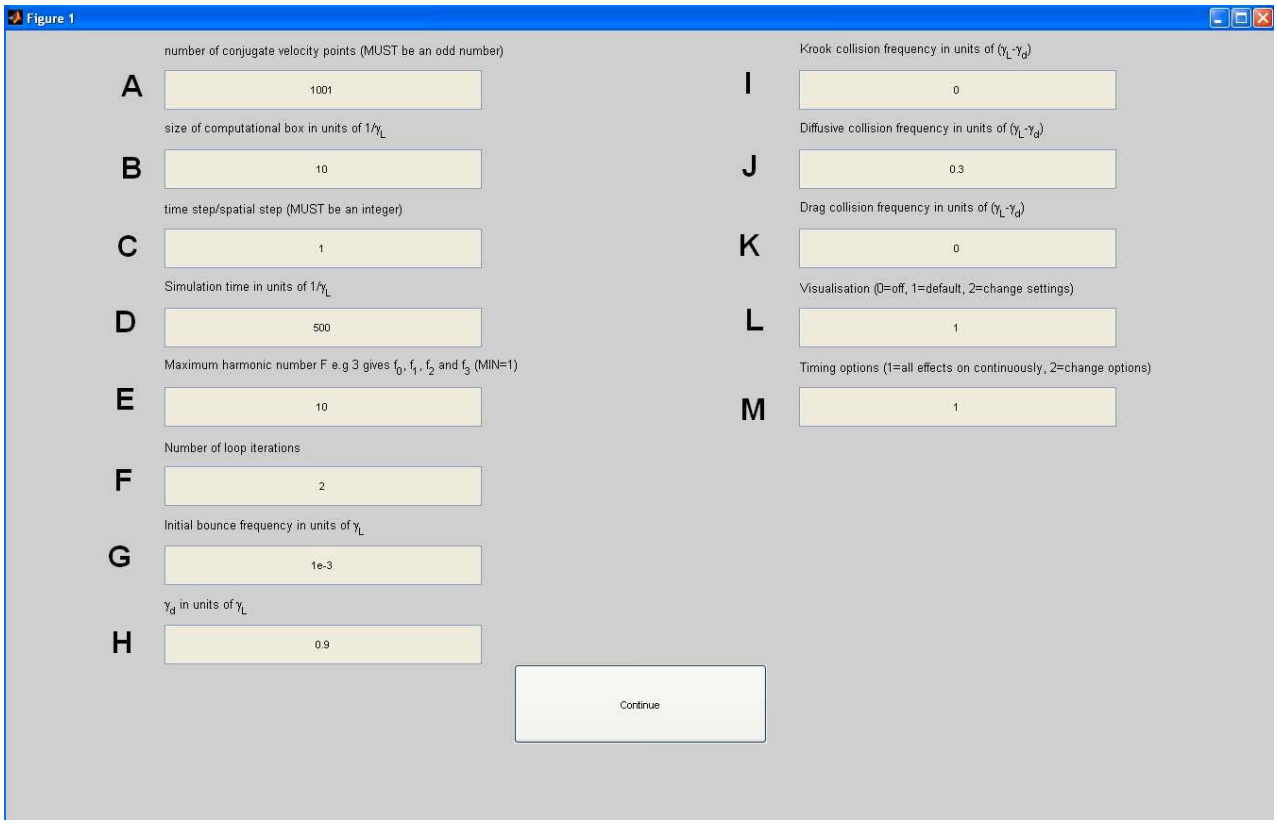
BOT (Bump On Tail) Manual

REQUIREMENTS

- Windows or Mac or Unix/Linux
- Matlab – specifically you will need the *specgram* routine if you wish produce spectrogram plots.

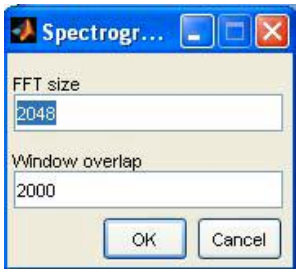
1- GETTING STARTED

There are many settings you can play with on the input screen. Some are physics parameters, others are for setting the accuracy of the simulations. It is instructive to begin by looking at physics parameters, since the default values give good accuracy for most situations you might wish to look at.



