



## १. Introduction

---

Welcome to BUBBLE Toolbox Release 2000. This manual provides an insight to expedite the efficiency of your bubbling analysis. This manual is targeted at the intermediate to advanced MATLAB user. Working knowledge of GUIs in MATLAB is assumed and knowledge of GUI design is a bonus. The entire BUBBLE Toolbox has been successfully converted to an object oriented approach from the conventional variable based approach. So, familiarity with basic object oriented programming structures is required. Let the reader be cautioned that the author of this toolbox has had no training in computer programming and the entire of the BUBBLE Toolbox has been an intuitive approach. No claims are made to the programming logic used, being the best or the most efficient. The author does seek the reader's comments and feedback on alternative programming strategies and bugs.

This manual has been divided into sections for convenient reference. The first section deals with installation and general set-up of the BUBBLE Toolbox. The second & third sections brings the user to a level of operating knowledge. The fourth section reveals the underlying logic behind the GUI, and a general programming philosophy used. The last section is for the advanced user, which will deal with the source code behind the BUBBLE Toolbox. It is targeted so that the user will be able to make their modifications to the BUBBLE Toolbox routines and customize the source code. The author strictly believes in survival of the fittest in the open source code arena. So the user is cautioned against the excessive customization of code and advised to keep one master back up of the original BUBBLE Toolbox!



2

## 2. Installation and set-up

---

The BUBBLE Toolbox is located in the root directory of the supplied CD-ROM under the folder name 'bubble'. (figure 1)



Figure 1.

This folder should contain the various toolbar folders, which contain the bubble subroutine files. The BUBBLE Toolbox can be run either from the CD-ROM or it can be copied to a local hard drive from where it can be accessed locally.



---

*The BUBBLE Toolbox requires MATLAB version 5.3 or higher and does not support earlier versions of MATLAB. This is because the newer MATLAB versions have capability to store GUI figures as '\*.fig' files instead of the conventional '\*.mat' files.*

---

The MATLAB engine has to be installed on a local drive. This toolbox will not run with MATLAB running on a server. Ideally, a local installation is best suited for efficient operation. But, running from a local CD-ROM also gives satisfactory results especially where disk space is at a premium.



## 2.2. Installation On Local Drive

Copy the BUBBLE Toolbox to the 'toolbox' folder under MATLAB root directory. A typical MATLAB root is shown in figure 2.2. This can be done by a simple drag and drop operation from the CD-ROM to the MATLAB toolbox root directory.

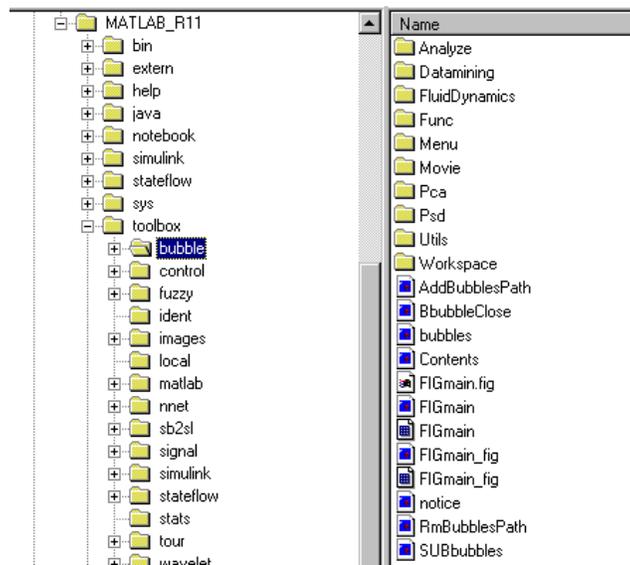


Figure 2. MATLAB directory structure

Once the toolbox folder is created, the user will need to add the path for this folder to the MATLAB search path. This can be done by using the path browser in the main command window. Click on 'Add to path' in the Path menu. Choose the folder you just copied and

click 'OK'. On closing the path browser by clicking on the top right hand-side corner button, MATLAB asks if the path definitions are to be saved for future references. It is recommended that the path definitions are saved for future references to avoid the agonizing process of adding path every time. So, click 'Yes'. On typing 'path' at the command window, the list of paths on MATLAB's search order will show the BUBBLE Toolbox folder on the top.

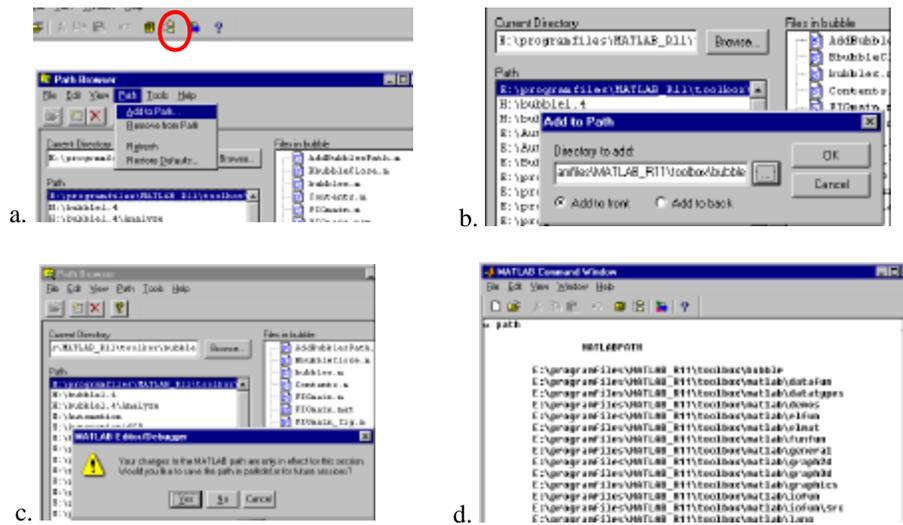


Figure 3. Adding BUBBLE Toolbox to MATLAB's search path

The MATLAB BUBBLE Toolbox is now ready for use. To run, type in 'bubbles' at the MATLAB main command window.

## 2.2. Running From Cd-Rom



*(or any other removable storage device)*

---

This option is especially attractive for one-time power demonstrations on mystic bubbles. This option also makes the execution a trifle slower than running locally on a hard drive. Once MATLAB is started, change directory to the BUBBLE Toolbox. Once in the BUBBLE Toolbox root directory, type in 'bubbles' and the BUBBLE Toolbox setup will ensure that all the relevant paths are added and scratch directory work paths set up.



*Note: For read-only devices, it is essential that at least one scratch directory with read/write access is provided. The BUBBLE Toolbox uses the MATLAB root directory as the default scratch directory. If that is read-only, then the user is prompted to enter a new scratch directory.*

---



## 3. Initialization

---

To initiate the BUBBLE Toolbox type in 'bubbles' at the main command window. The following demonstrates a typical BUBBLE Toolbox initialization sequence.

