Towards Computational Modelling of Multiphase Flow in and around Furnace Tap-holes due to Lancing with Oxygen: an Initial Computational and Cold Model Validation Study

Markus W. Erwee[1,2], Quinn G. Reynolds[1], Johan H. Zietsman[3], Robert D. Cromarty[3], Axel S. Lexmond[3]

[1] Mintek, Private Bag X3015, Randburg, South Africa, 2125
[2] Part time student at University of Pretoria, South Africa
[3] University of Pretoria, Pretoria, South Africa, 0002
markuse@mintek.co.za, quinnr@mintek.co.za
johan.zietsman@up.ac.za, robert.cromarty@up.ac.za
axel.lexmond@up.ac.za

Most pyrometallurgical furnaces have one or more tap-holes, from which slag, alloy and/or matte is tapped. Although tap-holes differ in design, most are closed using tap-hole clay and opened by a combination of mechanical drilling and lancing with oxygen. Lancing with oxygen involves a thin steel tube, through which oxygen is transported, which is inserted into the tap-hole. The steel tube melts and ignites in the oxygen, providing energy to melt the tap-hole clay. During lancing, reaction of oxygen with the slag, alloy, matte, refractory, and gas is possible. All of these interactions (chemical and thermal) can influence the process and the tap-hole assembly itself. This can lead to premature shutdown of the furnace.

To study the effect of lancing in a controlled manner, multiphase models are being developed. Both computational and physical models will be used to study tap-hole flow during lancing throughout the entire project. The initial physical model is a simple assembly using water and air. A high-speed camera and wall pressure measurements are used to observe flow phenomena when injecting an air jet into a static bath. The computational model is developed with OpenFOAM. This study serves as a 'building block' for more complex models, since very little measured data are available for this particular problem. It is envisaged that the study will be one of many, during which proper validation data will be generated to ensure that the modelling results are a true reflection of what occurs within a pyrometallurgical furnace, and that the datasets will serve the modelling community by providing confidence in computational models. The emphasis of this work is on confirming that real-life phenomena can be described properly with computational models before performing more complex studies.

1 INTRODUCTION

The tap-hole is considered one of the most important parts of a pyrometallurgical furnace. The main product, be it matte, alloy or slag, can only be produced if it can be tapped successfully and safely at reasonable rates from the furnace.

Tap-holes are high-maintenance, high-risk areas of the furnace as they are opened and closed continually, allowing flow of molten materials at high temperatures. Tap-hole systems are complex and a variety of designs exist [1].

The standard configuration includes a tap-hole refractory system (tap-block), cooling system, drill, oxygen lance, 'mud'-gun and, in more modern systems some form of monitoring equipment. The drill and oxygen lance are used for opening the tap-hole, and the mud-gun for closing the tap-hole by injecting refractory clay into it.

Tap-hole monitoring is critical: firstly, because incidents like uncontrolled tapping, furnace run-outs and water explosions due to tap-block failure are safety risks, and secondly because a failed tap-hole assembly can stop production with subsequent loss of revenue. Significant advances have been made in tap-hole system monitoring, aside from refractory material and cooling water temperature measurements. These include the use of thermal imaging cameras and fibre optics to monitor wear of tap-hole blocks [2].

Many furnaces require a combination of drilling and oxygen lancing to initiate tapping. In the case of blast furnaces, lancing is used less than 1% of the time [3]. In contrast, up to 36% of taps from South African platinum melting furnaces involve oxygen lancing [4].

Lancing remains mostly a manual process. This often leads to damaged tap-hole refractories due to unwanted thermal spikes, skew holes, and oxidation of carbon-based refractories, and potential contact with molten material in the furnace. There are also risks of tap-hole clay moisture explosions due to extreme lancing temperatures, and accidental lancing into water-cooled tap-hole assemblies and resulting water explosions. On the other hand, manual lancing by experienced operators can correct drilling errors with minimal damage to the tap-hole assembly. Experienced operators often develop a 'feel' for the furnace process and tap-hole systems, making lancing somewhat of an art.

Thermal modelling is used extensively in the design of furnace tap-block assemblies [5], often leading to significant improvements in tap-hole life and safety aspects. To date, the effect of lancing inside and around the tap-hole has received limited attention in open literature. This is most likely due to the complexity of the lancing and tapping process.

Modelling of oxygen lancing can become quite complex. Should the lance penetrate fully into the bath inside the furnace, enhanced heat transfer, gas-metal-slag mixing, and oxidation of alloy can occur. This may lead to additional wear and shock of the tap-hole system. Inside the tap-hole, the thermal effects of the lancing process, whether molten material is flowing or not, is not yet fully understood. Simulations to determine how large or small these effects are can lead to a better understanding of tap-hole assembly wear, and ultimately to prolonged tap-hole life.

The research reported in this paper forms part of a larger project focused on the effects of lancing inside and around the tap-hole. The project involves multiphase computational fluid flow modelling to describe the effect of lancing on the tapping process.

The 'holy grail' of the research is to have a fully integrated multiphysics model, incorporating fluid flow, energy transfer and thermochemical phenomena resulting in energy release, and formation of new phases that can contribute to chemical wear of the refractories. The work therefore aims at developing an improved understanding of the lancing problem by predicting the unseen effects inside the furnace tap-hole, where measurement is difficult to impossible. A better understanding of the lancing action could lead to new ways of establishing when it is safe to lance and when not, as well as when to stop lancing.

