One-Dimensional Models of Slugging in Long Subsea Pipes

G14DIS

Mathematics 4th Year Dissertation

2015/16

School of Mathematical Sciences

University of Nottingham

Will Feavyour

Supervisor: Professor J Billingham

Project code: JB D2

Assessment type: Investigation

I have read and understood the School and University guidelines on plagiarism. I confirm that this work is my own, apart from the acknowledged references.

Contents

1 Introduction **3**

2 Roll Waves **5**

3 Two-layer Flow Equations **7**

4 Linear Stability **13**

5 Numerical Solutions **18**

6 Periodic Travelling Waves **26**

7 Numerical analysis of the limit cycle **38**

8 Conclusions **51**

Appendix A Matlab code: KT Pipe Simulation **56**

Appendix B Matlab code: BVP Solver **58**

1 Introduction

Slugging is a symptom of water building up in gas pipelines, and causes flow and pressure irregularities. Being able to predict the formation of water slugs is of interest to parties such as offshore natural gas field operators, who transport their gas through seafloor pipelines which also contain some small amount of water. This report aims to reproduce and explore, in particular, the results found in [17] by Needham et al. which studies slug development in two-layer flows.

Figure 1.1 shows a slug developing from some thin layer of condensate running along the bottom of a pipe. Originally the two layer gas and liquid flow is moving with some small perturbations at the fluid interface. These perturbations can become unstable and grow into a larger wave form which, when it fills the height of the pipe, is commonly called a slug. The complete formation and propagation of slugs is not analysed here, but is an active area of research in both mathematics and engineering. Instead, this paper examines the stability and evolution of perturbations to the flow which may ultimately become slugs. This can be thought of as the transitioning middle stage in the process of slug development. Partial differential equations derived from the Navier-Stokes equations

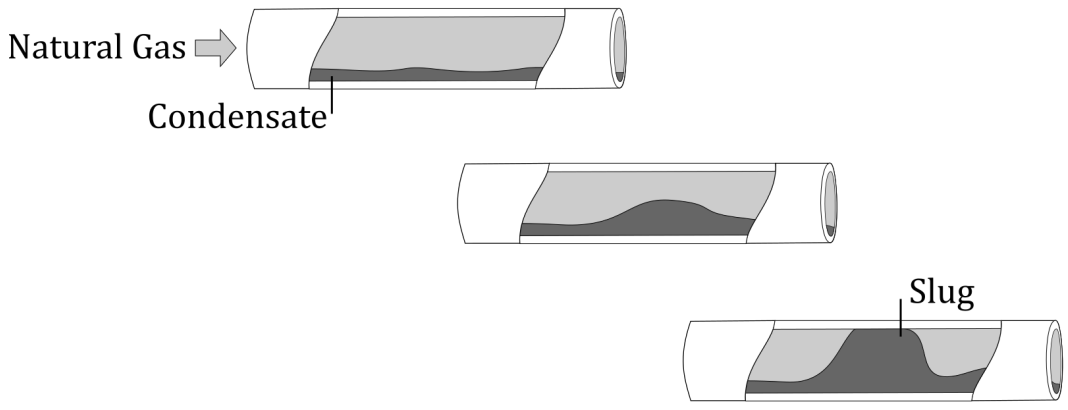


Figure 1.1: Evolution of a Slugging State

can be used to model the flows, with this study revolving around a one dimensional

simplification. The well established Shallow Water Equations (SWEs) are a related system of which this study is an extension. The SWEs have no shear terms, so to model the influence of the gas layer and pipeline wall on the liquid layer extra source terms are needed. This report begins by laying a route from a set of momentum and continuity equations to a pair of nondimensionalised flow equations for the liquid layer.

Waves forming at the interface can have a regular saw-tooth profile. They are easily likened to roll waves, which are explored more in section 2 as an introduction to some flow characteristics we expect to emerge in the rest of the report. Parallels are drawn between slugging flows and roll waves. By linearising the flow equations which are developed and non-dimensionalised in section 3, some instability criteria (similar to that for roll waves) are derived for perturbations to the uniform flow. In the derived equations, the Froude number describes some forcing in the system. Section 4 shows how a linear stability condition can be written in terms of this Froude number, and how varying it can greatly influence the upstream growth of waves.

As this model stems from the Navier-Stokes equations, exact solutions are elusive. Instead, numerical differentiation techniques are used to create simulations which, alongside analytical reasoning, give an insight into the slugging problem. Methods are discussed in section 5 for the spatial and temporal discretization, then a high order space-time method is implemented using MATLAB. This problem is first tackled for the homogeneous SWEs, then extended to take account of the extra source terms of importance to this problem.

A theory of periodic travelling waves is built up in section 6, and some extensive phase plane analysis is carried out. It aims to establish conditions for the existence of periodic solutions, which mirror the results seen in previous sections. Again there is some discussion of numerical and coding techniques to find approximate numerical solutions to the then established equations. In particular, some methodologies are developed to best utilise MATLAB's built in ode45 and bvp5c functions. This is the basis for section 7 where the behaviour of the periodic waves is analysed numerically.

2 Roll Waves

Before we look at the flow equations for the pipeline slugging problem, it is informative to briefly look at the behaviour of roll waves. Roll waves are an example of periodic travelling wave solutions to nonlinear partial differential equations. They arise when modelling a thin fluid flow, where the flow equations use some shallow water assumptions which are also used in section 3 for pipeline flow. Dressler [6] discusses in detail the formation of roll waves in a thin fluid flow with some resistive forces from the bed or channel. He derives discontinuous periodic solutions to the problem from the joining together of continuous solutions, causing shocks. It can be shown that the crests of long waves move faster downstream than the troughs, and so steepening wave fronts are formed. Lighthill and Whitham [15] show this, with the formation discontinuities in the solution being exactly the same as the formation of shock waves in a gas, hence why the term shock is used. In the extreme case, the steepening wave fronts form the shocks found in the solutions by Dressler.

Roll waves often occur naturally in open channel inclined flows, and although the pipeline study here only deals with a horizontal scenario there is some overlap in the theory. Hanratty and Hershman [8] note that there are several studies relating the open channel flow formation of roll waves to the two layer flow of gas and a liquid film. This film layer can be represented by the approximation that the fluid depth to channel depth ratio tends to zero, made in section 3 to arrive at equation (3.22). Hanratty and Hershman describe in the same paper how they experimented with the transition between a steady flow and roll wave flow in a laboratory setting, following from work by Hanratty and Engen [9]. These prior experiments generated regular ripple-like waves which filled the fluid region. This is a tell-tale characteristic of roll wave flows, tending to form the previously described saw-tooth like sharp waves in a very periodic way. A depiction of roll waves forming on an inclined channel can be seen in figure 2.1, alongside the analogous

formation because of some forcing gas in a pipe or channel.

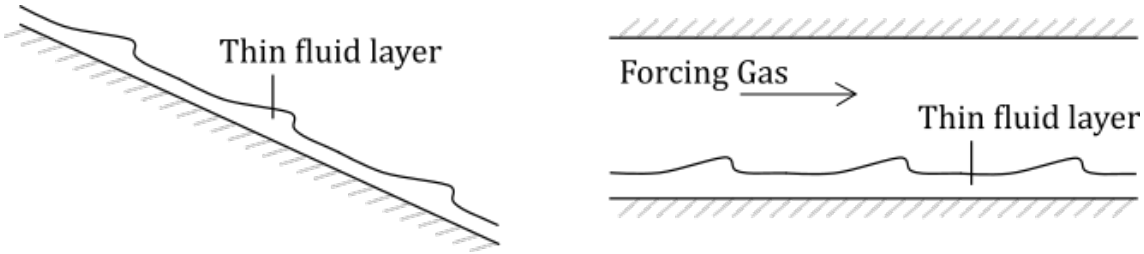


Figure 2.1: Roll wave flows for an inclined open channel (left) and pipeline flow (right)

Hanratty et al. suggest in these studies that unsteady flows can occur at much lower Reynolds numbers in the pipeline set-up than we would expect for a similar flow in an open channel. The flow speeds were increased and the Froude number measured empirically, in particular the effects of different gas flow rates were investigated. At the point of roll waves being formed, the Froude number was higher in slower, more stable flows. A hypothesis gained by imagining the converse of this is that a flow with a higher Froude number will give a more unstable flow. This is investigated in section 4, where linearisation shows it to be true.

Kadri et al. [10] discuss the different flow conditions governing whether roll waves or a slugging flow is created. The long waves common to both flows grow in two ways, vertically or horizontally. Vertical growth causes waves to travel faster and with a rising crest. The result from these experiments is corroborated numerically in section 7.2 of this report. The vertical growth continues at high enough wave velocities, until suction forces associated with the Bernoulli effect overcome the gravitational forces and a slug may form. Horizontal growth causes the crests to propagate faster than the average fluid speed, so roll waves form. Kadri et al. investigated the time required for each of these growth cases to come to its conclusion, and then which occurs first in a given flow.

An investigation into roll waves down an inclined channel by Needham and Merkin [18] has a lot of similar results to those developed later on in this study. Starting with a general thin fluid flow down an inclined channel, they derive a stability condition which is also

found in section 4 for pipeline flow. They also conduct phase plane analysis for periodic solutions, comparable to that in section 7. This is the final example which justifies taking cues from roll wave theory in later parts of this investigation, and shows how the two areas of study are closely interlinked.

3 Two-layer Flow Equations

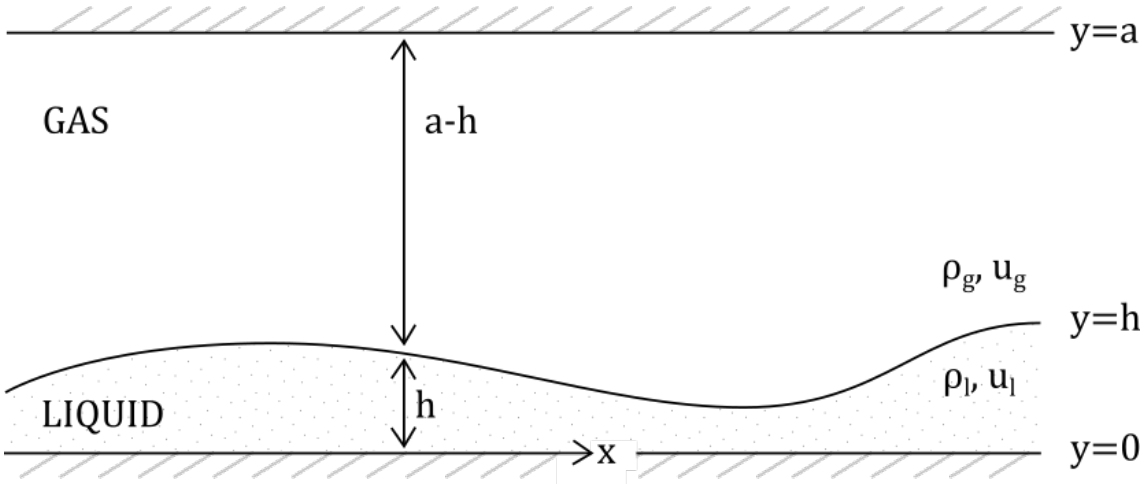


Figure 3.1: Definitions of quantities in flow equations

To begin with, there is a need to understand the flow under consideration. Figure 3.1 shows the cross-section of a rectangular channel, with some of the physical quantities used in the flow equations labelled, which will be used for the one dimensional model. The one dimensional hydraulic equations of motion in a channel are applicable to this set up. Whilst not accounting for the possible effects of flow in a circular pipeline, this much simpler model can be used to establish some valid theory and predictions for the full pipeline problem. The densities are ρ_l and ρ_g for the liquid and fluid layers respectively. Also the respective vertically averaged velocities are u_l and u_g. The vertical averaging uses similar assumptions to those used in the derivation of SWEs [3]. To get the depth

averaged velocity u_l, take the non averaged $\tilde{u_l}(y)$ and define

$$u_l = \frac{1}{h}\int_0^h \tilde{u_l}(y) \ \mathrm{d}y. \tag{3.1}$$

Next we can write down the equations of continuity and momentum, similar to those outlined by Needham et al. in [17], for both layers. The acceleration due to gravity is g, p_l and p_g are total pressures, $\bar{p}_l$ and $\bar{p}_g$ are dynamic pressures, μ_l and μ_g are constant eddy viscosities. The dimensionless Chézy coefficients between the walls and fluids are c_l and c_g, and at the interface is c_I. The Chézy formula for dimensional coefficients is

$$C = v/\sqrt{RS}, \tag{3.2}$$

where C is the coefficient, v the average velocity, R the hydraulic radius ($\sim$ water depth) and S the slope ratio. This can be derived analytically from an energy loss theory [11]. However, commonly used to determine the Chézy coefficient is the Manning formula

$$C = R^{1/6}/M, \tag{3.3}$$

where M is the Manning coefficient. These Manning coefficients are empirical measures of roughness, actual measurement of which is hard and one of the reasons why physically replicating theory can be difficult. The measurement is described as more of an art than a science by the previous reference, but detailed tables of collected values can be found, for example those by Chow [5] with accompanying exemplar illustrations. Dressler [6] notes that although these formulas are commonly used for non-uniform flows, they are strictly speaking valid only for uniform flows. This means drag terms approximated using the Chézy formula will only be accurate for flows which vary slowly in x and t. After the non-dimensionalisation of the flow equations in this section, there is a note on why this restriction is applicable to the slugging problem.

In the following equations, note that a subscript x or t denotes a partial derivative, whilst l and g differentiate between the liquid or gas layers respectively. The six flow

equations (three each for the liquid and gas flows) are

$$h_t + (hu_l)_x = 0, \tag{3.4}$$

$$(a-h)_t + ((a-h)u_g)_x = 0, \tag{3.5}$$

$$\rho_l \left\{(hu_l)_t + (hu_l^2)_x\right\} = -h\bar{p}_{lx} - \rho_l c_l u_l^2 + \rho_g c_I (u_g - u_l)|u_g - u_l| + \mu_l (hu_{lx})_x, \tag{3.6}$$

$$\begin{aligned}\rho_g \left\{((a-h)u_g)_t + ((a-h)u_g^2)_x\right\} \\ = -(a-h)\bar{p}_{gx} - \rho_g c_g u_g^2 - \rho_g c_I (u_g - u_l)|u_g - u_l| + \mu_g ((a-h)u_{gx})_x,\end{aligned} \tag{3.7}$$

$$p_l = -\rho_l g y + \bar{p}_l, \tag{3.8}$$

$$p_g = -\rho_g g y + \bar{p}_g. \tag{3.9}$$

At the interface between the fluids $y = h$, there must be continuity of pressure $p_g = p_l$. Using this to couple (3.8) and (3.9), we get

$$\bar{p}_l - \bar{p}_g = (\rho_l - \rho_g) g h. \tag{3.10}$$

Over this section, the aim is to derive a pair of nondimensionalized equations (which govern the liquid flow) from the five equations (3.4) - (3.7) and (3.10). Most of the work for this, clearly, will revolve around equations (3.6) and (3.7). To start, it is useful to combine them in such a way that the pressure terms are eliminated. This can be achieved

by subtracting (3.7) from $((a/h)-1)$ times (3.6), to obtain

$$\left(\frac{a}{h}-1\right)\rho_l\left\{(hu_l)_t+(hu_l^2)_x\right\}-\rho_g\left\{((a-h)u_g)_t+((a-h)u_g^2)_x\right\}$$

$$=-\left(\frac{a}{h}-1\right)h\bar{p}_{lx}+(a-h)\bar{p}_{gx}-\left(\frac{a}{h}-1\right)\rho_l c_l u_l^2+\rho_g c_g u_g^2$$

$$+\frac{a}{h}\rho_g c_I(u_g-u_l)|u_g-u_l|+\left(\frac{a}{h}-1\right)\mu_l(hu_{lx})_x-\mu_g((a-h)u_{gx})_x. \quad (3.11)$$

It can be seen that the motivation behind using the factor $((a/h)-1)$ is so that the pressure terms can be combined by

$$-\left(\frac{a}{h}-1\right)h\bar{p}_{lx}+(a-h)\bar{p}_{gx}=-(a-h)(\bar{p}_l-\bar{p}_g)_x. \quad (3.12)$$

