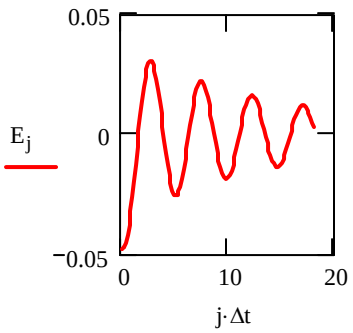
Vlasov - Poisson sytem of equations and Landau damping

The kinetic approach to finding the dispersion relation of electrostatic waves uses the Vlasov equation to find the evolution of the distribution function in one dimension:



$$\frac{\partial f(x,v,t)}{\partial t} + v\frac{\partial f(x,v,t)}{\partial x} + a\frac{\partial f(x,v,t)}{\partial v} = 0$$

where f(x,v,t) is the distribution function, x is distance, v is velocity, t is time and a is acceleration. Poisson 's equation is:

$$\varepsilon_0 \frac{dE}{dx} = nq\left[1 - \int f(x,v,t)dv\right]$$

where the ion density is assumed to have a constant value n, and q and and m have their usual meaning.

The Vlasov equation can be written $$\frac{Df(x,v,t)}{Dt} = 0$$

where D/Dt indicates a convective derivative. If we follow a group of particles, the vaue of f(x,v,t) remains constant on the trajectory in x,v space. For example, if a particle moves to x from x-dx, and simultaneously to v from v-dv, the value that f(x,v,t) will have at the new time and place is the value of f(x-dx, v-dv, t -dt). The following prescription implements the change and carries us forward one time step:

$$f(x,v,t) \leftarrow f(x - dx, v - dv, t - dt)$$

where dx = v dt, dv = a dt, E is the electric field, and a = qE/m.
This procedure is repeated to find the evolution of f(x,v,t).

Cheng and Knorr (see reference) developed the following method that is accurate to second order. The method takes a half step in the dx direction, calculates E, takes a full step in the dv direction, and then a half step in the dx direction:

1. $$f_{temp1}(x,v) \leftarrow f_0\left(x - \frac{dx}{2}, v - dv, t - dt\right)$$

2. $$E \leftarrow 1 - \int f_{temp1}(x,v)\,dv$$

3. $$f_{temp2}(x,v) \leftarrow f_{temp1}(x, v - dv)$$

4. $$f_0(x,v,t) \leftarrow f_{temp2}\left(x - \frac{dx}{2}, v\right)$$

where f_{temp} denotes a temporary function. At each step, a cubic spline function is fit to f(x,v), and the spline function is used to find the values at x - dx/2 and v - dv by interpolation. The interpolation adds significantly to the time needed to find a solution.

Dimensionless equations:

The following substitutions will make the Vlasov equation and Poisson's equation dimensionless:

$$v \leftarrow \frac{v}{v_t} \qquad x \leftarrow \frac{x \cdot \omega_p}{v_t} \qquad E \leftarrow \frac{q \cdot E}{m \cdot v_t \cdot \omega_p} \qquad \omega_p \leftarrow \sqrt{\frac{n \cdot q^2}{\varepsilon_0 \cdot m}} \qquad f \leftarrow v_t \cdot f \qquad \lambda_D \leftarrow \frac{v_t}{\omega_p \cdot \sqrt{2}}$$

where v_t is the thermal velocity and λ_D is defined but not used. The dimensionless equations are:

$$\frac{\partial f(x,v,t)}{\partial t} + v\frac{\partial f(x,v,t)}{\partial x} + E(x,t)\frac{\partial f(x,v,t)}{\partial v} = 0 \qquad \text{Dimensionless Vlasov}$$

$$\frac{dE}{dx} = \left[1 - \int f(x,v,t)dv\right] \qquad \text{Dimensionless Poisson}$$

The x and v grids

In x, our grid will span 12 Debye lengths: $L := 12$

Define the grid using 33 points: $kmax := 32 \quad k := 0 .. kmax \quad x_k := L \cdot \frac{k}{kmax} \quad \Delta x := x_1 - x_0$

$\Delta x = 0.375$

The v grid will be -3 to + 3 in dimensionless units:

$mmax := kmax \quad m := 0 .. mmax \quad v_m := -3 + 6 \cdot \frac{m}{mmax} \quad \Delta v := v_1 - v_0 \quad \Delta v = 0.188$

The v and x grids must have the same number of elements for the 2-d spline to work.