1. We will begin with the default settings. This will produce **holes and clumps.** Click **continue**.
2. You should see a figure with two subplots. The top is the spatially averaged part of the perturbation to the distribution function (f_0), which is where you will see holes and clumps produced. The bottom plot is the square of the bounce frequency, which is proportional to the amplitude of the electric field.
3. You should see a number printed in the command window. This tells you how many plots have been produced. The default number per simulation is **100**.

4. At the end of the simulation you will be presented with an input screen for plotting the spectrogram of the electric field. If you do not have **specgram** as part of your MatLab package this window will not be displayed. A spectrogram is a picture that combines many Fast Fourier Transforms (FFT) of successive intervals of data (whose size is defined here by **FFT size**) to give frequency as a function of time. Click **OK** to produce the spectrogram. The input screen will appear again along with the plot. You may play with this plot to determine if it is good. If not you may produce another. Try changing **FFT size** to **256** and **Window overlap** to **200**. The larger **FFT size** the greater the resolution in frequency but the poorer resolution in time. **FFT size** is typically set to be a power of 2 to make the FFT run fast. **Window overlap** must always be less than **FFT size**. Once you have finished click **cancel**. Note that specgram cannot handle the negative frequencies that would arise from doing a spectrogram of the electric field amplitude alone. Therefore an artificial carrier frequency of **10** has been included in order that the downward sweeping events are at a positive frequency.



5. Two files have now been created in the current folder, one is a **.mat** file containing data from the simulation and the other is a **.avi** file, which is a movie of the plots. Note that in MAC and UNIX this movie will be uncompressed (i.e large!). Windows typically has codecs which allow the movie to be compressed. BOT determines if you have an appropriate compressor.

2- COLLISIONALITY

We will now look at the effect of diffusion on the holes and clumps.

EX1

1. Make the simulation longer by changing setting **D** to **800** and increase diffusion by changing setting **J** to **1.** Click **continue**
2. When the simulation has finished click **OK** at the spectrogram input screen.

Observe that diffusion causes the holes and clumps to die after some time.

We will now look at the effect of Krook on the holes and clumps.

EX2

1. Turn off diffusion by setting **J** to **0** and turn on Krook collisions by setting **I** to **0.5.** Click **continue**
2. When the simulation has finished click **OK** at the spectrogram input screen.

Observe that with Krook collisions the holes and clumps are barely allowed to form. There is a tendency for frequency sweeping (you can see by zooming in) but it is much less than with diffusion.

Diffusion and Kook both act to kill holes and clumps, but since the diffusion timescale goes like $1/v^3$ and for Krook it is $1/\beta$, Krook is much more effective.

Now let's see what happens as we push the diffusive collisions higher.

EX3

1. Increase diffusion by changing setting **J** to **3.** Click **continue**
2. When the simulation has finished click **OK** at the spectrogram input screen.

Observe that no holes and clumps form. The amplitude has reached a steady state and the frequency does not chirp. The steady state amplitude was calculated analytically in [1,2] for the case of Krook and diffusion for the near threshold case ($\gamma_d / \gamma_L \approx 1$).

At some low level of Krook or diffusive collisions the steady state behaviour is lost, but holes and clumps do not yet form. The collisionality at which the steady state is lost was also calculated in [1,2]. Let's look at an example of such behaviour

EX4

1. Make the simulation longer by changing setting **D** to **2000.** Lower diffusion by changing setting **J** to **2**. Click **continue**
2. When the simulation has finished set **FFT size** to **16384** and **Window overlap** to **16000** and click **OK** at the spectrogram input screen.

The frequency can now be seen to split into side bands. This is called pitch fork splitting (zoom in to see this). The reason it was necessary to alter the spectrogram parameters is that this splitting is fine and so we needed a high frequency resolution to see it.

Now let's see what happens when we add drag collisions along with diffusion

EX5

1. Set diffusion by changing setting **J** to **2.** Turn on drag by setting **K** to **2.2**. Click **continue**
2. When the simulation has finished click **OK** at the spectrogram input screen.

Steady state holes are now formed with a constant frequency shift that can be observed by zooming in. A more striking example of this behaviour is seen at a lower collisionality over a longer time

EX6

1. Make the simulation longer by changing setting **D** to **4000**. Change diffusion setting **J** to **1.** Change drag setting **K** to **1.5**. Click **continue**
2. When the simulation has finished click **OK** at the spectrogram input screen.