Also balancing terms gives the relation

$$\rho_l c_l u_l^2=\mathcal{O}\left(\rho_g c_g u_g^2\right),$$

and therefore

$$\left(\frac{u_l}{u_g}\right)^2=\mathcal{O}\left(\frac{\rho_g}{\rho_l}\right).$$

In [17] it is stated without justification that the gas-liquid density ratio ρ_g/ρ_l is small. This leads to the simplification that within the interfacial shear term, u_g dominates and the approximation

$$(u_g-u_l)|u_g-u_l|\sim u_g^2 \quad (3.13)$$

can be made. In generating slugs with physical experiments, Kadri et al. [10] used a set up with density ratio of approximately 10^{-3}. This would support the simplification made in (3.13). However, later in this paper (as in [17]) the density ratio used is of order 10^{-1}, which an arguably large value for this simplification. The closer to one the ratio is, the more this model will overestimate the influence of the interfacial shear, since the term should vanish at $u_l=u_g$ or $\rho_g=\rho_l$. In fact the crude estimate can be made that if $\rho_g/\rho_l=\mathcal{O}(10^{-1})$ then $u_g/u_l=\mathcal{O}(10^{-1/2})$ and

$$(u_g-u_l)|u_g-u_l|\sim u_g^2(1-\mathcal{O}(10^{-1/2}))\sim\frac{1}{2}u_g^2.$$

We will proceed by making the approximation (3.13), but keeping in mind that there is a upper limit to the density ratio for this approximation to be relevant. By also using (3.10), equation (3.11) becomes

$$\begin{aligned}\left(\frac{a}{h}-1\right)\rho_l\left\{(hu_l)_t+(hu_l^2)_x\right\}-\rho_g\left\{((a-h)u_g)_t+((a-h)u_g^2)_x\right\}\\ =-(a-h)((\rho_l-\rho_g)gh)_x-\left(\frac{a}{h}-1\right)\rho_l c_l u_l^2+\rho_g c_g u_g^2\\ +\frac{a}{h}\rho_g c_I u_g^2+\left(\frac{a}{h}-1\right)\mu_l(hu_{lx})_x-\mu_g((a-h)u_{gx})_x. \qquad (3.14)\end{aligned}$$

Now introduce the dimensionless scales

$$h=h_0h', \qquad u_g=u_g^0u', \qquad u_l=u_l^0v', \qquad (3.15)$$

and use them to introduce the additional dimensionless parameters

$$\begin{aligned}&F_0=\frac{u_l^0}{\sqrt{gh_0}}, && x=\frac{h_0}{c_l}x', && t=\frac{h_0}{c_lu_l^0}t',\\ &\bar{p}_l=\frac{p_0h_0}{c_l}\bar{p}_l', && \bar{p}_g=\frac{p_0h_0}{c_l}\bar{p}_g', && \lambda=\frac{c_Ia}{c_gh_0}, \qquad (3.16)\\ &R_l=\frac{h_0\rho_lu_l^0}{\mu_lc_l}, && \rho=\frac{\rho_g}{\rho_l}, && \epsilon=\frac{h_0}{a}.\end{aligned}$$

Of these, some key physical interpretations are as follows. The Froude number F_0 relates to the upstream flow rate and ρ is the density ratio discussed previously. The effective Reynolds number $R_l=\mathcal{O}(10^2)$ is a measure of the ratio of inertial and viscous forces. For the turbulent liquid flow in the pipeline, the actual Reynolds number is orders of magnitude larger, for example $\mathcal{O}(10^5)$. Also λ measures the relative importance of the interfacial shear to flow resistance, while p_0 (which is used for the scaling of $\bar{p}_l'$ and $\bar{p}_g'$) is a positive constant reference pressure gradient.

When interpreting the later numerical results practically, these dimensionless scales are important to remember, as the time and distance units of the plots have little resemblance

of the physical system they model. Both scales are compressed in their dimensionless forms, for instance the pipelines being modelled are kilometres long but the dimensionless x makes them appear jagged over a small range in the numerical simulations. Part of the motivation behind all of this theorising and the numerical simulations is that it is difficult to set up physical experiments to model such large systems. In fact the possible slugging flows develop very slowly and over great distances. This makes the use of the Chézy formula for uniform flows valid, as previously mentioned.

Moving forward, only the dimensionless equations will be considered, so for convenience discard the primes of the dimensionless variables. After substitution of (3.13), (3.15) and (3.16) into (3.14) and some manipulation we arrive at

$$\begin{aligned}\frac{1}{h}(1-\epsilon h)\left\{(hv)_t+(hv^2)_x\right\}-\rho\left\{\frac{u_g^0}{u_l^0}((1-\epsilon h)u)_t+\left(\frac{u_g^0}{u_l^0}\right)^2((1-\epsilon h)u^2)_x\right\}\\ =-\frac{(1-\rho)}{F_0^2}(1-\epsilon h)h_x-(1-\epsilon h)\frac{v^2}{h}+\left(\frac{\epsilon\rho c_g}{hc_l}\right)\left(\frac{u_g^0}{u_l^0}\right)^2(h+\lambda)u^2\\ +\frac{1}{R_l h}(1-\epsilon h)(hv_x)_x-\frac{u_g^0\mu_g c_l}{\rho_l h_0(u_l^0)^2}((1-\epsilon h)u_x)_x. \qquad (3.17)\end{aligned}$$

It is not within the scope of this study to look at when a slug is fully formed, more pertinent is when instabilities arise and how they develop. Because of this, a simplification can be made by letting $\epsilon \to 0$, which from the definition of ϵ represents the fluid layer depth being much smaller than the channel height. This also relates to roll wave theory for a thin fluid layer as discussed previously. As this step is made, there are three things to consider to simplify the final form of (3.17).

Firstly, h and v have been scaled such that $h = v = 1$ is the equilibrium solution. At the equilibrium, equation (3.17) becomes

$$0=-(1-\epsilon)+\left(\frac{\rho c_g}{hc_l}\right)\left(\frac{u_g^0}{u_l^0}\right)^2(h+\lambda),$$

which can be rearranged to give the relation

$$\left(\frac{\epsilon\rho c_g}{c_l}\right)\left(\frac{u_g^0}{u_l^0}\right)^2 = \frac{1-\epsilon}{1+\lambda}. \tag{3.18}$$

This will be a useful relation for simplifying the third term on the right hand side of (3.17). Secondly, using (3.4) it can be seen that

$$\begin{aligned}\frac{1}{h}((hv)_t + (hv^2)_x) &= \frac{1}{h}(hv_t + h_t v + (hv)_x v + hvv_x)\\ &= v_t + \frac{v}{h}(h_t + (hv)_x) + vv_x\\ &= v_t + vv_x. \end{aligned}\tag{3.19}$$

Lastly, from nondimensionalizing (3.5),

$$-\frac{u_l^0}{u_g^0}(1-\epsilon h)_t + ((1-\epsilon h)u)_x = 0. \tag{3.20}$$

Now, taking $\epsilon \to 0$, equation (3.20) implies that $u_x = 0$. With $u \to 1$ as $x \to -\infty$ it must be the case that $u = 1$ for all x.

Using the results (3.18) - (3.20), we arrive at the final form of the momentum equation

$$v_t + vv_x + \frac{(1-\rho)}{F_0^2}h_x = \frac{h+\lambda}{(1+\lambda)h} - \frac{v^2}{h} + \frac{1}{R_l h}(hv_x)_x, \tag{3.21}$$

as $\epsilon \to 0$. It is much more simple to arrive at a nondimentionalized form of (3.4). By just substituting the new variables and rearranging we get

$$h_t + (hv)_x = 0. \tag{3.22}$$

These final two equations form the basis of the rest of this investigation.

4 Linear Stability

4.1 Linearised Equations

Guidance can be taken from Needham and Merkin [18] and Whitham [22] in analysing the linear stability of the flow equations. Assuming small amplitude disturbances in h

and v from the uniform state, with $\alpha \ll 1$ and $\bar{h}, \bar{v} = \mathcal{O}(1)$, we can write

$$h(x,t) = 1 + \alpha\bar{h}(x,t), \quad v(x,t) = 1 + \alpha\bar{v}(x,t). \tag{4.1}$$

Substituting into the flow equations (3.22) and (3.21) gives

$$\alpha\bar{h}_t + \alpha(\bar{h} + \bar{v})_x + \alpha^2(\bar{h}\bar{x})_x = 0, \tag{4.2}$$

$$\begin{aligned}\alpha\bar{v}_t + \alpha^2\bar{h}\bar{v}_t + \alpha\bar{v}_x + \alpha^2\bar{v}_x(\bar{h} + \bar{v}) + \alpha^3\bar{h}\bar{v}\bar{v}_x + \frac{(1-\rho)}{F_0^2}\left(\alpha\bar{h}_x + \alpha^2\bar{h}\bar{h}_x\right) \\ = \frac{1 + \alpha\bar{h} + \lambda}{(1+\lambda)} - \left(1 + 2\alpha\bar{v} + \alpha^2\bar{v}^2\right) + \frac{1}{R_l}\left(\alpha\bar{v}_{xx} + \alpha^2(\bar{h}\bar{v}_x)_x\right),\end{aligned} \tag{4.3}$$

so discarding terms $\mathcal{O}(\alpha^2)$ leads to

$$\bar{h}_t + (\bar{h} + \bar{v})_x = 0, \tag{4.4}$$

$$\bar{v}_t + \bar{v}_x + \frac{(1-\rho)}{F_0^2}\bar{h}_x = \frac{\bar{h}}{1+\lambda} - 2\bar{v} + \frac{1}{R_l}\bar{v}_{xx}. \tag{4.5}$$

Then $\bar{v}$ can be eliminated by differentiating and combining the two equations, leaving one equation in terms of $\bar{h}$. Using (4.4), we find that

$$\bar{v}_x = -\bar{h}_t - \bar{h}_x, \quad \bar{v}_{xt} = -\bar{h}_{tt} - \bar{h}_{xt}, \quad \bar{v}_{xx} = -\bar{h}_{tx} - \bar{h}_{xx}. \tag{4.6}$$

Taking $\partial/\partial x$ of (4.5) gives

$$\bar{v}_{xt} + \bar{v}_{xx} + \frac{(1-\rho)}{F_0^2}\bar{h}_{xx} = \frac{\bar{h}_x}{1+\lambda} - 2\bar{v}_x + \frac{1}{R_l}(\bar{v}_x)_{xx},$$

hence, by substitution of (4.6),

$$\bar{h}_{tt} + 2\bar{h}_{xt} + \bar{h}_{xx} - \frac{(1-\rho)}{F_0^2}\bar{h}_{xx} + \frac{\bar{h}_x}{1+\lambda} + 2(\bar{h}_t + \bar{h}_x) = \frac{1}{R_l}(\bar{h}_t + \bar{h}_x)_{xx}, \tag{4.7}$$

which will be analysed to give a linear stability condition.

4.2 Stability Condition

A solution to equation (4.7) can be sought of the form $\bar{h} = A(k)e^{ikx-\omega t}$, with $\omega \in \mathbb{C}$ and the wavenumber k. This gives

$$\omega^2 - \omega\left(2ik + 2 + \frac{k^2}{R_l}\right) - k^2\left(1 - \frac{(1-\rho)}{F_0^2}\right) + ik\left(2 + \frac{1}{(1+\lambda)} + \frac{k^2}{R_l}\right) = 0 \tag{4.8}$$

As $\omega \in \mathbb{C}$, it can be defined as $\omega = \omega_r + i\omega_i$ where $\omega_r, \omega_i \in \mathbb{R}$. Stability requires that $\omega_r > 0$, since otherwise the perturbation will grow exponentially in time. Substituting into (4.8) and considering the real and imaginary parts, (4.9) and (4.10) respectively, we obtain

$$\omega_r^2 + \omega_r\left(-2 - \frac{k^2}{R_l}\right) - \omega_i^2 + 2\omega_i k - \left(1 - \frac{(1-\rho)}{F_0^2}\right)k^2 = 0 \tag{4.9}$$

$$\omega_i\left(2\omega_r - 2 - \frac{k^2}{R_l}\right) - \left(2\omega_r k - \left(2 + \frac{1}{(1+\lambda)}\right)k - \frac{k^3}{R_l}\right) = 0 \tag{4.10}$$

These two equations can be combined to eliminate ω_i, producing a fourth-order polynomial in ω_r,

$$\begin{aligned} P(\omega_r, k) = &\left(\omega_r^2 + \omega_r\left(-2 - \frac{k^2}{R_l}\right) - \left(1 - \frac{(1-\rho)}{F_0^2}\right)k^2\right)\left(2\omega_r - 2 - \frac{k^2}{R_l}\right)^2 \\ &+ 2k\left(2\omega_r k - \left(2 + \frac{1}{(1+\lambda)}\right)k - \frac{k^3}{R_l}\right)\left(2\omega_r - 2 - \frac{k^2}{R_l}\right) \\ &- \left(2\omega_r k - \left(2 + \frac{1}{(1+\lambda)}\right)k - \frac{k^3}{R_l}\right)^2 \\ = &\ 0, \end{aligned}$$

which is also dependant on k. This admits the simplification

$$\begin{aligned} P(\omega_r, k) = &\left(\omega_r^2 + \omega_r\left(-2 - \frac{k^2}{R_l}\right) + \frac{(1-\rho)}{F_0^2}k^2\right)\left(2\omega_r - 2 - \frac{k^2}{R_l}\right)^2 \\ &- k^2\left[\left(2\omega_r - \left(2 + \frac{1}{(1+\lambda)}\right) - \frac{k^2}{R_l}\right) - \left(2\omega_r - 2 - \frac{k^2}{R_l}\right)\right]^2, \end{aligned}$$

where the square bracketed term simplifies greatly to leave

$$P(\omega_r, k) = \left(\omega_r^2 + \omega_r\left(-2 - \frac{k^2}{R_l}\right) + \frac{(1-\rho)}{F_0^2}k^2\right)\left(2\omega_r - 2 - \frac{k^2}{R_l}\right)^2 - \frac{k^2}{(1+\lambda)^2}. \quad (4.11)$$

To solve $P(\omega_r, k) = 0$ for ω_r is a non-trivial task. Reasoning can be applied that only the

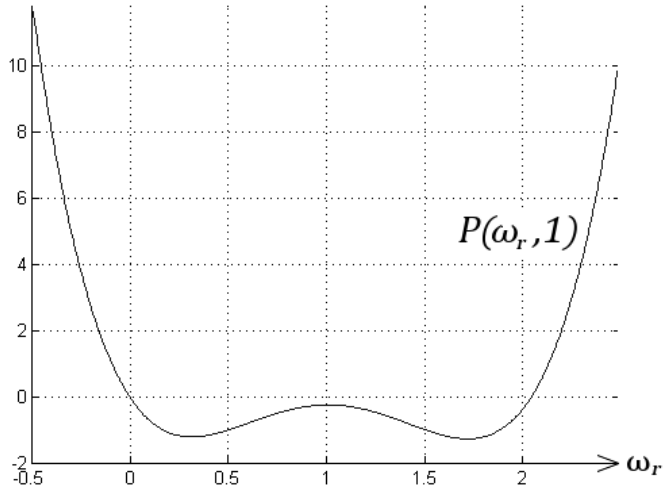


Figure 4.1: Representative plot of Equation (4.11) for $k = 1$

neutral curve $P(0, k)$ needs to be looked at, as $\omega_r = 0$ is the critical point between $\omega_r > 0$ and $\omega_r < 0$. With some interesting insights of $P(\omega_r, k)$ though, a condition to enforce $\omega_r > 0$ can be confidently established. It may be helpful to see figure 4.1 to visualise the following reasoning.