The research has a secondary objective of developing open-source software and hardware platforms to describe and solve the problem. Significant effort is also made to validate the computational work experimentally. This paper focuses on the validation of a very simple multiphase flow model.

2 Objectives of this Research Paper

The objectives of this initial part of the work are to:

- start development of a simple computational multiphase model on an open-source platform to describe the tap-hole lancing problem;
- design and build a simple physical model to generate data to validate the initial computational model against; and
- to implement a multiphase fluid flow model in open-source software to promote open access in the field of computational modelling and validation.

The media that were used in the simulations and experiments were water and nitrogen. These fluids are not representative of the materials in industrial furnaces. The purpose of this work was to establish the necessary tools required to validate a computational multiphase model against experimental data for a simple system. For future work fluids and flow characteristics that are more representative of industrial processes will be used.

3 Physical Model

3.1 Dimensions and Equipment

A physical model was constructed from 5 mm thick Perspex panels (Figures Figure 21 and Figure Figure 22). Table 10 contains the dimensions of the model as well as the position of the sensors used.

The physical model was fitted with a copper tube (representing the lance) at varying penetration depths for different experiments. The lance was connected to a gas bottle filled with nitrogen, and the gas flow rate was controlled by a mass flow controller (model: Bronkhorst, F-201C-FAC-22-V, range: 0-10 L/min).

Two pressure sensors (RS Components IPSL series) were fitted in the tank wall above the lance at distances of 35 mm (bottom sensor, range: 0-50 mbar gauge) and 70 mm (top sensor, range: 0-250 mbar gauge). An additional sensor was fitted in line with the mass flow controller to measure pressure variations inside the lance during gas injection.

The two wall pressure sensors needed to be flush against the tank wall, to ensure accuracy was not lost during measurements [6], and to ensure that flow was not affected by the sensors during experiments. The range of operation of the two wall sensors in the tank were chosen based on basic estimates for these experiments, and on work done by previous authors on similar problems [6][7]. There was, however, a decision made to choose sensors with different sensitivities to determine which sensitivity would be optimal. Sensors with a pressure range of 50 mbar and 250 mbar were selected for this purpose.

Data from the pressure sensors were captured using the Arduino Uno [8] open-source electronics platform coupled to a laptop computer. Python [9] scripts were employed for data processing and communication between devices.

A high-speed digital video camera (Olympus iSpeed 3)was triggered to record as soon as the first bubble appeared from the lance. Video imagery was recorded at a rate of 1000 frames per second. The video frame rate was chosen by trial and error, and an optimal rate will be investigated in future.

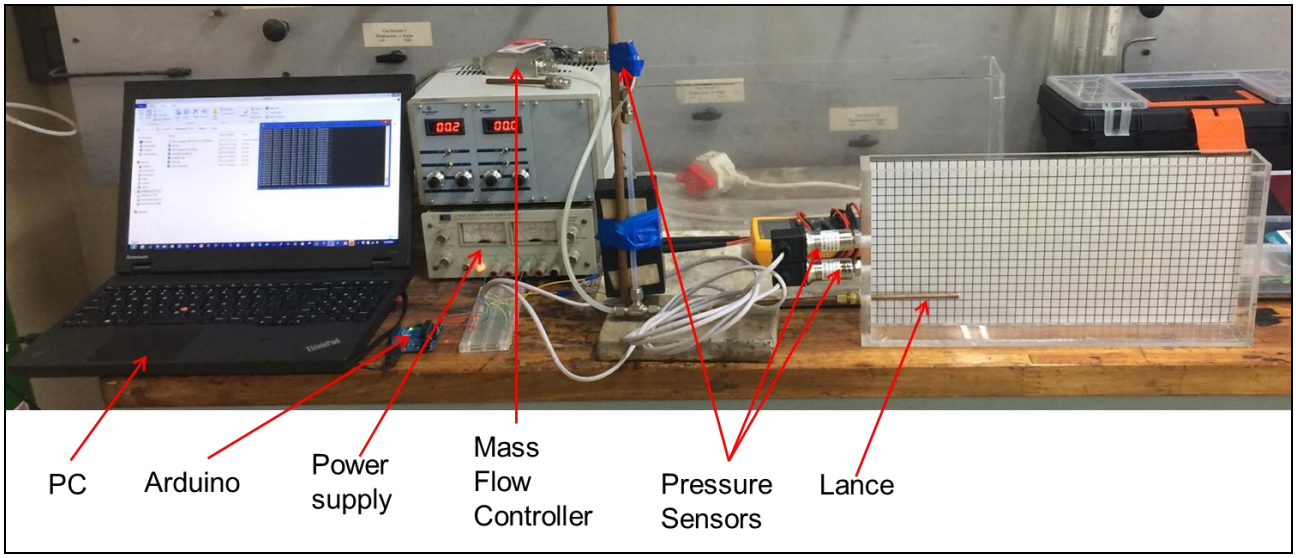

Figure 21: Photograph of the physical model with attachments, excluding camera and light