The electrostatic wave

The dimensionless wave amplitude will be δ: $\delta := 0.025$

$\delta = 0.03$ is is the largest value that does not make f(x,v) go negative (unphysical).

fstart(x_k,v_m) is the density-modulated distribution function that we use at the start :

$$fstart_{k,m} := \left(1 + \delta \cdot \cos\left(\frac{2 \cdot \pi \cdot x_k}{L}\right)\right) \cdot \frac{e^{-(v_m)^2}}{\sqrt{\pi}}$$ f is normalized to unity in 1-d.

There is one wavelength in the simulation region.

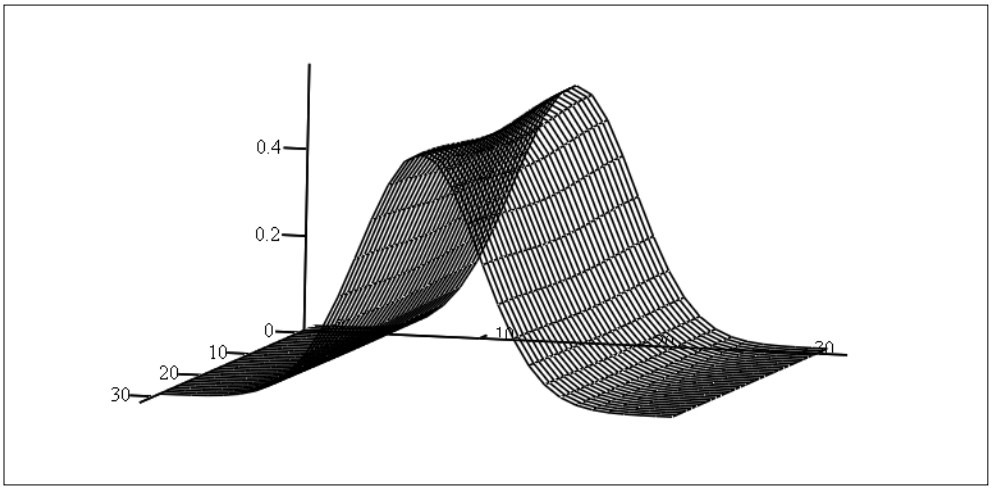
Check normalization to unity:

$$\Delta v \cdot \sum_m \left[\frac{e^{-(v_m)^2}}{\sqrt{\pi}}\right] = 0.999989$$

Check mean squared-velocity:

$$\Delta v \cdot \sum_m \left[\frac{(v_m)^2 \cdot e^{-(v_m)^2}}{\sqrt{\pi}}\right] = 0.499878$$

Plot of f(x,v): x is front to back and v is left to right



fstart

Poisson's equation: finding E from f(x,v)

Poisson equation finds dE/dx at each x_k: $dEdx_k := 1 - \sum_m \left(fstart_{k,m} \cdot \Delta v\right)$

E is found by integrating dE/dx, using Simpson's rule, which integrates across an interval using the average of the two derivative values at the ends of the interval. This method has second order accuracy.

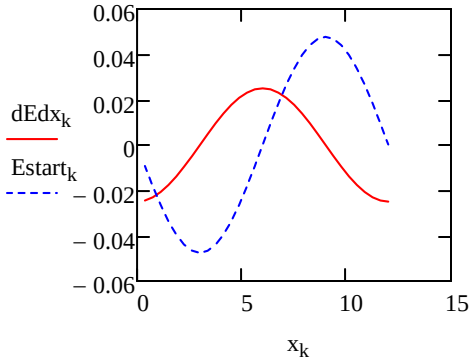
$Estart_0 := 0$ By using the cosine function for the density perturbation, we have a zero electric field at the left boundary.