Firstly, it can be noted that as the ω_r^4 term has a positive sign, so $P(\omega_r, k)$ is a positively valued, monotonically decreasing, function for large negative ω_r. Next it is informative to find the turning points, and so it is convenient that $\partial P(\omega_r, k)/\partial\omega_r = 0$ can be solved to find that they occur at three positive values. The first two (one each for the positive and negative square root term) are given by

$$\omega_r = 1 + k^2(2R_l)^{-1} \pm \left[(2 + k^2R_l^{-1})^2/8 - k^2(1-\rho)F_0^{-2}/2\right]^{1/2}$$

$$= \tfrac{1}{4}\left(2(2 + k^2R_l^{-1}) \pm \left[2(2 + k^2R_l^{-1})^2 - 8k^2(1-\rho)F_0^{-2}\right]^{1/2}\right) > 0, \quad (4.12)$$

and the third is

$$\omega_r = 1 + k^2(2R_l)^{-1} > 0. \tag{4.13}$$

For linear stability, we want all values of ω_r which satisfy $P(\omega_r, k) = 0$ to be positive. From evaluating $P(\omega_r, k)$ at the turning points (4.12) and (4.13) it can be determined that there are two roots. The minimum of these, say ω_1, must lie between $-\infty$ and some turning point, which are all greater than 0. Therefore a necessary and sufficient condition that ω_1 is positive, and therefore both roots are, is that $P(0, k) \geq 0$. Accordingly, we can now look at the neutral curve

$$P(0,k) = -k^2 \left[C_+ \left(2 + \frac{k^2}{R_l} \right) - \left(2C_0 + \frac{k^2}{R_l} \right) \right] \left[C_- \left(2 + \frac{k^2}{R_l} \right) - \left(2C_0 + \frac{k^2}{R_l} \right) \right], \tag{4.14}$$

where

$$C_+ := 1 + \frac{\sqrt{(1-\rho)}}{F_0}, \quad C_- := 1 - \frac{\sqrt{(1-\rho)}}{F_0}, \quad C_0 := 1 + \frac{1}{2(1+\lambda)}. \tag{4.15}$$

Then we can look for a stability condition relating k and F_0. Using $P(0, k) \geq 0$,

$$\left[C_+ \left(2 + \frac{k^2}{R_l} \right) - \left(2C_0 + \frac{k^2}{R_l} \right) \right] \left[C_- \left(2 + \frac{k^2}{R_l} \right) - \left(2C_0 + \frac{k^2}{R_l} \right) \right] \leq 0. \tag{4.16}$$

Of the two square bracketed terms, one must be negative and one positive. As $C_+ > C_-$, we have that

$$C_+ \left(2 + \frac{k^2}{R_l} \right) - \left(2C_0 + \frac{k^2}{R_l} \right) \geq 0,$$

which by substitution of (4.15) becomes

$$\left(1 + \frac{\sqrt{(1-\rho)}}{F_0} \right) \left(2 + \frac{k^2}{R_l} \right) \geq 2 + \frac{1}{(1+\lambda)} + \frac{k^2}{R_l}.$$

Since we are interested in this condition in terms of the Froude number F_0, a final rearrangement leaves

$$(1-\rho)(1+\lambda)^2 \left(2 + k^2 R_l^{-1} \right)^2 \geq F_0^2.$$

This stability inequality can be seen as a plot of F_0 against k in figure 4.2. The system is more unstable for lower wavenumbers, so letting $k = 0$ gives a sufficient condition that

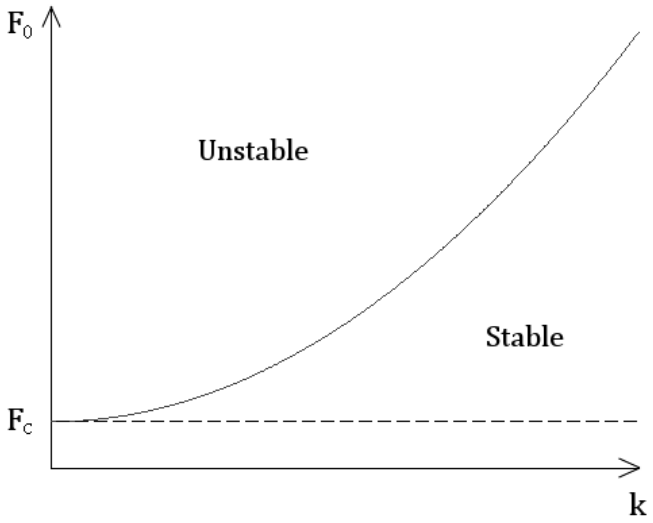


Figure 4.2: F_0 vs k, showing linear stability regions

the linear stability is always satisfied. Define this value of F_0 as the critical Froude number F_c, such that for an unconditionally linearly stable flow, F_0 satisfies

$$F_c^2 = 4(1-\rho)(1+\lambda)^2 \geq F_0^2. \tag{4.17}$$

Returning to ideas of slugging from the outset of this project, it can be proposed that slugs may develop when this stability condition is not satisfied. It would be expected that for large enough F_0, a given small perturbation would evolve into a large wave which could become a slug. Conversely, a small enough F_0 would see the perturbation decay and the flow remain stable. In the hopes of observing this, the focus of this study turns to numerical solutions of the full nonlinear system.

5 Numerical Solutions

Equations (3.22) and (3.21), can be written as the analogous single vector equation

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} = R(U). \tag{5.1}$$

Here U, $F(U)$ and $R(U)$ are defined as

$$U = \begin{bmatrix} h \\ q \end{bmatrix}, \quad F(U) = \begin{bmatrix} q \\ q^2 h^{-1} + \frac{1}{2}(1-\rho)F_0^{-2}h^2 \end{bmatrix}, \tag{5.2}$$

with $q = vh$, and

$$R(U) = \begin{bmatrix} 0 \\ (1+\lambda)^{-1} + \lambda((1+\lambda)h)^{-1} - v^2 h^{-1} + (R_l h)^{-1}[hv_x]_x \end{bmatrix}. \tag{5.3}$$

This system needs to be discretized in both space and time. Let Δx be the grid spacing between the discrete points in the x direction, then

$$U(x_j, t) = U_j(t) \quad \text{where} \quad x_j = j \cdot \Delta x \text{ for } j = 0, 1, 2, \ldots \tag{5.4}$$

Initially in the development of a full numerical solver, U remains continuous in t, and the method is only semi-discrete.

5.1 Kurganov and Tadmor (Homogeneous) Method

The Kurganov and Tadmor (KT) method [14] is an appropriate numerical solver, which can be fairly easy to implement for the homogeneous equation

$$U_t + F(U)_x = 0. \tag{5.5}$$

This simplification actually leaves the SWEs, with the extra source terms contained in $R(U)$ of the full system (5.1) being the key difference to simulating the pipeline problem. The source terms will be dealt with later on, for now the focus will be on applying the KT method to equation (5.5). It is a carefully developed second order central difference scheme, with deliberately small numerical viscosity. Developed initially as a semi-discrete scheme, it can be expressed as

$$\frac{\mathrm{d}}{\mathrm{d}t}U_j(t) = \frac{-1}{\Delta x}\left(H_{j+1/2}(t) - H_{j-1/2}(t)\right), \tag{5.6}$$

where

$$H_{j+1/2}(t) = \frac{F(U^+_{j+1/2}(t)) + F(U^-_{j+1/2}(t))}{2} - \frac{a_{j+1/2}(t)}{2}\left(U^+_{j+1/2}(t) - U^-_{j+1/2}(t)\right) \tag{5.7}$$

is the numerical flux, and the intermediate values $U^{\pm}_{j+1/2}$ are defined as

$$U^+_{j+1/2}(t) = U_{j+1}(t) - \frac{\Delta x}{2}(U_x)_{j+1}(t), \quad U^-_{j+1/2}(t) = U_j(t) + \frac{\Delta x}{2}(U_x)_j(t). \tag{5.8}$$

The maximal local speed $a_{j+1/2}$ is, strictly speaking, the maximum of the spectral radii of $\partial F/\partial U$ over the range of U between the two intermediate values (5.8). The spectral radius of a matrix being defined as its maximum absolute eigenvalue. However, in this application only the spectral radii *at* the intermediate values need to be considered, given by

$$a_{j+1/2} = \max\left\{\Lambda(U^+_{j+1/2}), \Lambda(U^-_{j+1/2})\right\}, \tag{5.9}$$

where $\Lambda(U_j)$ is the spectral radius of $\partial F/\partial U$ evaluated at at U_j. This can be directly calculated, with I as the identity matrix and eigenvalues $\hat{\Lambda}$, by

$$0 = \left|\frac{\partial F}{\partial U} - \hat{\Lambda} I\right| = \begin{vmatrix} -\hat{\Lambda} & 1 \\ -q^2h^{-2} + (1-\rho)F_0^{-2}h & 2qh^{-1} - \hat{\Lambda} \end{vmatrix}$$

$$= \hat{\Lambda}^2 - 2qh^{-1}\hat{\Lambda} + q^2h^{-2} - (1-\rho)hF_0^{-2}.$$

So the maximum absolute eigenvalue is

$$\Lambda = \frac{q}{h} + \left((1-\rho)\frac{h}{F_0^2}\right)^{1/2}. \tag{5.10}$$

Note that since the flow is upstream $v > 0$, and for physical sense $h > 0$, so $q = vh > 0$. Finally, the approximate derivatives $(U_x)_j$ are calculated with use of the *minmod* function

$$(U_x)_j = \frac{1}{\Delta x}\text{minmod}(U_j - U_{j-1}, U_{j+1} - U_j), \tag{5.11}$$

which essentially calculates the most shallow gradient over the neighbouring interval as

$$\text{minmod}(a, b) = \frac{1}{2}\left(\text{sign}(a) + \text{sign}(b)\right) \cdot \min\{|a|, |b|\}. \tag{5.12}$$

5.2 Time Discretization

Now requiring discretization in time for a complete numerical method, the new notation

$$U_j(t_n) = U_j^n \quad \text{where} \quad t_n = n \cdot \Delta t \text{ for } n = 0, 1, 2, ... \tag{5.13}$$

is introduced, where Δt is the time step between iterations.

Since this form of the scheme is yet to deal with the time derivative, there is scope to explore some different methods. Initially, it is good to look at the most simple method so that implementation of more complicated methods can be verified. For instance, the common first order Euler method

$$U_j^{n+1} = U_j^n - \frac{\Delta t}{\Delta x}\left(H_{j+1/2}(t) - H_{j-1/2}(t)\right) \tag{5.14}$$

would be a sensible starting point. However, using a first order temporal method might detract from the use of a second order spatial method. Shu and Osher [21] discuss using multi-level solvers, which have the advantages of being relatively simple to program as well as increasingly accurate. For ease, let the right hand side of (5.6) be defined as

$$L(U_j) := \frac{-1}{\Delta x}\left(H_{j+1/2}(t) - H_{j-1/2}(t)\right). \tag{5.15}$$

Then the first order Euler method becomes

$$U_j^{n+1} = U_j^n + \Delta t L(U_j^n).$$

The second order modified Euler method is an extension of this, using one correction step

$$(U_j^n)^1 = U_j^n + \Delta t L(U_j^n),$$

$$U_j^{n+1} = U_j^n + \frac{1}{2}\Delta t\left(L(U_j^n) + L((U_j^n)^1)\right),$$

where the intermediate step s is denoted $(U_j^n)^s$

Although Shu and Osher give a general formula for an m^{th} order method of this form, and explicitly state up to the fifth order case, there are diminishing returns on

implementing higher order methods at the expense of repeatedly computing $L(U_j)$. The third order case

$$(U_j^n)^1 = U_j^n + \Delta t L(U_j^n),$$

$$(U_j^n)^2 = U_j^n + \frac{1}{4}\Delta t \left[L(U_j^n) + L((U_j^n)^1)\right],$$

$$U_j^{n+1} = U_j^n + \frac{1}{6}\Delta t \left[L(U_j^n) + L((U_j^n)^1)\right] + \frac{2}{3}\Delta t L((U_j^n)^2),$$

is the limit of what was used in the study of this problem. It is worth noting that these methods require no storage of any U_j^{n-1}, which streamlines both the programming and storage requirements. Although for efficiency, not only the $(U_j^n)^s$ have to be stored but also the $L((U_j^n)^s)$ values which are reused. In the one dimensional model used here, these storage requirements are not detrimental. However for much larger systems, more consideration may have to be taken as to the cost of storing many large vectors. A MATLAB implementation of this last method can be seen in Appendix A.

Efficient coding is key in using this method. Because there are several intermediate vector equations, vector operations must be fully utilised instead of using loops to iterate over all j. This is not difficult to do in a mathematical programming language such as MATLAB, but worth being aware of. For comparison, substituting in just one loop to calculate all a_j in (5.9) can increase the computation time more than tenfold. Increasing the execution speed with better coding practices makes it more practical to use the higher order temporal methods alongside the spatial KT method, since it is then less expensive to compute $L(U_j)$.

5.3 Full Non-homogeneous Approximation

To use any solver previously tailored to solving a homogeneous equation like the SWEs, the term demanding particular attention is that containing the second order derivative

$[hv_x]_x$. Other terms in $R(U)$ do not require approximation. Using a second order central difference, the derivative term can be approximated using

$$[hv_x]_x \approx \frac{1}{\Delta x}\left[h_{i+\frac{1}{2}}\left(\frac{v_{i+1}-v_i}{\Delta x}\right) - h_{i-\frac{1}{2}}\left(\frac{v_i - v_{i-1}}{\Delta x}\right)\right]$$

$$= \frac{1}{2(\Delta x)^2}\left[(h_{i+1}+h_i)(v_{i+1}-v_i) - (h_i+h_{i-1})(v_i - v_{i-1})\right]. \tag{5.16}$$

Once the homogeneous step has been calculated (for instance with the KT method and a suitable time discretization) the approximated $R(U)$ term can be appended to the solution at each time step. An adapted version of the KT method or pre-existing packages such as the Conservation Laws Package (CLAWPACK, used in [17]) could also be used. CLAWPACK uses high-resolution Godunov-type methods, and requires an approximate Riemann solver - something which has been avoided by using the KT method.

5.4 Periodic Boundary Conditions

To best simulate a real pipeline flow, the simulation would ideally be run over a semi-infinite (or very long) x domain. Obviously computational limits mean this is not possible. In creating a small disturbance to a horizontal layer of fluid, a short wave form develops. Because it takes a reasonable time for the disturbances to propagate along a broad range, the simulation can be run with periodic boundary conditions over a relatively short domain without any self-interference. That is, although the wave form reappears on the left of the domain when exiting the right, there is a lot of undisturbed space between. See figure 5.1 as an example when $x \in (-30, 30)$ with the highest peak at the front of the waveform.

5.5 Linearised Theory vs. Numerical Solutions

There are three key cases to consider in comparing the linear stability to the numerical approximation to the full equations. Namely a stable, borderline and unstable case of F_0.

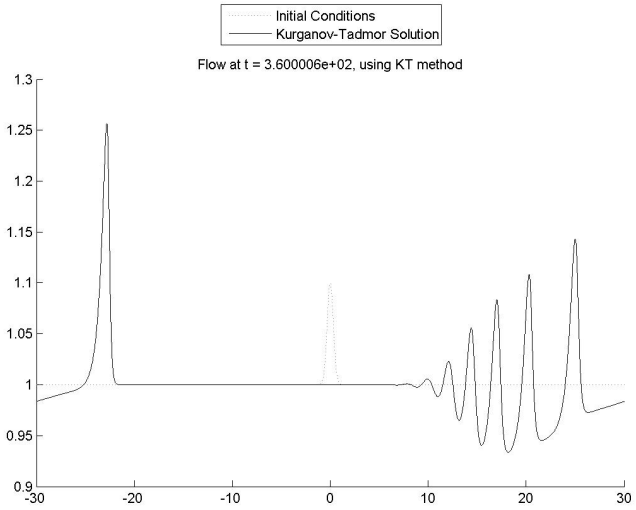


Figure 5.1: Periodic boundary conditions after long run, $t = 360$

Simulations can be run with all other parameters the same across the cases, choose

$$\lambda = 1, \quad \rho = 0.1, \quad R_l = 50, \tag{5.17}$$

then the critical Froude number is $F_c \approx 3.79$. With λ, ρ and R_l given and F_c calculated, F_0 can be chosen for a desired stability. Also an initial small disturbance

$$h(x_i, 0) = 1 + 0.1 \exp(-5x_i^2) \tag{5.18}$$

is created. The first case, where the perturbation is expected to die away by the linear stability theory, is when

$$F_0^2 < 4(1-\rho)(1+\lambda)^2. \tag{5.19}$$

This can be seen demonstrated in figure 5.2, with the dashed line of the initial condition clearly larger than the developed wave. Relatively quickly, the amplitude of the perturbation reduces and the flow tends to the uniform state.

