

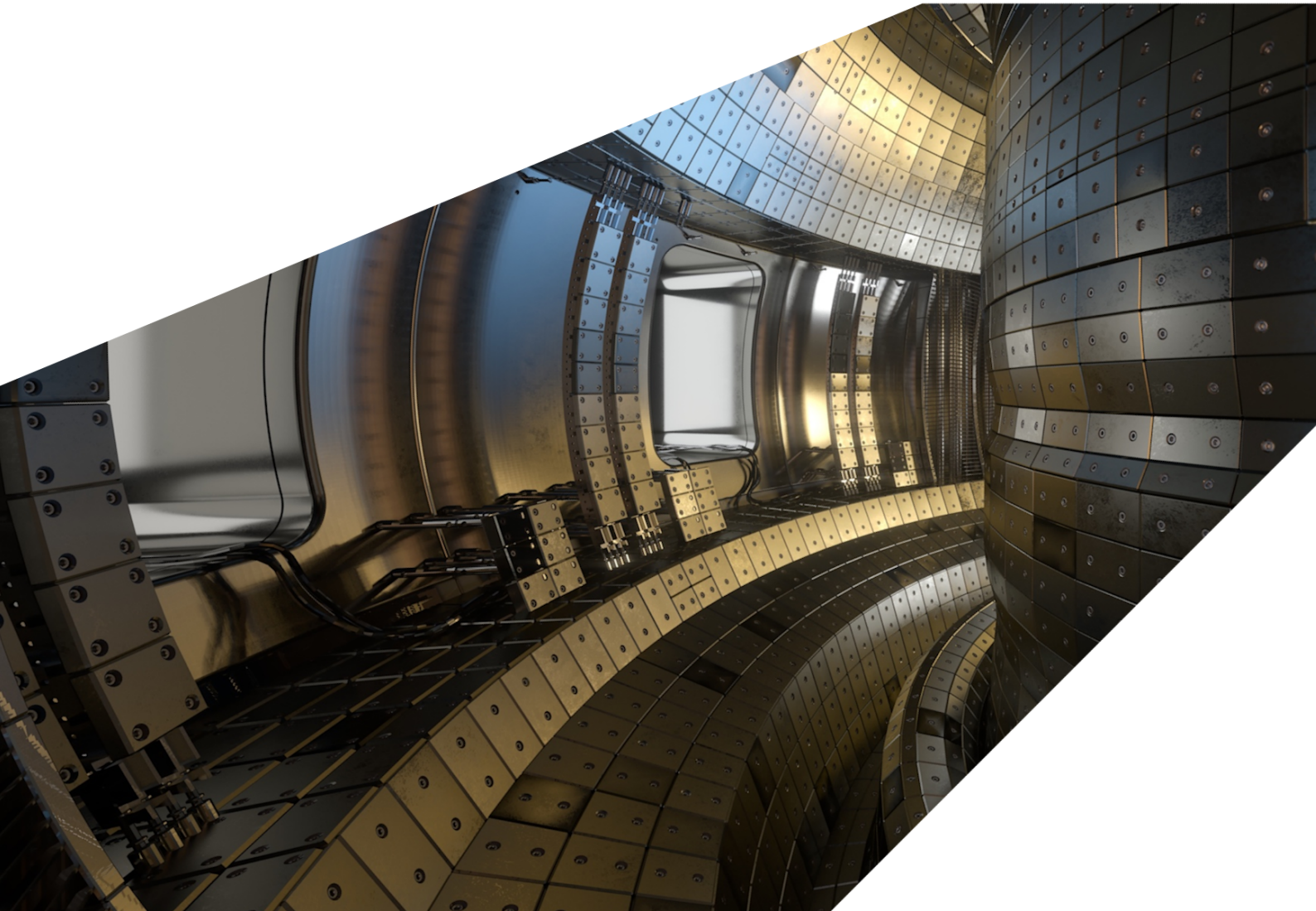
ExCALIBUR

Support High-dimensional Procurement

M4.1 Version 1.00

Abstract

The report describes work for ExCALIBUR project NEPTUNE at Milestone M4.1. It collates technical material to prepare call for high-dimensional procurement. In particular, it contains a description of a simple $1d1v$ kinetic solver implemented within the *Nektar++* spectral / *hp* element framework and the application of this solver to the textbook problem of the two-stream instability, work which is intended to prepare the ground for anticipated upgrades to *Nektar++* and also to deepen UKAEA knowledge of the workings of the *Nektar++* code.



UKAEA REFERENCE AND APPROVAL SHEET

	Client Reference:		
	UKAEA Reference:	CD/EXCALIBUR-FMS/0066	
	Issue:	1.00	
	Date:	18 March 2022	
Project Name: ExCALIBUR Fusion Modelling System			
	Name and Department	Signature	Date
Prepared By:	Ed Threlfall	N/A	18 March 2022
	Will Saunders	N/A	18 March 2022
	BD		
Reviewed By:	Wayne Arter	<i>W. Arter</i>	18 March 2022
	Project Technical Lead		

1 Introduction

Future extensions of the spectral / hp method used in the *Nektar++* framework for the fusion use case will need to target kinetic models ie. solutions for matter that is not accurately treatable by fluid dynamics as in eg. [1]). The extension of the framework to handle such models (physically, Boltzmann-type equations) in some generality is the subject of NEPTUNE grant T/AW084/21. In anticipation of the latter, and having collateral benefits in terms of learning *Nektar++*, a simple proxyapp to handle a $1d1v$ case has been prototyped, as described herein. This is applied to a classic kinetic problem in plasma physics: the two-stream instability in a system of collisionless charged matter. Analytic results for instability linear growth rates are seen to be replicated by this code (giving a non-trivial benchmark that can be applied immediately to the anticipated outputs of the aforementioned grant). In addition, the nonlinear evolution of the system is demonstrated. By virtue of the implementation of a model incorporating velocity-space effects, this report is relevant to Work Package FM-WP2 - Plasma Multiphysics Model.