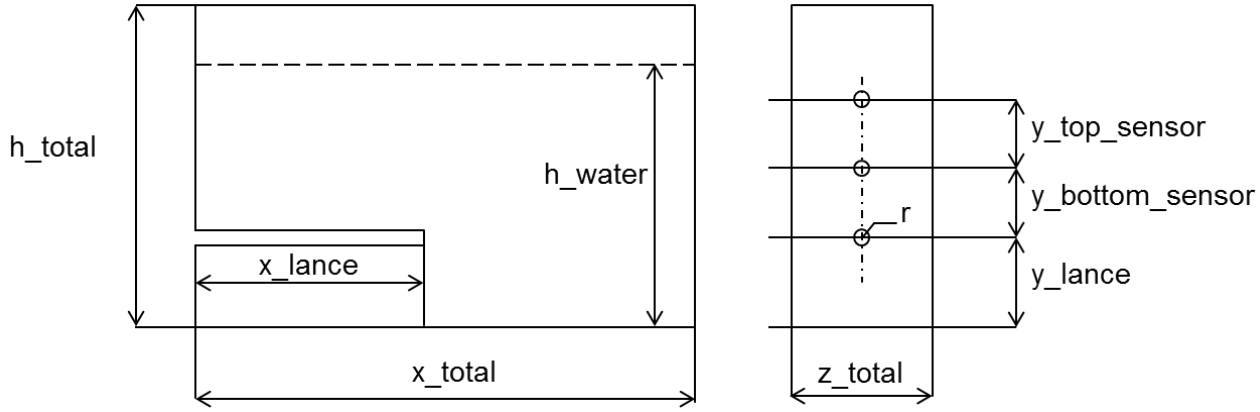

Figure 22: Schematic representation of the physical model: The figure on the left is the front view, and the one on the right is a side view. See Table 10 for dimensions.

Table 10: **Dimensions of the model**

Label	**Size (mm)**	**Label**	**Size (mm)**
h_total	200	y_top_sensor (pressure)	35
h_water	150	y_bottom_sensor (pressure)	35
x_lance	10, 50, or 100	y_lance, r_lance	45, 2.91 (internal radius)
x_total	400	z_total	45

3.2 Experimental Conditions, Noise Measurement, and Operating Method

The tank was filled with water up to a height (h_water) of 150mm. This provided a gauge pressure of P_{water} = ρgh_water = (1000*9.81*0.15)/100 = 14.72 mbar at the bottom of the tank. For the top and bottom sensor, the expected gauge pressure would therefore be P_{bottom} = 6.87 mbar and P_{top} = 3.43 mbar. These pressures were calculated and then measured with the pressure sensors to establish whether the sensor readings are accurate (i.e., whether the pressure sensors have the appropriate range for the experiment). The ambient pressure was 845 mbar and the temperature 27°C during experiments.

Once the tank was filled and the water was completely static, the baseline pressure was measured for 5 seconds in order to establish the noise levels on the sensors.(Note: In all plots, only 2 seconds worth of data is shown).

Figure 23 shows the top sensor, having a larger range, had a mean value of 2.88 mbar, with a standard deviation of 0.17 mbar and coefficient of variation of 0.059. The bottom sensor showed a value of 6.88 mbar, with a standard deviation of 0.05 mbar and coefficient of variation of 0.007. On this basis it was decided that sensors with a range of 0-50 mbar will be used for future work.

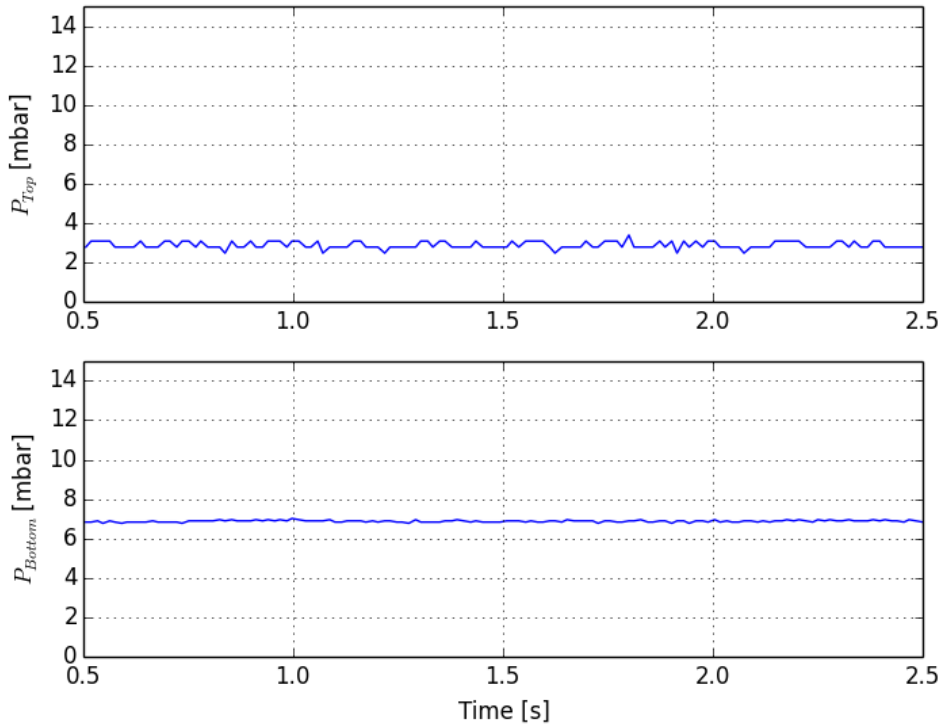

Figure 23: Baseline noise pressure readings on the top and bottom sensors mounted on the wall of the physical model when filled with water to a height of 150mm

Two variables were changed in the experiments conducted, namely lance penetration depth, and gas flow rate through the lance. The lance was positioned 1, 5 or 10 cm from the left wall and the gas flow rate was set to values of 1, 3.5, and 7.5 L/min.

