case have a mix of momentum and dissipation dominated behaviors, but then lose dissipation dominated behaviors once the shocks coalesce. This is also present in Figure 19, showing the integral length Reynolds number, although this is interrupted by the discontinuity where the shocks are formed. It is interesting that the maximum integral length Reynolds number stays roughly the same.

Finally, the non-dimensional stiffness, shown in Figure 20, reveals some interesting behaviors as the shocks form and then combine. It appears that the ratio of stiffness to stress ratio goes to zero when the shocks form and then rises to settle at a value for when the shocks combine. This indicates that the modeling of the stiffness affects the other properties of the flow, including the diffusion.

### Time Integration Comparison

It is helpful to directly compare different time integration schemes. The two that are currently programmed in is the Euler explicit method, results in Figure 21, and the RK-4 time integration, results in Figure 22. For this case, the CFL was turned down to 0.1. It is clear that the Euler integration contains content of the noise that is note present in the RK-4 results. It is not clear which is correct, as the Euler integration fluctuations make more intuitive sense that there is not a dissipation mechanism for these fluctuations. Additionally, the additional terms in RK-4 may produce a dissipation numerically that is not present physically. Alternatively, the Euler integration may not model wavenumbers that lead to dissipation. To determine which is correct, a comparison to a lower order, say 1st or 2nd order, implicit scheme would show the contribution of implicit solutions against the order of the integration.

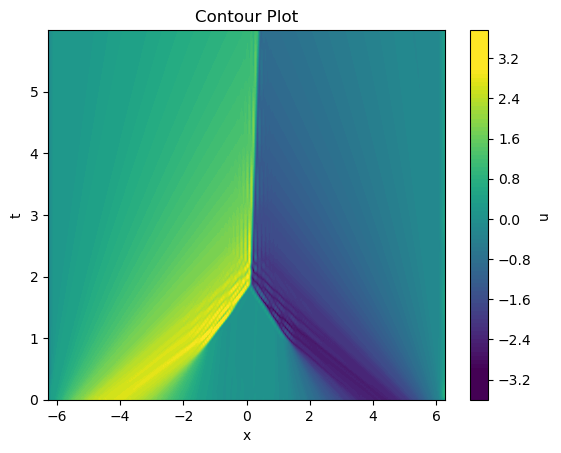


Figure . Contour Plot of Case 2 with Euler Explicit Time Integration.

A chart of a contour plot

Description automatically generated

Figure . Contour Plot for Case 2 with RK-4 Time Integration.

# Works Cited

1. Sivashinsky, G. I. (1977*). Nonlinear analysis of hydrodynamic instability of laminar flames – I. Derivation of basic equations*. Acta Astronautica, Vol 4, pg 1177-1206.
2. Kuramoto, Y. (1978). *Diffusion-Induced Chaos in Reaction Systems.* Supplement of the Progress of Theoretical Physics, No 64.
3. Kalogirou, A., Keaveny, E. E., and Papageorgiou, D. T. (2015). *An in-depth numerical study of the two-dimensional Kuramoto-Shivashinsky equation.* The Royal Society Publishing.
4. SciPY:Sparse. (2024). API Reference: scipy.sparse.linalg.spsolve. Available at: <https://docs.scipy.org/doc/scipy/reference/generated/scipy.sparse.linalg.spsolve.html#r1c4f91118c9e-1>
5. Davis, T. A., Gilbert, J. R., Larimore, S. I., and Ng, E. G. (2000). *Algorithm 8xx: COLAMD, a column approximate minimum degree ordering algorithm.* University of Florida: Technical Report TR-00-006.
6. Anderson, D. A., Tannehill, J. C., Pletcher, R. H., Munipalli, R., and Shankar, V. (2021). Computational Fluid Mechanics and Heat Transfer. CRC Press, 4th Edition
7. Frank, J. (2008). *Chapter 10: Stability of Runge-Kutta Methods.* Lecture from Numerical Modelling of Dynamical Systems. Hosted by Utrecht University.