PolyPublie
Polytechnique Montréal

POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

Titre: Experimental methods in chemical engineering: Optical fibre probes in multiphase systems
Title:

Auteurs: Shahab Golshan, Gregory Scott Patience, Reza Zarghami, Jamal Chaouki, & Bruno Blais
Authors:

Date: 2022

Type: Article de revue / Article

Référence: Golshan, S., Patience, G. S., Zarghami, R., Chaouki, J., & Blais, B. (2022). Experimental methods in chemical engineering: Optical fibre probes in multiphase systems. Canadian Journal of Chemical Engineering, 100(10), 2762-2777. https://doi.org/10.1002/cjce.24419
Citation:

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

URL de PolyPublie: https://publications.polymtl.ca/50792/
PolyPublie URL:

Version: Version finale avant publication / Accepted version
Révisé par les pairs / Refereed

Conditions d'utilisation: Tous droits réservés / All rights reserved
Terms of Use:

Document publié chez l'éditeur officiel
Document issued by the official publisher

Titre de la revue: Canadian Journal of Chemical Engineering (vol. 100, no. 10)
Journal Title:

Maison d'édition: Wiley
Publisher:

URL officiel: https://doi.org/10.1002/cjce.24419
Official URL:

Mention légale: This is the peer reviewed version of the following article: Golshan, S., Patience, G. S., Zarghami, R., Chaouki, J., & Blais, B. (2022). Experimental methods in chemical engineering: Optical fibre probes in multiphase systems. Canadian Journal of Chemical Engineering, 100(10), 2762-2777. https://doi.org/10.1002/cjce.24419, which has been published in final form at doi:10.1002/cjce.24419. This article may be used for non-commercial purposes in accordance with Wiley Terms and Conditions for Use of Self-Archived Versions. This article may not be enhanced, enriched or otherwise transformed into a derivative work, without express permission from Wiley or by statutory rights under applicable legislation. Copyright notices must not be removed, obscured or modified. The article must be linked to Wiley's version of record on Wiley Online Library and any embedding, framing or otherwise making available the article or pages thereof by third parties from platforms, services and websites other than Wiley Online Library must be prohibited.
Legal notice:

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal

https://publications.polymtl.ca

Received 4 April 2021; | Revised 4 April 2021; | Accepted 4 April 2021

DOI: xxx/xxxx

MINI-REVIEW

Experimental methods in chemical engineering: Application of optical fibre probes in multiphase systems

Shahab Golshan[1] | Gregory S. Patience[1] | Reza Zarghami[2] | Jamal Chaouki[3] | Bruno Blais*[1]

[1]Chemical Engineering, Polytechnique Montréal, C.P. 6079, Succ. "CV", Montréal, H3C 3A7 Québec, Canada

[2]Chemical Engineering, University of Tehran, Process Design and Simulation Research Centre, School of Chemical Engineering, College of Engineering, 11155/4563 Tehran, Iran

[3]Chemical Engineering, Polytechnique Montréal, Process Engineering Advanced Research Lab (PEARL), C.P. 6079, Succ. "CV", Montréal, H3C 3A7 Québec, Canada

Correspondence
*Corresponding author B. Blais, Polytechnique Montreal. Email: bruno.blais@polymtl.ca

Summary

As much as 50 % of consumer products and 75 % of the raw materials in the chemical industry are in the form of powders and granular solids. Gasification, pyrolysis, coating, granulation, drying, and mixing are examples of processes in which granular particles contact fluids. Researchers examine the hydrodynamics of these fluid-solid contactors with pressure signals, acoustics, computed tomography, radioactive particle tracking, optical fiber measurements, and spectroscopy. Among these techniques, fiber optic probes are simple, inexpensive, sensitive, with high response rates and accuracy. Optical probes measure local hydrodynamic properties including particle velocity, solids fraction, and voids, which are difficult for heterogeneous flow systems like spouted beds, risers, and turbulent fluidized beds. We discuss the theory, calibration, applications limitations and sources of uncertainty in optical probe measurements. Sample Matlab codes explain how to process the experimental data. We provide pieces of experimental data as a sample input for each application.

KEYWORDS:
fiber optics, fluidized bed, spouted bed, particle velocity, solids holdup

1 | INTRODUCTION

Fluidized and spouted bed reactors are fluid-solid contactors industry applies to gasify and pyrolyze organic substrates and waste, coat and granulate powders, mix active pharmaceutical ingredients, and dry slurries and pastes. Heat transfer rates in fluidized bed systems are an order of magnitude greater than fixed beds and and they are capable of processing enormous volumes of gases in vessels that reach 50 m at a lower pressure drop. Moreover, in catalytic reactors mass transfer rates between the solids and fluid is higher because typical particle sizes are on the order of 70 µm compared to pellets that reach 3000 µm.[1–3] Although they have many advantages over fixed bed reactor, they are chaotic systems and the entrance region, cyclones, diplegs, and bulk gulf-streaming represent operational challenges. To better design reactors, especially for systems with explosive mixtures, highly exothermic and endothermic reactions, cohesive/sticky powders,[4] and those that agglomerate easily (like for polymerization),[5] researchers build and operate large scale cold flow units (up to 1/3 the commercial size) and develop numerical techniques based on computational fluid dynamics (CFD) and discrete element methods (DEM).[6, 7] Both Eulerian-Eulerian (two-fluid model)[8] and Eulerian-Lagrangian (coupled computational fluid dynamics—discrete element method)[2, 6, 8] are applied to the micro-scale (particle dimension) and the marco-scale (diameter of the bed dimension).