For each run, the tank was allowed to settle before taking a new measurement. Measurements were repeated at least three times for each set of conditions.

Videocapturing was automatically triggered when the first movement in the tank was detected. Pressure readings were started once the gas flow on the mass flow controller reached the required value (this takes 0.5-1.0 seconds).

The results of these experiments are discussed later in this paper when compared to those from the computational model.

4 Computational Model

An equivalent computational fluid dynamics model of the physical model was implemented using OpenFOAM version 2.3.1 [10], an open-source framework for field solution of conservation equations. The computational model uses OpenFOAM's standard *interFoam* solver for two-phase flow, which consists of a flow model and a phase separation model. Subgrid turbulence modelling was not considered at this stage as the lance injection velocities were relatively low and fairly high mesh resolutions could be used. OpenFOAM discretizes the Navier-Stokes, continuity, and phase conservation equations governing the various fields using the finite-volume method [11].

The volume-of-fluid (VOF) method [12] was used to account for phase separation. In this method, solution of a separate convective transport equation is required for the volume fraction field of the primary phase, α. Water was used as the primary phase. This permits the phase regions and the interfaces between them to be calculated easily without the need for computationally expensive mesh movement calculations.

The Pressure Implicit with Splitting of Operators (PISO) predictor-corrector algorithm was used to calculate the velocity and pressure fields at each time step. Gradient-limited discretisations were used for all divergence terms with the exception of those related to the phase fraction field, which use the Multidimensional Universal Limiter for Explicit Solution (MULES) limiter to perform interface compression and capturing. Adaptive time-stepping was used to maintain a Courant number of 1 or below for numerical stability of the algorithm. A two-dimensional slice model was used for this preliminary work, with the option to extend to full 3D in the future.

Boundary conditions were supplied along the edges of the domain shown in Figure 22. The walls of the lance and vessel were treated as non-slip, non-permeable boundaries with velocity = 0, normal gradient of pressure = 0, and

normal gradient of $\alpha = 0$. The top surface of the model was assumed to be open to atmosphere, with a fixed gauge pressure = 0, velocity calculated from the pressure gradient, and an inlet-outlet boundary condition for α based on the direction of flow across the boundary ($\alpha = 0$ (air) if inflow, normal gradient of $\alpha = 0$ if outflow). At the lance tip, a fixed constant velocity in the horizontal direction was specified, together with the normal gradient of pressure = 0, and a fixed value of $\alpha = 0$.

Meshing of the model region with unstructured quadrilateral elements was accomplished using version 2.8.3 of GMSH [13]. Mesh resolutions varied from 20000 to 40000 elements, with substantial refinement in the region of the rising bubble plume originating from the end of the lance. Figure 24 shows an example mesh of the region.

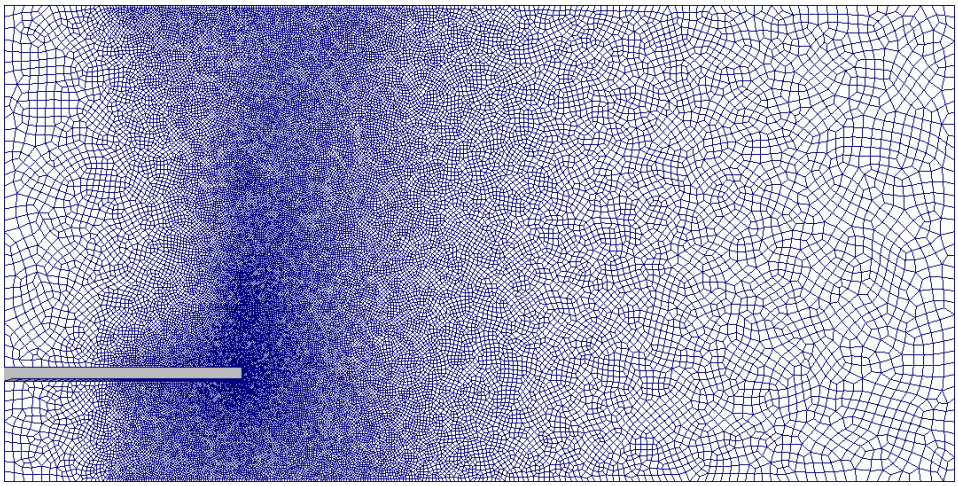

Figure 24: 2D unstructured mesh for computational models (40218 elements)

5 Comparison of Computational Model and Physical Model

5.1 Visual Comparison

For the purposes of this paper, only one set of visual results are shown (all others are available on request). The result shown here (Figure 25) is for a gas flow rate of 7.5 L/min (extreme) at a lance depth of 1 cm.