2 A spectral/ hp Vlasov-Poisson solver in *Nektar++*

2.1 Introduction

This work anticipates the outputs of grant T/AW084/21, which are explicitly to provide a framework for solving continuum kinetic problems within *Nektar++* and also a proxyapp capable of solving $1d1v$ and 1D-3V systems. Briefly, the plan involves adding a new library module `TensorRegions` which will involve the inclusion of a separate computational mesh for handling the velocity components of phase space. The results of this section, which are obtained with a simple modification to the existing *Nektar++* code, will provide a ready-made benchmark for the expected proxyapp outputs of the aforementioned grant. This work is, to the authors' knowledge, the first application of the *Nektar++* framework to a kinetic problem.

2.2 Theoretical background

The kinetic equation treated is the $1d1v$ Vlasov-Poisson system, in the phase-space coordinatized by (x, v) ,

$$\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} + E \frac{\partial f}{\partial v} = 0. \quad (1)$$

Here $E = -\frac{\partial \phi}{\partial x}$ is the electric field, depending only on x (not v), with the electrostatic potential ϕ found from the Poisson equation

$$\frac{\partial^2 \phi}{\partial x^2} = \omega_P^2 \left(\int f \frac{dv}{v_0} - 1 \right), \quad (2)$$

where the plasma frequency ω_P is the only physical parameter in the problem.

The subtracted term adds a uniform (in x and v) neutralizing background; v_0 is chosen so as to give global neutrality. The system may be interpreted as electrons moving in a background of much heavier, i.e. effectively stationary, ions. More broadly, this model is a statistical, continuum representation of charged matter in the non-relativistic limit - the particle dynamics are Newtonian and electrodynamics reduces to electrostatics. It is possible to include additional physics by inserting collision terms on the right-hand-side but that is left for future work, as classic and nontrivial results can be obtained in the collisionless limit. The definitions are as in [2]; note the electric field in the equations is rescaled by a factor of $\frac{e}{m}$ relative to the physical value. It is worth noting that the matter density function comes with the interpretation of something that is positive; $f(x, v) \geq 0$.

2.3 *Nektar++* implementation

The implementation is made possible within the existing *Nektar++* framework by realizing that the system 1 has the form of a two-dimensional advection problem.

The solver derives from the class `AdvectionSystem` and the advection velocity is simply $v_{adv} = (v, E)$. In fact the new solver was based on `nektar-driftwave` provided to the NEPTUNE community by the developers of *Nektar* and available at [3]. Most of the work involves writing the method `ExplicitTimeInt` (which in the implementation discussed here includes also the solution of the Poisson problem). The advection part of the problem is solved using a discontinuous Galerkin formulation with an upwind numerical flux.

The Poisson problem is solved using the API function `HelmSolve` (with zero Helmholtz constant); note that a continuous field is always used for the electric potential, as is the case in `nektar-driftwave`. During a prototyping phase, the Poisson problem was solved using a simple finite difference solver before being upgraded to the full spectral/*hp* method. Note that having all the fields in the proper spectral/*hp* form enables use of *Nektar++* API functionality to provide diagnostics of e.g. electrostatic field energy and matter kinetic energy using the `Integral` method to perform numerical quadrature accurate to the spectral order of the code.

The only non-API functionality that was necessary is for the code to integrate f over one dimension (v) in order to obtain the spatial charge density (*Nektar++* does not currently provide this capability - [4] - it can integrate along element boundaries but not along a line cutting through an element). This was done using Gauss-Legendre quadrature (see, for example, [5]) and involved use of the *Nektar++* API functions to obtain field values at the quadrature points. Given n_G such points over a (one-dimensional) element, polynomials up to order $2n_G - 1$ are integrated exactly; it is thus simple to ensure that the integrals are exact up to a given finite element order. The Gaussian integrals are done on a per-element basis so there is no question of inaccuracies due to attempting to integrate over the field discontinuities of the discontinuous Galerkin method; also, since the Gauss quadrature points lie strictly within the elements, there is no issue with the ambiguity associated to non-unique field values at element boundaries. Collectively, the integrals form a one-dimensional array indexed by the node positions along the x -direction; these values are converted to a two-dimensional array over the entire (x, v) domain in order to provide an appropriate right-hand-side for the `HelmSolve` function.

The spatial dimension is taken as periodic in the range $[-1, 1]$ with a periodic boundary condition. The velocity space is taken to be in the range $[-4, 4]$ with homogeneous Dirichlet boundary conditions on f and homogeneous Neumann conditions on ϕ . Note initial tests used a periodic boundary condition in v , which worked due to the density vanishing at that boundary, but which caused a crash when used with MPI (possibly a bug).

The solution was made compatible with the MPI support in *Nektar++* by using the shared array class `Nektar::Array`. The possibility of more than one thread contributing to the v -integral at a given x -point was countered by performing a ‘minimum’ reduction operation after the evaluation of the set of v -integrals. In addition, vector arithmetic API functions for operating on the shared arrays were utilized, e.g. `Vmath::Vvtvp` which is ‘vector times vector plus vector’. Note that the API contains MPI reduction commands for summation, maximum, and minimum (e.g. the latter is `AllReduce(Array, LibUtilities::ReduceMax)`); this knowledge means that the temperature-maximum filter in [1] can now be made MPI-compatible.