$k := 1 .. kmax$ Now for all the values k > 0, use Simpson's rule to integrate dE/dx:

$Estart_k := Estart_{k-1} + 0.5 \cdot \left(dEdx_k + dEdx_{k-1}\right) \cdot \Delta x$

Reset range of k: $k := 0 .. kmax$

Plot of initial E and dE/dx values



Define the time grid

Our dimensionless time scale is defined by the plasma period. We will follow three oscillations which will put the ending time at 3 x (2π) = 18, approximately.

$\text{EndTime} := 18$ For accuracy, we will use a time step of 0.25. $\Delta t := 0.25$

The number of iterations we will need is: $\text{jmax} := \text{ceil}\left(\frac{\text{EndTime}}{\Delta t}\right)$ $\text{jmax} = 72$ $\text{jmax}\cdot\Delta t = 18$

Periodic boundary conditions for x

If the particle moves to a value of x that is outside the simulations box, we must use the modulo function **mod** to put the x value into the box. This function will return the remainder after dividing by the length of the box L. If the value of x is less than zero, **mod** will return a negative number that is outside our box. So we must add L to the value of x before using the **mod** function:

$\text{mod}\left(L + x_k - 0.5\cdot v_m\cdot\Delta t, L\right)$ This returns a value for x that is always between 0 and L.

Number density of particles in the simulation box

We can find the number of particles in the box by integrating f(x,v) with dx and dv. The integral is performed by a double summation. We divide by L to find the number per unit length.

$$n(f) := \frac{1}{L}\cdot\left[\sum_{kk=1}^{kmax}\left[\left[\sum_{m}\left[0.5\cdot\left(f_{kk,m} + f_{kk-1,m}\right)\cdot\Delta v\right]\right]\cdot\Delta x\right]\right] \qquad n(fstart) = 0.999989$$

For accuracy, we used Simspon's rule when integratring over x. It is not necessary to do this for the v integral because the values of f(x,v) are nearly zero at the end points.

Total energy of particles in the simulation box

For the total kinetic energy W, we integrate (sum) 0.5 v_m^2 with the distribution function:

$$W(f) := \left[\sum_{kk=1}^{kmax}\left[\left[\sum_{m}\left[0.5\cdot\left(f_{kk,m} + f_{kk-1,m}\right)\cdot\left[0.5\cdot\left(v_m\right)^2\right]\cdot\Delta v\right]\right]\cdot\Delta x\right]\right] \qquad W(fstart) = 2.999$$

Total energy in the electric field

This is found by summing 0.5 E^2 with dx, using Simpson's rule:

$$\text{Eenergy}(Estart) := \sum_{kk=1}^{kmax}\left[0.5\cdot 0.5\left[\left(Estart_{kk}\right)^2 + \left(Estart_{kk-1}\right)^2\right]\cdot\Delta x\right] \qquad \text{Eenergy}(Estart) = 6.808\times 10^{-3}$$

Grid point information

The spline fitting routine needs to know the values of the x and v grid points, so we stack these vectors into a matrix with two columns: $xv := \text{augment}(x, v)$

The program loop j is time index, k is distance index, m is velocity index

$M :=$

$f0 \leftarrow fstart$

Initialize matrices f0, f1, f2

$f1_{kmax,\,mmax} \leftarrow 0$

$f2_{kmax,\,mmax} \leftarrow 0$

Initialize vectors E, Esave.

$E_{mmax} \leftarrow 0$

$Esave \leftarrow Estart$

for $j \in 0 .. jmax$

fit spline to f0

$SplineCoeffs \leftarrow cspline(xv, f0)$

for $m \in 0 .. mmax$

for $k \in 0 .. kmax$

find f1

$$f1_{k,m} \leftarrow interp\left[SplineCoeffs, xv, f0, \begin{pmatrix} mod\left(L + x_k - 0.5 \cdot v_m \cdot \Delta t, L\right) \\ v_m \end{pmatrix}\right]$$

fit spline to f1

$SplineCoeffs \leftarrow cspline(xv, f1)$

for $k \in 0 .. kmax$

find dE/dx

$$dEdx_k \leftarrow 1 - \sum_{m=0}^{mmax} \left(f1_{k,m} \cdot \Delta v\right)$$

find E_k

for $k \in 1 .. kmax$

$$E_k \leftarrow E_{k-1} + 0.5 \cdot \left(dEdx_k + dEdx_{k-1}\right) \cdot \Delta x$$