### **bubbles**

```
root =  
E:\programfiles\MATLAB_R11\toolbox\bubble  
Adding BUBBLY Toolbar directories to MATLABpath...  
  
E:\programfiles\MATLAB_R11\toolbox\bubble\Analyze  
E:\programfiles\MATLAB_R11\toolbox\bubble\Datamining  
E:\programfiles\MATLAB_R11\toolbox\bubble\FluidDynamics  
E:\programfiles\MATLAB_R11\toolbox\bubble\Func  
E:\programfiles\MATLAB_R11\toolbox\bubble\Menu  
E:\programfiles\MATLAB_R11\toolbox\bubble\Movie  
E:\programfiles\MATLAB_R11\toolbox\bubble\PCA  
E:\programfiles\MATLAB_R11\toolbox\bubble\PSD  
E:\programfiles\MATLAB_R11\toolbox\bubble\Utils  
E:\programfiles\MATLAB_R11\toolbox\bubble\Workspace  
E:\programfiles\MATLAB_R11\toolbox\bubble  
Done!
```

---

The BUBBLE Toolbox for Chaotic Analysis of Time Series

Version 1.4 Release 2000

Requires MATLAB version 5.3 or higher

Thank you for using this toolbox  
You may freely distribute this package  
Report bugs to [ssarnobat@hotmail.com](mailto:ssarnobat@hotmail.com)

Contact us at:

Department of Chemical Engineering  
419 Dougherty Hall  
The University of Tennessee  
Knoxville TN 37996, USA

Telephone: 865 974 5317  
Fax: 865 974 7076

Copyright 2000, Sachin Udaya Sarnobat  
The University of Tennessee, Knoxville  
All rights reserved

---

-----Scratch directory set as: E:\programfiles\MATLAB\_R11-----

The BUBBLE Toolbox initialization sets the root as the directory in which the BUBBLE Toolbox is installed. It then proceeds to add all the Toolbox directories to the MATLAB path to access the functions. The scratch directory is set as the MATLAB root directory. If that directory is found to be read-only, the user is prompted to enter a new scratch directory. The 'bubblesetup.mat' created in the MATLAB root directory stores the workpath preferences for the BUBBLE Toolbox.

The BUBBLE Toolbox main window displayed is shown. It should be noted that the main window is screen resolution dependent. For screen resolution of 800x600 the GUI displayed is shown in figure 4. Window will appear maximized.

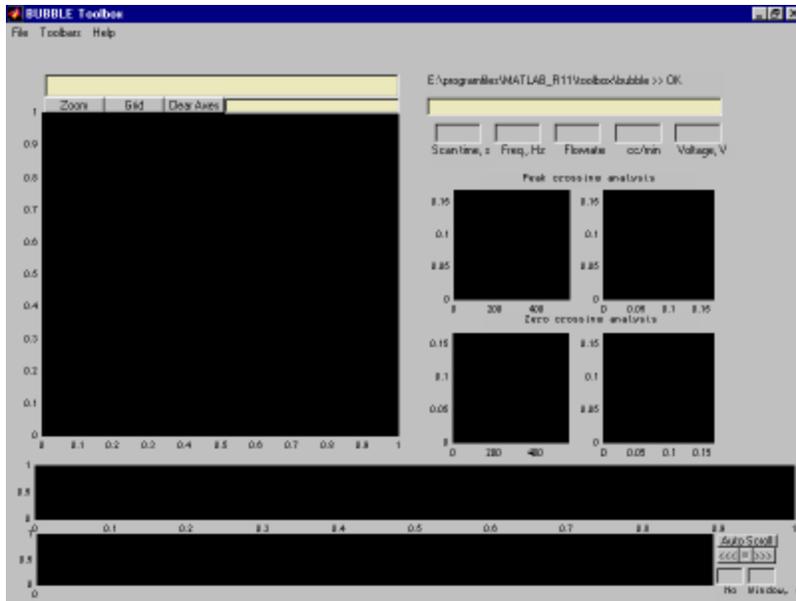


Figure 4. Screen resolution 800x600

For the higher screen resolution of 1024x800, the window is shown in figure 5 where the window covers only a part of the screen and can be maximized. The BUBBLE Toolbox retains full functionality both in the semi-minimized and maximized mode. On maximizing the window, additional toolbar features are accessible in the main window. It should be noted though that the same toolbar features are available as floating windows.

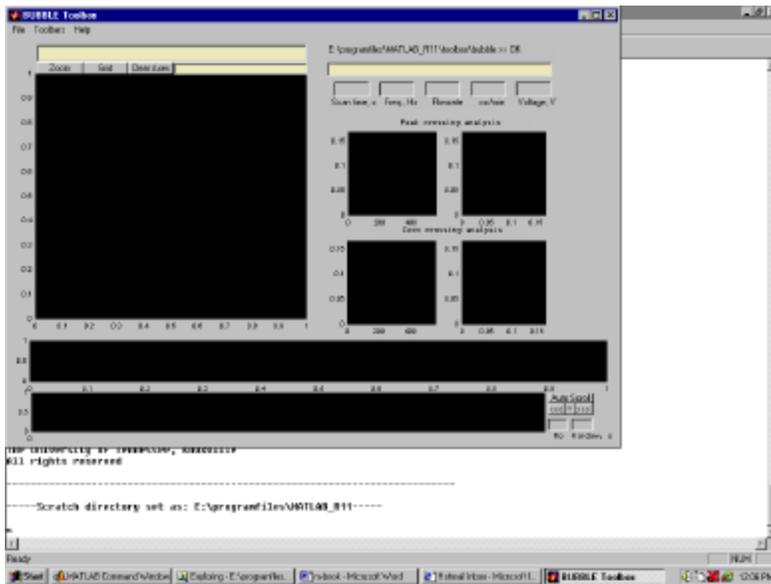


Figure 5. Screen resolution 1024x800



Figure 6. Maximized screen for a screen resolution of 1024x800

The best way of explaining the working of this toolbox will be through a demonstration. To open a file there are several ways:

- Under the 'File' menu choose 'Open...'
- The infamous 'Cntrl-O'
- Click the 'Load' button on the toolbar to the right. If the window toolbars are not available, then call the floating 'Workspace' toolbar by selecting Toolbars|Workspace (or 'Cntrl+W')
- Use the command line on the BUBBLE Toolbox. (For advanced users)