There are a number of properties about this implementation that are non-optimal from the point of view of efficiency: the Poisson problem is solved in two dimensions when it only depends on one of those dimensions; the electric field energy integral is done over two dimensions when again the

electric field only varies in one dimension. Note that these (and other) performance concerns in *Nektar++* will be addressed by the outputs of Call T/AW084/21.

The (entirely C++) code was written and debugged using the Microsoft Visual Studio 2019 Community Edition IDE [6] running under Microsoft Windows (note that the author's experience with the *Nektar++* community suggests that Windows users are in something of a minority). It is intended that the new solver will be made available to the NEPTUNE community soon, though there are areas of incompleteness e.g. dependencies on the particular mesh (the problems treated in this section all used the same regular computational mesh, which comprised 1600 squares of side 0.1).

2.4 Two-stream instability

A classic test case for the system 1 is the case of two counter-propagating beams of like-charged particles (in a static uniform neutralizing background) interacting via electrostatic forces only. The initial condition is spatially-homogeneous and has zero electrostatic field - such a configuration is trivially seen to be a time-independent solution of Eq.1 - and the question is the stability and subsequent time-evolution of the system.

This problem has a long history of study. The paper [2] is an early (1967) numerical investigation using the contour dynamics technique to track the time-evolution of boundaries between $f = 1$ and $f = 0$ regions of phase space, thereby reducing the dimensionality of the problem by one and making it tractable on '60s computers; the limitation is that only values of 1, 0 are allowed for f . The latter is rather complementary to the spectral/ hp method, which, without special modifications, is not expected to work well for non-smooth functions (this is a general feature of spectral methods; basically, the efficacy of the method results from the presupposition that the solution is smooth, and if this is not the case then numerical problems arise via Gibbs / Runge phenomena). A basic synopsis of the results in [2] is that the initial condition is not stable and the system evolves in phase space to a distinctive form, for which a subjective justification is now given.

In view of the spatial periodicity of the system, it seems reasonable to expand the electric field in a Fourier series. In fact, some insight into the steady-state solution can be found by keeping only the leading term i.e. $E(x) = E_0 \cos(\pi x)$. Changing variables, one finds that steady solutions are given by functions of the local *particle* energy density, $\mathcal{E} \equiv E_0 \frac{\sin(\pi x)}{\pi} + \frac{1}{2}v^2$. Plotting contours of constant \mathcal{E} (Fig.1) is certainly suggestive of the numerical solutions exhibited in the sequel. Note the exact evolution of f is determined by the initial data and the value of ω_P^2 - some setups give more than two of the equilibrium points that are now described. Note the two-equilibrium cosine electric field presented here gives a mathematical problem equivalent to a rigid pendulum in a uniform gravitational field with the solution given in terms of an elliptic integral; the stable equilibrium point corresponds to the pendulum bob at the minimum height and the unstable point has the bob at the maximum. The two equilibrium points - zeros of the electric field - correspond to a phase-space attractor and a repeller (near these respective points, the governing equation is locally $\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} \mp x \frac{\partial f}{\partial t} = 0$, with the negative sign case representing simple harmonic motion).

It will be seen that these features are evident in the numerical solutions.

Initial data: the system was initialized with two spatially-uniform counter-propagating beams that

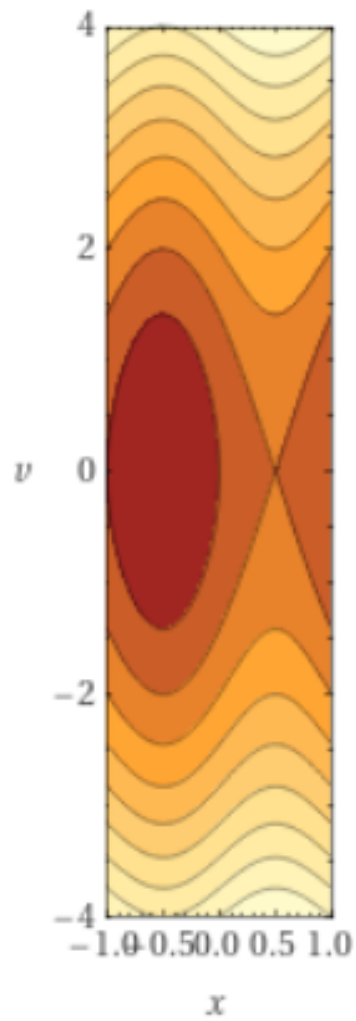


Figure 1: Contours of constant energy assuming $E(x) \propto \cos(\pi x)$, for one period of solution. Figure generated by [7]. In terms of the phase-space attractor / repeller discussed in the main text, there is an attractive point on the x -axis at $x = -0.5$ and a repeller at $x = 0.5$.

were both Gaussian in velocity space: the function used was

$$\frac{1}{\sqrt{8\pi\sigma^2}} \left(e^{-\frac{(v-v_0)^2}{2\sigma^2}} + e^{-\frac{(v+v_0)^2}{2\sigma^2}} \right), \quad (3)$$

with $\sigma = 0.2$ chosen so the initial data is fairly smooth (recall it is expected that delta or step function initial data will not work well with the spectral / hp element method) and $v_0 = 1$. This function is normalized to unity when integrated over v at any given x (by coincidence, it also has maxima which are very close, but not equal to unity). Physically, σ^2 may be taken as the temperature for each of the non-interacting beams (and note the total energy of the two-beam configuration is given by $v_0^2 + \sigma^2$). A small x -dependent initial perturbation was applied to the initial data Eq.3 in the form of the multiplicative factor $1 + 0.01 \sin \pi x$ in order to seed the instability.

