Ultrasound Contrast Bubble Simulation. *Bubblesim*

Lars Hoff *

December 23, 2004

Contents

1 Warning **2**

2 Function of the program **2**

3 System Requirements **2**

4 Installation **2**

5 Running **2**

6 Results **3**

List of Figures

1 Graphical user interface to operate the program. 4
2 Example of result of *Bubblesim* calculation 4

List of Tables

1 Contents of *struct*-variables. 3
2 Fields of Matlab struct *pulse* 5
3 Fields of Matlab struct *particle* 5
4 Fields of Matlab struct *linear* 5
5 Fields of Matlab struct *simulation* 6
6 Fields of Matlab struct *graph* 6
7 File naming convention. 7

*Vestfold University College, NO-3103 Tønsberg, Norway. Tel.: +47 33 03 77 13. E-mail: lars.hoff@hive.no

1 Warning

Please be aware that this program is still under development. I attempt to keep this file up to date, but some changes may still not have been included yet. Any comments are welcome!

2 Function of the program

The program 'Bubblesim' simulates the response of a bubble exposed to an ultrasound pulse. The main purpose of the program is to simulate contrast agents for medical ultrasound imaging. It is aimed at micrometer-sized bubbles exposed to Megahertz-frequency ultrasound. Only the radial oscillation mode is included, i.e. the oscillation is spherically symmetric. Nonlinear terms are fully included. A visco-elastic shell is included. Thickness and visco-elastic parameters of the shell must be specified by the user.

3 System Requirements

The program was originally written using Matlab version 5.2 and Microsoft Windows NT 4.0. It has been verified to work also under Matlab version 6.5 and Windows 95, 98 and XP.

Any computer running Matlab 5.2 or later should be able to run the program, but the graphical input and output may need modifications on non-Windows computers.

A Pentium or equivalent processor is required, to obtain reasonable run-time of the simulations. The memory requirements depend on the size of the traces to be simulated.

Functions from the Signal Processing Toolbox are used for filtering and resampling. This is not critical for the calculations. If the Signal Processing Toolbox is not found, the calculations will be performed without filtering, and a warning message is issued.

The graphical displays were written for a 1280x1024 pixels screen, running Windows 'Large Fonts'. The program has also been tested and works acceptable at screen resolutions 1024x768 and 800x600 pixels, at both 'Large Fonts' and 'Small Fonts'. However, several different screens exist, and the visual appearance of the graphical windows may look strange at some other screen/font settings.

4 Installation

Unzip the files to selected folder. Do not extract to a path name containing spaces, (e.g. 'c:/Program Files/'), as this may cause problems with Matlab. Recommended is to unzip to the root directory, e.g. 'c:/ A new folder 'Bubblesim' will be created under the selected folder.

5 Running

1. Start Matlab
2. Change to directory containing simulation files, e.g. 'cd c:*Bubblesim*'
3. Running from graphical user interface
   (a) Run startup-program 'startup.m'. This sets up paths and starts the programs. Alternatively, enter Matlab command 'Bubblesim'

(b) Enter particle and pulse parameters.

(c) Commands

Display pulse Display selected pulse.

Calculate Simulate bubble response.

(d) Options

Inverted pulse Include inverted pulse in simulation.

Plot linear calculations Include results of a linearized model, for comparison.

4. Running from batch file. Program 'ManualBubblesim.m'
   - Written for running batch jobs
   - Allows several bubble diameters, pulse amplitudes etc.
   - Run manually by changing the Matlab-code in 'ManualBubblesim.m', then calling the program from Matlab.
   - Results are stored in .mat-files. These are loaded for viewing and plotting results.

6 Results

Results are stored in the Matlab *struct*-variables listed in Table 1. These variables, except *graph*, are automatically stored to disk after each simulation. The fields of the struct-variables are explained in the tables on the following pages, Table 2 to Table 6.

Result files are named automatically according to the conventions in Table 7.