The competition between drag and diffusion produces a diverse range of behaviours, another example of which is hooked frequency chirping

EX7

1. Make the simulation longer by changing setting **D** to **4000**. Change diffusion setting **J** to **1.3.** Change drag setting **K** to **1.5**. Click **continue**
2. When the simulation has finished click **OK** at the spectrogram input screen

In general the drag is promoting the formation of holes and clumps whereas diffusion and Krook do the opposite [3,4]

3- DISSIPATION

Dissipation is controlled by setting **H**. If $\gamma_d > \gamma_L$, i.e. if dissipation is greater than the drive, then no instability will develop if the initial wave amplitude is sufficiently low. The initial amplitude is controlled by setting **G**.

EX8

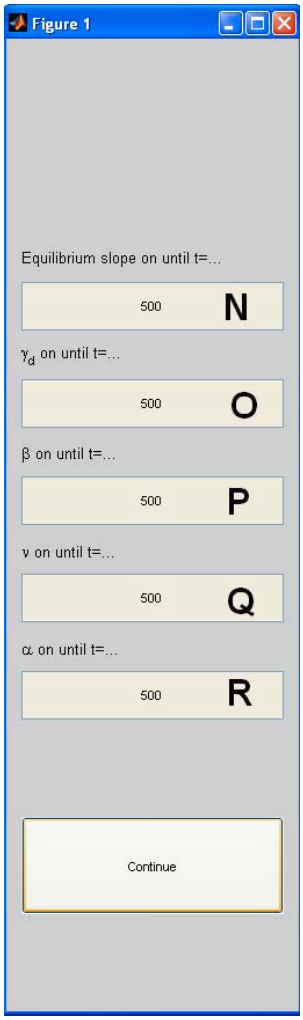
1. Turn off collisions by setting **J** to **0**. Make the dissipation greater than the drive by setting **H** to be **1.5**. Click **continue.**
2. When the simulation has finished click **OK** at the spectrogram input screen

The amplitude dies on a time scale given by $1/\gamma_d$. Now start off with a large amplitude wave.

EX9

1. Turn off collisions by setting **J** to **0**. Make the dissipation greater than the drive by setting **H** to be **1.5**. Create a large amplitude wave by setting **G** to **1**. Click **continue.**
2. When the simulation has finished click **OK** at the spectrogram input screen

Observe that holes and clumps develop even though the system was linearly stable! The initial bounce frequency needs to be comparable to the dissipation rate in order for this effect to be seen. This is not the situation we normally think about. We are normally interested in an instability that has grown out of some low level noise.



We shall now see how dissipation controls the chirping of holes and clumps near the threshold when $\gamma_d / \gamma_L \approx 1$. In the absence of collisions, holes and clumps will in principle persist indefinitely without dying.

EX10

1. Switch off collisions by setting **J** to **0**. Change setting **M** to **2**, which allows us to change how long certain effects are active for. Click **continue**
2. Change option **O** to **300**, this will switch off the damping after 300 in time. Change setting **N** to **300**, this will switch off the equilibrium slope of the distribution function to ensure the original resonance does not produce a very large instability that will destroy the nice holes and clumps that have just formed.
3. Click **continue**
4. When the simulation has finished click **OK** at the spectrogram input screen

Observe that chirping only continues up to the point when the dissipation is switched off. This is because dissipation is the reason behind chirping. The holes and clumps move in order to release kinetic energy to balance dissipation to a first approximation, so without dissipation they will not move and the frequency will remain fixed.

Far away from the instability threshold the story is quite different. Holes and clumps are not produced.

Now let us see what happens in the absence of dissipation and collisions

EX11

1. Make the simulation time shorter by changing setting **D** to **50**. Turn off collisions by setting **J** to **0**. Turn off dissipation by setting **H** to **0**. Click **continue.**
2. You will see a prompt in the command window informing you that the code has **LOST STABILITY**

From a physics point of view the electric field should have saturated. The problem is that without dissipation the saturation level is relatively large. The BOT code relies on an iterative method to calculate various quantities and this iterative method has a stability range. The larger the field becomes the smaller time step that is required to maintain stability. Stability is also affected by other things that we have yet to discus. To resolve this matter immediately simply change setting **A** to **8001** and try again. Observe that you don't quite get saturation, but you have solved the stability problem. Finally change setting **E** to **30** and try again. You will find you get nice saturation.

Now at this point you don't know what you just did to make everything better. It is therefore necessary to introduce you to the other settings on the input screen and explain where they come from and how the code works in more detail. But before we can do that you need to understand more about the physics problem we are looking at and the equations.