Note that the periodic boundary conditions mean that the Poisson problem is not well-posed unless the total charge within one period is zero (equivalently, the mean electrostatic field is zero).

For a modern study of this system see [8].

2.5 Numerical results for two-stream instability

The numerical performance of the solver was tested by running simulations on a 20×80 grid of square elements with an element size of 0.1. The level of computational accuracy was varied by using different global values for the number of modes (per dimension) per element - varied from 2 (equivalent to using linear polynomials within each element) to 7. A standard four-stage fourth-order Runge-Kutta (`ClassicalRungeKutta4` in `Nektar++`) was used to perform explicit time-stepping.

It is seen that the solution represented by the initial data Eq.3 is not stable, and that the time-evolution leads to the formation of a distinctive phase-space pattern (Fig.5) similar to that seen in [2]. Note it is indicated in that paper that the solution depends on the externally-applied periodic boundary condition. The precise form of the solution is dependent on the value of ω_P and the form of the initial data - an exposition of this is given in the following subsection, for the linear regime.

Conservation of particle number - the amount of matter is quantified by $\int f(x, v) dx dv$. It was found that, for all $p > 1$, the simulation conserved particle number to 2×10^{-13} (relative to unity) over a period of 40 time units.

Positivity of $f(x, v)$ - since this function is essentially a probability density, negative values are unphysical (as well as liable to give difficulty when evaluating entropy, which involves forming $\ln f$). Improvements can potentially be made using the techniques outlined in [9]. The table 1 gives the magnitude of the most negative value at the end of a 40 time unit simulation - note that these do not seem to reduce as the element order p is increased. The figure 4 shows, however, that the negative values are confined to relatively small regions of phase space.

Energy conservation - energy conservation is exhibited, to a degree, but there is currently a growth in the total energy with time (Fig.2). The expected scenario was a slow decay of the total energy (because upwind DG schemes are known to be slightly dissipative - indeed, the lower- p simulations shown slightly suppressed energy growth, presumably for this reason). This is being

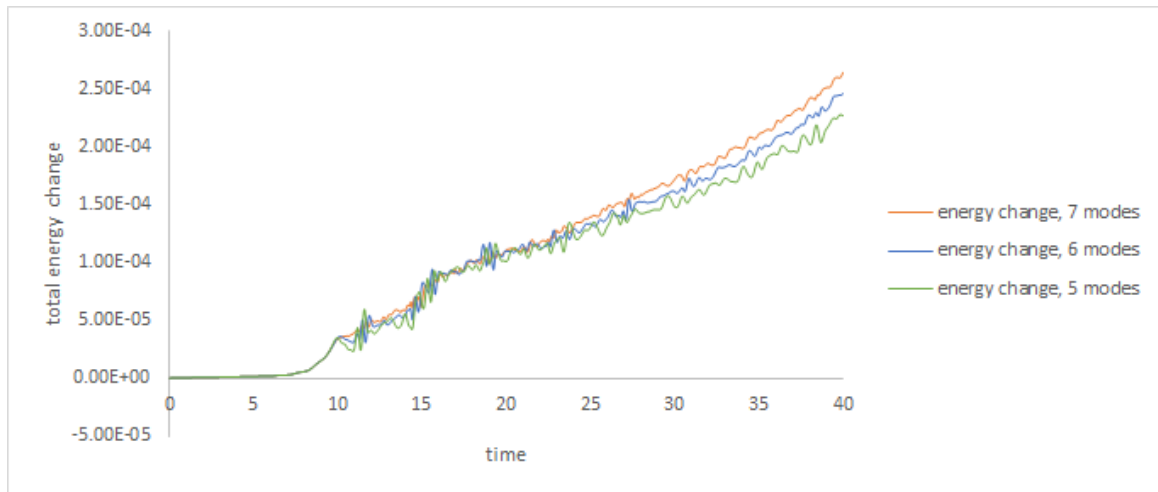


Figure 2: Time-evolution of the total energy change for 5, 6, 7-mode solutions (based on a total energy of approx. unity).