The next clear case to look at is a large F_0, choosing an example for

$$F_0^2 > 4(1-\rho)(1+\lambda)^2. \tag{5.20}$$

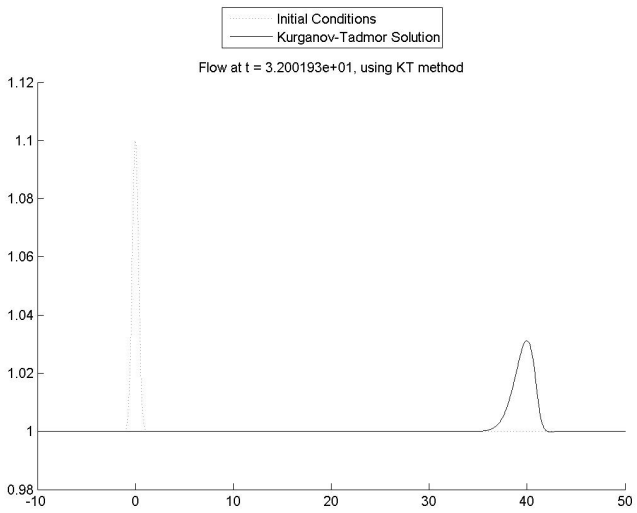


Figure 5.2: Stable perturbation for $F_0 = 3$

For demonstrating this case, it is worth remembering that $F_0 > F_c$ is not a condition for instability, despite its opposite being a condition for stability, see figure 4.2. To be confidently within the region of instability, a sufficiently large value is chosen. It can be seen in figure 5.3 that the perturbation grows very large as expected.

The final case is when

$$F_0^2 \approx 4(1-\rho)(1+\lambda)^2. \tag{5.21}$$

In line with the linear theory, we are expecting no quick development of the initial condition near the stability boundary. Because of this, the numerical simulation must be run over a much longer time domain, say until some large time $t = T$. This will incur larger inaccuracies as any local errors at each time step will be compounded over the simulation. It is put forward by Atkinson et al. [2] that Richardson's extrapolation [20] could be used to estimate the global error in terms of two simulations using different step sizes. It is also noted, however, that even if methods were used to approximate the global error, it is not as simple as decreasing Δt when this error rises above some tolerance. As an initial value problem, the whole simulation must be restarted with a smaller Δt to achieve the desired accuracy at time T.

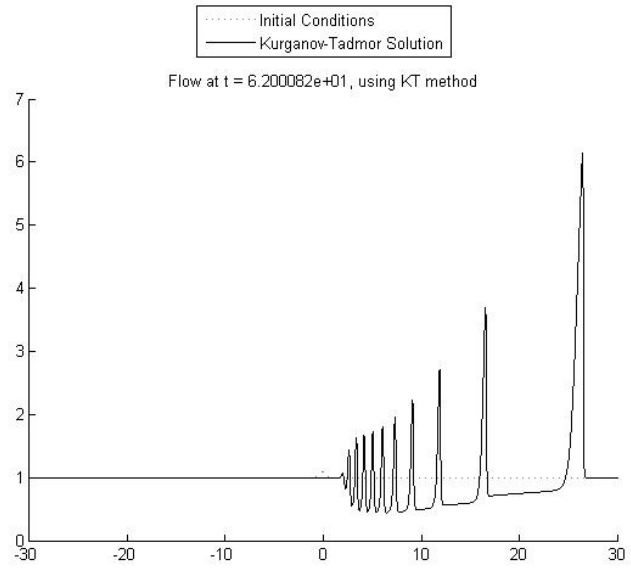


Figure 5.3: Unstable perturbation for $F_0 = 10$

By using the higher order methods for time discretization (see section 5.2) it can be seen that these accrued errors appear to be of little concern with even the first order approximation, assuming Δx is sufficiently small. Because of the sharp shock structure of the waves, it appears that a small Δx is more crucial for an accurate model than a small Δt, assuming the CFL condition for convergence holds.

So, with more confidence than before, long simulations can be run near the linear stability boundary to observe the growth of a perturbation. Figure 5.4 shows, for a relatively large T and an F_0 just slightly in the predicted instability region, an unstable and growing wave developing slowly.

6 Periodic Travelling Waves

6.1 Deriving a 2D system

The waves arising from the numerical simulations appear have a similar saw-tooth profile to the periodic roll waves discussed in section 2. Taking cues from this and other sources

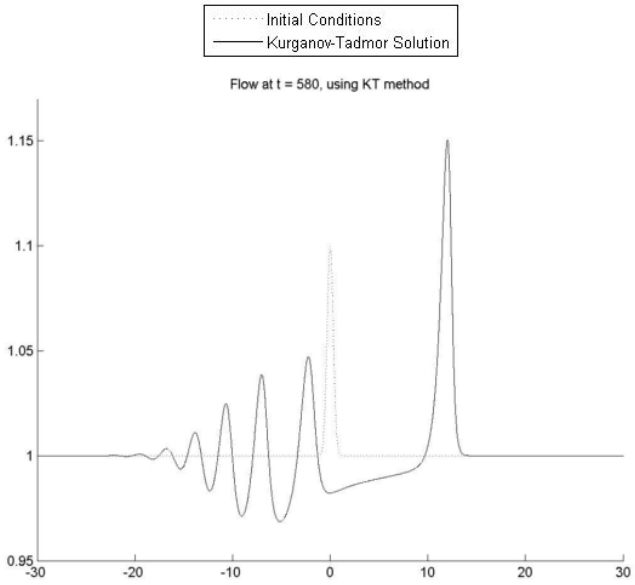


Figure 5.4: Barely unstable perturbation for $F_0 = 4$

([17], [18]), we move on to explore the stability behaviour of the full non-linear system in the case of periodic travelling wave solutions. In particular taking the case $F_0 > F_c$ as defined in (4.17), as otherwise we expect disturbances to tend to a uniform state rather than a slugging state due to previous theory and numerical simulations. Introduce the travelling coordinate $z = x - Ut$, in terms of which $h = h(z)$ and $v = v(z)$ for $z \in (-\infty, \infty)$. Both h and v are periodic in z, with $h > 0$ as before. By substitution into (3.22), we obtain

$$0 = \frac{\partial h(z)}{\partial t} + \frac{\partial (h(z) \cdot v(z))}{\partial x},$$

which by the chain rule is

$$0 = \frac{\mathrm{d}h}{\mathrm{d}z}\frac{\partial z}{\partial t} + \frac{\mathrm{d}(hv)}{\mathrm{d}z}\frac{\partial z}{\partial x},$$

and therefore

$$0 = [h(v - U)]_z. \tag{6.1}$$

Also, by a similar substitution and reduction of (3.21) we obtain

$$\frac{\partial z}{\partial t}\frac{\mathrm{d}v}{\mathrm{d}z} + v \cdot \frac{\partial z}{\partial x}\frac{\mathrm{d}v}{\mathrm{d}z} + \frac{(1-\rho)}{F_0^2}\frac{\partial z}{\partial x}\frac{\mathrm{d}h}{\mathrm{d}z} = \frac{h+\lambda}{(1+\lambda)h} - \frac{v^2}{h} + \frac{1}{R_l h} \cdot \frac{\partial z}{\partial x}\frac{\mathrm{d}}{\mathrm{d}z}\left[h \cdot \frac{\partial z}{\partial x}\frac{\mathrm{d}v}{\mathrm{d}z}\right],$$

which simplifies to

$$(v-U)v_z + \frac{(1-\rho)}{F_0^2}h_z = \frac{h+\lambda}{(1+\lambda)h} - \frac{v^2}{h} + \frac{1}{R_l h}[hv_z]_z. \tag{6.2}$$

By integrating (6.1), it is clear that for some real constant B, we have

$$h(v-U) = B. \tag{6.3}$$

With the aim of eliminating v from (6.2), it is first useful to look at some of the individual terms and how v can be replaced using (6.3). By a simple rearrangement it is apparent that $v = B/h + U$, so

$$v_z = \frac{-B}{h^2}h_z, \quad v^2 = \left(\frac{B}{h} + U\right)^2. \tag{6.4}$$

The first and last terms are to have v eliminated using

$$(v-U)v_z = \frac{B}{h}\left(-\frac{B}{h^2}h_z\right) = \frac{-B^2}{h^3}h_z \tag{6.5}$$

and

$$(hv_z)_z = \left[\frac{-B}{h}h_z\right]_z = \frac{B}{h^2}h_z^2 - \frac{B}{h}h_{zz} \tag{6.6}$$

respectively. Then directly substituting (6.4)-(6.6) into equation (6.2), it becomes

$$-\frac{B^2}{h^3}h_z + \frac{(1-\rho)}{F_0^2}h_z = \frac{h+\lambda}{(1+\lambda)h} - \frac{(B/h+U)^2}{h} + \frac{1}{R_l h}\left(\frac{B}{h^2}h_z^2 - \frac{B}{h}h_{zz}\right),$$

which rearranges to

$$-B^2h_z + \frac{(1-\rho)h^3}{F_0^2}h_z + \frac{h}{R_l}\left(Bh_{zz} - \frac{B}{h}h_z^2\right) + (B+Uh)^2 - \frac{(h+\lambda)h^2}{(1+\lambda)} = 0.$$

By collecting derivative terms, we arrive at the second order ordinary differential equation in $h(z)$

$$Bh_{zz} - \frac{B}{h}h_z^2 + \frac{R_l}{h}\left(\frac{(1-\rho)h^3}{F_0^2} - B^2\right)h_z + \frac{R_l}{h}\left((B+Uh)^2 - \frac{(h+\lambda)h^2}{(1+\lambda)}\right) = 0, \tag{6.7}$$

which can be rewritten as the 2D first order system

$$h' = w \tag{6.8}$$

$$Bw' = B\frac{w^2}{h} - \frac{R_l}{h}\left[\frac{(1-\rho)h^3}{F_0^2} - B^2\right]w - \frac{R_l}{h}\left[(B+Uh)^2 - \frac{(\lambda+h)h^2}{(\lambda+1)}\right]. \tag{6.9}$$

For all $z \in \mathbb{R}$, where the prime denotes $\mathrm{d}/\mathrm{d}z$. Periodic orbits are sought in the (h, w) phase plane, for $U > 0$ and $(U, B) \in \mathbb{R}^2$. For physical sense, these should lie entirely within $h > 0$.

6.2 Case $B = 0$

To start with, it is most simple to consider the case $B = 0$, then (6.8) and (6.9) become

$$h' = w, \quad 0 = -\frac{R_l}{h}\left[\frac{(1-\rho)h^3}{F_0^2}\right] w - \frac{R_l}{h}\left[U^2 h^2 - \frac{(\lambda + h)h^2}{(\lambda + 1)}\right] \tag{6.10}$$

Combining these,

$$h' \frac{(1-\rho)h^3}{F_0^2} + U^2 h^2 - \frac{(\lambda + h)h^2}{(\lambda + 1)} = 0$$

$$\Rightarrow h' \frac{(1-\rho)}{F_0^2} + \frac{U^2}{h} - \frac{(\lambda + h)}{(\lambda + 1)h} = 0$$

So

$$h' = \frac{-F_0^2}{(1-\rho)h}\left(U^2 - \frac{(\lambda + h)}{(\lambda + 1)}\right) \tag{6.11}$$

Now, for some reasoning about the behaviour of h', define

$$\phi = U^2 - \frac{(\lambda + h)}{(\lambda + 1)} \tag{6.12}$$

Note that $h'/\phi < 0$. To show that all solutions to (6.11) are monotone in z, two cases can be quickly considered. First, if $h' \leq 0$, then $\phi \geq 0$. We have h decreasing and so ϕ increasing. Therefore it remains true that $\phi \geq 0$ and so $h' \leq 0$. The converse argument is true for the case that $h' \geq 0$. Because h' is monotonic, (6.8) and (6.9) cannot poissibly have any periodic solutions when $B = 0$.

6.3 Nature of Equilibrium Points for $B \neq 0$

At the equilibrium points $(H, 0)$ where $h' = w' = 0$, equation (6.9) becomes

$$(B+UH)^2 - \frac{(\lambda+H)H^2}{(\lambda+1)} = 0, \tag{6.13}$$

so solutions in terms of U are

$$U_{\pm} = \pm\frac{(\lambda+H)^{\frac{1}{2}}}{(\lambda+1)^{\frac{1}{2}}} - \frac{B}{H}. \tag{6.14}$$

For analysis of the behaviour at the equilibria for $B \neq 0$, it is useful to establish equations for the eigenvalues of the linearised system associated with (6.8) and (6.9). This is because, by the Hartman-Grobman Theorem [19], if neither eigenvalue is purely imaginary (the fixed points are hyperbolic) then the linearised system has the same qualitative structure as the nonlinear system. To get the linearised system near an equilibrium point $(H, 0)$, define

$$f(h,w) = h', \quad g(w,h) = w', \quad \hat{h}, \hat{w} \ll 1, \tag{6.15}$$

such that

$$f(h,w) = f(H,0) + \hat{h}f_h(H,0) + \hat{w}f_w(H,0) + O(\hat{w}^2) + O(\hat{h}^2), \tag{6.16}$$

$$g(h,w) = g(H,0) + \hat{h}g_h(H,0) + \hat{w}g_w(H,0) + O(\hat{w}^2) + O(\hat{h}^2). \tag{6.17}$$

By definition $f(H,0) = g(H,0) = 0$, so the linear approximation can be made

$$\begin{bmatrix} \hat{h}' \\ \hat{w}' \end{bmatrix} = \begin{bmatrix} f_h(H,0) & f_w(H,0) \\ g_h(H,0) & g_w(H,0) \end{bmatrix} \begin{bmatrix} \hat{h} \\ \hat{w} \end{bmatrix}. \tag{6.18}$$

Here the matrix is the Jacobian, J, evaluated at the equilibrium point. The partial derivatives of f are simple, those of g at $(H,0)$ are

$$g_h = \frac{R_l}{BH^2}\left((B+UH)^2 - \frac{(\lambda+H)H^2}{(\lambda+1)}\right) - \frac{R_l}{BH}\left[(B+Uh)^2 - \frac{(\lambda+h)h^2}{(\lambda+1)}\right]_h, \tag{6.19}$$

$$g_w = \frac{-R_l}{BH}\left(\frac{(1-\rho)H^3}{F_0^2} - B^2\right). \tag{6.20}$$

Where the second term of (6.19) is yet to be evaluated at $(H, 0)$ and, from (6.13), the first term is zero. The Jacobian can now be written as

$$J(H,0) = \begin{bmatrix} 0 & 1 \\ \frac{-R_l}{BH}\left[(B+Uh)^2 - \frac{(\lambda+h)h^2}{(\lambda+1)}\right]_h & \frac{-R_l}{BH}\left(\frac{(1-\rho)H^3}{F_0^2} - B^2\right) \end{bmatrix}. \tag{6.21}$$

For ease in analysing the eigenvalues of J, define

$$J(H,0) = \begin{bmatrix} 0 & 1 \\ a & b \end{bmatrix}. \tag{6.22}$$

Eigenvalues of J are needed in order to examine the nature of the equilibria. Eigenvalues k satisfy

$$|J(H,0) - kI| = \begin{vmatrix} -k & 1 \\ a & b-k \end{vmatrix} = 0, \tag{6.23}$$

which gives the eigenvalue equation

$$k^2 - bk - a = 0. \tag{6.24}$$

Denoting the two eigenvalues as k_+ and k_-, we obtain the relations

$$-a = k_+ k_- \quad , \tag{6.25}$$

$$b = k_+ + k_- \quad . \tag{6.26}$$

By determining the signs of a and b we will be able to use (6.25) and (6.26) to determine the signs of the eigenvalues.

