

NESS Brass - User Documentation

Reg Harrison
Acoustics and Audio Group
The University of Edinburgh
`r.l.harrison-3@sms.ed.ac.uk`

September 21, 2016

1 Overview

This document aims to give an overview of how to use the Brass instrument software developed as part of the NESS project. This document is structured as follows:

- Use of Code - how to run the code from the command line or terminal
- Instrument File - how the instrument files are constructed and a discussion on how construction affects the produced sound
- Score File - how the score files are constructed and how the parameters within need to be set to get the instrument to play
- Trouble Shooting - possible error messages and how to fix them
- Example Files - example input files

2 Use of Code

Standard use of the brass code is from the command line or terminal¹. When in the directory that the code lies in, it can be run by inputting:

```
ness-brass -i <instrumentfile> -s <scorefile>
```

The `-i` option specifies that the following input is the instrument file name and the `-s` option specifies that the following input is the score file. The full list of options is:

`-i <filename>`: Specifies the instrument filename.
`-s <filename>`: Specifies the score filename.
`-r`: Enables saving of raw output data into .f64 files. Off by default.
`-e`: Run first 1000 timesteps and print out an estimate of how long it would take to run complete simulation.
`-o <basename>`: Base name for output files. Default is 'output', so by default the files will be named 'output-mix.wav', 'output1.wav', 'output2.wav', etc.

¹For Linux and Mac OSX this needs to start with `./` to run.

Instrument and score files can be specified in either .m format, which uses the vector representation used by MATLAB, or .xml format, which uses a node based database style. These are explained in the in the Instrument File and Score File sections.

3 Instrument File

The instrument file contains parameters relating to the simulation, air and instrument properties which include: sample rate (Hz), air temperature ($^{\circ}\text{C}$), valve position (mm), default and bypass valve tube lengths(mm), and bore profile (distance (mm) and diameter (mm)).

The air temperature affects thermodynamic properties, such as the speed of sound in air and the air density, and therefore the resonances of the instrument.

Valve positions must be given in ascending order. The number of entries for the valve position, default tube lengths, and bypass tube lengths must be the same, e.g. if 3 valve positions are entered into the instrument file there must also be 3 default tube lengths and 3 bypass tube lengths. The sum of a valve position and default tube length pair must lie within the total length of the instrument.

The profile of the default valve path is set by the original bore profile and the amount that the valve is considered open and is considered constant along that section; the actual cross sectional surface area is original section multiplied by the valve opening factor. The bypass section can be considered in 3 sections: 2 at the beginning and end of the tube that are the same length as the default tube (the valve sections), and a middle, longer piece of tubing. The overall profile of the bypass tube is a linear interpolation between the surface areas it joins to the main tubes but the opening and closing sections are modified by the valve opening parameter.

The bore profile can specified in one of two ways. Setting the `custominstrument` option to be 0 uses a manual input of position diameter pairs, setting it to be 1 uses the custom bore function.

3.1 Measured Bore

Using the measured bore option requires inputting position-diameter pairs into the bore breakpoint function. The position entries must be inputted in ascending order.

3.2 Custom Bore

To use the custom instrument function, dimensions relating to sections must be specified. There are three general sections: the mouthpiece, the middle sections, and the flare. The length of the flare is determined by the overall length of the instrument and the previous sections length. There can be several middle sections defined of differing type. All lengths and diameters are in units of mm.

3.2.1 The Mouthpiece

The mouthpiece is constructed using a raised cosine which ramps from the opening diameter to the first diameter specified in the middle sections over a specified length.

3.2.2 Middle Sections

Multiple middle sections of differing profile can be specified to lie between the mouthpiece and the instrument flare. The length of each middle section is specified in a length array and a second array contains diameter and profile information for each section. This second array has 3 columns and the same number of rows as there are middle sections. The third column entry specifies the type of profile used for that section, labelled by a number, and the first two columns contain diameters in mm used in constructing the section. The labels correspond to:

- 1 - linear - the bore profile linearly changes from the first entry diameter to the second
- 2 - sinusoid bump - a squared sine function is used to add a bulge to the profile. The entrance and exit of the section have a diameter equal to the first entry of this section diameter, the bulge has a diameter equal to the second entry.
- 3 - raised cosine ramp - a raised cosine is used to ramp from the first entry to the second. Same as for the mouthpiece section.

3.2.3 Flaring Section

The flare of the instrument is controlled by a final opening diameter and an exponent which controls the steepness of this section. The length of the flare is equal to the total length of the instrument minus the lengths of the mouthpiece and middle sections.

4 Score File

The score file contains the information related to how the instrument is played. This includes: normalisation factor of output, length of gesture (s), lip parameters (including effective area (m^2), width (m), and mass (kg); damping; and equilibrium separation (m)), lip frequency (Hz), mouth pressure (Pa), noise amplitude, and valve openings. There are also modulation control parameters for lip frequency, mouth pressure and valve opening which include the rate and strength of modulation. All control parameters are breakpoint functions of time - value pairs.

The normalisation factor sets the range of the outputted .wav file, e.g. for a normalisation factor of 1, the output will have a range between -0.5 and 0.5.

4.1 Lip Parameters

The lip parameters control how the lip model interacts with the instrument and therefore how well the instrument plays. The `scoretemplate.m` file contains

lip parameters that should work with most instruments but extremely large instruments will need modification. The lip stiffness parameter is perhaps the easiest to modify where a higher stiffness can make it more difficult to lock onto an instrument resonance. A high stiffness can improve attack of the note.