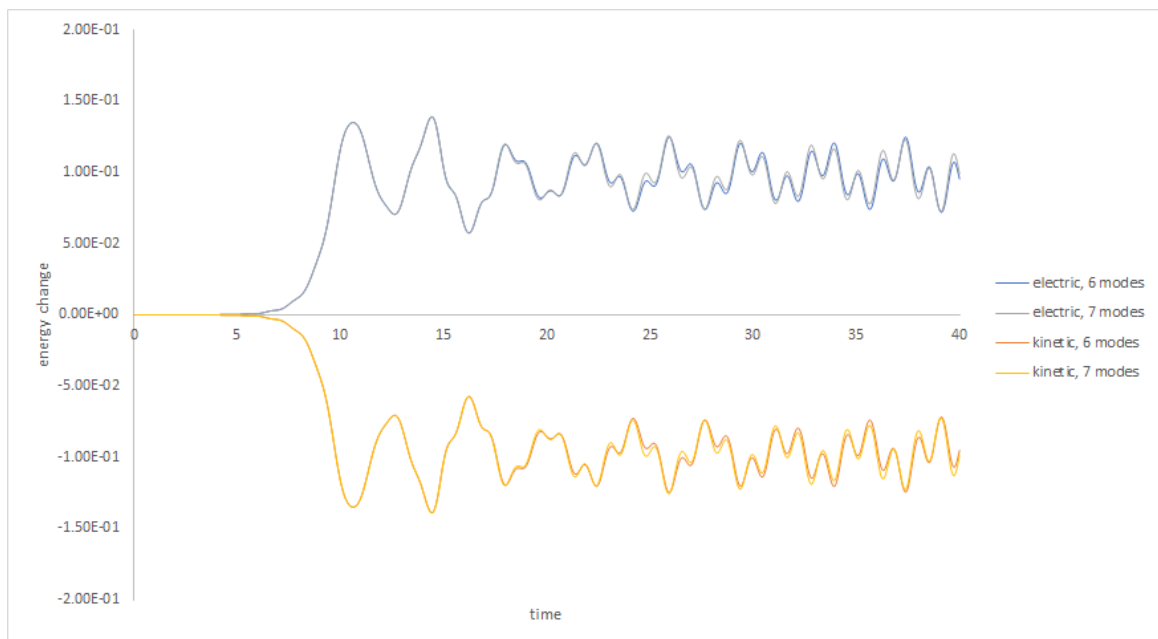


Figure 3: Change in electric and kinetic energy from start of simulation; data shown for 6-mode and 7-mode solutions.

Element order p	Magnitude of most negative value $f(x, v)$
1	1.1×10^{-1}
2	2.0×10^{-3}
3	2.4×10^{-3}
4	4.1×10^{-3}
5	5.2×10^{-3}
7	8.1×10^{-3}

Table 1: Table of maximum negative values of $f(x, v)$ from *Nektar++* Vlasov-Poisson solver.

investigated at the time of writing. Note that the energy conservation problem seems to arise from an imbalance between the kinetic energy lost and the electric field energy gained during the time stepping: the latter is consistently larger than the former.

2.6 Dispersion relations and linear instability growth rates

Linear dispersion relations can be calculated for the growth rates of harmonic instabilities over initial data that is uniform in position space and has a specified distribution in velocity space. This is based on a linearized theory that can be found in e.g. [10]. Note all initial data here are normalized to a total of unity (i.e. each beam is normalized to one half). The linear dispersion relation for a single species can be expressed as

$$1 = \omega_P^2 \int_{-\infty}^{\infty} \frac{f(v) dv}{(\omega - kv)^2}. \quad (4)$$

The simplest initial data case is two delta functions in v , for which the dispersion relation is $x^2 = \frac{1}{2} (1 + 2u^2 \pm \sqrt{1 + 8u^2})$ for $x \equiv \frac{\omega}{\omega_P}$, $u \equiv \frac{kv_0}{\omega_P}$.

A second example echoes [2] by using a background of two counterpropagating Heaviside functions in velocity space, normalized to unity in total, centred on v_0 and extending to $\pm\sigma$. The dispersion relation for this is $x^2 = \frac{1}{2} (1 + 2u^2 + 2s^2 \pm \sqrt{16s^2u^2 + 8u^2 + 1})$, with the delta function case recovered if $s = 0$.

The above simple cases are not directly useful as the spectral / hp implementation is not expected to perform well with non-smooth initial data. It is, however, possible to obtain an analytic dispersion relation for the initial data in Eq.3, as follows.

The NRL Plasma Formulary [11] gives

$$\frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \frac{dt e^{-t^2}}{t - x} = e^{-x^2} \left(i\sqrt{\pi} - 2 \int_0^x dt e^{-t^2} \right). \quad (5)$$

An expedient (non-rigorous, though it will be seen to give the correct answer) path to a closed-form solution involves differentiating Eq.5, giving

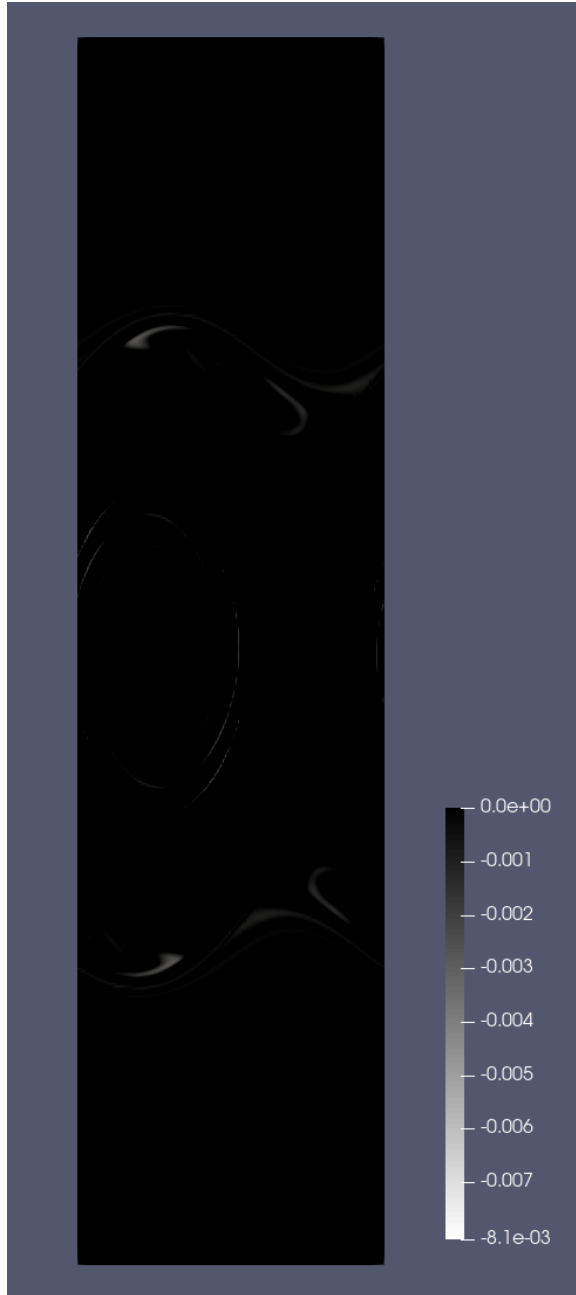


Figure 4: Negative values of $f(x, v)$ in the 7-mode solution.