Pressure signals,[9, 10] acoustic signals[11], computed tomography (CT)[12], radioactive particle tracking (RPT)[13] and fiber optics[13, 14] are examples of invasive and non-invasive experimental techniques.[13–16] In non-invasive methods, the probes lie outside the system while invasive probes protrude into vessels and contact the fluid and powder, which msy disturb the hydrodynamics. Fiber optic probes are invasive devices that are highly accurate, simple, sensitive,

with a fast response time and low cost compared to most non-invasive methods.[15–20] Researchers have measured bubble rise velocity,[15, 21] bubble diameter,[15, 21] bubble holdup,[15] particle velocity,[14, 22–28] local holdup,[14, 22–24, 28–33] local solids flux,[14] and spout diameter[14, 24, 28–30, 34] with fiber optic probes. The fiber diameter (generally in the range of 10 µm to 20 µm), should be larger than the particle diameter to measure particle velocity.[14, 23] Bubble diameter, local holdup and spout diameters are much greater than this but to minimize perturbing the system, the outer probe diameter shoud be less than 5 mm.

Maurer et al.[15] compared bubble diameter and rise velocity using X-ray tomography and optical probe measurements. Pugsley et al.[16] compared fluidized bed holdup (i.e. voidage) using CT and fiber optic measurements. Xu et al.[35] compared time-averaged particle velocity using optical prope, RPT and Positron Emission Particle Tracking (PEPT). These research,[15, 16, 35] confirmed the high accuracy, less than 10 % discrepancy in holdup measurements compared to CT[16], of fiber optic measurements. Furthermore, optical measurements are less expensive compared to the other methods (at least 50 % cheaper than CT, for example), and more straightforward to implement[36].

Chemical engineers employ several optical methods, including fiber optic measurements[37], Raman spectroscopy[38], near-infrared spectroscopy (NIR), phototubes, and laser spectroscopy[39]. in process engineering. These optical methods are unrelated and each requires a distinct experimental protocol. Here, we focus on fiber optic applications in multiphase systems as part of a series of tutorial reviews on experimental methods in chemical engineering.[40] We first introduce the theory of fiber optic measurements in multiphase contactors and discuss operational challenges. Furthermore, we demonstrate how to calibrate probes and process the data for several applications.

2 | OPTICAL FIBER PROBES

Fiber optic probes are produced in various types and forms, including two parallel light emitting and receiving optical fibers, parallel-optical fiber bundle probes, cross-optical fiber probes, and parallel-optical bundle fibers, which all have their advantages and drawbacks.[22] Researchers generally classify these probe types into two groups of transmition and reflection type probes.[17] In transmition type, a gap separates the illuminating and reflecting fibers, while the reflection type intermingle illuminating and reflecting fibers into groups of bundles (a.k.a. channels). Several bundles make up a probe, while each bundle consists of multiple arrays of fibers, whose

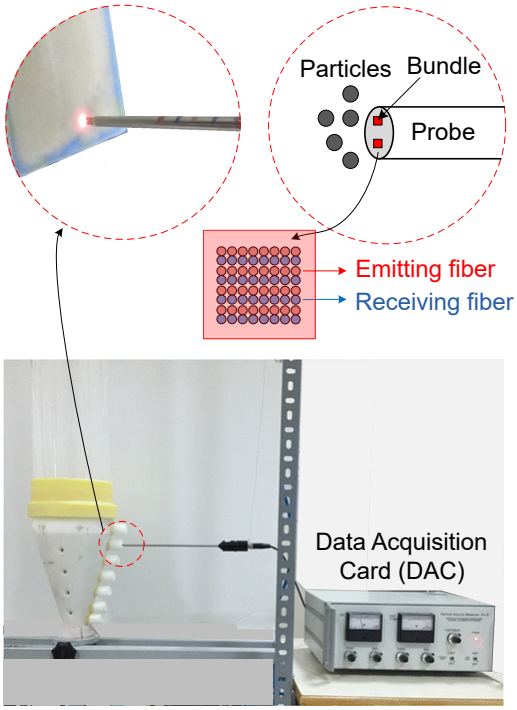

FIGURE 1 Schematic diagram of fiber optic probes.[14, 22] Each bundle consists of several sets of emitting and receiving fibers. We insert the optical probe into the multiphase bed, so that the probe tip lies in the bulk of particles. Emitting and receiving fibers are responsible for the emission of photons and transmission of the reflected photos from the surface of particles to a photo transistor. This photo transistor converts the light into an electrical voltage signal, which is recorded by a data acquisition card (DAQ).

diameter is about 200 µm. In recent fibre optic designs (reflection or parallel-optical fiber bundle probes), the arrays send and receive light alternately.[22] Parallel-optical fiber bundle probes reduce blind spots regions[41], which are areas that the light fails to illuminate or fails to return light to the probe. For other types of probes, a glass window at the tip of the probe is necessary to reduce blind regions. After receiving the reflection of the emitted light by the receiving fibers, photo-multipliers (generally one photo-multiplier for each probe bundle) convert the light signals into electrical signals.[42] Data acquisition cards save the signals. For simple applications, like solids holdup measurements, a single bundle in the probe is adequate, particle and bubble velocities require at least two light-emitting, -receiving bundles. Taofeeq et al.[22] described several fiber optic probe designs and enumerated their advantages and drawbacks. When the emitting light from the probe is in the range of visible light, we recommend to block all other light sources, including sunlight, to minimize its contribution to the overall signal (baseline shift and possibly noise).

The experimental setup with parallel-optical fiber bundle probes, Figure 1, shows a probe inserted in a bed together with the geometery of the tip. Emitting fibers in each bundle project light (emit photons) that shine on the particles located in front of the probe tip. Receiving fibers transmit the reflected photons from the surface of particles to a photo transistor, which converts the light signal into an electrical voltage, which is recorded by a data acquisition card (DAQ). To improve the spatial distribution, we repeat the experiments at different tip locations. For example, to determine the radial distribution of solids holdup in a spouted bed, we collect datasets by changing the position of the probe tip from the spout to annulus.

To interpret the voltage signals, we calibrate it against known systems to for local holdup, solids velocity and flux, bubble holdup, rise velocity and size.