6.4 Case $B > 0$

Now that $B = 0$ has been eliminated, the next case to consider is $B > 0$. It can easily be seen that $U_- < 0$, as both its terms are negative, and so it has no a valid solution. There

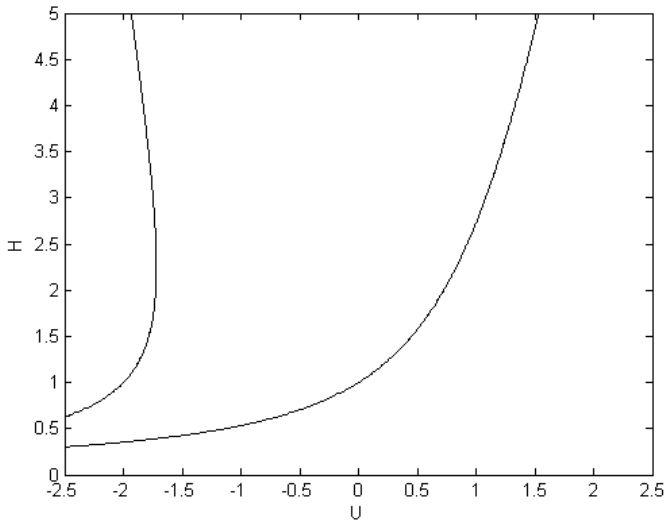


Figure 6.1: $U_\pm$ as defined in (6.14) for $B = 1, \lambda = 1$

is one valid positive solution $H = h_e(U)$ for U_+, as can be seen in figure 6.1 for a typical case with $B > 0$.

To establish the signs of the eigenvalues, it is sufficient to look at the sign of

$$\left[(B + Uh)^2 - \frac{(\lambda + h)h^2}{(\lambda + 1)}\right]_h . \tag{6.27}$$

By evaluating at $h = h_e$ and eliminating U using (6.13) and (6.14), the expression (6.27) becomes

$$
\begin{aligned}
&2U(B+Uh_e) - \frac{(2\lambda+3h_e)h_e}{\lambda+1} \\
&= 2\left(\frac{(\lambda+h_e)^{\frac{1}{2}}}{(\lambda+1)^{\frac{1}{2}}} - \frac{B}{h_e}\right)\left(\frac{(\lambda+h_e)^{\frac{1}{2}}h_e}{(\lambda+1)^{\frac{1}{2}}}\right) - \frac{(2\lambda+3h_e)h_e}{\lambda+1} \\
&= 2\left(\frac{(\lambda+h_e)h_e}{\lambda+1}\right) - \frac{B(\lambda+h_e)^{\frac{1}{2}}}{(\lambda+1)^{\frac{1}{2}}} - \frac{(2\lambda+3h_e)h_e}{\lambda+1} \\
&= \frac{(2\lambda+2h_e)h_e - (2\lambda+3h_e)h_e}{\lambda+1} - \frac{B(\lambda+h_e)^{\frac{1}{2}}}{(\lambda+1)^{\frac{1}{2}}} \\
&= \frac{-h_e^2}{\lambda+1} - \frac{B(\lambda+h_e)^{\frac{1}{2}}}{(\lambda+1)^{\frac{1}{2}}} \\
&< 0.
\end{aligned}
$$

Since $(-R_l/(BH)) < 0$ and $(6.27) < 0$, the product a is positive and

$$k_+k_- = -a < 0. \tag{6.28}$$

This implies the eigenvalues are real of opposite sign, so the equilibrium point $(h_e, 0)$ is a saddle point. As the periodic orbit entirely within $h > 0$ would be a closed loop with no fixed points, index theory can be applied. As noted in [23], the index of a center (single equilibrium within a periodic orbit) is $+1$. The index of a saddle point is -1 so it is not possible for $(h_e, 0)$ to be a center and we can eliminate the possibility that $B > 0$.

6.5 Case $B < 0$

For the case $B < 0$, more possibilities arise. The following observations can be seen in a typical plot of H against U for $B < 0$, such as figure 6.2. For small enough U there is still only one solution $h = h_e$, along U_-. It is also clear though (from the plot, but

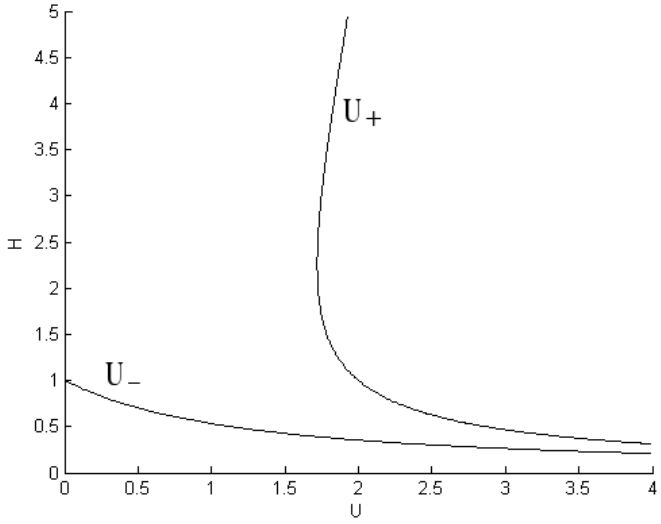


Figure 6.2: $U_{\pm}$ as defined in (6.14) for $B = -1, \lambda = 1$

more importantly because both terms in (6.14) are positive) that $U_+ > 0$, giving rise to up to two new solutions. A saddle node bifurcation, where two new equilibrium points come into existence, occurs at $U = U_{SN}$ which is the minimum U in the U_+ curve. This minimum can be found by differentiating (6.14) to obtain

$$U'_+(H) = \frac{1}{(2(\lambda+1)^{1/2}(\lambda+H)^{1/2}} + \frac{B}{H^2} = 0, \tag{6.29}$$

so

$$B_{SN} = \frac{-H_{SN}^2}{2(\lambda+1)^{1/2}(\lambda+H_{SN})^{1/2}}. \tag{6.30}$$

Again using (6.14) with (6.30) gives

$$U_{SN} = \frac{(\lambda+H)^{1/2}}{(\lambda+1)^{1/2}} + \frac{H_{SN}}{2(\lambda+1)^{1/2}(\lambda+H_{SN})^{1/2}}. \tag{6.31}$$

Where all subscripts SN denote occurrence at the saddle node bifurcation. So (6.30) and (6.31) are together parametric equations for U and B in terms of H at the saddle node bifurcation.

If $U > U_{SN}$, there are in fact there two more solutions, say h_- and h_+. From figure 6.2 it can be seen that $h_e < h_- \leq h_+$, where $h_- = h_+$ is a repeated solution only occurring at

$U = U_{SN}$. Since h_- and h_+ are monotonically decreasing and increasingly respectively, this size order will hold for all U.

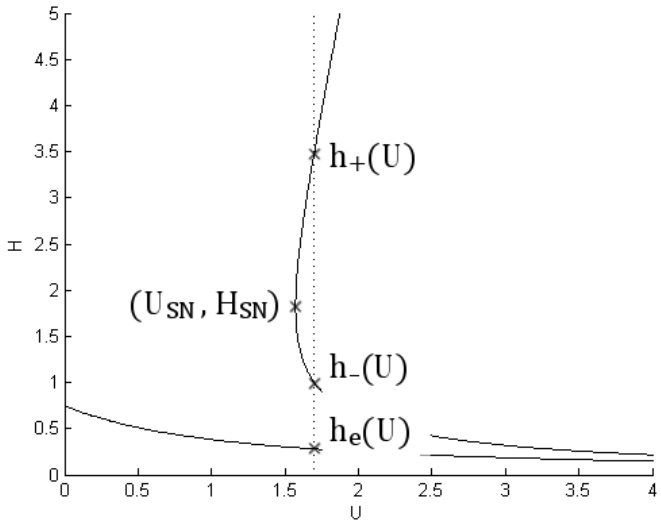


Figure 6.3: $U_\pm$ with key points labelled for $B = -1, \lambda = 1$

Next the nature of the three equilibrium points is looked at with the hope that at least one will admit periodic orbits. The remainder of this section follows the elimination of the cases $h = h_e$ and $h = h_-$, before establishing the existence of a Hopf bifurcation, and therefore periodic orbits, in the case $h = h_+$. The sign of a is less obvious when $B < 0$, as we cannot draw an immediate conclusion from (6.27) like before. Instead, it is better to use some reasoning about cubic functions to draw a conclusion. Of course the partial derivative operator in (6.27) treats the other parameters as constant. Graphically, (6.27) evaluated at h_e, h_- and h_+ can be interpreted as the gradient of the cubic function

$$(B + Uh)^2 - \frac{(\lambda + h)h^2}{(\lambda + 1)}, \tag{6.32}$$

evaluated at its roots. These can be seen labelled on figure 6.4, which is three demonstrative plots to help visualise the different possible cases for the form of (6.32). As a negative cubic the gradient at the smallest root, h_e, must be negative. This gives

$$\left[(B + Uh)^2 - \frac{(\lambda + h)h^2}{(\lambda + 1)}\right]_h \Bigg|_{h_e} < 0, \tag{6.33}$$

so

$$k_+k_- = -a > 0 \quad . \tag{6.34}$$

As the product of the eigenvalues is positive, they must have the same sign. For completeness it is worth noting that the gradient at h_e cannot be 0. This would imply that $U_- = U_+$, which in turn is only possible if $\lambda + h_e = 0$. An impossibility since both quantities are positive. The only possible repeated root is when $h_- = h_+ = H_{SN}$, as can be seen in figure 6.4. This is also a good visualisation of the saddle node bifurcation occurring when two equilibrium points collide and eliminate each other.

Now knowing that the eigenvalues have the same sign, we must determine whether $k_-, k_+ > 0$ or $k_-, k_+ < 0$. This is given by the sign of b, which is equivalent to the sign of

$$\frac{(1-\rho)H^3}{F_0^2} - B^2 \tag{6.35}$$

at $H = h_e$. Since h_e lies on U_-, and $F_0^2 > F_c^2 = 4(1-\rho)(1+\lambda)$,

$$\frac{(1-\rho)h_e^3}{F_0^2} - B^2 < \frac{(1-\rho)h_e^3}{4(1-\rho)(1+\lambda)} - B^2 = \frac{h_e^3}{4(1+\lambda)} - B^2 \tag{6.36}$$

$$= \frac{h_e^3}{4(1+\lambda)} - \left[\frac{(\lambda+h_e)h_e^2}{(\lambda+1)} + \frac{2U(\lambda+h_e)^{1/2}h_e^2}{(\lambda+1)^{1/2}} + U^2h_e^2\right] \tag{6.37}$$

$$= \frac{h_e^3/4 - h_e^3 - \lambda h_e^2}{\lambda+1} - \frac{2U(\lambda+h_e)^{1/2}h_e^2}{(\lambda+1)^{1/2}} - U^2h_e^2 \tag{6.38}$$

$$< 0. \tag{6.39}$$

Now it is established that $b < 0$, so both eigenvalues $k_\pm$ are negative and so h_e is a stable node or spiral for any $U > 0$.

For $U > U_{SN}$, the nature of h_- can be assessed. Again easy to see by the graphical reasoning,

$$\left[(B+Uh)^2 - \frac{(\lambda+h)h^2}{(\lambda+1)}\right]_h\Bigg|_{h_-} > 0, \tag{6.40}$$

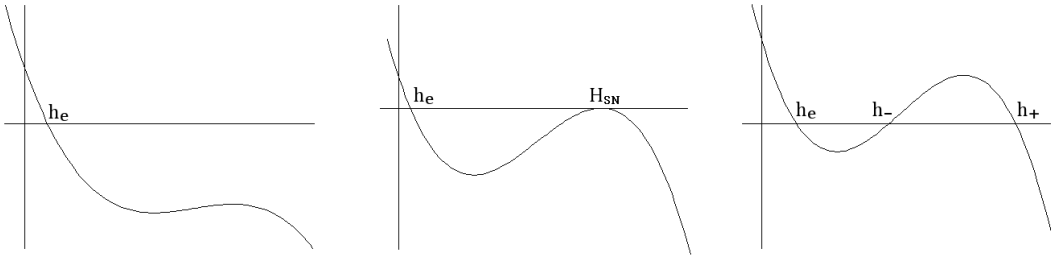


Figure 6.4: One ($U < U_{SN}$), two ($U = U_{SN}$) or three ($U > U_{SN}$) roots of (6.27)

so

$$k_+ k_- = -a < 0 \quad . \tag{6.41}$$

The eigenvalues are therefore of opposite sign, and h_- must be a saddle point. The only remaining case to explore for periodic orbits is $U > U_{SN}$, $h = h_+$. Again it can be established that

$$\left[(B + Uh)^2 - \frac{(\lambda + h)h^2}{(\lambda + 1)}\right]_h \Bigg|_{h_+} < 0, \tag{6.42}$$

so

$$k_+ k_- = -a > 0 \quad . \tag{6.43}$$

A Hopf bifurcation, if it exists, will occur at the point when a periodic orbit is created. Periodic orbits have purely imaginary eigenvalues k_+, k_-, so

$$\frac{(1-\rho)h_+^3}{F_0^2} - B^2 = 0. \tag{6.44}$$

A parametric form of the point in parameter space (U, B) where the Hopf bifurcation occcurs can now be written using (6.44) and (6.14), bearing in mind that h_+ lies on U_+

$$B_{HB} = -\frac{(1-\rho)^{1/2}}{F_0} h_+^{3/2}, \tag{6.45}$$

$$U_{HB} = \frac{(\lambda + h_+)^{1/2}}{(\lambda + 1)^{1/2}} + \frac{(1-\rho)^{1/2}}{F_0} h_+^{1/2}. \tag{6.46}$$

Subscripts HB denote the values at the Hopf bifurcation. If B_{HB} and B_{SN} were concurrent

we would get $h_+ = H_{SN}$ and

$$-\frac{(1-\rho)^{1/2}h_+^{3/2}}{F_0} = -\frac{h_+^2}{2(\lambda+1)^{1/2}(\lambda+h_+)^{1/2}},$$

so

$$\frac{(1-\rho)}{F_0^2}\cdot 4(\lambda+1)(\lambda+h_+) = h_+ \quad .$$

Collecting the h_+ terms this gives

$$\frac{(1-\rho)}{F_0^2}\cdot 4(\lambda+1)\lambda = h_+\left(1-\frac{(1-\rho)}{F_0^2}\cdot 4(\lambda+1)\right),$$

and the final value of h_+ is given purely in terms of flow parameters as

$$\frac{4(1-\rho)(\lambda+1)\lambda}{F_0^2-4(1-\rho)(1+\lambda)} = h_+ \quad . \tag{6.47}$$

Since $h_+(U_{HB}) \geq h_+(U_{SN}) = H_{SN}$, and therefore $B_{HB} \leq B_{SN}$, we get

$$h_+(U_{HB}) \geq \frac{4(1-\rho)(\lambda+1)\lambda}{F_0^2-4(1-\rho)(1+\lambda)}, \tag{6.48}$$

and also a condition on B_{HB} from substitution into (6.45) to obtain

$$B_{HB} = -\frac{(1-\rho)^{1/2}}{F_0}h_+(U_{HB})^{3/2} \leq \frac{-8(1-\rho)^2(\lambda+1)^{3/2}\lambda^{3/2}}{F_0\left[F_0^2-4(1-\rho)(\lambda+1)\right]^{3/2}} \quad . \tag{6.49}$$

7 Numerical analysis of the limit cycle

The limit cycle which is created at $h_+(U_{HB})$ grows in amplitude in the (U,h) plane. The actual amplitude of the cycle is very small, and grows monotonically in U. At some point, $U = U_{HOM}$, the amplitude of the cycle is large enough that the minimum point collides with the saddle point at h_-. Upon this happening, the cycle becomes a homoclinic orbit originating at h_-, joining the saddle equilibrium to itself, as pictured in figure 7.1. By choosing some initial value h_0 greater than and in the region of h_-, it would be expected that the solution either gets captured by an orbit around h_+ if $U < U_{HOM}$, or escapes to infinity if $U > U_{HOM}$. The code developed in this section can be seen in appendix B.