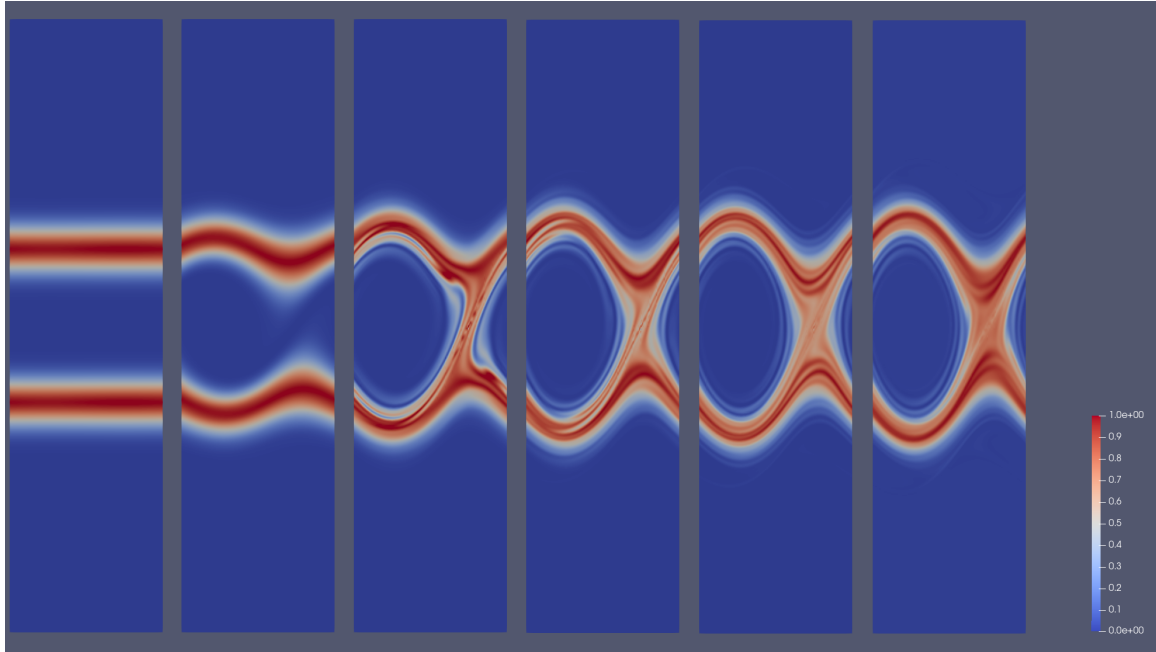


Figure 5: Time-evolution of the bi-Gaussian initial data into a state showing an phase-space attractor and a repeller (here $\omega_p^2 = 10$). The simulation time interval between successive plots is 8.0 units. Compare Fig.1. Note the vertical axis is v and the horizontal x .

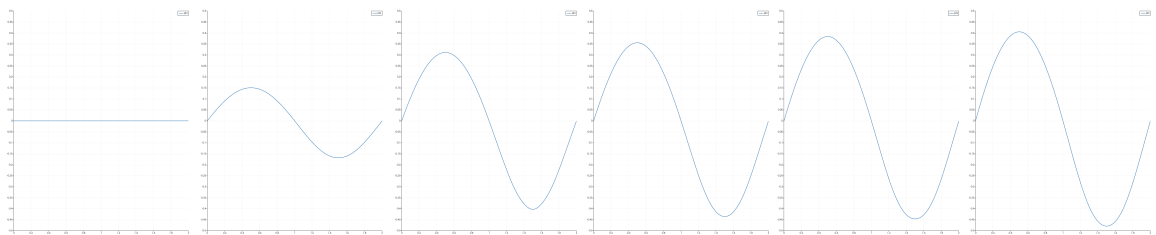


Figure 6: Time-evolution of the electric potential plotted as a function of x from the simulations in Fig.5 - plots correspond to the same intervals. Note the closeness of the solution to a simple harmonic wave.

$$\frac{d}{dx} \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \frac{dt e^{-t^2}}{t-x} = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \frac{dt e^{-t^2}}{(t-x)^2} = -2 \left(1 + \sqrt{\pi} x e^{-x^2} (i - \operatorname{erfi}(x)) \right) \quad (6)$$

wh. $\operatorname{erfi}(x) \equiv \frac{2}{\sqrt{\pi}} \int_{-\infty}^x dt e^{t^2}$.

Inserting Eq.3 into Eq.4 leads to quadratures of the above form and straightforwardly to the dispersion relation

$$1 = -\frac{\omega_P^2}{2\sigma^2 k^2} \left(2 - \sqrt{\pi} A e^{-A^2} (\operatorname{erfi}(A) - i) - \sqrt{\pi} A' e^{-A'^2} (\operatorname{erfi}(A') - i) \right) \quad (7)$$

with $A \equiv \frac{\omega - kv_0}{\sqrt{2\sigma k}}$, $A' \equiv \frac{\omega + kv_0}{\sqrt{2\sigma k}}$.