4- BOT PROBLEM AND EQUATIONS

The physics system consists of a single electrostatic wave in a 1-D periodic plasma containing three species. The first two are the static background ions and `cold' electrons (characterised by mass m_e an equilibrium density n_e, perturbed fluid velocity V). The cold electrons respond linearly to the wave field and are subject to a small friction force providing a damping mechanism for the wave. The third is a low density population of fast electrons that are subject to `weak' (much less than the background) collisions and whose distribution function F is treated kinetically. The resulting closed system of equations is:

$$\begin{aligned}
&\frac{\partial F}{\partial t}+\mathrm{v}\frac{\partial F}{\partial x}-\frac{|e|E}{m_e}\frac{\partial F}{\partial \mathrm{v}}=\left.\frac{dF}{dt}\right|_{\mathrm{coll}}\\
&\frac{\partial V}{\partial t}=-\frac{|e|E}{m_e}-\nu_c V \qquad (1.1)\\
&\frac{\partial E}{\partial t}=4\pi|e|\left(n_e V+\int \mathrm{v}\left(F-F_0\right)d\mathrm{v}\right)
\end{aligned}$$

where ν_c is the cold electron collision frequency, F_0 is the unperturbed distribution function of the fast electrons (assumed to have a constant positive slope $\partial F_0/\partial \mathrm{v}>0$ that determines the linear growth rate γ_L) and the right hand side of Ampere's Law is a sum of the perturbed currents from

the cold and fast electrons. The constant slope assumption is appropriate as long as the velocity range of interest is significantly more narrow than the overall width of the distribution function F_0. The appropriate collision operator for the problem ($dF/dt|_{\text{coll}}$) involves three relaxation processes (diffusive, drag and Krook-type), and is taken to be of the following form

$$\left.\frac{dF}{dt}\right|_{\text{coll}} = D\frac{\partial^2}{\partial \mathrm{v}^2}\left(F-F_0\right)+S\frac{\partial}{\partial \mathrm{v}}\left(F-F_0\right)-K\left(F-F_0\right) \tag{1.2}$$

D, S and K are constants characterising the velocity space diffusion, dynamical friction (slowing down), and Krook operators respectively.

Periodicity of the plasma naturally leads to the use of a Fourier description in space for the physical quantities.

$$\begin{aligned} F &= F_0(\mathrm{v}) + f_0(\mathrm{v},t) + \sum_{n=1}^{N}\left[f_n(\mathrm{v},t)\exp(in\xi) + \mathrm{c.c}\right] \\ E &= \frac{1}{2}\times\left[\hat{E}(t)\exp(i\xi)+\mathrm{c.c}\right] \\ V &= \left[\hat{V}(t)\exp(i\xi)+\mathrm{c.c}\right] \end{aligned} \tag{1.3}$$

where $\xi \equiv k\mathrm{x}-\omega_{pe}t$, $k \equiv 2\pi/\lambda$ and where the envelope functions $\hat{E}$ and $\hat{V}$ are assumed to vary on a time scale that is long compared to $1/\omega_{pe}$. The electric field is assumed to be sinusoidal, which remains a good approximation as long as the frequency of the wave remains close to the plasma frequency. Setting **E** controls the maximum harmonic number of the distribution function F.

The resulting algebraic nature of the spatial derivatives transforms the kinetic equation from a partial differential one in (x,v,t) to a coupled system in (v,t). In the wave reference frame, moving with the phase speed, this looks like

$$\begin{aligned} &\frac{\partial f_0}{\partial t} - \nu^3\frac{\partial^2 f_0}{\partial u^2} - \alpha^2\frac{\partial f_0}{\partial u} + \beta f_0 = \frac{1}{2}\frac{\partial}{\partial u}\left(\omega_B^2 f_1^* + \mathrm{c.c}\right) \\ &\frac{\partial f_1}{\partial t} + iuf_1 - \nu^3\frac{\partial^2 f_1}{\partial u^2} - \alpha^2\frac{\partial f_1}{\partial u} + \beta f_1 = \frac{1}{2}\frac{\partial}{\partial u}\left(\omega_B^2\left(F_0+f_0\right)+\omega_B^{2*} f_2\right) \\ &\frac{\partial f_2}{\partial t} + 2iuf_2 - \nu^3\frac{\partial^2 f_2}{\partial u^2} - \alpha^2\frac{\partial f_2}{\partial u} + \beta f_2 = \frac{1}{2}\frac{\partial}{\partial u}\left(\omega_B^2 f_1 + \omega_B^{2*} f_3\right) \end{aligned} \tag{1.4}$$

where $u \equiv k\mathrm{v}-\omega_{pe}$, $\omega_B^2 \equiv |e|k\hat{E}/m_e$ and $\nu^3 = Dk^2$, $\alpha^2 = Sk$, $\beta = K$. Note $\sqrt{|\omega_B^2|}$ gives the trapped particle bounce frequency.