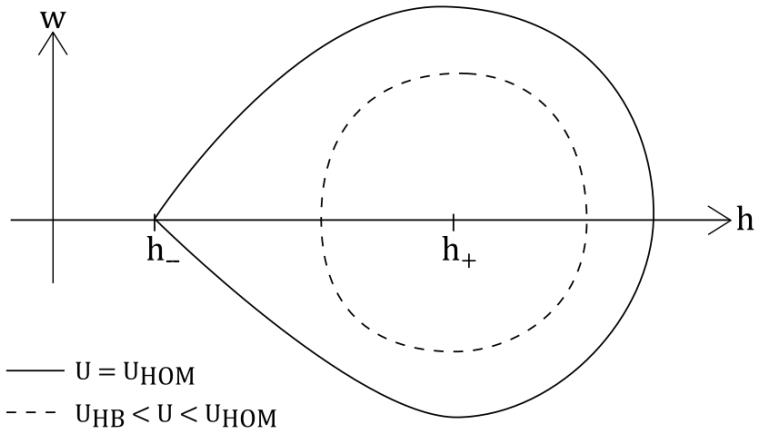


Figure 7.1: Creation of homoclinic orbit around h_+ by collision with saddle point h_-

The value of U_{HOM} can be calculated numerically for a given B, using some numerical iteration which will be described shortly and a built in MATLAB ordinary differential equation (ODE) solver. Specifically, the most commonly applicable ode45 is a variable step Runge-Kutta solver for ODEs. Integrating (3.17) using ode45, with an initial value h_0 for $U < U_{HOM}$ will give some bounded periodic solution. In the case that $U > U_{HOM}$, the solution will grow exponentially, and likely cause the solver to fail. These two cases can easily be identified by looking at the maximum value of the solution, and a bisection approach can be used to calculate the value of U_{HOM} to some given accuracy.

7.1 Bisection Approach to Calculate U_{HOM}

Analogous to the root finding bisection method, we can choose two initial values which straddle U_{HOM} and then take a further value as the midpoint of them. Then discard the one in the original pair which is on the same side of U_{HOM} as the new point. This process requires a sufficiently accurate method for determining whether a given value of U is greater than or less than U_{HOM}. If that is available then the steps can be iterated to approximate the value of U_{HOM}. Let the steps of the bisection approach to approximate U_{HOM} be

$$U_0, U_1, U_2, ..., U_N \quad \text{where} \quad U_N \to U_{HOM} \quad \text{as} \quad N \to \infty. \tag{7.1}$$

It has been established that

$$U_{HB} < U_{HOM}, \tag{7.2}$$

and later numerical calculation shows that

$$U_{HOM} - U_{HB} = \xi, \tag{7.3}$$

for some small ξ. Let $U_0 = U_{HB}$, then choose an initial guess for $U_1 > U_{HOM}$. This does not need to be very large because of (7.3), but of course needs to be large enough that $U_{HOM} \in (U_0, U_1)$. The method to be used during the iterations can be used to check whether U_1 is a valid (large enough) guess. Then implement a standard bisection approach as described above. For root finding, we would retain a left value for U which is positive, and a right value which is negative. In this application, we should retain a left value of U for which ode45 gives a periodic solution and a right value for which it gives an exponential solution. The main limitation of this approach is that it relies on the approximate solution being qualitatively representative of the true solution. For the examples trialled this appears to not be an issue, but could be for some parameters which are (even more) sensitive to change.

The calculated approximation to $U_{HOM}(B)$ can be seen alongside $U_{HB}(B)$ and $U_{SN}(B)$ in figure 7.2. With this visual, it is apparent how narrow the region is between $U_{HB}(B)$ and $U_{HOM}(B)$, as mentioned in (7.3).

The solutions gained from ode45 in calculating U_{HOM} can be used to approximate the amplitude of the periodic solutions, which can be visualised as a function of U. By zooming in to the small region for which the periodic solution can exist, found above, figure 7.3 shows the amplitude of the periodic waves as given by ode45. It can also be seen how the amplitude grows to the point where it collides with h_-.

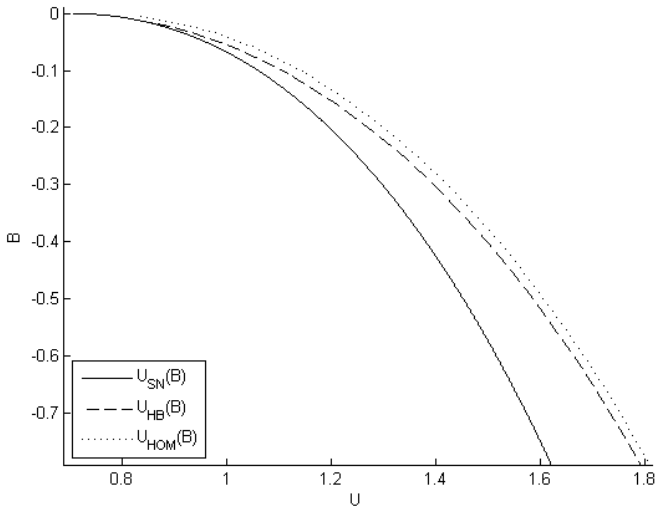


Figure 7.2: Bifurcation curves in (U, B) for $\lambda = 1, F_0 = 10, \rho = 0.1, R_l = 50$

7.2 Calculating Periodic Solutions

Instead of using ode45 over a large domain to look for convergence to a steady periodic solution, which would be very inefficient, it would be better to solve (6.8) and (6.9) subject to periodic boundary conditions. For some unknown wavelength L, let $\hat{z} = z/L$, with $\hat{z} \in [0, 1]$. Then

$$h_{\hat{z}} = Lw, \tag{7.4}$$

$$w_{\hat{z}} = \frac{L}{B}\left\{B\frac{w^2}{h} - \frac{R_l}{h}\left[\frac{(1-\rho)h^3}{F_0^2} - B^2\right]w - \frac{R_l}{h}\left[(B+Uh)^2 - \frac{(\lambda+h)h^2}{(\lambda+1)}\right]\right\}. \tag{7.5}$$

For a periodic solution, the standard and necessary boundary condition on h is

$$h(0) = h(1). \tag{7.6}$$

Various sensible boundary conditions on w (taking the total to three) could be chosen, for example

$$w(0) = w(1) = 0. \tag{7.7}$$

For calculation of a true periodic solution, another in-built MATLAB function can be

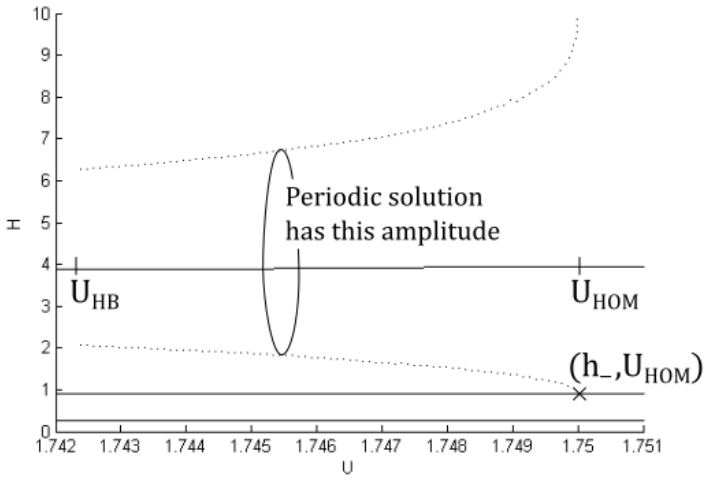


Figure 7.3: Amplitude of periodic solution for $U \in [U_{HB}, U_{HOM}]$

utilised. The bvp5c function can be used to solve boundary value problems for ODEs, and has the capacity to also handle unknown parameters which will be needed for calculating the wavelength L. Its lower order predecessor bvp4c appears to be the standard bearer in documentation, using very similar syntax, the development of which is documented in [12]. Both functions use a variant of the Lobatto IIIa collocation formula which gives a C^1 continuous solution. Implemented as implicit Runge-Kutta methods (examples of which can be seen in [7]) they are A-stable and accurate to order $2s - 2$ for an s stage implementation [1]. The older function bvp4c uses a 3 stage implementation so is 4^{th} order. The newer bvp5c has some minor advantages, one of which is that it implements a 4 stage Lobatto IIIa method, but contrary to the expectation of it then being 6^{th} order the MATLAB documentation states it is 5^{th} order. In the paper outlining its derivation, Kierzenka and Shampine [13] detail how the scheme is implemented as an implicit Runge-Kutta solver. They note in the same paper that for moderate convergence tolerances, bvp4c and bvp5c have comparable performance speeds, while for strict tolerances bvp5c far outperforms.

Theory aside, bvp5c can be used as a black-box with little understanding of its inner workings. It requires as inputs only three things; the ODE to solve, the boundary condi-

tions, and an initial guess to the solution. The first two of these are simple and already given in equations (7.4) - (7.7). For the two variables h and w, we would ordinarily require two boundary conditions. However, the third boundary condition is required when also asking bvp5c to solve for the unknown parameter L. The initial guess is slightly more complicated, and involves an approximate function for the wave as well an an estimate for L.

As the sensitivity around calculating U_{HOM} might suggest, the initial guess must be very good to get a correct solution. If it is not then we simply get a trivial constant h solution (or the solver fails to converge), which does satisfy the boundary conditions but is essentially useless. By using the solution to the ode45 result described above, attributes of the wave after some short time can be used to construct a sine wave which might be a good enough initial guess. This is done by taking the amplitude, wavelength and offset, giving something like in figure 7.4

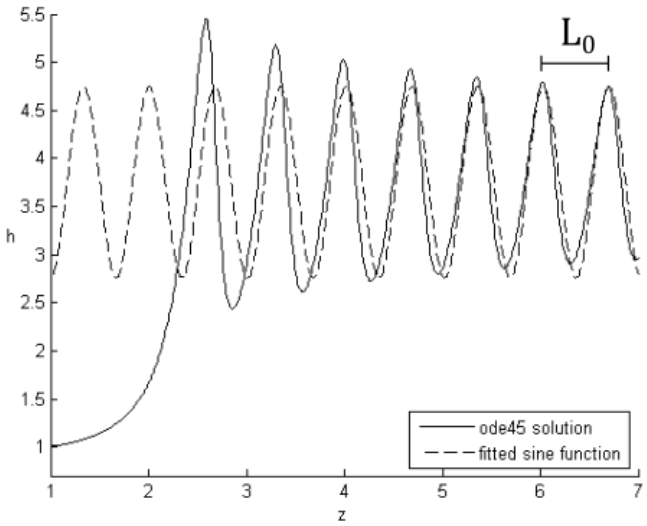


Figure 7.4: Fitting a sine wave to the approximated solution of (6.8) and (6.9)

It can be seen that this sinusoidal function fits reasonably well as z increases. The initial guess for the wavelength to be used with bvp5c is labelled L_0. For given flow parameters, U_{HB} and U_{HOM} can be calculated. Then a U in this range can be chosen

to find the corresponding initial conditions, and ultimately the wavelength and waveform solution to (7.4) and (7.5). For example, let

$$B = -0.7, \qquad R_l = 50, \qquad \rho = 0.1, \qquad \lambda = 1, \qquad F_0 = 10. \tag{7.8}$$

Then the region for possible solutions is

$$[U_{HB}, U_{HOM}] \approx [1.73229, 1.75004], \tag{7.9}$$

by the parametric equations (6.45) and (6.46) and the previously discussed calculation method for U_{HOM}. So choosing $U = 1.73276$ from within this range we get figure 7.5, a small amplitude wave with $L = 0.65$. Note that the wave solution has been plotted over $\hat{z} \in [0, 2]$, but that the boundary conditions hold at $z = 0, z = L$ because of the scaling with $\hat{z}$.

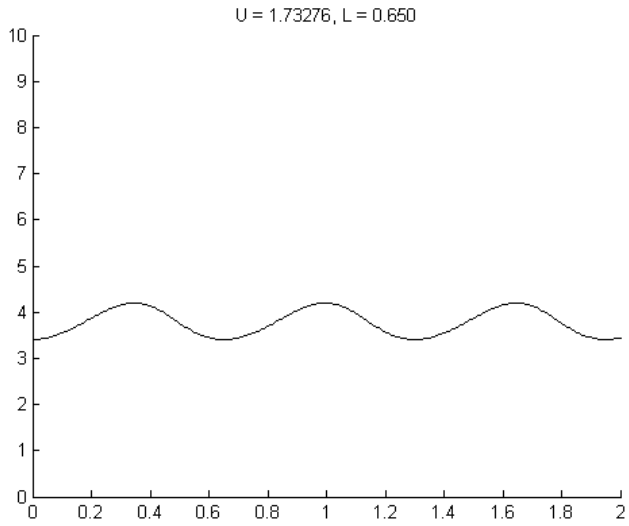


Figure 7.5: Example periodic solution for $U = 1.73276$

This result can be verified against previous results from Needham et al. [17]. Confident in the established method, we might try an increased U, say $U = 1.74858$. However, because of the aforementioned sensitivity to the initial guess, the solver fails and no result can be gained. It seems fairly unpredictable for when the initial guess is accurate enough

using the sinusoidal approximation. It would undoubtedly be good enough more often if the approximation was taken around a larger z, but then we stray into the territory of simply relying on the inefficient method of only using the ode45 solution. Instead, numerical continuation can be used to extrapolate a working solution across the desired U domain.

By taking, say, 100 points within the range (7.9), the solution from the (successful) previous value of U can be used as the approximation to the next value of U. This means that the sinusoidal approximation need only be used once, and then not relied upon for subsequent iterations of increasing U. This gives meaningful periodic solutions as can be seen in figure 7.6. The saw-tooth profile of the waves mirrors that seen in the developed numerical simulations in section 5.5.

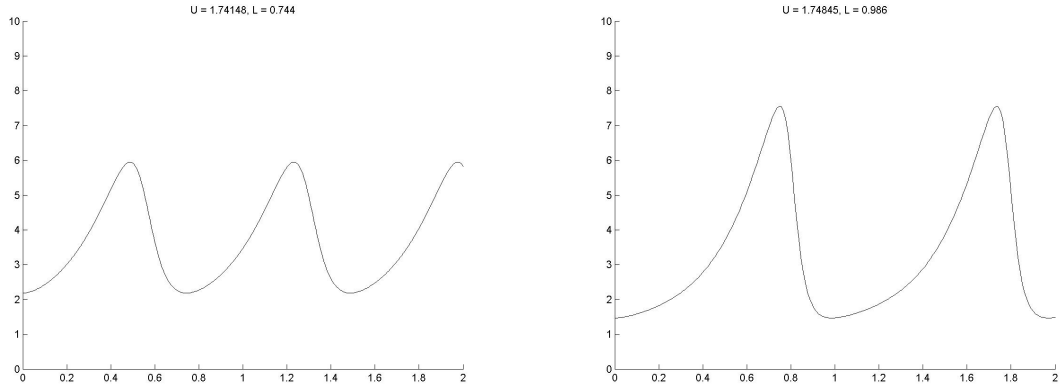


Figure 7.6: Examples of periodic solutions for larger U, $U = 1.74148, 1.74845$

With longer, taller waves arising from increased U, figure 7.7 shows the iterated wave solutions for increasing U by only a few thousandths at each step and recycling the solution. it can be seen that neighbouring waves clearly make a good initial guess for a given U.