for $m \in 0 .. mmax$

for $k \in 0 .. kmax$

find f2

$$f2_{k,m} \leftarrow interp\left[SplineCoeffs, xv, f1, \begin{pmatrix} x_k \\ v_m + E_k \cdot \Delta t \end{pmatrix}\right]$$

fit spline to f2

$SplineCoeffs \leftarrow cspline(xv, f2)$

for $m \in 0 .. mmax$

for $k \in 0 .. kmax$

find the new f0

$$f0_{k,m} \leftarrow interp\left[SplineCoeffs, xv, f2, \begin{pmatrix} mod\left(L + x_k - 0.5 \cdot v_m \cdot \Delta t, L\right) \\ v_m \end{pmatrix}\right]$$

$Esave \leftarrow augment(Esave, E)$

$augment(Esave, f0)$

Augment(Esave, E) is used to stack the E values at each time step into an array Esave.

Augment (Esave,f0) is used to stack the final values for f(x,v) into the matrix Esave that has the E values. The first thing we do below is unstack the E values and f(x,v) into separate matrices:

This statement puts the E(x,t) values into the matrix Ext: $Ext := submatrix(M, 0, kmax, 0, jmax + 1)$

This statement recovers the values of f(x,v) into a new matrix FinalF:

$FinalF := submatrix(M, 0, kmax, jmax + 2, cols(M) - 1)$

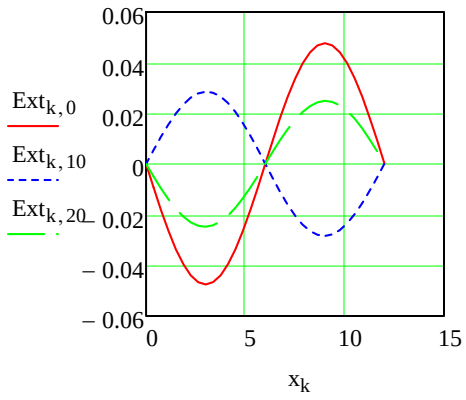
This submatrix has E(x,t):

$Ext =$

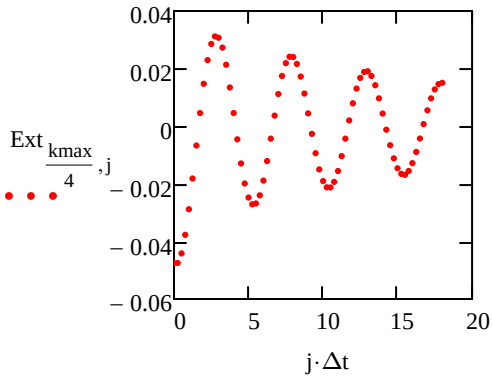
	0	1	2	3	4	5
0	0	0	0	0	0	0
1	$-9.281 \cdot 10^{-3}$	$-9.271 \cdot 10^{-3}$	$-8.6 \cdot 10^{-3}$	$-7.325 \cdot 10^{-3}$	$-5.569 \cdot 10^{-3}$	$-3.493 \cdot 10^{-3}$
2	-0.018	-0.018	-0.017	-0.014	-0.011	...