3 | APPLICATIONS

In 2019, Web of Science Core Collection (WoS) indexed 18 257 documents that contained fibre optics in a search query in the field *Topic* (2). Fiber optic probes coupled with Fourier-transform infrared spectroscopy measure the local volume fraction and gaseous species composition simultaneously in gas-solid contactors.[43]

3.1 | Solids holdup

A single bundle is sufficient to calculate local solids holdup while for probes with multiple bundles, the holdup is an average of all the signals from all the bundles. Shabanian et al.[17] reviewed the optical probe calibration methods including solids dropping in a downer, liquid-solid suspension, fluidized bed, solids-polymer blocks, and theoretical methods for holdup. Among these methods, solids dropping in a downer is the most popular method for gas-solid systems (Figure 3). The calibration is based on signals acquired from particle mixtures with various holdup values. The mixtures comprise various concentrations of white and black coloured powder that are used in the experiments (Figure 3).[14, 44, 45] The black particles represent voids, since the optical probe is unable of detecting them.

The calibration procedure consists of pouring various concentrations of black and white powder through a tube with an optical probe protruding from the side (Figure 3). To maximize the signal to noise ratio for visible light experiments, the entrance and exit are sealed to minimize contributions from extraneous light. We first pour the particles into the pipe from above (top valve) and allow the particles to accumulate on the bottom valve. A few seconds after the solids level rises to the height of the probe, we open the bottom valve slowly for the particles to exit the pipe and record the signals from the probe.

This setup simulates loosely packed motion of particles in front of the probe.[14, 44, 45] A sample with only black particles represents a completely empty region, a solids hold up equal to zero. A sample with only white particles represents a loosely packed bed close the hold-up of the Scott density or poured density:[46] solids flow through a series of baffles and collect in a graduated flask below. The density is simply the ratio of the mass of solids collected and the volume of the receptical ($\rho_{\text{Scott}} = m/V$). The solids fraction, α is the ratio of the Scott density and particle density, $\rho_{\text{Scott}}/\rho_{\text{p}}$ and the void fraction is

$$\epsilon_{\text{w}} = 1 - \frac{\rho_{\text{Scott}}}{\rho_{\text{p}}} = 1 - \alpha \tag{1}$$

Every probe bundle must be calibrated as ϵ_w depends on geometry and other factors. Consequently, the solids holdup for any operating condition ϵ_s, is a function of the packed void fraction, ϵ_{w} and the signals collected during the experiment, $\overline{S}_{\text{s}}$, the signal of an empty bed, $\overline{S}_{\text{b}}$, and the signal of the packed bed $\overline{S}_{\text{w}}$

$$\epsilon_s = \epsilon_{\text{w}} \left(\frac{\overline{S}_s - \overline{S}_{\text{b}}}{\overline{S}_{\text{w}} - \overline{S}_{\text{b}}} \right) \tag{2}$$

Generally, the voltage signals varies linearly with concentration of black/unpainted powder (Figure 4). In some cases, however, like when the probe tip is damaged due to collision with rough particles, this relation becomes non-linear. Figure 4 shows schematics of a typical average voltage - solids holdup calibration. Using curve fitting, we derive an equation to link the average voltage and solids holdup for each probe bundle. We use these equations for measuring holdup in the main experiments in fluidized or spouted beds.

Based on the calibration equations in Figure 4, we present a sample Matlab code (Appendix A) to process and convert experimental signals to solids holdup (experimental data in the supplementary material). First, the code loads the experimental data of bundles. Then it divides the data into segments. The main reason of this segmentation is to remove possible problematic data sections during the elimination. In other words, we increase the number of data for the elimination steps. Then the code converts the data segments into holdup from the raw voltage signals using calibration curves of each bundle. Finally, the code removes outliers and calculates the local holdup by averaging data segments and bundle values. When we run the code using these data, and input parameters of $n_{\text{ds}} = 40$ and $\sigma^{\text{t}} = 3$, the code calculates the holdup = 0.203.

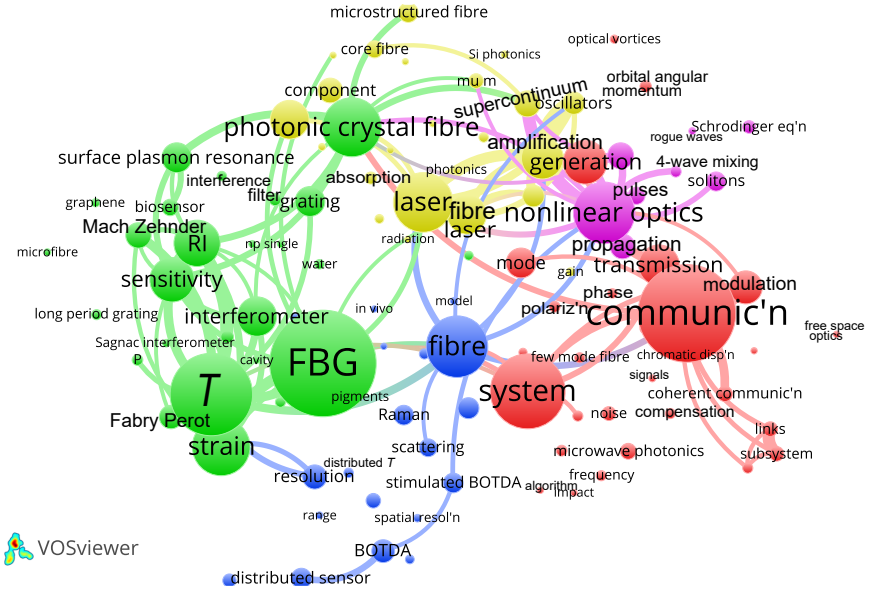

FIGURE 2 VOSviewer bibliometric co-citation map of keywords derived from a WoS topic search .