By calculating the current from the cold background perturbatively, using the assumed smallness of ν_c and $\partial\left(log\left(\hat{V}\right)\right)/\partial t$ with respect to ω_{pe}, Amperes law is then given by

$$\frac{\partial \omega_B^2}{\partial t} - \frac{4\pi |e|^2}{m_e} \frac{\omega_{pe}}{k} \int f_1 du + \gamma_d \omega_B^2 = 0 \tag{1.5}$$

where $\gamma_d \equiv \nu_c / 2$ is the damping rate of the wave. Eq 1.5 takes into account that the dominant contribution to the perturbed current comes from the resonant electrons with $\mathrm{v} = \omega_{pe} / k$, which allows the fast electron current to be written as

$$-|e| \int \mathrm{v} f_1 d\mathrm{v} \approx -|e| \frac{\omega_{pe}}{k} \int f_1 d\mathrm{v} \tag{1.6}$$

The kinetic equation is well known to generate secularly growing velocity space derivatives as small scales develop over time. This `small scale' problem can be dealt with using artificial numerical diffusion, but a simpler method is to Fourier transform in velocity space. The smallest scale can then be controlled by a suitable choice of numerical box size in conjugate velocity space 's'. Another advantage of Fourier space is that the collision operator simplifies to an algebraic one. Setting **B** controls the shortest length scale in velocity space that can be resolved (larger number resolves shorter scales). Setting **A** controls the resolution in conjugate velocity space, which relates to the maximum velocity in the simulation (larger number gives higher resolution and larger maximum velocity).

Using the normalised variables $C = \omega_B^2 / \gamma_L^2$, $G = \left(2\pi |e|^2 \omega_{pe} / m\gamma_L^2 k\right) F$, $\tau = \gamma_L t$, $\hat{\gamma}_d = \gamma_d / \gamma_L$, $\Omega = \left(k\mathrm{v} - \omega_{pe}\right) / \gamma_L$, $\bar{\nu} = \nu / \gamma_L$ and similarly for $\bar{\alpha}$ and $\bar{\beta}$, with $\gamma_L \equiv (\omega_{pe}^3 \pi / 2k^2 n_e) \partial F_0 / \partial \mathrm{v}$ the Fourier transformed equations are then

$$\frac{\partial \mathcal{G}_n}{\partial \tau} - n \frac{\partial \mathcal{G}_n}{\partial s} + \left[\bar{\nu}^3 s^2 - i\bar{\alpha}^2 s + \bar{\beta}\right] \mathcal{G}_n = R_n(s,\tau) + \delta_{1,n} \frac{1}{\sqrt{2\pi}} C\delta(s) \tag{1.7}$$

$$\frac{\partial C}{\partial \tau} + \hat{\gamma}_d C = 2\sqrt{2\pi} \mathcal{G}_1(0,\tau) \tag{1.8}$$

where $\mathcal{G}_n = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} G_n \exp(-i\Omega s) d\Omega$ is the Fourier transform of G_n, $\delta_{1,n}$ is the Kronecker delta and $R_n(s,\tau)$ represents the non-linearity in the equations and is given by

$$\begin{aligned} R_0 &= \frac{is}{2}\left[C^* \mathcal{G}_1(s,\tau) + C\mathcal{G}_1^*(-s,\tau)\right] \\ R_n &= \frac{is}{2}\left[C^* \mathcal{G}_{n+1}(s,\tau) + C\mathcal{G}_{n-1}(s,\tau)\right] \end{aligned} \tag{1.9}$$

The delta function $\delta(s)$ arises from the constant slope approximation of the equilibrium distribution function F_0. The fast particle current is now represented as a single point of the Fourier transformed distribution function so that no numerical velocity integration needs to be performed .