Table 1. Contents of *struct*-variables where parameters and results of the simulation are stored. These variables, except *graph* are automatically stored to disk after the simulation.

pulse	Driving ultrasound pulse pulse(1): Original pulse data pulse(2): Data of inverted pulse, if selected
particle	Parameters of the contrast agent particle, or bubble
linear	Results of linear calculation
simulation	Results of nonlinear simulation simulation(1): Original pulse simulation(2): Inverted pulse, if selected
graph	Plotting and saving parameters

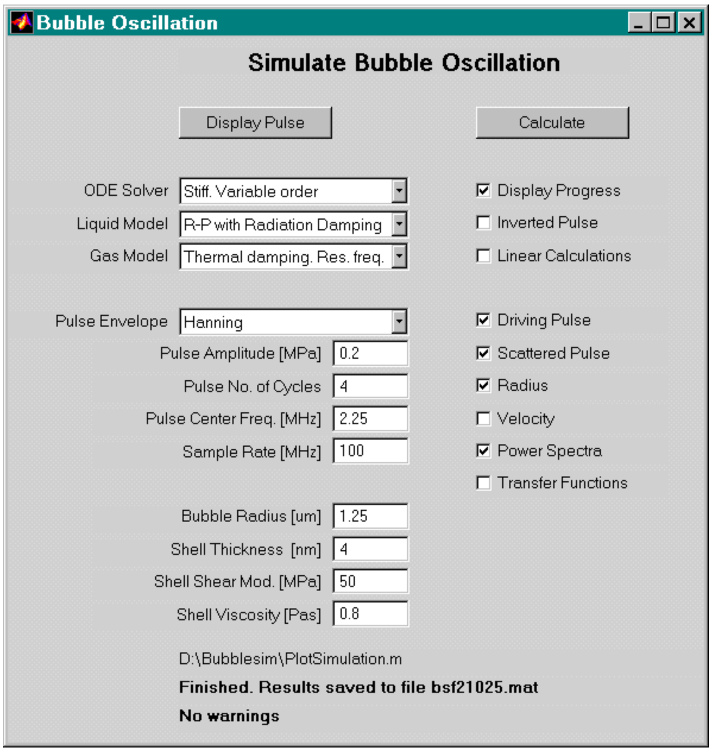

Figure 1. Graphical user interface to operate the program.

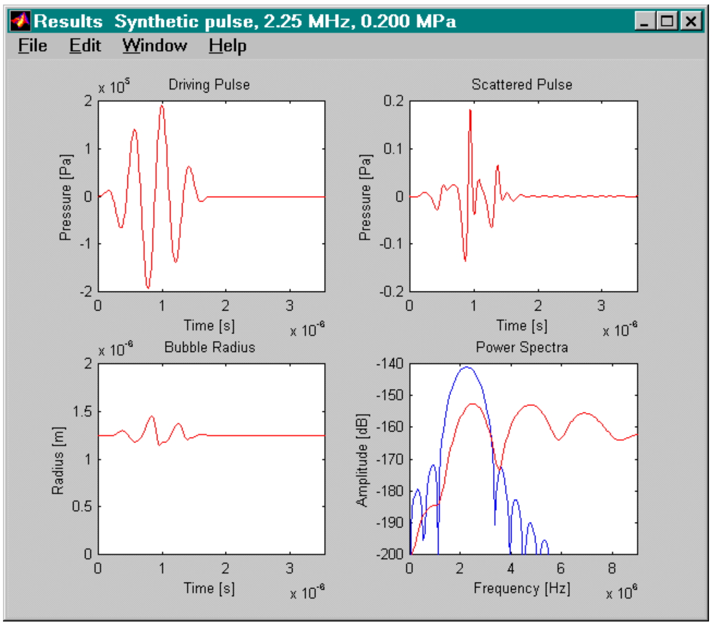

Figure 2. Example of result of *Bubblesim* calculation .

Table 2. Fields of the variable *pulse* containing parameters of the driving ultrasound pulse.

pulse(1)	Data of original pulse
pulse(2)	Data of inverted pulse, if selected

pulse	envelope	char[ ]		Pulse envelope
	A	double[1]	Pa	Pulse amplitude
	Nc	int[1]	No. cycles	Pulse length
	f0	double[1]	Hz	Pulse center frequency
	fs	double[1]	1/s	Pulse sample rate
	invert	int[1]	Boolean	Include inverted pulse
	t	double[N]	s	Time vector
	p	double[N]	Pa	Driving pressure pulse vector
	source	char[ ]		Name of pulse source: file name or "Synthetic'.