Note that the delta-function dispersion relation can be obtained from this in the limit $\sigma \rightarrow 0$ and using $\operatorname{erfi}(x) \sim \frac{e^{x^2}}{\sqrt{\pi}} \left(\frac{1}{x} + \frac{1}{2x^3} \right) - i$ (keep the two leading terms and neglect the i as the exponential dominates it).

This form of the dispersion relation does not seem invertible by analytic means. In practice, roots are straightforwardly obtained by Newton's method; it is found in the cases examined that the roots lie on the imaginary axis i.e. the complex frequencies have zero real part. The open-source library *Faddeeva* [12] was used to compute these functions (limited tests of this library were done e.g. comparing outputs with those calculated via [7]). Note that it is computationally better to use the Dawson function $D(z) \equiv \frac{\sqrt{\pi}}{2} e^{-z^2} \operatorname{erfi}(z)$ than directly use the imaginary error function $\operatorname{erfi}(z)$.

Numerical results to compare with this theory were obtained using a small initial perturbation amplitude, $1 + 10^{-7} \sin n\pi x$ (n is the wavenumber of the perturbation), chosen in order to prolong the duration of the linear regime. See Fig.8 for an illustration. Simulations were run long enough that a linear growth region was seen to be established beyond the initial transient and the growth rates were obtained by simple gradient measurement. The growth rates were assessed from the integral of the electric field energy, for which the growth rate is twice the linear mode growth rate on account of bilinearity. These simulations were found to be sufficiently accurate using only four modes and the agreement between theory and numerics is good (Fig.7), with the instability type concurring with the wavenumber used in the initial data perturbation.

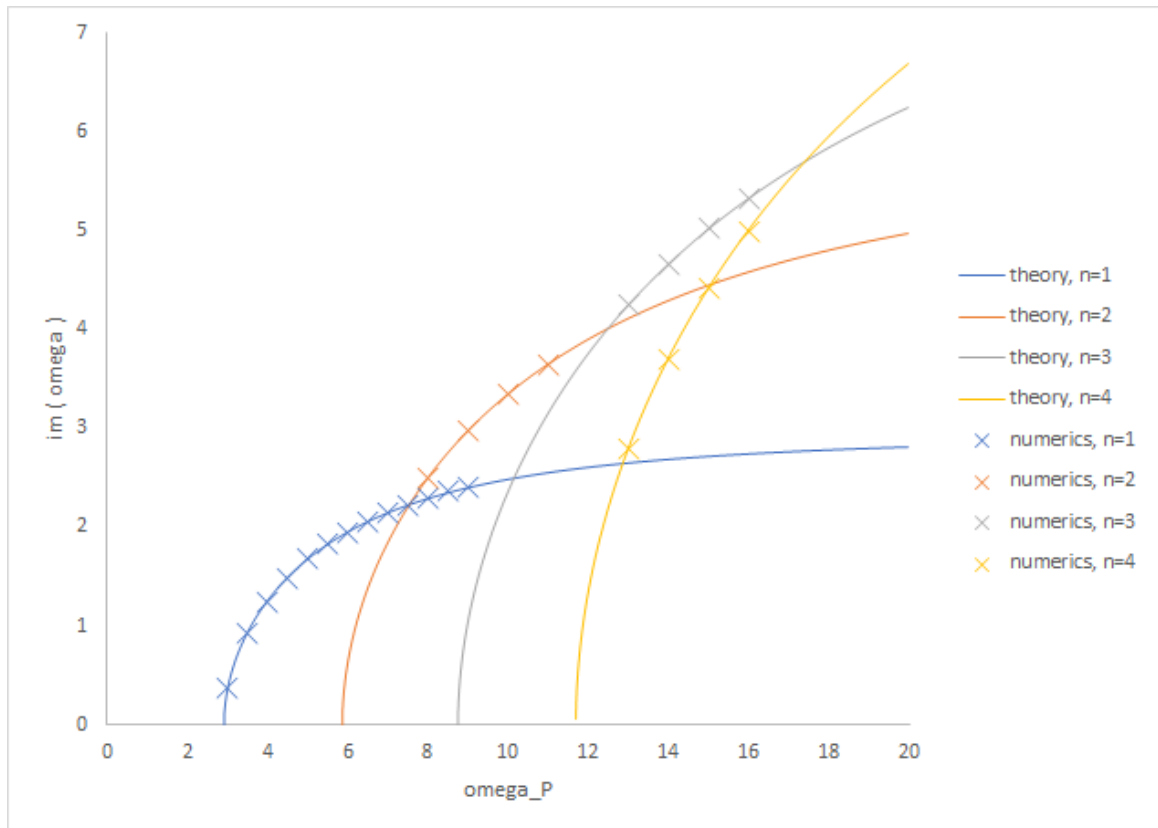


Figure 7: Theoretical predictions for the growth rates of modes $n = 1 - 4$ (spatial wavenumber $k = n\pi$) as a function of the plasma frequency ω_P . Overlain are numerical results from the *Nektar++* implementation, showing good agreement.