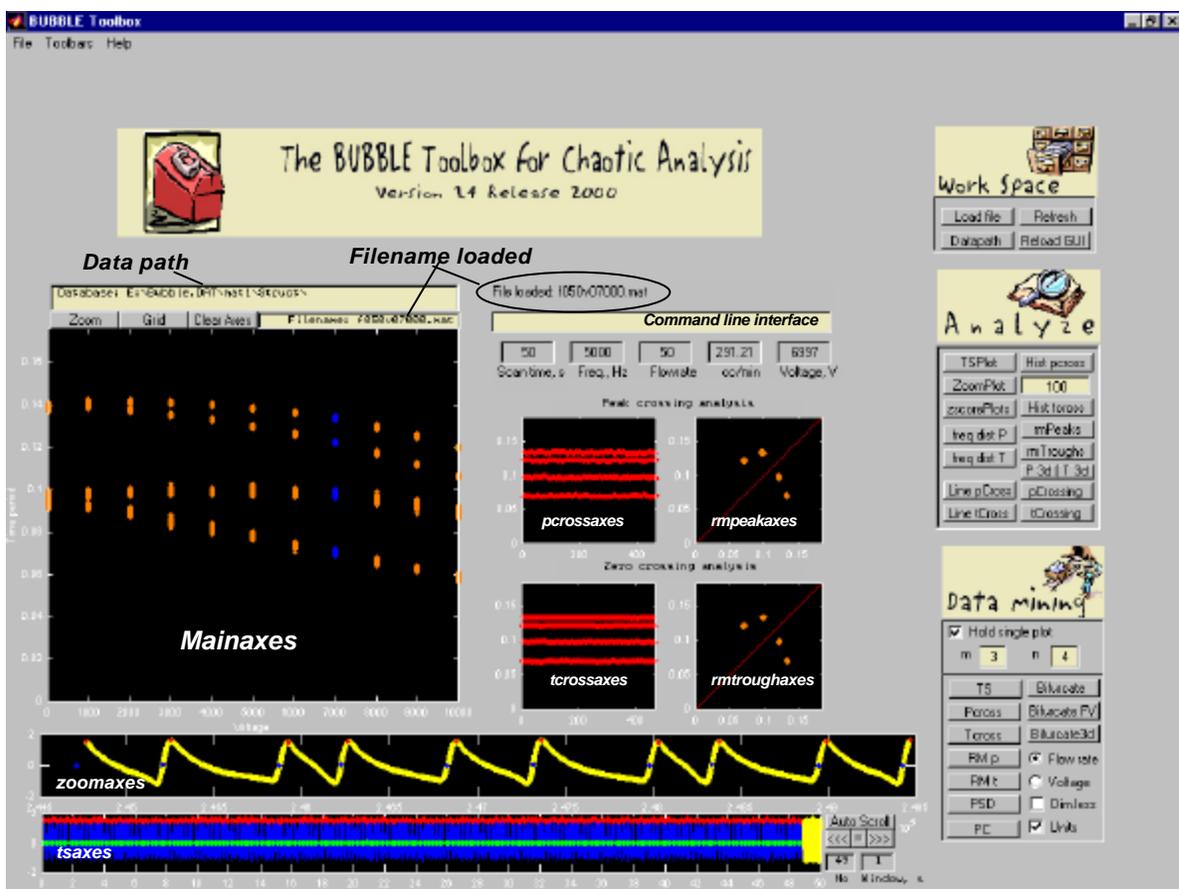


Figure 7. BUBBLE Toolbox main window

Once the data file has loaded the main BUBBLE Toolbox window displays all the information about the datafile. Upon loading a file, the mainaxes display the bifurcation diagram at constant flowrate. The current file loaded is marked in blue. The 'tsaxes' displays the whole of the pressure time series with the red dots marking the peaks and the green dots marking the zero-crossing points. The threshold for the zero-crossing is taken as the mean of the entire pressure time series. The yellow highlight on the tsaxes is magnified on the zoomaxes. The pcrossaxes and tcrossaxes show the crossings from peak and zero-crossing analysis respectively. The time return maps obtained from crossings are shown on the rmpeakaxes and rmtroughaxes respectively.

Information about the run such as the scan time (the total length of the data file), frequency at which the data was collected, the flow rate (both in absolute rotameter units as well as in units of cc/min) and the voltage applied to the nozzle is shown (Figure 8).

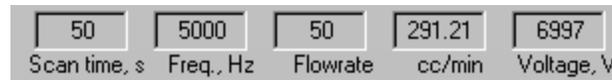


Figure 8. Run information

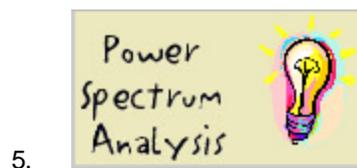
The data is now loaded and ready for analysis.



## 8. Toolbars

---

Various analysis approaches are covered in the BUBBLE Toolbox. Different toolbars are provided for detailed analysis. The default toolbars, which are shown on the main BUBBLE Toolbox window, are:



8.2.

## Workspace



---

This toolbar primarily targets the workspace and the path handling (figure 9).

The **Datapath** button sets the path of the database in study.

In case anything goes wrong and plots on the wrong plot, or 'rotate3d' messes up (as always), you can safely reload the GUI without having to load the data again.

1. Hit **Reload GUI** and then the GUI reloads with a blank axes.
2. Hit **Refresh** to refresh the axes.



Figure 9. The Workspace toolbar

### **Legend**

---

<b>Load file</b>	Load workspace
<b>Datapath</b>	Set data path for database
<b>Reload GUI</b>	Reload GUI only without clearing the workspace
<b>Refresh</b>	Refresh the GUI axes with workspace data

---

Table1. List of Workspace toolbar commands

## 8.2. Analyze

The analyze toolbar is primarily for analyzing the current run in question (figure10). It provides the tools to analyze different aspects of the current run like crossings, the time series, and offers zoom capability. All the analysis is carried out on the *mainaxes*. The analyze toolbar is primarily to view the data for the current run.

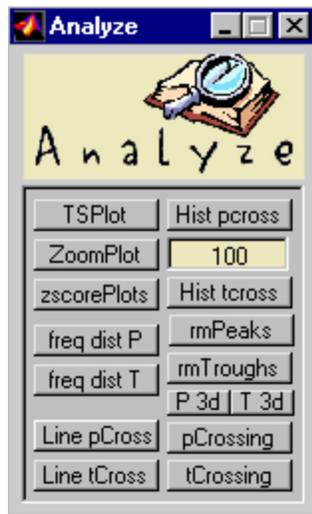
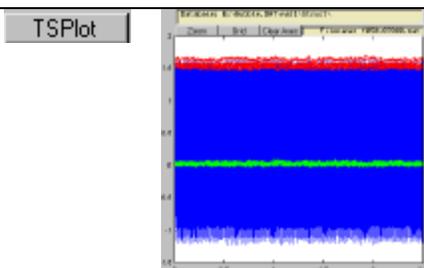


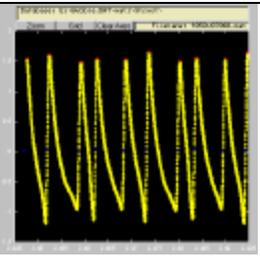
Figure 10. The analyze toolbar

### **Legend**



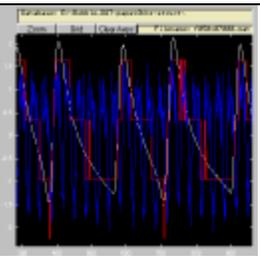
Plot the entire time series on mainaxes. The red dots mark the peaks and the green dots mark the zero crossings with the mean of the entire time series used as the threshold. It will be noted though that the threshold is almost zero. This is because the bubbling pressure fluctuates around the ambient pressure.