3.2 | Particle velocity

To calculate the particle velocity, v_p, requires at least two sending-receiving on the probe. It equals the ratio of the effective distance between the two bundles, l_e, and the time delay, τ,[14, 22, 44, 45]

$$v_p = \frac{l_e}{\tau} \tag{3}$$

The effective length is approximately equal to the distance between bundles on the probe but changes when collisions with particles erode the probe tip. We apply a chopper disk to calibrate the probe for velocity measurements (Figure 5).[14, 22, 44, 45] Rather than a circular chopper, we also apply a belt-driven stepper that moves along a horizonal plane linearly (Figure 6).[22]

Although patterned disks are applied to calibrate velocity (Figure 5 bottom middle),[44] we recommend fixing the actual particles to the disk surface.[14, 45]. Before starting the calibration procedure, we measure the distance between the center of disk and the probe tip on the disk, to convert rotating speed to linear velocity, and use it to calculate effective length of the probe (Eq. 3). The procedure requires rotating the disk at various speeds and recording the signal. We choose the highest rotational speed based on the predicted velocity of particles in the main experiment. For instance, in a spouted bed, the highest particle velocity appears inside the spout zone and is close to the superficial gas velocity at the inlet.[2] Consequently, the highest limit of the rotational rotating speed (after converting to linear velocity), should be slightly higher than the superficial gas velocity in the spout. Since particles pass in front of the probe tip frequently (depending on the rotating speed), sections of high and low voltage appear in the signals. Figure 7 shows a typical data for three rotations of the disk and a blown up section of one rotation.

High voltage sections in Figure 7 correspond to particles passing in front of the probe bundles. A delay (n_d in Figure 7-b) relates to the real distance between the probe bundles (effective length, l_e). A cross-correlation of the signals (Figure 8) from two bundles gives this delay in data number (n_d). Cross-correlation, also known as sliding inner product, is a measure of similarity of two signals as a function of the time lag of one relative to the other. In other words, for different lag numbers in one of the signals, cross-correlation measures the correlation coefficient of the signals. For discrete time-series (signals) x_i and y_i, Equation 4 calculates the correlation coefficient,[44]

$$R_{xy} = \frac{\sum(x_i - \overline{x})(y_i - \overline{y})}{(N_e - 1)\sigma_x \sigma_y} \tag{4}$$

where σ_x and σ_y are standard deviations of x and y signals, respectively. Once we have the corresponding time delay (by converting data number, n_d, to time delay based on the sampling frequency) to the maximum correlation coefficient (Figure 8), Eq. 5 converts it to probe effective length

$$l_e = v_p \tau \tag{5}$$

In this equation, v_p is the linear velocity of particles at the probe tip and for the chopper disk experiment it is the product of the radial distance of the probe from the centre, r, and the rotational velocity of the chopper disk, ω

$$v_p = r\omega \tag{6}$$

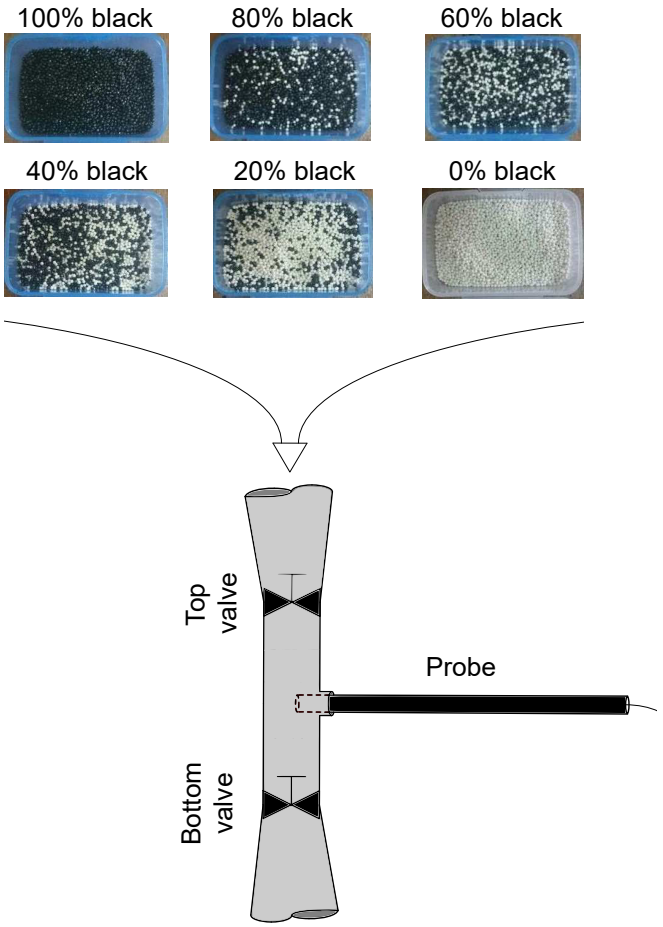

FIGURE 3 Optical probe calibration setup for solids holdup. Mixtures of various concentrations of black and white powder are poured into a downer with the probe protruding from the side through a hole. Valves at the bottom and top of the downer control the flow rate of the particles. We generate a calibration curve of the electrical signal and concentration of black powder, which represents the void fraction.

Researchers choose high sampling frequencies in areas where the velocities are high, like in the spout, and lower frequencies in the annular region, where velocities can be 1000 times lower. In the main experiments, we calculate the cross-correlation using time lags in both x and y signals. Finding cross-correlation using time lags in both signals guarantees that the motion of particles in both directions is taken into account. For example, in spouted beds, the dominant motion directions of particles in the spout and annulus are in the opposite direction. Once we obtain the cross-correlation in both directions, cross-correlation values specify the dominant motion direction, and we apply the maximum time-lag in the dominant direction for further calculations. Appendix B illustrates a sample Matlab code to process and convert experimental signals to particle velocity.