Figures 7.8 and 7.9 show how increasing U affects the structure of the periodic wave solution for a fixed B. The amplitude grows steadily for U in the most part of the range (7.9), but exponentially as $U \to U_{HOM}$. This aligns well with the previous solution seen in figure 7.3. Mees and Chua [16] discuss the maximum amplitude of a limit cycle determined

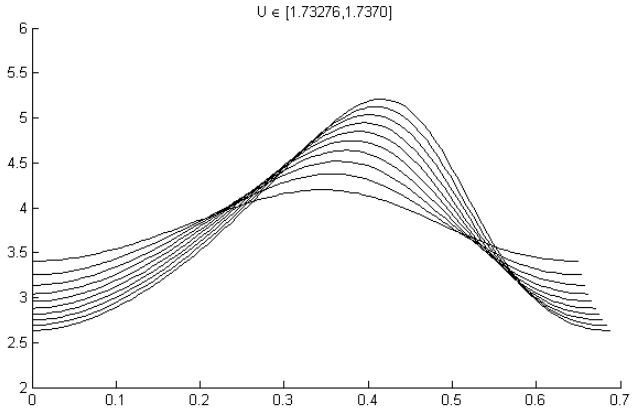


Figure 7.7: Growing wave by iterating U and recycling solution

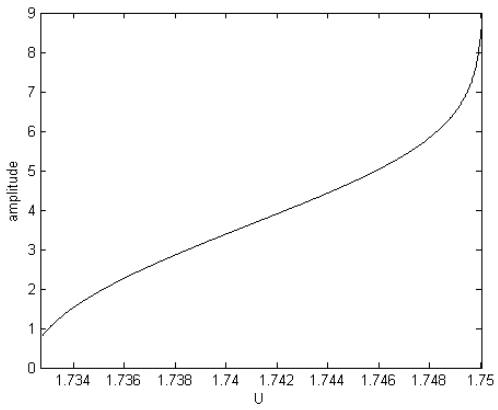


Figure 7.8: Amplitude against U

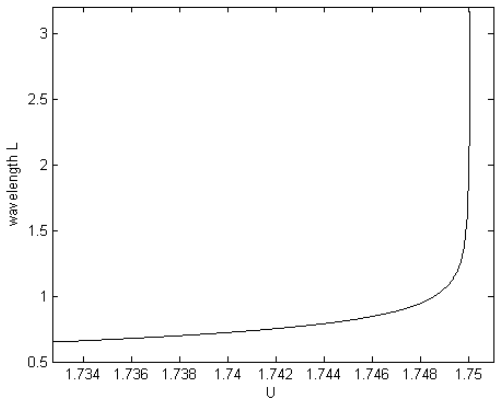


Figure 7.9: Wavelength (L) against U

by the Hopf bifurcation theorem. For a general system, which is applicable to ours, they show that there is a parabolic 'bowl' region within which a limit cycle is stable, at the extremum of which the Hopf bifurcation occurs. Since the Hopf bifurcation occurs at $U = U_{HB}$ and the stability region is parabolic, the amplitude A of the periodic solution as $U \to U_{HB}$ is

$$A(U,B) = \max_{z\in[0,L]}\{h(U,B)\} - \min_{z\in[0,L]}\{h(U,B)\}$$

$$= \mathcal{O}\left((U - U_{HB}(B))^{1/2}\right). \tag{7.10}$$

For values of U in the small region (U_{HB}, U_{HOM}), we have $U - U_{HB} < 1$ and so

$$U - U_{HB}(B) < (U - U_{HB}(B))^{1/2}.$$

Therefore the amplitude growth is much quicker than $\mathcal{O}(U - U_{HB}(B))$. This would suggest that more accurately approximated amplitudes than those used in sections 7.1 should give a bell shape growing quickly from $A(U_{HB}, B) \approx 0$ tending to nearer what was calculated. The influence of the equilibrium point at h_- appears to have a greater influence on the amplitude as $U \to U_{HOM}$ to give the increasing $\partial A/\partial U$. The wavelength is also monotonically increasing in U, with

$$L \to +\infty \quad \text{as} \quad U \to U_{HOM}. \tag{7.11}$$

This (as pointed out by Chow et al. [4]) is to be expected as the periodic orbit must disappear, giving way to a single non periodic solution. Combining these results, figure 7.10 shows the amplitude against wavelength. As L increases, A tends to some finite value.

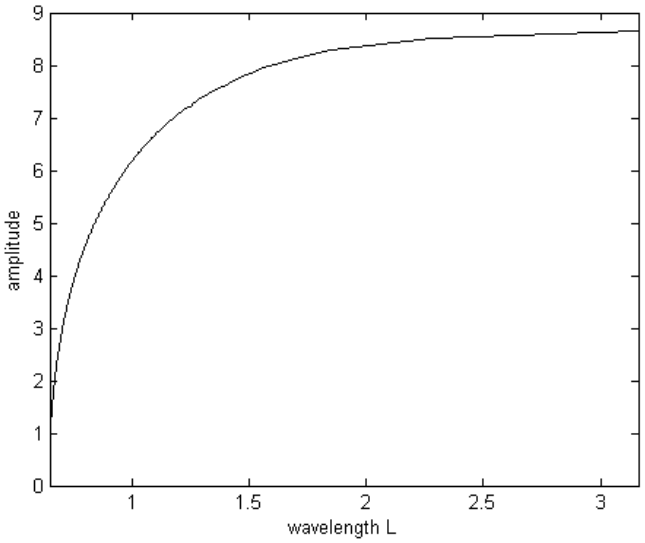


Figure 7.10: Amplitude against wavelength (L)

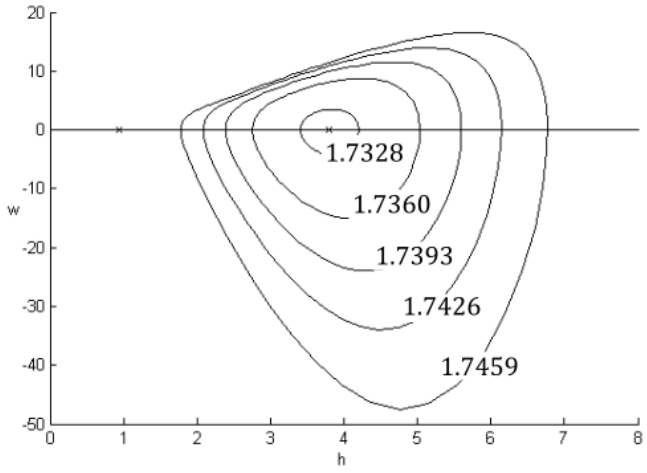


Figure 7.11: Phase plane for parameters in (7.8), with U values labelled on each orbit

7.3 Parameter Effects on Periodic Solutions

Example periodic orbits in the (h, w) phase can be seen in figure 7.11. It is clear that they are centred around h_+, with the left hand side of the orbit gradually forming a wedge, tending to h_-. The orbits become increasingly close on the top side, which leads to an issue in replicating the convergence to a homoclinic orbit as $U \to U_{HOM}$ which is illustrated in figure 7.1. This is because, as the phase lines grow close, small numerical errors from either computation or the methodology can cause non convergence as the orbit collides with the saddle lines originating at h_-.

Once a solution has been gained for a given wavelength, the parameters can be gradually changed to give new solutions with interesting characteristics. Figure 7.12 shows how varying the parameters F_0, R_l and λ changes the properties of the periodic solution for a fixed wavelength $L = 1$. Increasing F_0 mainly has the effect of increasing the amplitude of the waves. This reflects what was found in the numerical solutions where a higher F_0 gave rise to larger waves propagating upstream from a disturbance. It makes physical sense since F_0 is the dimensionless flow rate, so when it is higher there is a larger force acting on the wave. Increasing R_l also increases the amplitude, though it has a much smaller effect.

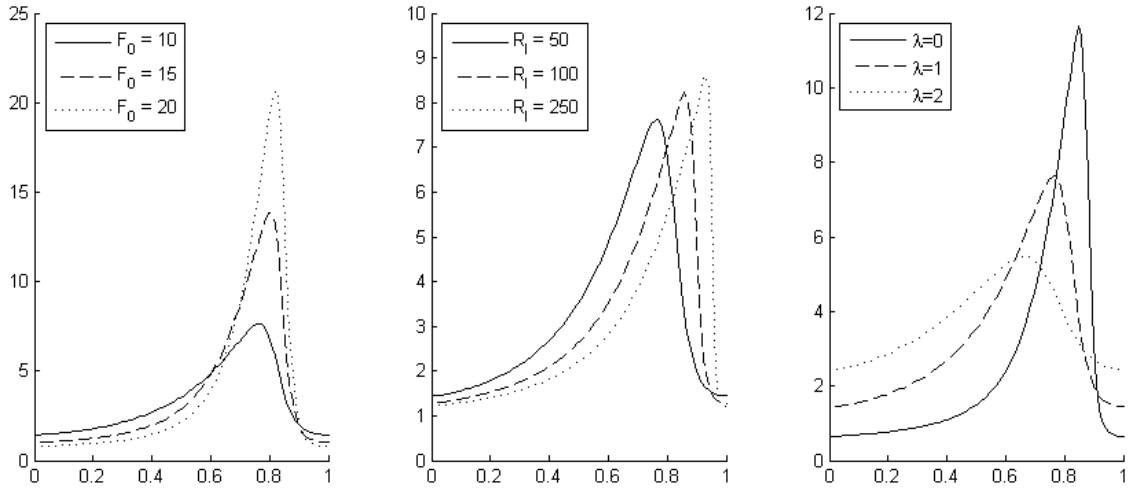


Figure 7.12: Effect of varying F_0, R_l and λ for other parameters fixed as in (7.8), $L = 1$

More significant is how a higher R_l gives a much steeper front to the wave. In equation (3.21), R_l^{-1} in acts on the viscous term and smooths the solution. So the higher R_l is the less shock-like it becomes, which can be seen in the figure. The rightmost plot in the same figure shows how increasing λ appears to have the combined but inverse effect of F_0 and R_l by decreasing both the amplitude and steepness of the wave. The critical Froude number F_c as defined in (4.17) increases linearly with λ. As seen in previous sections, this causes increased stability, so the outcome of a damped periodic solution follows as expected.

Figures 7.13 - 7.15 show the relationships just described, but over a broad range for each changing parameter. Firstly in figure 7.13, an almost direct proportionality can be seen between F_0 and the amplitude or U. Next, figure 7.14 shows how the amplitude grows with R_l quickly to begin with, but that for values of $\mathcal{O}(10^2$ the growth slows down as discussed. For small values, U also grows rapidly. However it soon peaks, and by the time the amplitude is stabilising it is decreasing with R_l. Figure 7.15 shows the descent in amplitude and U as λ increases.

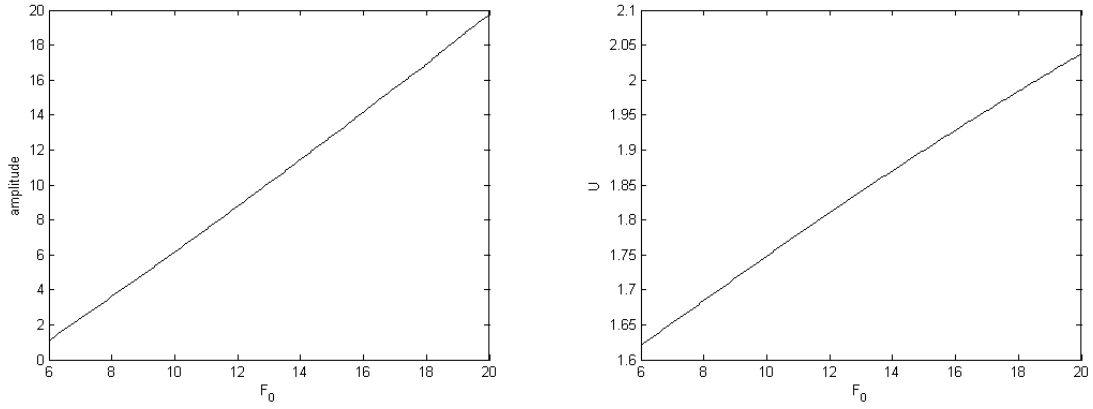


Figure 7.13: F_0 against amplitude (left) and U (right). Parameters B, R_l, ρ, λ as in (7.8)

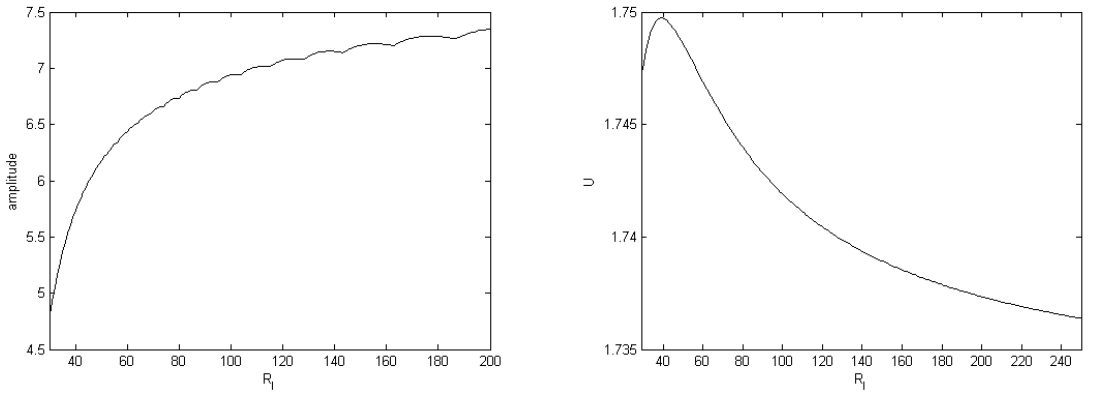


Figure 7.14: R_l against amplitude (left) and U (right). Parameters B, ρ, λ, F_0 as in (7.8)

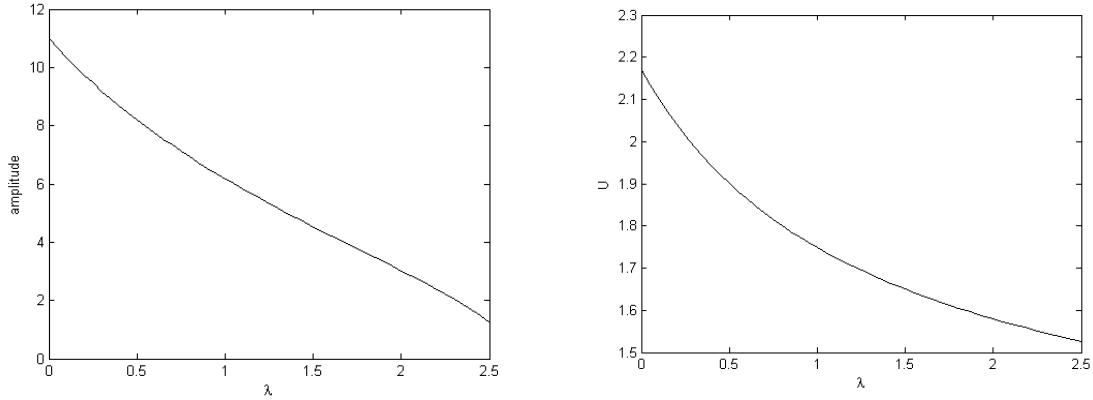


Figure 7.15: λ against amplitude (left) and U (right). Parameters B, R_l, ρ, F_0 as in (7.8)

8 Conclusions

Initially, flow equations were established for a horizontal gas-liquid pipeline flow. Using a series of simplifications, including the continuity of pressure and a thin layer approximation, two dimensionless equations can be derived which govern the liquid flow.

Considered as a similar system, the relationship between roll waves and pipeline flows was laid out. It is outlined by several sources where the similarities lie between the slugging problem and a thin flow down an inclined channel. A particular idea which is informed by this comparison is that the Froude number, the dimensionless flow rate in the system, has an effect on the stability of roll waves. This helps guide investigations into linear stability and numerical simulations later on.

By linearising the equations developed at the start, they could be combined into one partial differential equation in terms of the height of a perturbation to the uniform flow. By observing the growth of this perturbation, a stability condition can be developed which encompasses many of the flow parameters. This can be put in terms of the Froude number, and for the most unstable wave-number a critical Froude number for stability can be defined.

The notions of stability developed to this point can be verified using numerical approximations of the full nonlinear system. The second order semi discrete solution described by Kurganov and Tadmor [14] is a highly suitable method to use in this problem, and can be implemented alongside various methods for temporal discretization. This high order scheme was developed since the system under examination exhibits shock-like peaks which require a method with little numerical dissipation. After optimising for use with the available computing power, several interesting simulations were run and their characteristics analysed. In particular, the stability condition developed for a linearised model was explored. Propagating waves did indeed mimic the expected behaviour, a satisfying result jointly strengthening the validity the linearised and numerical results.