ZoomPlot



Plot the highlighted portion of the time series on the mainaxes.  
 The zero crossings are plotted in blue. The time series is plotted as points as opposed to lines in **TSPlot**

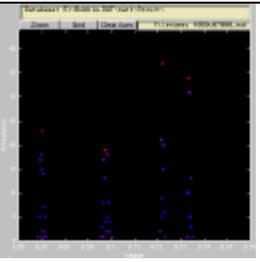
zscorePlots



Plot the profiles of pressure, voltage and current normalized by the standard deviation on the mainaxes to analyze correlation. The voltage is in blue, the pressure time series is in white and the current is plotted in red.

freq dist P

freq dist T

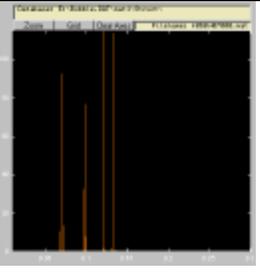


The time return map can have a number of points on a single visible point on the map. To identify the number of points, the frequency distribution of the points is given by this function. The red points are the peaks identified. The bin resolution is 0.0001, which is the same resolution of the data taken from LABView. Histogram plots can also be plotted with variable resolution (variable number of bins in the range of the period of formation).

Hist pcross

100

Hist tcross

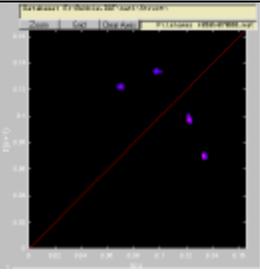


This function specifies the number of points in each frequency bin which between the maximum and the minimum period of formation of the bubbles. The spectrum is divided into a user-specified number of equal frequency bins and the number of points in each is calculated.

rmPeaks

rmTroughs

P 3d | T 3d



Plots the intensity return maps for the crossings. The lighter points indicate higher density of points. The fraction of points below a certain threshold is marked by a constant darker color.

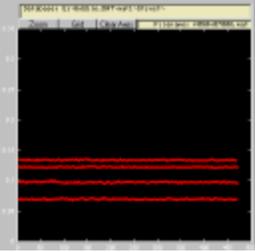
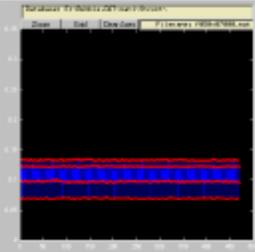
<p>pCrossing tCrossing</p>		<p>Plots the crossings vector on the mainaxes. The x-axis marks the index number of bubbles formed.</p>
<p>Line pCross Line tCross</p>		<p>To visualize the pattern of periods of formation, this function gives the periods of formation connected with lines. This function is exactly similar to the pCross and tCross functions, except that it also connected the points with lines.</p>

Table 2: List of Analyze toolbar commands

## 8.3. Data-mining



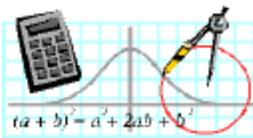
The data mining tools are used to visualize the current run with respect to the entire database (figure 15).

The plots can be made to visualize the run either at constant flow rate or at constant voltage.

There are two basic plots in this toolbar:

- Subplots
- Bifurcation plots

Both can be made with either constant flow rate or constant voltage.



## Subplots

If the 'Hold single plot' option is not checked, then the toolbar plots the series of runs in the database either at constant voltage or at constant flow rate, which is chosen by the respective radio buttons. The number of subplots is chosen by the 'm' and 'n' variables on the toolbar, which represent 'm' rows and 'n' columns on the subplot. The run information is mentioned as the title of each subplot. The nature of the title can be chosen with the 'Dim. Less', (which stands for 'dimensionless'), and the 'Units' checkboxes.

Example:

The following example shows the subplot option for the peak return maps (RM p) at constant flow rate. The units' checkbox was not selected, so the titles on the subplots are in absolute rotameter units. The plots are color coded so that the lighter colors indicate the runs with higher voltages. The colors vary with both voltage and flow, so that the plots for different voltages at different flow rates will have different colors. Each run is thus identified by a unique color.

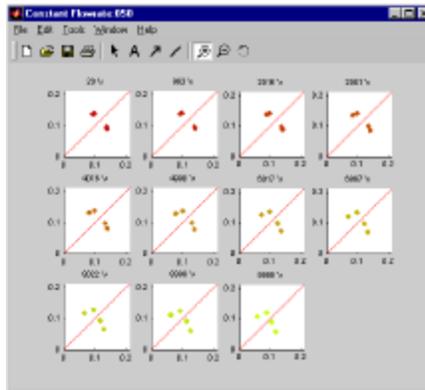


Figure 11. Subplots

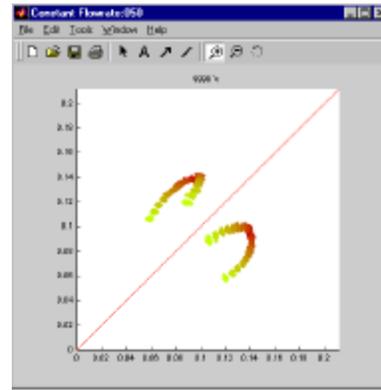


Figure 12. Hold plot

The units checkbox is chosen if SI units are desired as titles. This converts the flow rate, which are in absolute rotameter units, to cc/min. The units of the applied electrostatic potential are 'volts' in both cases.

If the dimensionless checkbox is chosen then for the constant flow rate case, Tsuge's flow rate number is plotted as the title for each subplot. For the constant voltage case, the electric Bond number is used as the title.



## Bifurcation Diagrams

Bifurcation diagrams are visualization tools to manifest the effect of the control variable on the periodicity of the system. In order to determine the relative effect of each of the bifurcation variables on the system, bifurcation tools are provided for analyzing the current with respect to the whole database.