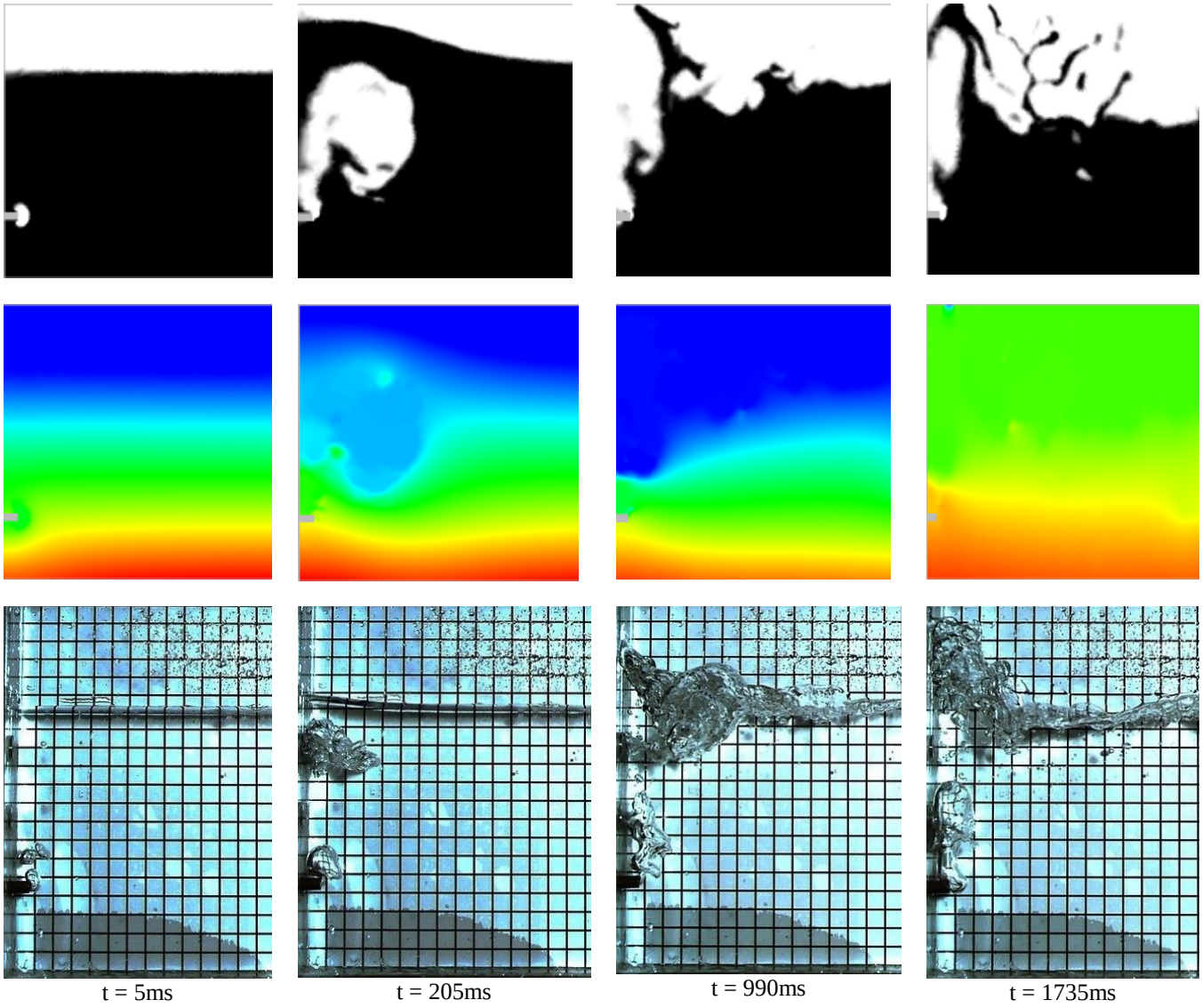

Figure 25: Computational model results and high-speed video images taken for a lance depth of 1cm and a gas flow rate of 7.5 L/min. Top row: The parameter α in the computational model (α = volume fraction of water; white matches the condition α=0, black matches α=1). Middle row: The computed pressure field (blue is 0mbar, red is 14.72 mbar; the colour scale is linear). Bottom row: frames taken from the high-speed video footage.

It appears that the computational model, at least on a visual basis, behaves similarly to the physical model. To validate the computational model to the experimental results, pressure sensor readings were compared to one another (next section).

5.2 Comparison of Pressure Signals

5.2.1 Typical Pressure Signals

Typical pressure fluctuations with time from the computational model and experimental results are shown in Figure 26.