$j := 0 .. jmax$

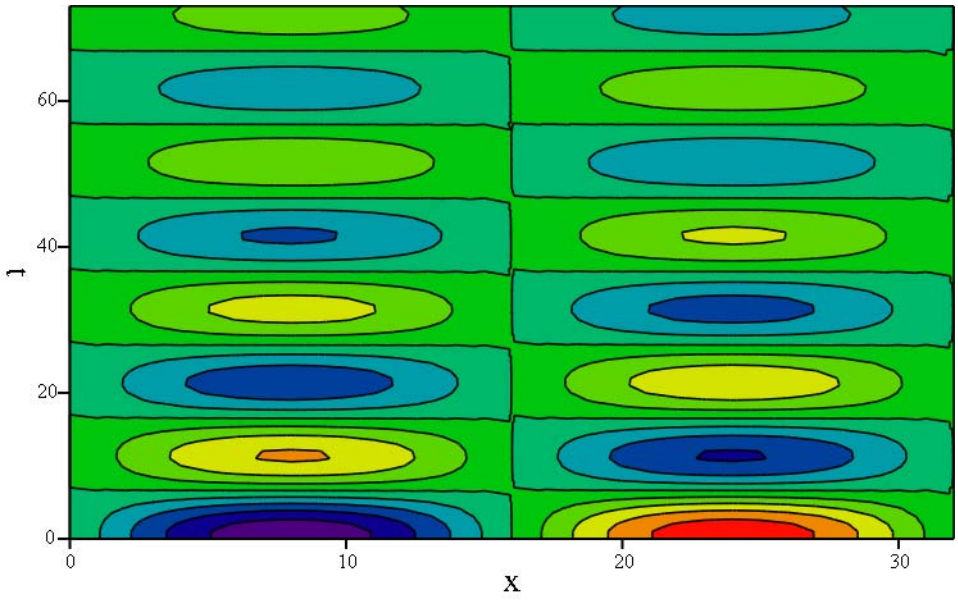
E as a function of x at 3 times



Peak value of E as a fcn of time



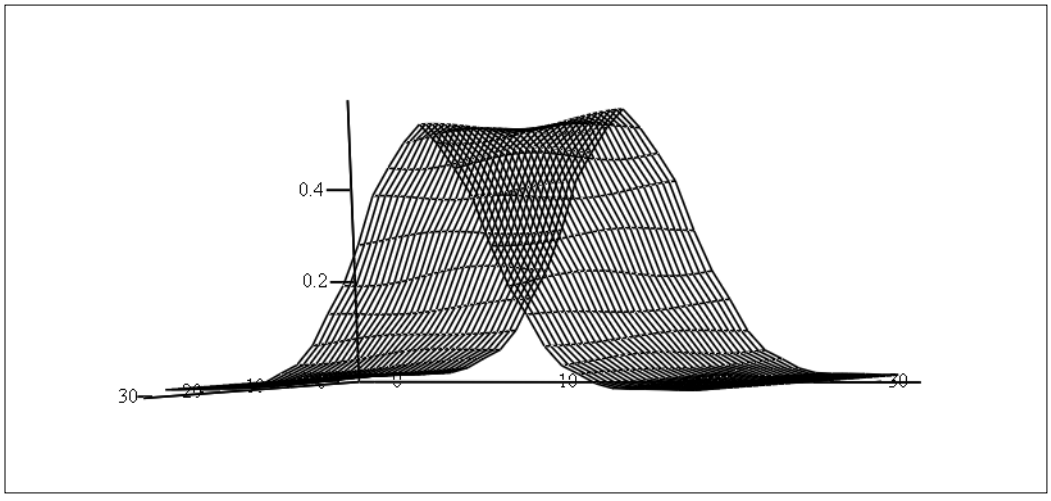
Plot of E(x,t).



Ext

Distance is along the horizontal axis and time is on the vertical axis.
The E field "seesaws" as time increases.

This plot is of f(x,v) at the end time



(FinalF)

Check here to see if f has gone negative:
If so, the wave amplitude δ must be reduced. $\min(\text{FinalF}) = 9.342 \times 10^{-6}$

Did the reduced wave energy show up as increased particle energy?

The final E field is: $Efinal_k := Ext_{k,jmax}$ because jmax is the end time.