Table 3. Fields of the variable *particle* containing parameters of the contrast agent bubble, shell, and physical constants of the surrounding liquid.

particle	p0	double[1]	Pa	Internal equilibrium pressure
	rho	double[1]	kg/m3	Density of the surrounding liquid
	eL	double[1]	Pas	Viscosity of the liquid
	c	double[1]	m/s	Speed of sound in the liquid
	Kg	double[1]	W/(mK)	Thermal conductivity of the gas
	rg	double[1]	kg/m3	Density of gas
	Cp	double[1]	J/(kgK)	Heat capacity of the gas, $p = const.$
	gamma	double[1]	1	Adiabatic constant of the gas. $\gamma = C_p/C_v$
	a0	double[1]	m	Particle radius
	ds	double[1]	m	Shell thickness
	Gs	double[1]	Pa	Shell shear modulus
	es	double[1]	Pas	Shell viscosity

Table 4. Fields of the variable *linear* containing parameters and results of linear model calculations.

linear	t	double[N]	s	Time vector
	x	double[N]	1	Relative radial displacement
	a(:,1)	double[N]	m	Particle radius
	a(:,2)	double[N]	m/s	Particle velocity
	p	double[N]	Pa	Scattered pressure
	f	double[N]	Hz	Frequency vector
	Hxp	double[N]	m/Pa	Transfer function $H_{xp}(\omega) = \frac{x(\omega)}{p_{in}(\omega)}$
	Hpp	double[N]	1	Transfer function $H_{pp}(\omega) = \frac{p_{out}(\omega)}{p_{in}(\omega)}$
	hxp	double[N]	m/Pas	Impulse response $p_{in}(t) \rightarrow x(t)$
	hpp	double[N]	1/s	Impulse response $p_{in}(t) \rightarrow p_out(t)$

Table 5. Fields of the variable *simulation* containing parameters and results of the nonlinear ODE simulation.

simulation(1)	Original pulse data
simulation(2)	Data of inverted pulse, if selected

simulation	solver	double[N]		ODE solver. Name and Matlab function
	model	double[N]		Bubble model. Name and Matlab function
	thermaldamping	double[N]		Thermal model for the gas
	displayprogress	double[N]	boolean	Plot progress during simulation
	etime	double[N]	s	Elapsed CPU time for simulation
	Unevenly sampled vectors, as returned from ODE solver			
	t	double[N]	s	Time vector. Unevenly sampled
	a(:,1)	double[N]	m	Particle radius. Unevenly sampled
	a(:,2)	double[N]	m/s	Particle velocity. Unevenly sampled
	p	double[N]	Pa	Scattered pressure. Unevenly sampled
	Vectors resampled to constant rate fs			
	tr	double[N]	s	Time vector. Constant sample rate fs
	pr	double[N]	Pa	Scattered pressure. Constant sample rate fs
	fs	double[N]	1/s	Actual sample rate of simulation

Table 6. Fields of the variable *graph*. This variable controls display and plotting of results.

graph	plotlinear	int[1]	Boolean	Include linear calculations
	include	int[N]		Graphs to include in plot
	title	char[N]		Simulation title
	resultfile	char[ ]		File to store simulation results into
	tmax	double[1]	s	Max. time to plot
	fmax	double[1]	Hz	Max. frequency to plot
	figure	cell[2]		Figure no. where results are plotted
	incoming	cell[2]		Graph for driving pulse(s) figure, graph
	scattered	cell[2]		Graph for scattered pulse figure, graph
	radius	cell[2]		Graph for bubble radius figure, graph
	velocity	cell[2]		Graph for wall velocity figure, graph
	spectra	cell[2]		Graph for power spectra figure, graph
	transfer	cell[2]		Graph for linear transfer functions figure, graph
	symbol	cell[2]		Line styles used in plots

Table 7. Explanation of file naming convention.

Filename: CS*yymmdd_nnn*.mat

Symbol	Description	Values
CS	File code	*Contrast Bubble Simulation*
mm	Month	01, 02, . . . , 12
dd	Day of month	01, 02, 03, . . . , 31
nnn	Measurement no.	001, 002, 003 . . . , 999
mat	Extension	Matlab format

Example: The 15th simulation result on June 25, 2001, is stored in a file named *CS010625_015.mat*.