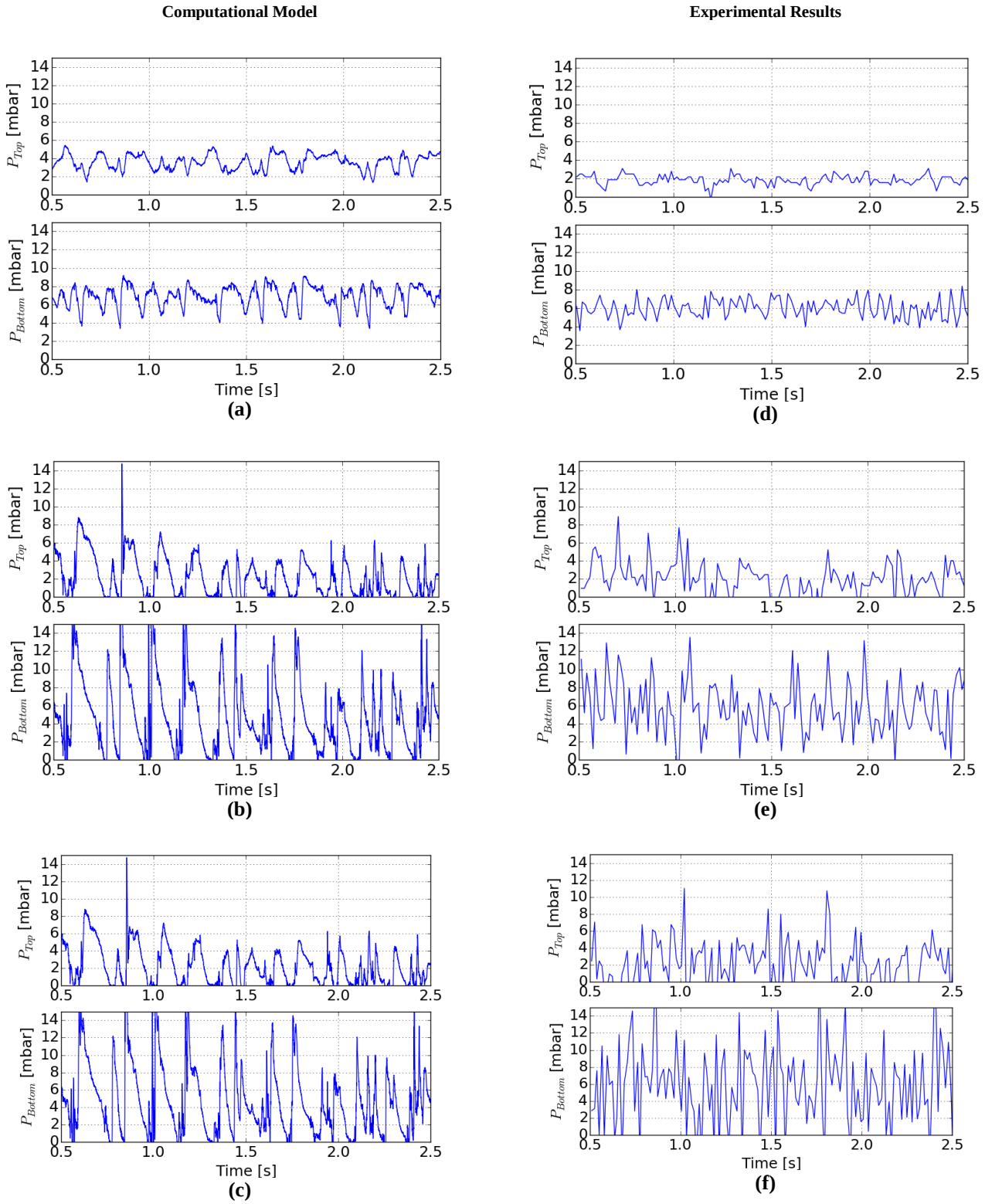

Figure 26: Pressure signals for lance depth of 1cm.Gas flow: (a) and (d) = 1.0 L/min; (b) and (e) = 3.5 L/min; (c) and (f) = 7.5 L/min

Compared to the computational model, the pressure signals from the physical model are at a much lower sampling rate, limited by the code used with the Arduino as data logger. Modification of the hardware platform and/or communication between the computer and the Arduino will be required to increase the sampling rate during high-flowrate experiments, to ensure that a maximum amount of detail can be obtained. This said, the overall results using

the Arduino were good for lower flowrates (Figure 26 (a) and (d)); with only partialloss in resolution at higher flowrates (Figure 26 (e) and (f)). The amplitude variation is also higher in the measurements at higher flowrates.

5.2.2 Effect of Gas Flow Rate and Lance Depth on Mean Wall Pressure

The wall pressure mean and standard deviation were evaluated against the gas flowrate, for each penetration depth of the lance for a fixed length of time. It is common in, for example, fluidized bed applications [14][15] to compare the standard deviation in pressure rather than the absolute mean. The terms standard deviation and pressure variation are used interchangeably here. It should be noted that sensor noise has been taken into account in all plots shown from this point on (error bars).

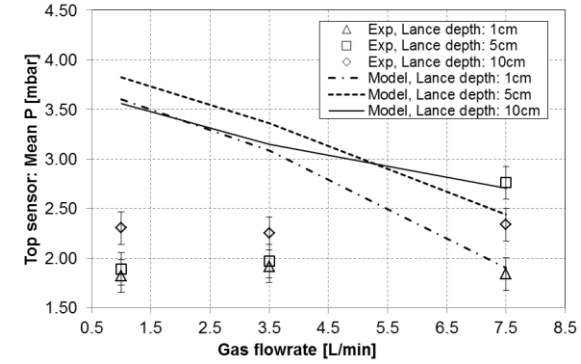

Figure 27: Mean pressure for top sensor as a function of gas flowrate and lance depth

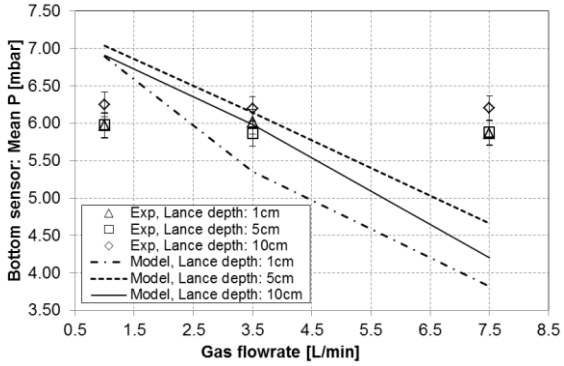

Figure 28: Mean pressure for bottom sensor as a function of gas flowrate and lance depth

In the computational model, the mean pressure decreases with increasing gas flow, whereas there is very little change in mean pressure for the experimental model (Figure 27 and Figure 28). This change could only occur if the water level changes significantly over time, which is not applicable in the physical model.