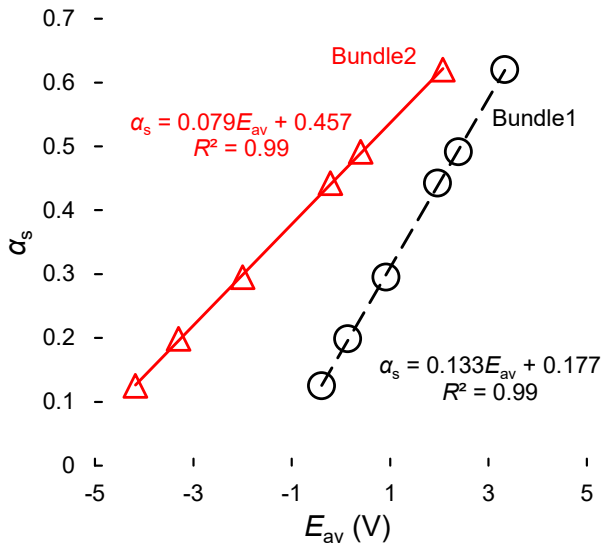

FIGURE 4 Typical solids holdup calibration curve. The holdup calibration curves relate (generally with a linear relation) the average voltage of the signals and the holdup values for each bundle.

First the code asks input parameters from the operator and reads the data of both bundles. It divides the data of each bundle into several segments using a similar segmentation method in Appendix A. Then, the code finds the maximum cross-correlation coefficients for each data segment. From these maximum cross-correlation coefficient, Eq. 3 calculates raw particle velocity. These raw velocities must pass four verification steps (elimination steps). In the first, we investigate the number of data in each segment. If the maximum cross-correlation occurs at the end of a data segment, the segments are too short and we should divide the data into a smaller number of segments (i.e. more data in each segment). The next elimination step deals with deleting velocities with weak maximum cross-correlation coefficients. For this purpose, we define a threshold. If the maximum cross-correlation coefficient of a segment is smaller than the defined threshold, the code removes the corresponding particle velocity. Defining this threshold requires prior knowledge of the experimental system: In spouted beds, the maximum values of the cross-correlation coefficient in the spout region are mainly larger than that of the annulus. The reason is due to the lower solids holdup in the spout and motion of separate particles, which makes it easier for the probe to detect them. In the annulus, however, the particles are packed and they move in bulk, making it more difficult for the probe to identify the motion of single particles. Consequently, the threshold values are larger compared to the annulus.

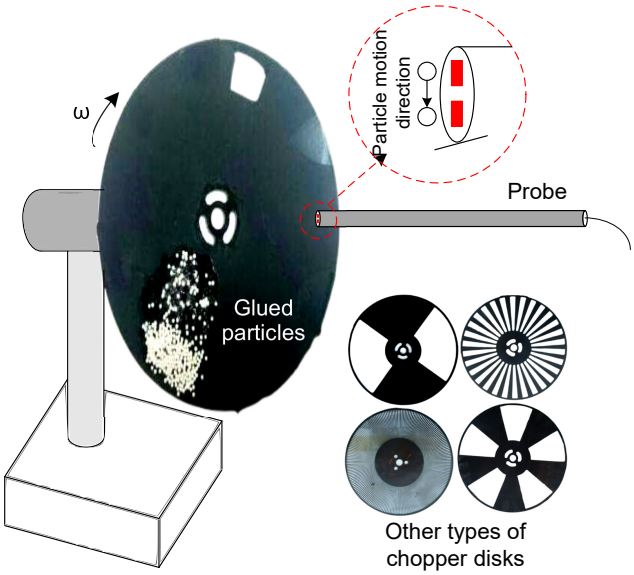

FIGURE 5 Particle velocity calibration experiments using a chopper disk comprised of black and white particles glued to the surface. A rotor rotates the disk at a rotational speed ω. We place the probe tip orthoganal to the rotating disk at a distance of 1 mm to 2 mm.

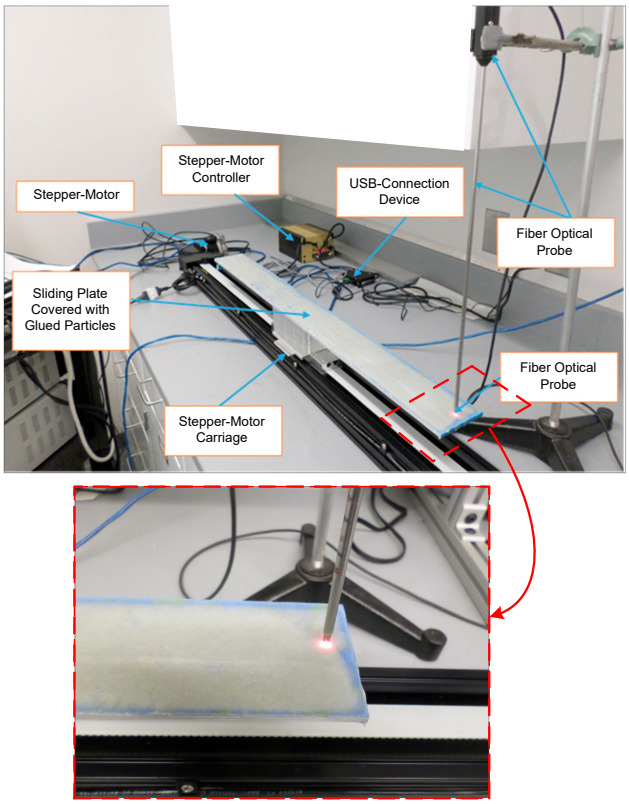

FIGURE 6 Calibration of fiber optic probe for velocity measurements of small particles with a belt-driven stepper motor. [22]