The change in the E field energy is: $Eenergy(Efinal) - Eenergy(Estart) = -6.125 \times 10^{-3}$

where: $Eenergy(Estart) = 6.808 \times 10^{-3}$ $Eenergy(Efinal) = 6.823 \times 10^{-4}$

The change in the particle energy is: $W(FinalF) - W(fstart) = 6.401 \times 10^{-3}$

where: $W(fstart) = 2.999$ $W(FinalF) = 3.006$

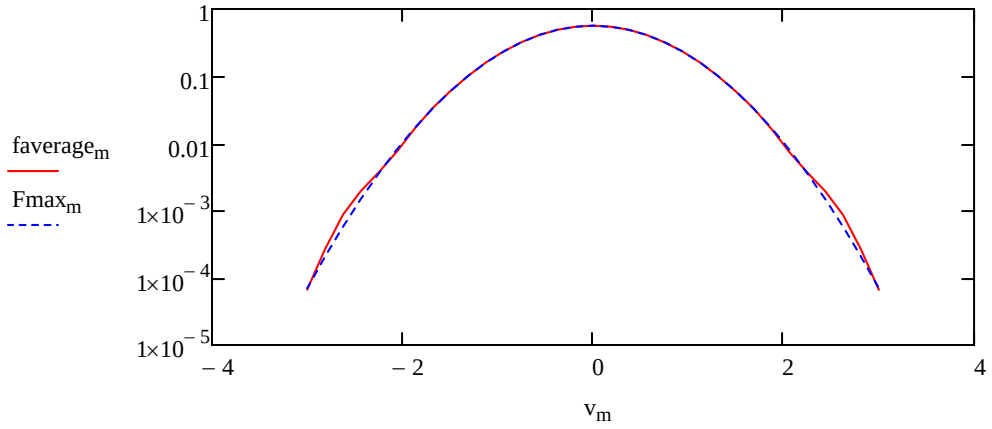
Energy is poorly conserved because we have used a rather coarse spacing in x, v and t in order to reduce the calculation time.

Try it: Change Δt from 0.25 to 0.1 and notice that energy conservation is improved.

Modification of the distribution function

The damping of the waves flattens the region of f(x,v) where v is near to the phase velocity of the wave. For a nicer plot, we will average the final value of F over all x.

$$faverage_m := \frac{1}{kmax + 1} \cdot \left(\sum_k FinalF_{k,m} \right)$$ Compare this to: $$Fmax_m := \frac{e^{-(v_m)^2}}{\sqrt{\pi}}$$

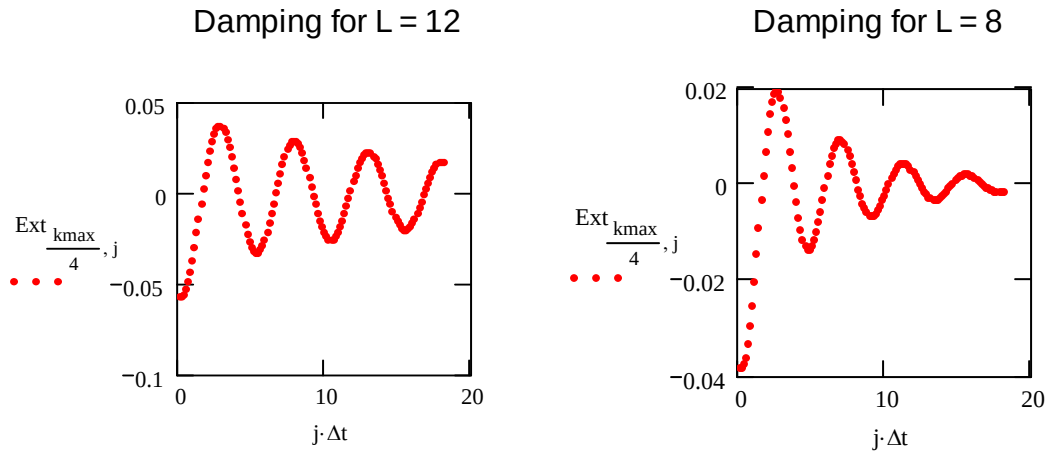


A "bulge" in the distribution function is visible at a velocity near 2.5.

Try it: Examine the E(t) and calculate the wave frequency. The wavenumber is $2\pi/L$. Is the phase velocity ω/k near to the velocity where there is a bulge in the distribution function?

k dependence of Landau damping

The figures below show that Landau damping is more severe for shorter wavelengths:



Try it: How does the observed damping compare with the damping from theory? How heavily damped is the wave with L = 6? 4?

Reference: C. Z. Cheng and G. Knorr, J. Comp. Phys. 22, 330 (1976).