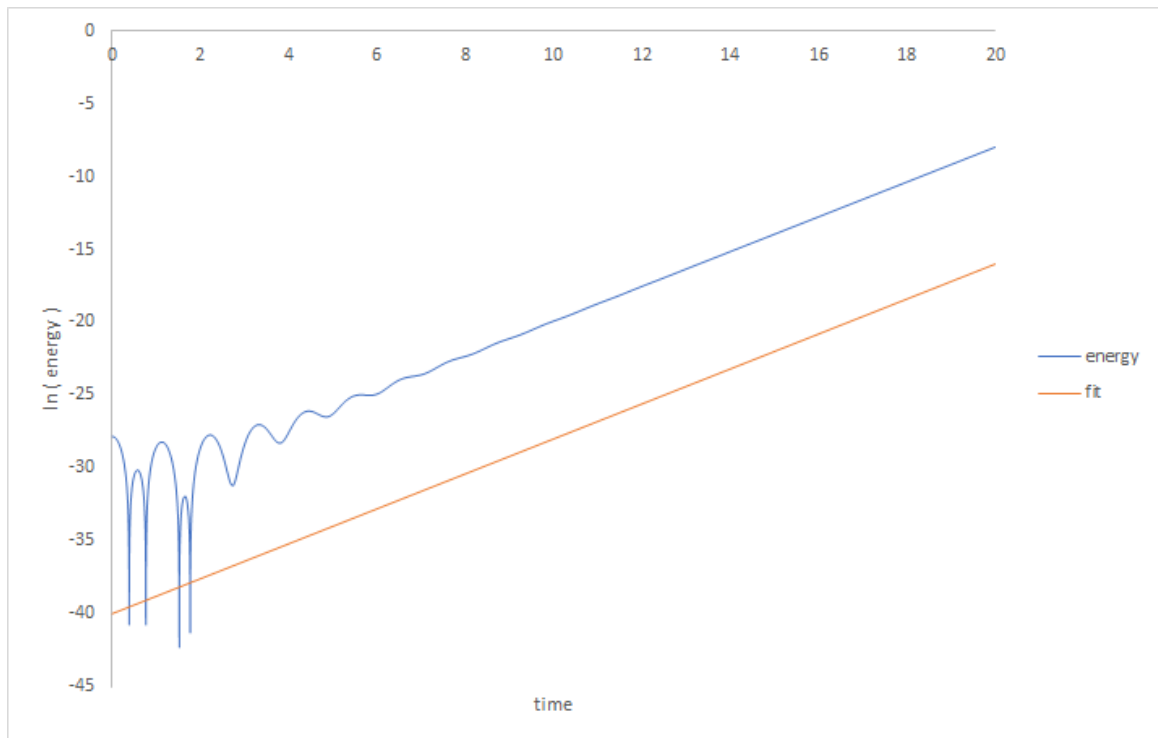


Figure 8: Linear growth of the electric field energy for $\omega_p^2 = 10$ and the bi-Gaussian initial condition Eq.3. The linear growth saturates at later times when the nonlinear dynamics dominates it.

3 Summary

This report has provided further numerical results using the *Nektar++* framework, in which an initial attempt at a *1d1v* kinetic solver has been implemented and tested. Qualitative agreement with the nonlinear dynamics of the two-stream instability in the literature was shown. More quantitative results were given for the growth rates of linear wave instabilities for the same system, which exhibited good agreement with theory. The study justifies the altering of emphasis from particles to finite elements in the high-dimensional procurement.

The author would like to thank the developers of *Nektar++* for useful discussions and Dr James Cook and Dr Joseph Parker of UKAEA for helpful discussions regarding kinetic methods.

Acknowledgement

The support of the UK Meteorological Office and Strategic Priorities Fund is acknowledged.

References

- [1] E. Threlfall and W. Arter. Finite Element Models: Complementary Activities I. Technical Report CD/EXCALIBUR-FMS/0051-M6.1, UKAEA, 2021. https://github.com/ExCALIBUR-NEPTUNE/Documents/blob/main/reports/ukaea_reports/CD-EXCALIBUR-FMS0051-M6.1.pdf.
- [2] K.V. Roberts and H.L. Berk. Nonlinear evolution of a two-stream instability. *Phys. Rev. Lett.* Vol. 16 No. 9, August 1967, pages 297–300, 1967.
- [3] Nektar++ solver for Hasegawa-Wakatani equations. <https://github.com/ExCALIBUR-NEPTUNE/nektar-driftwave>, 2021. Accessed: September 2021.
- [4] D. Moxey. Private conversation, 2021.
- [5] Gaussian quadrature wiki. https://en.wikipedia.org/wiki/Gaussian_quadrature.html. Accessed: March 2022.
- [6] Microsoft Visual Studio. <https://visualstudio.microsoft.com>, 2022. Accessed: March 2022.
- [7] WolframAlpha. www.wolframalpha.com. Accessed: March 2022.
- [8] J.T. Parker and P.J. Dellar. Fourier-Hermite spectral representation for the Vlasov-Poisson system in the weakly collisional limit. *J. Plasma Physics* Vol. 81, 305810203; doi:10.1017/S0022377814001287.
- [9] V. Zala, R.M. Kirby, and Narayan A. Structure-preserving nonlinear filtering for continuous and discontinuous Galerkin spectral/*hp* element methods. *SIAM Journal on Scientific Computing* 43 (6), A3713-A3732, 2021.

- [10] T.H. Stix. *Waves in Plasmas*. American Inst. Phys., 1992.
- [11] J.D. Huba. *NRL Plasma Formulary*. US Naval Research Laboratory, 2013.
- [12] Fadeeva library. http://ab-initio.mit.edu/wiki/index.php/Fadeeva_Package. Accessed: March 2022.