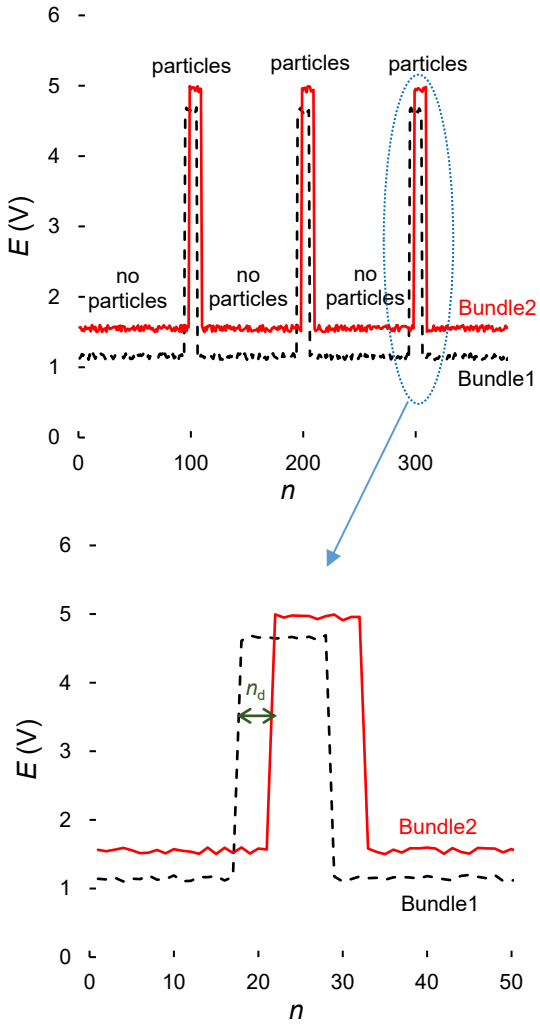

FIGURE 7 Voltage signals from three rotations of the chopper disk (top) and one rotation (bottom). When the glued particles pass in front of the probe, voltage signal is highest. Black and red lines show data of bundles 1 and 2, respectively. The lag, n_d, between the bundles 1 and 2 is the basis to calculate the effective length, l_e.

After the elimination of weak cross-correlated values, the code eliminates the data due to inappropriate sampling frequency. In this elimination step, we check in each segment whether the maximum value of the cross-correlation coefficient occurs at the first element. If this is true, the sampling frequency is too small to capture the real particle velocity. There is a trade-off between the size of the data file and the sampling frequency. Larger data files require more memory and when the files are too large for the computer, this compromises the processing procedure. As a result, we have to select the sampling frequency at each location of a bed according

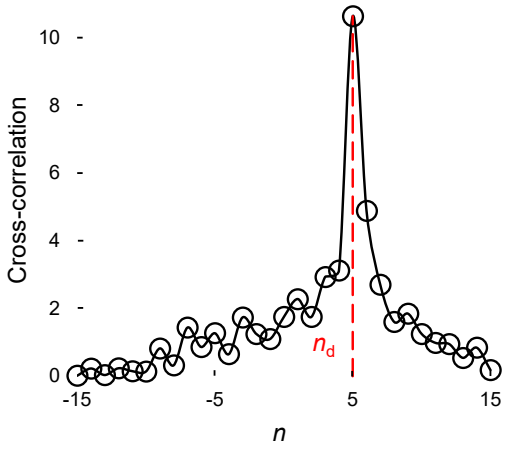

FIGURE 8 Cross-correlation curve for velocity calibration. The maximum cross-correlation occurs at data lag = n_d.

to its maximum velocity of particles. Higher particle velocities require higher sampling frequencies. In the last elimination step, the code removes the data outliers using their average and standard deviation. For the case in the supplementary material the code calculate $v_p = 0.202\,\mathrm{m\,s^{-1}}$ with the input parameters $f_s = 10000$, $l_e = 0.002$, $R^t_{xy} = 5000$, $\sigma^t = 3$, $n_{ds} = 40$.

3.3 | Instantaneous solids flux

The local instantaneous solids flux is the product of the solids holdup (Eq. 2) and particle velocity (Eq. 3)

$$G_s(t) = \rho_p \alpha_s(t) v_p(t) \tag{7}$$

3.4 | Spout diameter

The spout diameter, d_{spout}, is derived from the time-averaged radial distribution of particle velocity or the time-averaged radial distribution of solids holdup.[14, 24, 25] In the case of the former, d_{spout} is the distance from the centre at which $v_p = 0$ (Figure 9. In the latter, ϵ_s, it is the inflection point as the solids holdup goes from close to zero to the packed bed density. In the case of the example in Figure 9, d_{spout} is at the non-dimensional distance $r/R = 0.4$ while it is slightly lower at $r/R = 0.4$ based on ϵ_s.[14] To establish the variation of d_{spout} with height requires multiple measurements vertically.

3.5 | Bubble size and rise velocity

The bubble rise velocity and size are interconnected. Consider a fluidized bed with a single bubble rising from the bottom (Figure 10). Initially, the probe is in the emulsion phase (dense

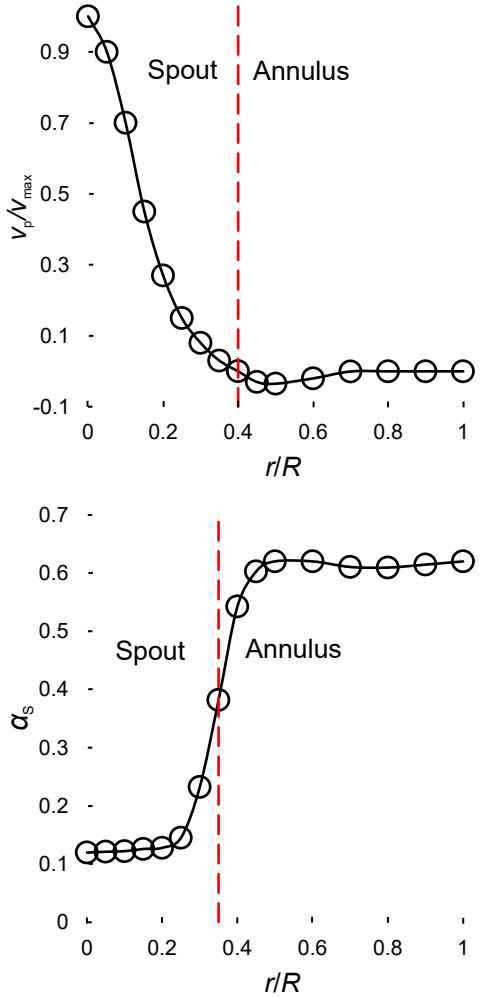

FIGURE 9 Typical particles velocity (a) and holdup (b) distributions in spouted beds. r denotes the radial distance from the spouted bed center.