THE BOT CODE SOLVES EQS 1.7 AND 1.8.

Eqs 1.7 are integrated using the method of characteristics. This procedure allows the contribution from the equilibrium distribution function F_0 (the delta function in Eq 1.7) to be integrated analytically. This greatly reduces the resolution required in s, since one is not forced to resolve a strongly peaked function around $s=0$. Specifically, by changing variables to $s'=s+n\tau$, $\tau'=\tau$ and integrating in τ', by using the trapazoid rule, one finds the following relation

$$\mathcal{G}_n(s,\tau+\Delta\tau) = \mathcal{G}_n(s+n\Delta\tau,\tau)\,\mathrm{EXP} + \delta_{1,n}\sigma + \frac{\Delta\tau}{2}\left[R_n(s,\tau+\Delta\tau) + R_n(s+n\Delta\tau,\tau)\,\mathrm{EXP}\right] \tag{1.10}$$

Where

$$\mathrm{EXP} \equiv \exp\left[-\frac{\bar{\nu}^3\Delta\tau}{3}\left(3s^2+3sn\Delta\tau+n^2\Delta\tau^2\right)+\frac{i\bar{\alpha}^2\Delta\tau}{2}\left(2s+n\Delta\tau\right)-\bar{\beta}\Delta\tau\right] \tag{1.11}$$

and where

$$\sigma = \begin{cases} 0 & \text{if} \quad s>0 \\ \dfrac{C(\tau+\Delta\tau)}{2\sqrt{2\pi}} & \text{if} \quad s=0 \\ \dfrac{C(\tau+\Delta\tau+s)}{\sqrt{2\pi}}\exp\left(\dfrac{\nu^3 s^3}{3}-\dfrac{i\alpha^2 s^2}{2}+\beta s\right) & \text{if} \quad -\Delta\tau<s<0 \\ \dfrac{C(\tau)}{2\sqrt{2\pi}}\exp\left(\dfrac{\nu^3 s^3}{3}-\dfrac{i\alpha^2 s^2}{2}+\beta s\right) & \text{if} \quad s=-\Delta\tau \end{cases} \tag{1.12}$$

noting that $C(\tau+\Delta\tau+s)$ should be obtained by linear interpolation to be consistent with the trapazoid rule used to evaluate the integrals. The contribution from the delta function to $\mathcal{G}_1$ creates a discontinuity. To avoid interpolating discontinuous functions in the scheme, the time step $\Delta\tau$ is set to be an integer multiple of Δs so that $\mathcal{G}_n(s+n\Delta\tau,\tau)$ is a known value. Setting **C** controls the ratio of time step to conjugate velocity step. The wave equation Eq 1.8 is also integrated in time using the trapazoid rule to obtain

$$C(\tau+\Delta\tau) = C(\tau)\exp(-\hat{\gamma}_d\Delta\tau) + \frac{\Delta\tau}{2}2\sqrt{2\pi}\left[\mathcal{G}_1(0,\tau+\Delta\tau)+\mathcal{G}_1(0,\tau)\exp(-\hat{\gamma}_d\Delta\tau)\right] \tag{1.13}$$

To solve for the distribution function at the next time-step $\mathcal{G}_n(s,\tau+\Delta\tau)$ one should invert a matrix equation of the form $A\mathcal{G}_n(s,\tau+\Delta\tau) = B\mathcal{G}_n(s,\tau)$ for each point s. However, since $\mathcal{G}_n$ couples only to neighbouring harmonics, the matrix A will be sparse and so computational time will be wasted by performing such an inversion. By recognising that the non-linear piece R_n in Eq 1.10 scales as $\Delta\tau$ as compared to the linear piece (and even smaller for small values of s which is where most of the accuracy is needed), an efficient way to solve for the distribution function is to use an iterative method. The numerical scheme then takes the form

$$\mathcal{G}_n\left(s,\tau+\Delta\tau\right)^{(p+1)} = c + P\mathcal{G}_n\left(s,\tau+\Delta\tau\right)^{(p)} \tag{1.14}$$

where P is called the iteration matrix, c is related to $\mathcal{G}_n\left(s,\tau\right)$, and $\mathcal{G}_n\left(s,\tau+\Delta\tau\right)^{(p+1)}$ is the improved solution based on the previous guess $\mathcal{G}_n\left(s,\tau+\Delta\tau\right)^{(p)}$. The same applies to the wave equation. The natural first `guess' is then the solution to the linear problem, i.e neglecting all the R_n terms. Setting **F** controls the maximum number of iterations. Note this iterative scheme has a stability range. The BOT code is stable as long as