The **Bifurcate** button, bifurcation for the current run at either constant voltage or constant flow rate is shown. Here the bifurcation diagram is a 2-d plot with the period of formation plotted against either voltage or flow rate. The units of the control variable can be chosen by using the check boxes for 'Units' and 'Dim.less' so that the control variable is represented in terms of dimensionless units, in regular units or absolute measurement units. (figure 13)

The **Bifurcate FV** button is exactly similar to the 'Bifurcate' button, except that it plots both constant voltage and constant flow rate bifurcation diagrams on the same plot. (figure 14)

The **Bifurcate3d** button is used to visualize the whole database at the same time on a single plot. This button plots the data one run at a time creating a neat animation effect showing how the database is built. Here, the runs with the current voltage are marked in blue so that the current run at constant flow rate with respect to the entire database can be studied.

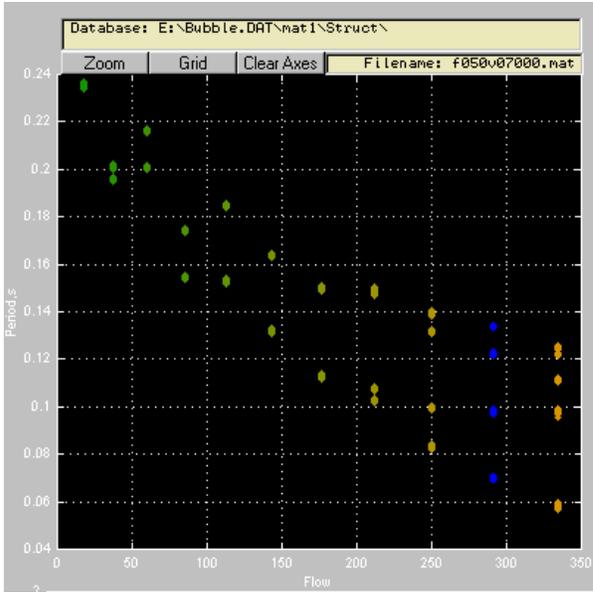


Figure 13. Bifurcation at constant voltage

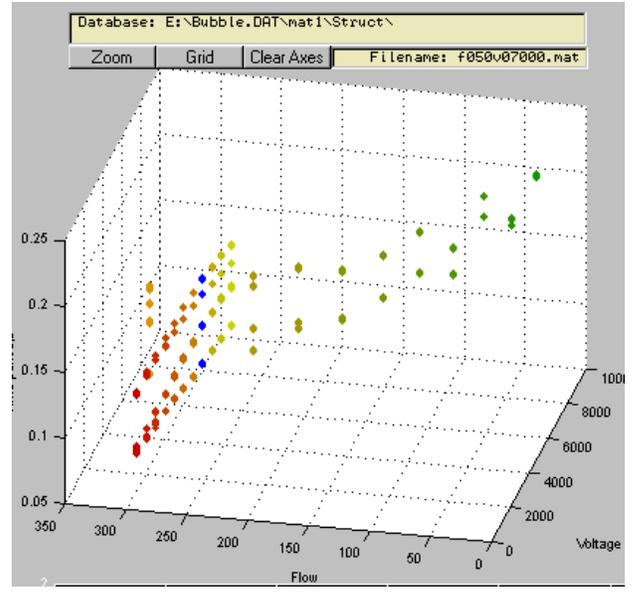


Figure 14. Bifurcation with both voltage and flow rate

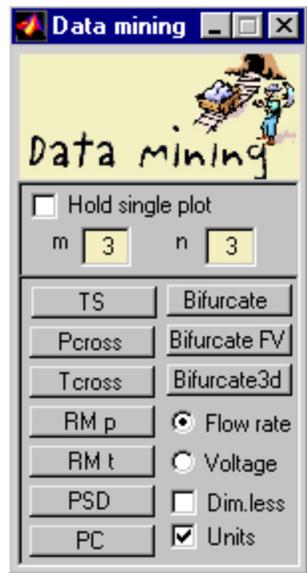


Figure 15. The Data-mining Toolbar

**Legend**

---

TS	Subplot with pressure time series
Pcross Tcross	Subplot with crossings
RM p RM t	Subplot with return maps
PSD	Subplot with power spectrum distribution
PC	Subplot with principal component analysis
Bifurcate	2-d Bifurcation at either constant flow or voltage depending on the radio button
Bifurcate FV	3-d bifurcation with both flow and voltage variation at the current run
Bifurcate3d	3-d bifurcation diagram of the entire

	database
<input checked="" type="radio"/> Flow rate <input type="radio"/> Voltage	Radio button to set either constant flow or constant voltage
<input type="checkbox"/> Dim.less <input checked="" type="checkbox"/> Units	Checkbox for dimensionless units, use of CGS units as the title. If both are left unchecked, absolute units are used.
m <input type="text" value="3"/> n <input type="text" value="3"/>	Array of subplots
<input type="checkbox"/> Hold single plot	Hold a single 'subplot' and plot all the runs in a single figure axis rather than creating several sub-plots

Table 3. List of Data-mining toolbar commands

## 8.8. Phase space analysis

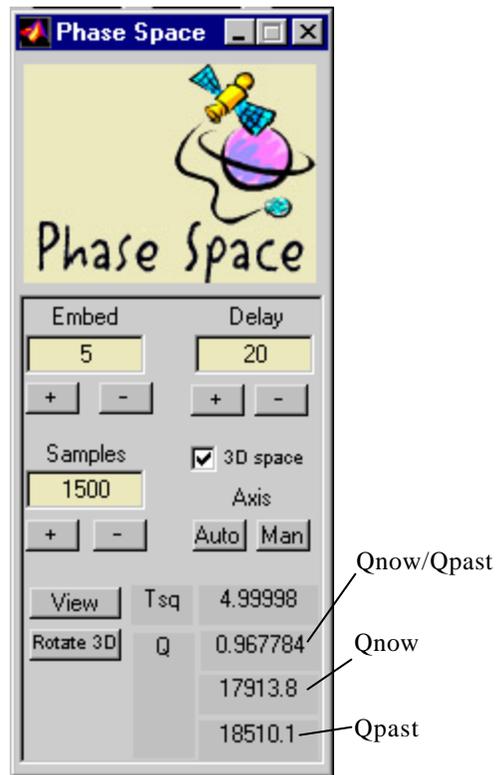


Figure 16. Phase Space Analysis (PCA) Toolbar

The buttons for this toolbar are self-explanatory (figure 16). Use the **View** button to view the phase space. The text labels give the  $T^2$  and the Q statistic. The Q labels also give the ratio of the current Q to the last Q, the value of the current Q and the value of the past Q.