phase) and the corresponding voltage signal is high (5 for bundle 1 and 4.8 for bundle 2). When the probe is inside the bubble, the signal drops to below 1 for bundle 1 and to 1.7 for bundle 2. Finally, when the bubble has gone passed the probe, the signals of the bundles return to the emulsion density value. The bubble length reported by the probe is not necessarily the maximum bubble length. For instance, in Figure 10-b, the probe reports d_{bub}, while the maximum bubble length is d_{max}. A more appropriate classification groups the voltage signals into three phases: bubble (dilute phase), particles (dense solid phase), and a transition region between these two.[47] This classification uses two thresholds to distinguish these three phases. Signals below the first threshold (with solids holdup below $\approx$ 0.2) belong to the bubble phase, signals between these thresholds (with solids holdup between $\approx$ 0.2 and 0.45) belong to

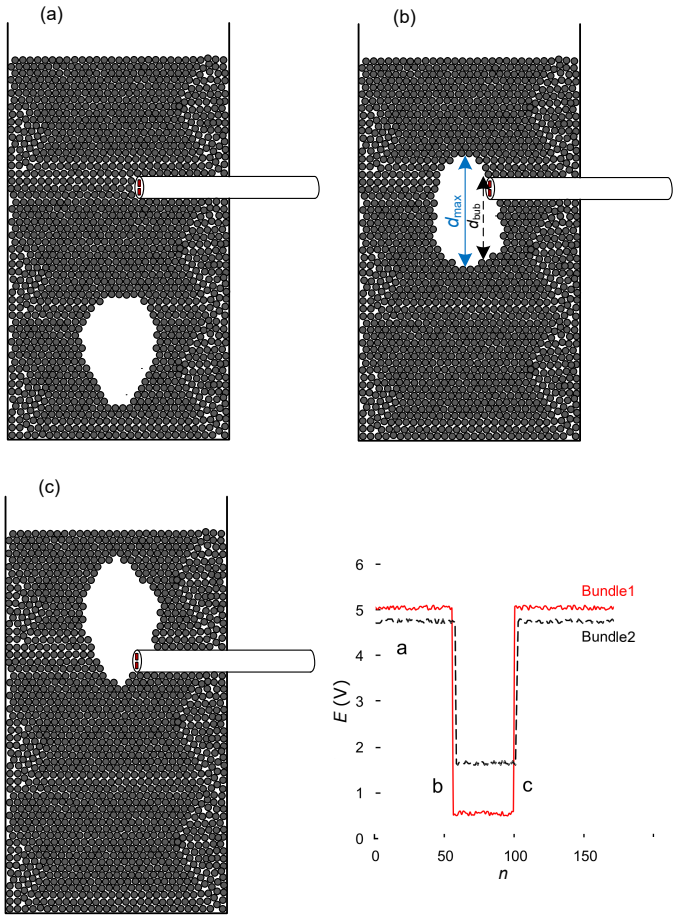

FIGURE 10 Possible positions of probe tip and a single bubble in a fluidized bed and corresponding voltage signals: (a) the bubble below the probe, (b) the probe inside the bubble, and (c) bubble above the probe. Red and black lines correspond to bundles 1 and 2, respectively.

the transition phase, and signals higher than the second threshold (with solids holdup higher than $\approx$ 0.45) show the dense phase.[47]

The bubble length, d_{bub} is the product of the bubble velocity, v_{bub} and the time interval from when the signal drops from a high level to a low level and then returns to the high level, t_{bub}

$$d_{bub} = v_{bub} t_{bub} \tag{8}$$

For each bundle, we find a threshold voltage to determine whether the probe tip is inside or outside a bubble. We estimate these threshold values using the calibration experiments of solids holdup (Figure 3). Voltage signals inside a bubble are close to the voltages of calibration with 100 % black particles, while calibration using 100 % white particles estimates voltages outside a bubble. We choose the thresholds between these two limits to the number of data and the corresponding time, the probe spends inside a bubble. When the signal value is below the threshold, the probe is inside a bubble. Calculation of bubble velocity follows a similar method to the calculation of v_p. When a bubble reaches the probe (Figure 10-b), we calculate the time lag between the data of two bundles with the maximum cross-correlation coefficient and convert it to velocity. Similar to the particles velocity, calculate the bubble velocity, v_{bub} based on the effective probe length of the probe l_e and time delay, τ

$$v_{bub} = \frac{l_e}{\tau} \tag{9}$$

The same method applies to fluidized beds with several bubbles.[48]

Appendix C presents a sample Matlab code to calculate bubble size and rise velocity. First the users enter the required parameters minimum acceptable bubble size, sampling frequency, the maximum number of data for cross-correlation, signals thresholds for detection of bubbles for each bundle, and threshold cross-correlation coefficient for elimination. The code only considers bubbles larger than the specified minimum acceptable bubble size. In the next step, the code calculates the velocities of detected bubbles using the cross-correlation function in the range of the maximum number of data. In the elimination steps, the code removes weak cross-correlated values and checks the sampling frequency, similar to the code for calculation of particle velocity. Finally, the code calculates the bubble sizes using Eq. 8.

3.6 | Bubble holdup

Particles may fall inside the bubbles in fluidized beds from above.[49] Due to this phenomenon, the holdup of bubbles is not equal to zero (pure gas), and it generally increases as a bubble rises inside the bed. Prior to the calculation of the bubble holdup, we need to detect distinct bubbles according to the method explained in the previous section. Then, for each bubble, we obtain the average value of voltage signal. This average voltage value converts into holdup, using holdup calibration curves of each bundle.