$$\Delta\tau < \frac{2}{|s|_{\max}|C|} \tag{1.15}$$

5- UNDERSTANDING HOW MUCH ACCURACY YOU NEED

If the electric field remains small such that $\omega_B / \gamma_L << 1$ then the system is called weakly nonlinear. This is the regime that you looked at in exercises *EX3 and EX4*. If the system is weakly nonlinear then we do not need to keep many Fourier harmonics of the distribution function.

EX12

1. Increase diffusion by changing setting **J** to **3**. Decrease setting **E** to **1**. Click **continue**
2. When the simulation has finished click **OK** at the spectrogram input screen.
3. Compare the amplitude for this case and for the *EX3* case

The simulation ran much faster and the amplitudes are almost identical.

We can go further. The weakly nonlinear regime only perturbs a small region in phase space around the resonance, so we don't need a large velocity range in the simulation. In Fourier space this corresponds to having a lower resolution and so a smaller value in setting A.

EX13

1. Increase diffusion by changing setting **J** to **3.** Decrease setting **E** to **1.** Decrease setting **A** to **251**. Click **continue**
2. When the simulation has finished click **OK** at the spectrogram input screen.
3. Compare the amplitude for this case and for the *EX12* case

The simulation again ran faster with barely any difference in the final result.

The larger the electric field the more nonlinear the system becomes and the more iterations are required. For the weakly nonlinear scenarios changing setting **F** to **1** is good enough. In more strongly nonlinear regimes the default setting is good enough.

EX14

1. Increase diffusion by changing setting **J** to **3.** Decrease setting **E** to **1.** Decrease setting **A** to **251**. Decrease setting **F** to **1**. Click **continue**
2. When the simulation has finished click **OK** at the spectrogram input screen.
3. Compare the amplitude for this case and for the *EX13* case

Again, barely any change.

Note that Making setting **E** to **1** and **F** to **1** should reproduce the analytical results from [1,3]. This is because the code is then only capturing the first nonlinear correction to the wave amplitude by performing 1 iteration and only taking terms up to f_1 in the distribution function.

What about the box size in Fourier space, or in other words, what about $|s|_{\max}$, how large should it be? The larger $|s|_{\max}$ the more accurate the simulation will become, because you are resolving shorter scales in velocity space, so you are capturing more physics. Note however in Eq 1.15, that $|s|_{\max}$ comes into the stability criterion. So if you increase the size of your numerical box, you will make the code more unstable unless you decrease the time-step. This will naturally make the code slower. What this tells you is that you don't want to have a bigger box than you really need. Krook and diffusive collisions naturally remove short scales in velocity space. A rough guide is:

$$|s|_{\max} \sim \frac{1}{\nu},\ \frac{1}{\beta},\quad \text{collision frequencies must be in units of } \gamma_{\mathrm{L}} \tag{1.16}$$

In *EX14* setting **B** was **10**. Was this too much or too little? $\nu/(\gamma_L - \gamma_d) = 3$ and $\gamma_d/\gamma_L = 0.9$, so $\nu/\gamma_L = 0.3$, which gives us $|s|_{\max} \sim 3.3$. So apparently we were being overly accurate in *EX14*. Let's see this explicitly.

EX15

1. Increase diffusion by changing setting **J** to **3.** Decrease setting **E** to **1.** Decrease setting **A** to **251**. Decrease setting **F** to **1**. Decrease setting **B** to **5**. Click **continue**
2. When the simulation has finished click **OK** at the spectrogram input screen.
3. Compare the amplitude for this case and for the *EX14* case
4. Repeat but this time reduce setting **B** to **3**

The results are not that different with $|s|_{\max} = 5$. However if we reduce it to as low as $|s|_{\max} = 3$ we see a difference, which tells us that we did not resolve short enough scales in velocity space. This is consistent with the estimate in Eq 1.16.

For the collisionless case we should formally keep all scales in velocity space, i.e we should have $|s|_{\max} = \infty$. However, over time the fluctuations in velocity space get shorter and shorter and do not contribute to the perturbed current that drives the wave. This means we can capture the physics with a finite box even in the collisionless case. You must check to see how the box size affects your results however. $|s|_{\max} = 10$ seems to be good enough.