The difference between the results is attributed to the fact that, in the current iteration of the computational model, the water level changes over time. When water "splashes" beyond the top boundary of the computational model, it is not replaced (i.e., it leaves the system completely). Subsequently, the overall pressure at the location of the sensors decreases, as the pressure head (ρgh_water) decreases. In the experimental model, the majority of the water splashing beyond the top of the tank falls back into it, maintaining a constant level. The difference might also occur because the plume formed in the computational model moves up against the wall of the tank, causing the pressure drop. This will be investigated further in subsequent studies.

In the computational model, the mean pressure drop is the greatest when the lance is closest to the wall (1 cm), which in part could be the "loss" of water. It does seem, however, that a significant amount of air moves up against the wall (see Figure 25). In the physical model, the mean pressure remains roughly the same for a change in lance depth.

To improve the computational model, the boundary conditions will need some refinement to match the experimental results. This is achievable and will be included in future work.

5.2.3 Effect of Gas Flow Rate and Lance Depth on Wall Pressure Variation

A more effective means of comparing the pressure measurements is by using the standard deviation of pressure (Figure 29 and Figure 30).

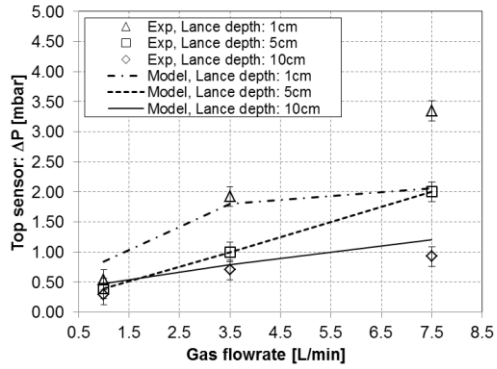

Figure 29: Pressure standard deviation for top sensor as a function of gas flowrate and lance depth

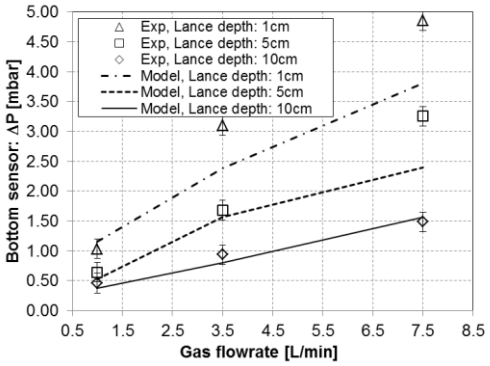

Figure 30: Pressure standard deviation bottom sensor as a function of gas flowrate and lance depth

In both the computational and experimental results, the pressure variations increase in magnitude as the flow rate of gas is increased. The agreement is less at high flow rates, most probably because of the boundary conditions on the computational model. Water that "splashes" out leaves the system boundary and the overall water level drops.

In summary, a plot of the predicted pressure variations for all lance depths and gas flow rates is shown in Figure 31 for both wall sensors. A reasonable good correlation was found for the bottom sensor, while the top sensor showed slightly poorer correlation.

Considering the extensive approximations and assumptions made in the development of the 2D computational model, the correlation with experimental results from the physical 3D model is good.

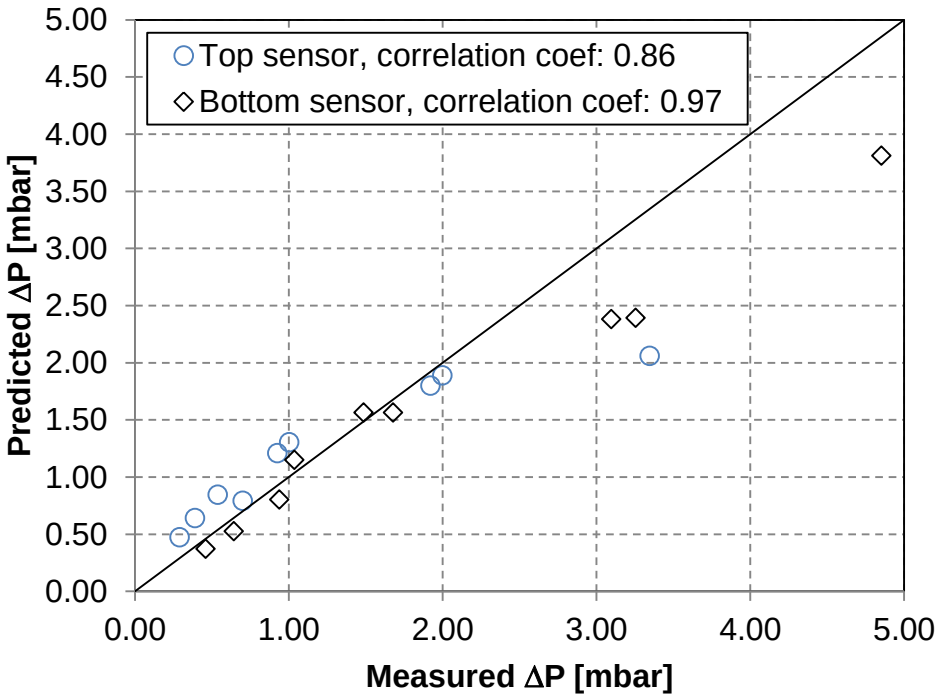

Figure 31: Comparison of all the pressure standard deviation data for both sensors