Example figures:

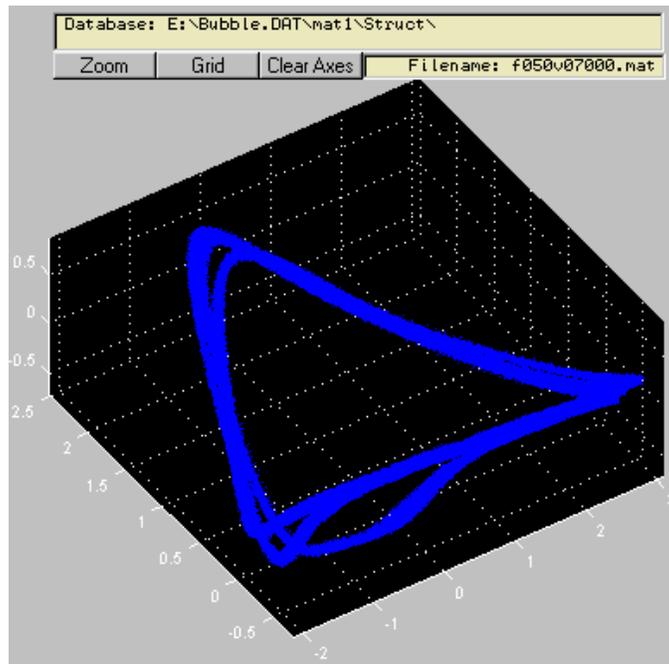


Figure 17. Principal component space

Note: The variation of the principal component space can be observed through the data mining toolbar (figure 18)

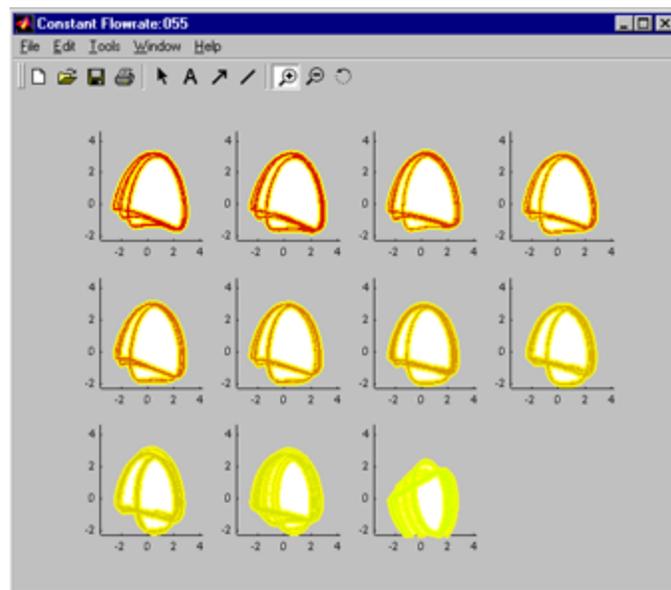


Figure 18. Variation of principal component space at constant flow rate

## 8.4. Power Spectrum Analysis



This toolbar basically allows convenient PSD analysis (figure 19). It has a built in periodicity-detector which can suggest the periodicity of the current run making use of the PSD information.

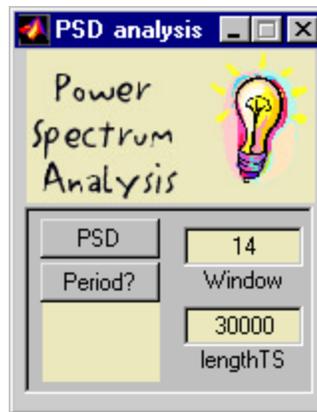


Figure 19. Power Spectrum Analysis

### **Legend**

PSD	Show power spectrum distribution on mainaxes
Period?	Suggest periodicity of the current run
14 Window	Exponent for base 2 for the window size. Default is $2^{14} = 16,384$ . The default length of the time series is 30,000 data points.
30000 lengthTS	

Table 4. List of PSD toolbar commands

Example:

The power spectrum distribution is shown in figure 20. The yellow and blue dots on the power spectrum indicate the peaks identified by the peak identification routine. Blue dots indicate the peaks suggested by their location on the frequency axis and the yellow dots show the refined peaks. A maximum limit of 40 Hz on the frequency axis is chosen because bubbling frequencies above 40 Hz are not observed.

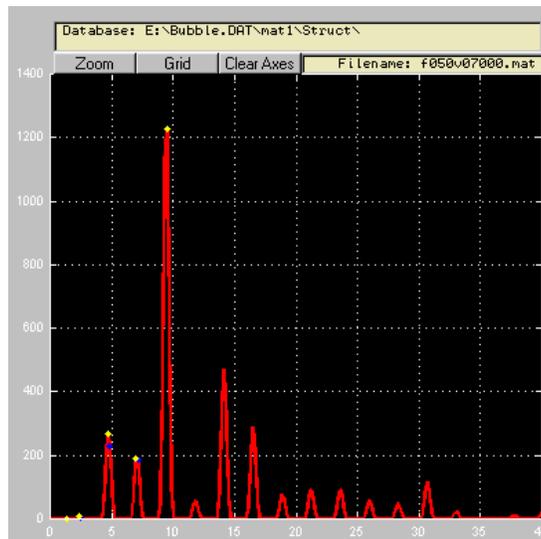


Figure 20. Power spectrum distribution



4

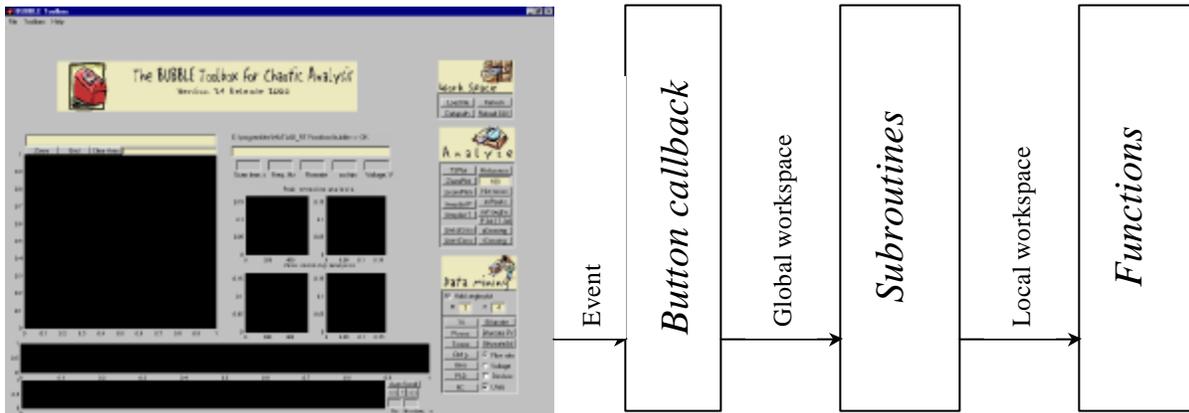
## 4. Back-stage

---

This section gives the programming and logic philosophy behind the BUBBLE Toolbox. The BUBBLE Toolbox is configured so that the GUI is just a front-end tool, which can be customized and changed to suit the needs of the advanced bubble expert once the underlying order of the BUBBLE Toolbox is understood.