Now we return to *EX11*.

EX11

1. *Make the simulation time shorter by changing setting **D** to **50**. Turn off collisions by setting **J** to **0**. Turn off dissipation by setting **H** to **0**. Click **continue.***
2. *You will see a prompt in the command window informing you that the code has **LOST STABILITY***

In this example the amplitude of the wave became large and the system became numerically unstable. According to Eq 1.15 we should make the time-step smaller, or the box size smaller. Making the box size smaller will limit the amount of physics we can capture, so we opt for the time-step option. Since the time-step Δv is linked to the conjugate velocity step Δs, the only way to make the time-step smaller is make setting **C** as low is it can be (i.e **1**) and then to increase the number of s points. Hence when we changed setting **A** to **8001** the system stabilised. Changing this setting also increased the effective velocity range of the simulation, which seems somewhat wasteful if we don't have moving holes and clumps, but this is the price we pay for avoiding interpolation. Since the system is strongly nonlinear many harmonics of the distribution function are required. We can now understand why changing setting **E** to **30** had a large effect.

A general guide to the number of harmonics in the strongly nonlinear regime is the following

$$N = 4\omega_B s_{\max} / \pi^2 \gamma_L \tag{1.17}$$

Holes and clumps are an example of a strongly nonlinear regime. It is therefore recommended that the user of the BOT code use at least the default settings for studying them. These default settings are a compromise between speed and accuracy. In this way the user can get a feel for the physics very quickly and then if the user wants to be more precise then they can be.

6- VISUALISATION

The first note about visualisation is that it does slow the code down. This is why the plots are only produced at certain regular intervals which the user can define, rather than at every time-step. The longer the simulation time the less of an effect the visualisation has on the total computer run time. But for a quick simulation the run time is very different with and without the visualisation.

EX16

1. Run *EX14* with the visualisation setting **L** set to **0**
2. Compare the run time with the *EX14* case

Now let's look in more detail at the visualisation options available in the BOT code.

EX17

1. Decrease the damping by changing setting **H** to **0.7**. Change setting **L** to **2.** Click **continue**
2. Decide on how many **frames** you will see in the resulting visualisation. Use the default for now and click **OK**
3. Decide whether you want to see a spectrogram at the end of the simulation
4. Select **YES** to phase space visualisation. This will go on to produce contours of the distribution function.
5. Here you can choose the range of the axes for the upcoming plots. The y axis is effectively velocity measured from the resonance and the x axis is space measured in units of normalised wavelength (2π is a single wavelength). Use the default values for now, and Click **OK**

You will see holes and clumps after a short time. You may wish to make the plots more continuous, and to do this you would simply increase the number of frames in the visualisation, try **500**.

EX18

1. Change setting **L** to **2.** Click **continue**
2. Decide on how many **frames** you will see in the resulting visualisation. Use the default for now and Click **OK**
3. Decide whether you want to see a spectrogram at the end of the simulation
4. Select **NO** to phase space visualisation.
5. You can now decide whether you would like to see the amplitude in your plots. Choose **NO** for now.
6. You can now decide whether you would like to see the distribution function in your plots. Select **YES**
7. You then have a choice of whether you want to see the Fourier transformed $\mathcal{G}_n(s,\tau)$, or if you wish to see a Real space plot that will show the spatially averaged part of the perturbation to the distribution function $G_0(u,\tau)$. Select **Fourier Space**.
8. Now you decide which harmonic you wish to look at. Choose the default **1** for now.
9. The harmonics in general have real and imaginary parts. Choose the **real part**.

You will see a single plot generated where you can see the discontinuity that arises in the distribution function due to the delta function from the equilibrium part that was discussed earlier in the Manual.

REFERENCES

[1] H. L. Berk, B. N. Breizman, and M. Pekker, Phys. Rev. Lett. **76**, 1256(1996)

[2] B. N. Breizman, H. L. Berk, M. S. Pekker, F. Porcelli, G. V. Stupakov, and K. L. Wong, Phys. Plasmas **4**, 1559 (1997)

[3] M. K. Lilley, B. N. Breizman, and S. E. Sharapov, Phys. Rev. Lett. **102**, 195003 (2009)

[4] M. K. Lilley, B. N. Breizman, and S. E. Sharapov, Phys. Plasmas. Lett. **17**, 092305 (2010)