6 Conclusion

A simple multiphase computational model was validated against a physical model as a first step in studying the effect of lancing in furnace tap-holes. A rectangular tank was constructed with a lance penetrating through the lower sidewall, injecting a gas stream into a set volume of water. The parameters that were changed include lance penetration depth and gas injection rate.

A two-dimensionalcomputational model was developed using the OpenFOAM open-source framework to represent the three-dimensional physical model. Pressure measurements at two points in the physical model were compared to equivalent signals produced by the computational model. The variation in pressure along the wall of the experimental rig was found to be in agreement with the values computed in the OpenFOAM model, with slight deviations observed at

higher flow rates. Correlation coefficients from 0.93 to 0.98 were found between the computational and experimental results when comparing pressure deviations. Superficially, mean pressures did not correlate well. This is attributed the boundary conditions used in the computational model.

The Arduino open-source hardware platform was used to log data. It was found that the sampling rate for the Arduino and pressure sensors used were not optimal, but this did not affect the quality of data gathered except at high flowrates of gas. Communication between the Arduino and PC will be optimized in future to increase the sampling rate.

Future work will involve improved dimensional scaling of the experimental rig and fluids used to represent industrial furnaces, and expansion of the computational model. In addition, frequency response analysis will be carried out on the pressure signals. Using Fast Fourier Transforms, it might be possible to uniquely identify the type of flow regime in the system.

7 ACKNOWLEDGEMENTS

This paper is published by permission of Mintek. The authors wish to thank Mr Steve McCullough, Mr Martin Sitefane and Mr Gideon Mbebe of Mintek for their assistance in the laboratory.

8 REFERENCES

[1] **Dienenthal, A.***A short history of the development of tapping equipment.* SAIMM, 2014. Furnace Tapping Conference. Vol. S80, pp. 203-216, Muldersdrift, South Africa.

[2] **Braun, W., Gebski, P., MacRosty, R., Puta, R., Sadri, A., Ying, W.***Tap-hole monitoring technologies.* SAIMM, 2014. Furnace Tapping Conference. Vol. S80, pp. 169-182, Muldersdrift, South Africa.

[3] **Nelson, L.R., Hundermark, R.J.***'The tap-hole' - key to furnace performance.* SAIMM, 2014. Furnace Tapping Conference. Vol. S80, pp. 1-32, Muldersdrift, South Africa.

[4] **Nolet, I.***Tapping of PGM-Ni mattes: an industry survey.* SAIMM, 2014. Furnace Tapping Conference. Vol. S80, pp. 223-232, Muldersdrift, South Africa.

[5] **McDoughall, I.** *Water-cooled tap-hole blocks.* SAIMM, 2014. Furnace Tapping Conference. Vol. S80, pp. 183-192, Muldersdrift, South Africa.

[6] **Souto-Iglesias, A., Botia-Vera, E., Martín, A., Pérez-Arribas, F.***A set of canonical problems in sloshing. Part 0: Experimental setup and data processing.* Ocean Engineering, Vol. 38, 2011, pp. 1123-1830.

[7] **Bulian, G., Botia-Vera, E., Souto-Iglesias, A.***Experimental sloshing pressure impacts in ensemble domain: Transient and stationary statistical characteristics.* Physics of Fluids, Vol. 26, 2014, pp. 1-29.

[8] **ArduinoUno**, http://arduino.cc/en/Main/ArduinoBoardUno , accessed November 2014.

[9] **Python**, https://www.python.org/ , accessed November 2014.

[10] **OpenFOAM**,http://www.openfoam.org , accessed November 2014.

[11] **Patankar, S.***Numerical Heat Transfer and Fluid Flow.* New York : Hemisphere Publishing Company (USA), 1980.

[12] **Hirt, C.W. and Nichols, B.D.***Volume of fluid (VOF) method for the dynamics of free boundaries.*, Journal of Computational Physics, Vol. 39, 1981 pp. 201-225.

[13] **GMSH**, http://guez.org/gmsh/ , accessed November 2014.

[14] **Guizhi, Q., Jiamin, Y., Haigang W., Wuqiang, Y.***Investigation of flow hydrodynamics and regime transition in a gas–solids fluidized bed with different riser diameters.* Chemical Engineering Science, Vol. 116, 2014, pp. 195-207.

[15] **Luckos, A., Reynolds, Q.G., den Hoed, P.***An Analysis of Pressure Fluctuations in a CFB of Heavy Minerals.* The 12th International Conference on Fluidization - New Horizons in Fluidization Engineering, Vancouver, Canada : ECI, 2007. pp. 147-152.