4.2 Pitch and Articulation

The lip frequency determines the pitch of the note but there is not an exact mapping between the lip frequency and the frequency of the outputted sound; a lip frequency of 400Hz does not guarantee an outputted sound of 400Hz!. Therefore to get a desired note some trial and error with the instrument in use is required.

The mouth pressure determines when the note is played and the volume of sound produced (before the output is normalised). For a good attack a short transient of about 1ms is recommended but this will vary depending on instrument and desired sound.

4.3 Modulation

Noise can be added to the mouth pressure signal to imitate the effects of turbulence generated inside a real instrument. The noise amplitude is specified in the score as a fraction of the original mouth pressure, e.g. for a constant mouth pressure of 3000Pa and a noise amplitude score file entry of 0.1 will add a noisy signal to the mouth pressure with a range of -300Pa to 300Pa.

The vibrato and tremolo functions modulates the lip frequency and pressure signals using the following transformation:

$$g \rightarrow g(1 + A_m \sin(2\pi f_m t))$$

where g corresponds to the lip frequency and mouth pressure signal that is modulated, and A_m and f_m is the corresponding modulation amplitude and frequency (Hz). As with the noise amplitude, the vibrato and tremolo amplitudes correspond to fractions of the original lip frequency and mouth pressure inputted into the score file.

4.4 Valve Control

The valve opening entry into the score file specifies how much of the air flow passes through the default tube at the valve section, with 1 minus this entry giving how ‘open’ the bypass tube is. An entry of 1 corresponds to the default tube being the only possible path, an entry of 0 corresponds to the bypass tube being the only possible path. Entries between 0 and 1 correspond to a partially open valve configuration. All valve entries must be specified at each time entry.

Modulation of valve openings is performed in a similar way to the vibrato and tremolo functions but the score entry for valve modulation amplitude is the actual amplitude of modulation, not a fraction of the original amplitude. The transformation is as follows:

$$q \rightarrow q + A_v \sin(2\pi f_v t)$$

where q is the original valve opening, and A_v and f_v are the valve modulation amplitude and frequency.

5 Trouble Shooting

Unable to get duration from score file!

- Missing semicolon in score file to specify change in time entry
- Entry in score parameter is not a number
- Missing score parameter

Error parsing brass instrument file: vpos, vdl and vbl must all be same length Unable to parse instrument file

- Different number of valve sections specified for the position of the valves and the lengths of the default and bypass tubes

No output from code

- Valve is positioned too close to another valve or to the ends of the instrument for its default tube length - consider repositioning valve

6 Example Files

6.1 .m Format

Score File

```
% template for score file

maxout=0.95;%max normalised value of instrument output
T=2;%Length of score s

Sr=[0,1.46e-5];%time s - effective lip surface area m^2
mu=[0,5.37e-5];%time s - effective lip mass kg
sigma=[0,5];%time s - effective lip damping
H=[0,0.00029];%time s - lip equilibrium separation m
w=[0,0.01];%time s - effective lip eidth m

lip_frequency=[0,400];%time s - lip frequency Hz

pressure=[0,0;
          1e-4,];%time s - mouth pressure pa, useful to have
               short pressure ramp

%vibrato
vibamp=[0,0.1];%time s - vibrato amplitude as fraction of
          normal frequency
vibfreq=[0,2];%time s - frequency of vibrato Hz
%tremolo
tremamp=[0,0.1];%time s - tremolo amplitude as fraction
          of normal mouth pressure
tremfreq=[0,2];%time s - frequencny of tremolo Hz

noiseamp=[0,0.1];%time s - noise amplitude as fraction of
          normal mouth pressure

valveopening=[0,1,1,1];%time s - fractional valve opening
          between 0 and 1 for each valve
valvevibfreq=[0,0,0,0];%time s - valve vibrato frequency
          Hz
valvevibamp=[0,0,0,0];%time s - valve vibrato amplitude
          between 0 and 1 (not a fraction of original opening
```

Custom Instrument File

```
%3 valved instrument
custominstrument=1;%this is 0 if you have a real
    instrument bore
FS=44100;%sample rate
temperature=20;%temperature in C
vpos=[600,630,660];%position of valve in mm
vdl=[20,20,20];%default (shortest) tube length mm
vbl=[130.7,57.5,199.8];%bypass (longest) tube length mm

xmeg=10;%mouthpiece length mm
x0eg=[80,200,545];%middle section lengths mm
Leg=1400;%total length of instrument mm
rmeg=20;%mouthpiece diameter mm
r0eg=[4,9,3;9,12,1;12,12,1];%middle section diameter (mm)
    and profile specifications
rbeg=130;%end diameter mm
fbeg=6;%flare power
```

Manual Instrument File

```
%3 valved instrument
custominstrument=0;%this is 0 if you have a real
    instrument bore
FS=44100;%sample rate
temperature=20;%temperature in C
vpos=[600,630,660];%position of valve in mm
vdl=[20,20,20];%default (shortest) tube length mm
vbl=[130.7,57.5,199.8];%bypass (longest) tube length mm

%bore specified in axial distance - diameter pairs both
    in mm
bore=[0,17.34;
    1,16.3;
    6,7.38;
    7,5.53;
    10,4.16;
    67.3,6.8;
    87.4,9.7;
    88.3,9.6;
    160.3,10.1;
    210.3,10.8;
    361.3,11.75;
    396.3,11.75;
    726.3,11.75;
    741.3,11.8;
    1131.3,18.3;
    1141.3,18.8;
    1358.8,66.7;
    1367.8,81.4;
    1381.3,127];
```