The main bubble workspace has been configured such that it is minimally cluttered with global variables. The earlier versions of the bubble toolbox, up to version 1.3 had all the variables in the global workspace. But then in view of better programming etiquette and a faster execution time for the bubble functions, a semi-object oriented approach was adopted.

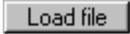
The GUI is created using the 'guide' command in MATLAB. Though the utility has bugs galore, it gives excellent results with a little patience and some creativity. Every object on the GUI such as a 'pushbutton' has a 'callback' property, which has a character array, which is executed when the object is activated. Instead of writing the callback scripts in the GUI callback editor, the GUI is created with a single script name in each of the callback



properties of every object. The callback script is then modified to change the required callback properties of every object. This not only saves on time by allowing access of the callback scripts without calling the guide utility, it also circumvents the ulterior motives of MATLAB, which seek to hamper any modifications of the GUI.

Each of these button callbacks will be referred to as 'events' and the scripts are referred to as 'buttons'. The scripts are named with a certain philosophy in mind. The letter 'B' is added to the name of the push button and a script 'm' file calls other functions and subroutines.

For example: Bloodfile.m.

Here Bloodfile is the callback for the pushbutton, . This takes variables from the global workspace and passes them to subroutines in the local workspace. Subroutines are functions called by buttons and cater only to those events. Subroutines further call functions which are more generic and can be used by other toolboxes.

For example: Bloadfile calls SUBloadfile.m

The subroutines may contain local functions, which are called by the subroutine. The advantage with functions over scripts is mentioned in MATLAB's reference manual, which says that functions are parsed and executed faster over scripts, which are interpreted line by line.

The data required for the BUBBLE Toolbox is in a specific format and can be created using the BUBBLE automation and database creation software included as a part of the BUBBLE Suite.

The files have a data object aptly named 'data'. It has the following variables:

**data**

```
data =  
      TS: [250000x1 double]  
      pcross: [1x259 double]  
      tcross: [1x326 double]  
      pat: [1x259 double]  
      tat: [1x327 double]  
      f: 25  
      v: 2015  
      currentProfile: [250000x1 double]  
      voltageProfile: [250000x1 double]  
      i: 115  
      iseed: 120  
      frequency: 2000  
      filename: 'f025v2015'  
      pathname: 'e:\automation'  
      user: []
```

A User field has been added to save any user defined objects. The main objective of this is to clear the global workspace.

With the passing of the 'data' object to each function, consolidating the Bubble GUI variables into objects is done. The object is named 'main' with a view to be in the workspace of the *main* calling function.

Capitalized names are chosen with a view that the variables were global in nature in the earlier toolbox release and so as to indicate the universal applicability of the paths to all the subroutines and functions.

```
main
main =
    root: 'f:\bubble'
    DATAPATH: []
    WORKPATH: 'D:\MATLABR11'
    TEMPPATH: []
    CURRENTPATH: 'f:\bubble'
    filename: []
    data: []
    color: [1x1 struct]
    user: []
```

The object 'main' is created when the bubble toolbox is loaded into the memory. This is the variable, which is passed to all the functions. The data from files is assigned to the 'data' field in the object and remains constant until changed by the 'open' function for a new file load i.e. this is the core work variable and is unaffected by the operations carried out by inside the toolbox.

## 4.2. bubble root

---

The bubble root has the following files

```
dir *.m
```

```
AddBubblesPath.m  RmBubblesPath.m  contents.m
BbubbleClose.m    SUBbubbles.m      notice.m
FIGmain.m         bubbles.m
```

## 4.2. workspace

---

The workspace menu has the following file structure:

```
dir *.m
```

```
Bbifurcatef.m  Brefresh.m  FUNplotTS.m  SUBreprocess.m
Bcdpath.m      Breload.m   FUNzoomTS.m  SUBscrollAUTO.m
Bclear.m       Breprocess.m  FUNzoomaxes.m  SUBscrollLEFT.m
Bclearall.m    Bscroll.m   SUBclear.m    SUBscrollRIGHT.m
Bclearaxes.m   BscrollAUTO.m  SUBclearall.m  SUBscrollSTOP.m
Bclearmainaxes.m  BscrollLEFT.m  SUBclearaxes.m  contents.m
Bdatapath.m    BscrollRIGHT.m  SUBdatapath.m
Bloadfile.m    BscrollSTOP.m  SUBloadfile.m
Bloadmenu.m    FIGloadmenu.m  SUBrefresh.m
```

*buttons*

1. Bloadfile; SUBloadfile
2. Bdatapath; SUBdatapath
3. Breload
4. Bcdpath
  
5. Bclear; SUBclear
6. Bclearaxes; SUBclearaxes
7. Bclearall; SUBclear
  
8. Bscroll;
9. BscrollAUTO; SUBscrollAUTO;

```
10.BscrollRIGHT;SUBscrollRIGHT
11.BscrollLEFT;SUBscrollLEFT
```

*functions*

```
12.FUNzoomaxes
13.FUNplotTS
14.FUNzoomTS
```

*figures*

```
15.FIGloadmenu
```

## 4.3. data mining

---

The Data Mining Menu has the following files:

**dir \*.m**

BRflow.m	BfTS.m	FIGdatabasemenu.m	SCRplotRMp.m
BRvoltage.m	Bfpcross.m	FUNbifurcate.m	SCRplotRMt.m
Bbifurcate.m	Bftcross.m	FUNbifurcate3d.m	SCRplotT.m
Bbifurcate3d.m	BrefreshDim.m	FUNextractFV.m	SCRplotTS.m
BbifurcateFV.m	BrefreshFV.m	FUNf.m	SCRtcross.m
Bdatabasemenu.m	BrefreshHold.m	FUNgetmn.m	SUBbifurcateFV.m
BfFFT.m	BrefreshUnits.m	FUNpcspace.m	SUBrefreshFV.m
BfPC.m	BsyncDim.m	SCRpcross.m	
BfPSD.m	BsyncHold.m	SCRplotP.m	
BfRMp.m	Bsyncmn.m	SCRplotPC.m	
BfRMt.m	Bsyncunits.m	SCRplotPSD.m	

*buttons*

1. Bbifurcate; FUNbifurcate
2. BbifurcateFV; SUBbifurcateFV(main);FUNbifurcate(3,main);
3. Bbifurcate3d; FUNbifurcate3d(main);

The Buttons for subplots call a central function FUNf.m which further calls script files, which are preceded with the letters 'SCR'. The SCR files operate in the local workspace of the FUNf function.

