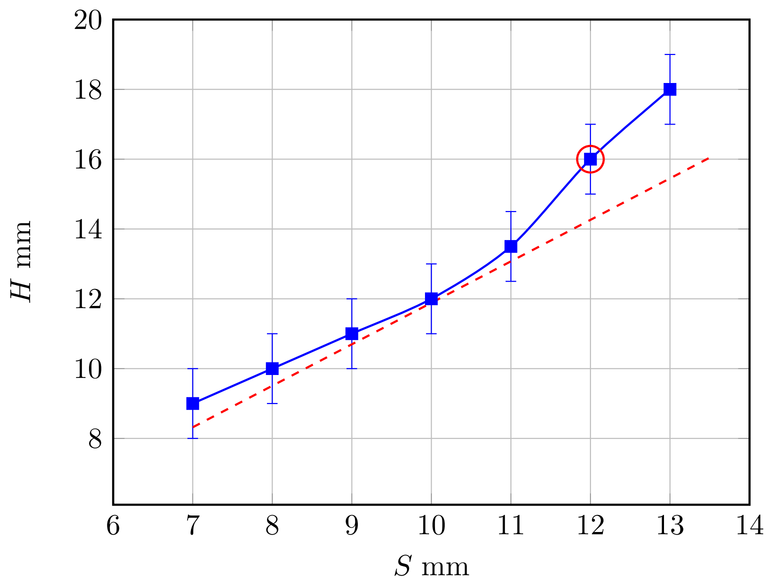

FIG. 9.

V. STUDENT EXPERIENCES

VI. CONCLUSIONS

Appendix A: Installation of olaFlow

In this work, we use the olaFlow to simulate the generation of water waves. These are a suite of codes which can be run in the OpenFOAM computational framework. Downloading and installing OlaFlow is not trivial, as successful implementation of olaFlow relies on the use of an earlier version of OpenFOAM. Hence, in this Appendix we outline the steps necessary to install OlaFlow and execute simple test cases. We describe this for users using a Windows Operating System, however, most of the commands for these tasks will be the same in Ubuntu.

First, we install an Ubuntu virtual machine which will run in a Windows environment, this is done using the Windows Power Shell and the command:

```
wsl -d Ubuntu-18.04
```

We next install the appropriate version of OpenFoam, which is OpenFoam6. The appropriate commands are obtained from https://openfoam.org/download/6-ubuntu/ and are repeated here:

```
sudo sh -c "wget -O - https://dl.openfoam.org/gpg.key | apt-key
   add -"
sudo add-apt-repository http://dl.openfoam.org/ubuntu
sudo apt-get -y install openfoam6
sudo apt-get update
```

At the end of the installation, we must add the following line to our `bashrc` file (a prompt appears to do this):

```
. /opt/openfoam6/etc/bashrc
```

We are now able to download olaFlow. We do this simply by downloading a zip file from https://github.com/phicau/olaFlow, this will produce the zip file `olaFlow-master.zip`, which we move in to the relevant directory and unzip (`home/username`). We next modify the permissions on the unzipped folder:

```
sudo chmod -R 777 olaFlow-master
```

We install ‘make’:

```
sudo apt-get install make
```

We can now use the OpenFoam command ‘all make’ to generate the appropriate olaFlow executables. First:

```
cd olaFlow-master
./allMake
```

Then:

```
cd genAbs
./allMake
```

To run the various codes that generate transient boundary conditions in OlaFlow, we require Python. Hence, we install `pip`:

```
sudo apt install python-pip
```

and finally, `numpy`:

```
sudo install numpy
```

Once these installations are complete, olaFlow is ready to be used. For instance, to run a simple flume model from the tutorials, we change into the relevant directory:

```
cd tutorials
cd wavemakerFlume
```

The relevant OpenFOAM case can be run using the appropriate file. For instance, for the piston wave maker, we type:

```
. runCasePiston
```

The details concerning this case (mesh generation, initial conditions, etc.) can be found by opening the file of the same name.

Appendix B: Mesh-Refinement Study

The standard simulations reported in the main paper use a uniform blockmesh of size $(N_x, N_y, N_z) = (500, 1, 50)$. Here, we report very briefly on simulation results where the blockmesh has been increased to $(1000, 1, 100)$. We carry out simulations for a stroke length $S = 0.015\,\mathrm{m}$ and a forcing frequency $\omega = 137.8\,\mathrm{RPM}$.