Taking cues partly from the nature of roll waves and numerical solutions, theory about periodic travelling waves can be developed. By introducing a travelling coordinate, a 2D system of first order equations was established which underwent extensive phase plane analysis in search of periodic orbits. In determining the nature of the three possible equilibria, several possible cases can be eliminated, mainly by analysing the eigenvalues of the linearised system. Periodic orbits exist, given certain restraints, in some narrow region between a Hopf bifurcation and homoclinic bifurcation. It was clear that the solutions are extremely sensitive to changes in parameter space, with for example the region between the bifurcations being extremely narrow. Numerical methods were again used to compute the periodic solutions, with methods developed in some detail. In particular, the MATLAB functions ode45 and bvp5c were essential in finding solutions efficiently. Carefully constructed methods did allow analysis of how changing some parameters affects the development of potential slugging states. For faster travelling waves, the wave length grows increasingly quickly, as does the amplitude. These approximated results agreed with theory, and it could be proposed that the growth of these waves is in kind with what we would expect from a forming slug. By varying other flow parameters, a more in depth understanding of the periodic solutions' characteristics was developed.

There is scope for many extensions to this work. Firstly the model could be developed for investigating the flow in a circular pipe, as this would potentially give greater insight into the physical problem. Given more time in the area of numerical simulation, a natural progression would be to implement methods with non-uniform grid spacing. As can be seen from figure 5.3, the waves have high fidelity areas at the peaks, but also large uniform areas. An adaptive grid method could take advantage of this wave trait to increase efficiency and accuracy. Lastly, methods could be applied in pursuit of more precise description of the effects seen in section 7.3.

Overall, this investigation has laid a route from the derivation of suitable flow equations to an in depth analysis of their periodic travelling wave solutions. Many results gained

throughout match those found in the guiding paper by Needham et al. [17], despite some different or independent approaches being taken. The wide availability of reference material spanning the last century shows that this is not only a historically interesting problem, but that it is a current area of active research with room for further development.

References

[1] Ascher U. M., Mattheij, R. M. M., Russel R. D. (1994) *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, SIAM, Classics in Applied Mathematics

[2] Atkinson K., Han W., Stewart D. E. (2011) *Numerical Solution of Ordinary Differential Equations*, John Wiley & Sons

[3] Billingham J., King A. C. (2000) *Wave Motion* Cambridge University Press

[4] Chow S. N., Deng B., Fiedler B. (1989) *Homoclinic Bifurcation at Resonant Eigenvalues*, Journal of Dynamics and Differential Equations, Vol. 2, Issue 2, pp. 177-244

[5] Chow V. T. (1959) *Open-Channel Hydraulics*, McGraw-Hill

[6] Dressler R. F. (1949) *Mathematical Solution of the Problem of Roll-Waves in Inclined Open Channels*, Communications on Pure and Applied Mathematics, Vol. 2, Issue 2-3, pp. 149-194

[7] Ehle B. L.(1968) *High order A-stable methods for the numerical solution of systems of differential equations*, BIT Numerical Mathematics, Vol. 8, pp. 276-278

[8] Hanratty T. J., Hershman A. (1961) *Initiation of Roll Waves*, AIChE Journal, Vol. 7, Issue 3, pp. 488-497

[9] Hanratty T. J., Engen J. M. (1957) *Interaction Between a Turbulent Air Stream and a Moving Water Surface* AIChE Journal, Vol. 3, Issue 3, pp. 299-304

[10] Kadri. U et al. (2009) *Prediction of the transition from stratified to slug flow or roll-waves in gasliquid horizontal pipes* International Journal of Multiphase Flow, Vol. 35, Issue 11, pp. 1001-1010

[11] Kay M. (2007) *Practical Hydraulics*, CRC Press

[12] Kierzenka J., Shampine L. F. (2001) *A BVP Solver Based on Residual Control and the MATLAB PSE* ACM Transactions on Mathematical Software, Vol. 27, Issue 3, pp. 299316

[13] Kierzenka J., Shampine L. F. (2008) *A BVP solver that controls residual and error*, http://faculty.smu.edu/shampine/finalbvp5c.pdf

[14] Kurganov A. and Tadmor E. (2000) *New High-Resolution Central Schemes for Nonlinear Conservation Laws and Convection-Diffusion Equations*, Journal of Computational Physics, Vol. 160 Issue 1, pp. 241-282

[15] Lighthill M. J., Whitham (1955) *On Kinematic Waves. I. Flood Movement in Long Rivers*, Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, Vol. 229, Issue 1178

[16] Mees A. I., Chua L. O. (1979) *The Hopf Bifurcation Theorem and its Applications to Nonlinear Oscillations in Circuits and Systems* IEEE Transactions on Circuits and Systems, Vol. 26, Issue 4, pp. 235-254

[17] Needham D. J. et al. (2007) *The Development of Slugging in two-layer hydraulic flows*, IMA Journal of Applied Mathematics, Vol. 73 Issue 1, pp. 274-322

[18] Needham D. J., Merkin J. H. (1984) *On Roll Waves Down an Open Inclined Channel*, Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences, Vol. 394 No. 1807, pp. 259-278

[19] Perko L. (2013) *Differential Equations and Dynamical Systems*, Springer Science & Business Media

[20] Richardson L. F. (1911) *The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, with an Application to the*

Stresses in a Masonry Dam, Philosophical Transactions, Royal Society London, DOI: 10.1098/rsta.1911.0009

[21] Shu, C. W., & Osher, S. (1987) *Efficient implementation of essentially non-oscillatory shock capturing schemes*, Journal of Computational Physics, Vol. 77 Issue 2, pp. 439-471

[22] Whitham G. B. (1974) *Linear and Nonlinear Waves*, John Wiley & Sons

[23] Wiggins S. (2013) *Introduction to Applied Nonlinear Dynamical Systems and Chaos*, Springer Science & Business Media

Appendix A Matlab code: KT Pipe Simulation

This code shows the Kurganov and Tadmor method for spatial discretization with a multi-level third order time discretization method. It is discussed in section 5.

```
function KT_Pipe_dt3(length,N,tend,dtout,F0,rho,lam,Rl)
    % Typical input variables
    % F0 = 4, rho = 0.1, lambda = 1, Rl = 50

    % Setup other variables for method
    x = linspace(-length,length,N);
    dx = x(2)-x(1);
    amp = 0.1;          % amplitude of disturbance
    b = 10;             % spread of disturbance
    U = ones(2,N);      % U = [h; q]
    U(1,:) = U(1,:) + amp*exp(-b*x.^2); % Initial perturbation
    U1 = U;             %U1, U2 needed for 3rd order time discretisation
    U2 = U;
    u_init = U;         % Store initial conditions for plotting later
    t = 0;
    tout = 0;

    % Iterations
    while t < tend
        % Kurganov & Tadmor (2000) Spatial Method
                % 3rd order Shu & Osher (1987) Temporal Method

        dt = dx/(4*max(abs(df(U))));    % Satisfy CFL condition

        L0 = L(U,dx,N);
        U1(:,2:N-1) = U(:,2:N-1) + dt*L0;
        L1 = L(U1,dx,N);
```

```
    U2(:,2:N-1) = U(:,2:N-1) + 0.25*dt*(L0 + L1);
    U(:,2:N-1) = U(:,2:N-1) + (1/6)*dt*(L0 + L1) + (2/3)*dt*L(U2,dx,N);

    % Now add time-stepped RHS terms
    U(2,2:N-1) = U(2,2:N-1) + ...
        dt*( ...
            ((U(1,2:N-1)+lam)./((1+lam).*U(1,2:N-1))) - ...
            U(2,2:N-1).*U(2,2:N-1)./(U(1,2:N-1).*U(1,2:N-1).*U(1,2:N-1)) + ...
            ( ...
                (U(1,3:N)+U(1,2:N-1)).*(U(2,3:N)./U(1,3:N) - U(2,2:N-1)./U(1,2:N-1)) - ...
                (U(1,2:N-1)+U(1,1:N-2)).*(U(2,2:N-1)./U(1,2:N-1) - U(2,1:N-2)./U(1,1:N-2))    ...
            )./((2*dx*dx*Rl).*U(1,2:N-1)) ...
        );

    % Periodic boundary conditions
    U(:,1) = U(:,N-1);
    U(:,N) = U(:,2);

    %Plotting
    t = t+dt;
    tout = tout+dt;
    if tout>dtout
        clf;
        str = sprintf('Flow at t = %d, using KT method', t);
        hold on
        title(str);
        plot(x,u_init(1,:),'Color',[0.5, 0.5, 0.5],'LineStyle',':')
        plot(x,U(1,:),'k')
        legend('Initial Conditions','Kurganov-Tadmor Solution','Location','northoutside')
        hold off
        drawnow();
        %Save image to file in current directory
        savefig(strcat('F0',num2str(F0),'_Rl',num2str(Rl),'_t',num2str(round(t))));
        tout = tout-dtout;
    end
end

function m = minmod(a,b)
            % minmod function
    m = 0.5*(sign(a)+sign(b)).*min(abs(a), abs(b));
end

function f = f(u)
            % Pipeline flow terms
    f = [u(2,:) ; (u(2,:).^2)./u(1,:) + 0.5*(1/(F0*F0))*(1-rho)*u(1,:).^2];
end

function df = df(u)
            % Wavespeed (max eigenvalue of df/du)
    df = u(2,:)./u(1,:) + sqrt(((1-rho)/(F0*F0))*u(1,:));
end

function L = L(U,dx,N)
    % As defined in Kurganov & Tadmor (2000), pg 18
    umx = zeros(2,N);
```

```
        ux = [diff(U(1,:))/dx; diff(U(2,:))/dx];
        umx(:,2:N-1) = minmod(ux(:,2:N-1),ux(:,1:N-2));
        uplus = U(:,2:N)-0.5*dx*umx(:,2:N);
        uminus = U(:,1:N-1)+0.5*dx*umx(:,1:N-1);
        a = max(abs(df(uplus)),abs(df(uminus)));
        H = 0.5*(f(uplus)+f(uminus))-0.5*[a;a].*(uplus-uminus);

        L = -(1/dx).*(H(:,2:N-1)-H(:,1:N-2));
    end
end
```

Appendix B Matlab code: BVP Solver

B.1 Periodic Solution and Wavelength Approximation

This code shows the periodic wave solver using bvp5c. It takes a guess from ode45, with the other parameters given.

```
function pipeline_bvp(Uin)
        % Calculate wavelength L using numerical continuation for gradually
        % incrementing U in vector Uin.

        % Standard parameter values
        B = -0.7;
        Rl = 50;
        rho = 0.1;
        lambda = 1;
        F0 = 10;

        % Can be used to calculate range of valid U
        [U_HB, U_hom] = U_bounds2(B,Rl,rho,lambda,F0);
        % Use a ode45 based function to guess initial solution
        [z,h,f,L_guess] = h_guess(B, Uin(1), Rl, rho, lambda, F0);
        solinit = [];
        for U = Uin
                if isempty(solinit)
                        solinit = bvpinit(linspace(0,1,20), f, L_guess);
                end
                % Use bvp5c to calculate L and waveform for each given U
                sol = bvp5c(@bvp2ode, @bvp2bc, solinit);
                solinit = sol;
                L = (sol.parameters);

                % Plot wave solution
                xint = linspace(0,1);
                sxint = deval(sol,xint);
                hold on
                plot(L*xint,sxint(1,:));
                title(num2str(U))
```

```
                hold off
                drawnow();
        end

%---------------------

    function dydx = bvp2ode(x,y,P)
                % ODE system to be solved
        dydx =  [P.*y(2)
            (P/B).*( ...
            B*(y(2).^2)./y(1) - ...
            (Rl./y(1)).*((1-rho)*(y(1).^3)/(F0^2) - B^2).*y(2) - ...
            (Rl./y(1)).*((B+U.*y(1)).^2 - (lambda + y(1)).*(y(1).^2)/(lambda+1)) ...
            )];
    end

%---------------------

    function res = bvp2bc(ya,yb,P)
                % Implements boundary conditions for res = 0
        res = [ ya(1)-yb(1)
            yb(2)
            ya(2)];
    end

%---------------------

end
```

B.2 Estimate Initial Sinusoidal Wave and Period

Used as an initalisation for pipeline_bvp.m.

```
function [z,h,f,period] = h_guess(B,U,Rl,rho,lam,F0)
    % Calculate h+, h-, he. Used to initialise ode45 with some h near h-
        h_plus = (B^2*F0^2/(1-rho))^(1/3);
    U_HB = (lam + h_plus)^(1/2)/(lam+1)^(1/2) + ((1-rho)*h_plus)^(1/2)/F0;
    F = @(H)-(lam+H)^(1/2)/(lam+1)^(1/2) - B/H - U_HB;
    h_e = fzero(F,[1e-5 h_plus]);
    F = @(H)(B+U_HB*H)^2-(lam+H)*H^2/(lam+1);
    h_minus = fzero(F,[h_e + 1e-3, h_plus - 1e-3]);

    [z,y] = ode45(@d_dx,linspace(0,10,1000),[h_minus + 0.01, 0.01]);

        % Calculalte period of wave by distance between peaks,
        % from changes of gradient +ve to -ve or -ve to +ve.
    N = numel(z);
    z1 = 0;
    z2 = 0;
    z3 = 0;
    for k = round(N/2):N-1
        if y(k,2)*y(k+1,2) < 0
```

```
            if z1 == 0
                z1 = (z(k)+z(k+1))/2;
            elseif z2 == 0
                z2 = (z(k)+z(k+1))/2;
            elseif z3 == 0
                z3 = (z(k)+z(k+1))/2;
            end
        end
        if z1 ~= 0 && z3 ~= 0
            break
        end
    end
        % Calculate key wave characteristics
    period = z3-z1;
    amplitude = max(y(z>5,1))-min(y(z>5,1));
    center = mean(y(z>5,1));

    % Approximate periodic wave using a cosine function
    f = @(z)[-(amplitude/2).*cos(2*pi*z/period) + center
             (amplitude*pi/period).*sin(2*pi*z/period)];

    h = y(:,1);
    %------------------------
    function v = d_dx(z,y)
        % y = [h,w], this is the ODE to solve
        v = zeros(2,1);
        v(1) = 1.*y(2);
        v(2) = (1/B).*( ...
                B*(y(2).^2)./y(1) - ...
                (Rl./y(1)).*((1-rho)*(y(1).^3)/(F0^2) - B^2).*y(2) - ...
                (Rl./y(1)).*((B+U.*y(1)).^2 - (lam + y(1)).*(y(1).^2)/(lam+1)) ...
               );
    end

end
```

B.3 Calculating U_{HOM}

This code uses a bisection approach to approximating U_{HOM}.

```
function [U_HB, U_hom] = U_bounds2(B,Rl,rho,lam,F0)
    % Omitted -- calculate h-, h+, he as in "h_guess.m"
        % surpess (non catastrophic) warning which is issued if ode45 doesn't converge
    warning('off', 'MATLAB:ode45:IntegrationTolNotMet')

    U_shift_a = 0.00;   % Initial left value for bisection
    U_shift_b = 0.5;    % Initial right value for bisection
    h_maxs = [];                % Maximum value of h for plotting
    h_mins = [];                % Minimum value of h for plotting
    tol = 1e-5;                 % Tolerence between left and right U in bisection
    while U_shift_b - U_shift_a > tol
                % Iterate bisection until tolerence reached
```

```
        U_shift = (U_shift_a + U_shift_b)/2;
        U = U_HB + U_shift;
        % Solve ODE system over the domain [0,10]. This is long enough for
                % solution to begin to stabilise to some periodic wave
        [z,y] = ode45(@d_dx,linspace(0,10,1000),[h_minus + 0.01, 0.01]);

        h_maxs = [h_maxs; [U, max(y(:,1))]];
                % Try used as h_min does not exist if U>U_HOM, since solution exponential
        try
            h_mins = [h_mins; [U, min(y(300:end,1))]];
        end

        if min(y(:,2)) > 0
                        % If no wave solution was found, w > 0 for all z, so U > U_HOM
            U_shift_b = U_shift;
        else
                        % Else wave solution found so U < U_HOM
            U_shift_a = U_shift;
        end
    end

    warning('on', 'MATLAB:ode45:IntegrationTolNotMet') % Reinstate disabled warning

    U_hom = U;
    %------------
    function v = d_dx(z,y)
        % Omitted -- ODE function as in "h_guess.m"
    end
end
```