4. BfTS; SCRplotTS
5. Bfpcross; SCRplotP
6. Bftcross; SCRplotT
7. BfRMp; SCRplotRMp
8. BfRMt; SCRplotRMt

9. BfFFT; SCRplotPSD
10. BfPC; SCRplotPC

These refresh the value of user defined attributes for plotting

11. BrefreshDim
12. BrefreshFV
13. BrefreshHold
14. BrefreshUnits

The BR files (where R stands for 'radio' and 'B' for 'button'), create the radio button effect for either constant flow or constant voltage

15. BRflow
16. BRvoltage

The 'sync' files synchronize the floating toolbars and the toolbars in the main window

17. BsyncDim
18. BsyncHOLD
19. Bsyncmn
20. Bsyncunits

*subroutines*

21. SUBbifurcateFV
22. SUBrefreshFV

*functions*

23. FUNbifurcate
24. FUNbifurcate3d

25. FUNf

The FUNf function searches the database for files with either the same flowrate or the same voltage through 'dir f\*v\*.mat'-type commands and generates a list of files. Then it proceeds to load each of those files into memory and calls the SCR script file which operates on those variables.

26. FUNgetmn
27. FUNpcspace

*scripts*

28. SCRpcross.m
29. SCRplotP.m
30. SCRplotPC.m
31. SCRplotPSD.m
32. SCRplotRMp.m
33. SCRplotRMt.m
34. SCRplotT.m
35. SCRplotTS.m
36. SCRtcross.m

## 4.8. analyze

---

The analyze toolbar has the following file structure:

`dir *.m`

<code>BfreqdistP.m</code>	<code>BplotTS.m</code>	<code>BsyncFreqbins.m</code>	<code>SUBcrossing.m</code>
<code>BfreqdistT.m</code>	<code>BplotZSCORE.m</code>	<code>Btcrossing.m</code>	<code>SUBfreqdist.m</code>
<code>Bhistpcross.m</code>	<code>Bplotmenu.m</code>	<code>BzoomPlot.m</code>	<code>SUBhistcross.m</code>
<code>Bhisttcross.m</code>	<code>Brmpeaks.m</code>	<code>Bzoomaxes.m</code>	<code>SUBlinecrossing.m</code>
<code>Blinepcrossing.m</code>	<code>Brmpeaks3.m</code>	<code>FIGplotmenu.m</code>	<code>SUBrm.m</code>
<code>Blinetcrossing.m</code>	<code>Brmtroughs.m</code>	<code>FUNfdrmpplot.m</code>	<code>SUBrm3.m</code>
<code>Bpcrossing.m</code>	<code>Brmtroughs3.m</code>	<code>FUNzplots.m</code>	

*buttons*

1. `BfreqdistP; SUBfreqdist`
2. `BfreqdistT; SUBfreqdist`
3. `Bhistpcross; SUBhistcross`
4. `Bhisttcross; SUBhistcross`
5. `Blinepcrossing; SUBlinecrossing`
6. `Blinetcrossing; SUBlinecrossing`
7. `BplotZSCORE; FUNzplots`
8. `Brmpeaks; SUBrm`
9. `Brmtroughs; SUBrm`
10. `Brmpeaks3; SUBrm3`
11. `Brmtroughs3; SUBrm3`
12. `BzoomPlot; FUNzoomaxes`
13. `Btcrossing; SUBcrossing`
14. `Bpcrossing; SUBcrossing`
15. `BplotTS; FUNplotTS`

*subroutines*

16. `SUBcrossing`
17. `SUBfreqdist`
18. `SUBhistcross`
19. `SUBlinecrossing`
20. `SUBrm`
21. `SUBrm3`

*functions*

22. `FUNfdrmpplot`
23. `FUNzplots`

## 4.4. power spectrum analysis

---

The PSD analysis menu has the following files

```
dir *.m
```

```
BanalysisPSD.m  Btagpsdaxes.m  FUNispeak.m  SUBanalysisPSD.m  
Bperiod.m      FIGPSDToolbar.m  FUNperiod.m  contents.m  
Bpsd.m         FUNfundpeaks.m  FUNpsd.m
```

*buttons*

1. BanalysisPSD.m
2. Bperiod.m
3. Bpsd.m
4. Btagpsdaxes.m

*subroutines*

5. SUBanalysisPSD.m

*functions*

6. FUNfundpeaks.m
7. FUNperiod.m
8. FUNpsd.m

*figures*

9. FIGPSDToolbar

## 4.6. phase space analysis

---

The Phase space analysis toolbar has the following file structure

`dir *.m`

B3dspace.m	Bdelay.m	Bembed.m	Bsamples.m	FIGPCToolbar.m
BanalysisPCA.m	Bdelaym.m	Bembedp.m	Bsamplesm.m	FUNembed.m
BaxisAuto.m	Bdelayp.m	BpcSpace.m	Bsamplesp.m	SUBpcSpace.m
BaxisMan.m	Bembed.m	Brotate3d.m	Bview.m	

*buttons*

1. B3dspace.m
2. BanalysisPCA.m
3. BaxisAuto.m
4. BaxisMan.m
5. Bdelay.m
6. Bdelaym.m
7. Bdelayp.m
8. Bembed.m
9. Bembedm.m
10. Bembedp.m
11. BpcSpace.m
12. Brotate3d.m
13. Bsamples.m
14. Bsamplesm.m
15. Bsamplesp.m
16. Bview.m

*subroutines*

17. SUBpcSpace.m

*functions*

18. FUNembed.m

*figures*

19. FIGPCToolbar.m
20. FIGPCToolbar.mat

## 4.9. Func

---

The func folder has the following functions, which are called by various toolbars. A detailed listing of the functions can be found in the code listing section.

`dir *.m`

```
Bzoom.m          gcfbubbles.m    previewFFT.m    readtxt.m
activateaxes.m  mcf.m           previewFundae.m tagaxes.m
clb.m            mycd.m          previewRM.m     write.m
disptxt.m       myload.m        previewTS.m
```

The utils folder has several utility functions, which are especially to be used via the BUBBLE Toolbox command line interface. The 'preview' features are especially useful while looking at files. The mycd command is extremely user friendly as it provides a GUI to change working directory. A welcome change from MATLAB's command line interface 'cd'.

## 4.6. Fluidynamics

---

The fluidynamics folder contains custom functions, which were created for this specific experiment.

```
dir *.m
```

```
allweberf.m          switchscript_fvwe.m  webersurf.m
bifurcateweber.m    switchscript_nwfreq.m weflow.m
nWeber.m            switchscript_we2d.m
rotameter.m         weber.m
```