4 | UNCERTAINTY

4.1 | Sources of error

The main sources of error for fiber optic probe measurements are:

- Light nuisance: For optical probes operating with visible light requires the experimental setup to be isolated from external light sources like including sunlight.
- Calibration uncertainty: A small discrepancy in the calculation of effective length (l_e) propagates measurement

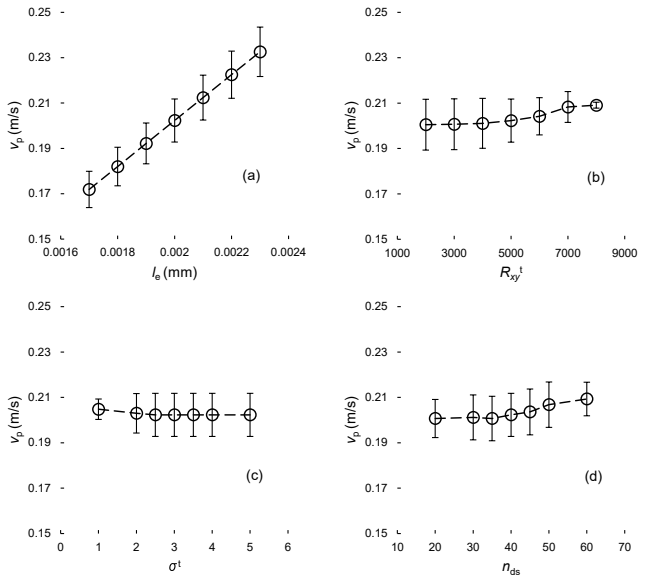

FIGURE 11 Sensitivity analyses on the effects of (a) effective length of the probe, (b) cross correlation threshold value, (c) standard deviation elimination threshold, and (d) number of data segments on particles velocity. We used the code in Appendix B, and the corresponding data for the sensitivity analyses. Error bars show the standard deviation of the particles velocity.

errors for velocity. Figure 11-(a) shows a sensitivity analysis on the effect of the effective length of the probe on the particles velocity. Average particles velocity linearly increases more than 36%, when the effective length increases from 1.7 mm to 2.3 mm. This shows strong dependency of the measured particles velocity on the effective length of the probe. In other words, the probe calibration for velocity measurements (to obtain l_e), should be performed with high accuracy to ensure the correctness of the measured velocities.

- Probe tip erosion: The probe tip is in continuous contact with moving particles, which erodes the probe and alters the holdup calibration curve and l_e. High density particles like zirconia, totally destroy the probe. To mitigate errors due to erosion, we recommend to calibrate the probe frequently (every 10 h or so of operation).

- Rotation of probe during experiment: During experiments (especially velocity measurements), the probe should be located perpendicular to the direction of particle motion (probe bundles locate parallel to the motion direction of particles, similar to Figure 5).

- Disturbance of bed by the probe: The presence of the probe should not affect the hydrodynamics of the experimental apparatus.

- Variable bubble velocities: Bubble velocities increase as they rise through the bed. We recommend taking the average of the bubble velocity as it first hits the probe and when the wake reaches the probe.

- Cross-correlation coefficient threshold: Choosing a low cross-correlation coefficient threshold leads to inclusion of weak data or loss of acceptable data for average velocity. Figure 11-(b) shows a sensitivity analysis on the effect of cross-correlation coefficient threshold on the particles velocity. Average particles velocity slightly (4.2%) increases by increasing the cross-correlation coefficient threshold from 2000 to 8000. Furthermore, the standard deviation of the measured velocity decreases with increasing the cross-correlation coefficient threshold. With increasing this threshold, less velocity data (with higher cross-correlation coefficients) pass the elimination steps and consequently, the standard deviation of velocity decreases.

- Standard deviation multipliers in the elimination step: we recommend that researchers use a standard deviation multiplier in the range of 3-5.[44, 45] Smaller or larger values may exclude or include outliers. Figure 11-(c) shows a sensitivity analysis on the effect of standard deviation multiplier on the particles velocity. We conclude that the effect of standard deviation multiplier in the range of 2-5 is negligible on the particles velocity. Using $\sigma^t = 1$ however, leads to smaller standard deviation. The reason is that less data with smaller deviations from the average value pass the elimination steps.

- Sampling frequency: Sampling frequency in the fiber optic measurements varies according to the system properties, especially local velocity of particles. Incorrect sampling frequency leads to inaccurate velocities.

- Number of data segments: With increasing the data segments, the number of data in each segment decreases. Consequently, we have to choose a number of data segments that leads to adequate number of data in each segment. This adequate number of data varies in different applications according to the particles velocity. As a rule of thumb, each data segment should contain at least signals of motion of 2-3 distinct particles. Figure 11-(d) shows a sensitivity analysis on the effect of number of data segments on the particles velocity. Average particles velocity increases slightly (4.3%) with increasing the number of data segments from 20 to 60.

4.2 | Limitations

- Fiber optic probes only measure particle and bubble velocities in the perpendicular direction (perpendicular to the probe, similar to Figure 5). They are incapable of measuring other components of particle or bubble velocity. This limits the application of fiber optic probe to systems in which the motion of particles or bubbles are dominant in a single direction.
- Fiber optic probes are unable to detect particles smaller than their fiber diameter.
- Some probe designs are unable to detect transparent particles like uncolored glass beads.
- Bubble size measurements using fiber optic probe do not necessarily obtain the maximum bubble size (see Figure 10).

5 | CONCLUSIONS

About 75 % of the raw materials and 50 % of the products in the chemical industries are in the form of granular solids. Chemical, petrochemical, polymer, pharmaceutical, mineral, metallurgical, food, etc. are examples of the industries deal with powders and solid-fluid multi phase flows. There are several experimental methods for studying the hydrodynamics of solid-fluid systems, including pressure signals, acoustics, computed tomography, radioactive particle tracking, and optical fiber measurements. Fiber optic probes are inexpensive, simple and accurate methods, which measure valuable information in multi phase systems, including solids holdup, particle velocity, solids flux, spout diameter, bubble size and rise velocity and bubble holdup. These wide range of applications make them a useful experimental method to study the hydrodynamics of granular systems and fluid-solid contactors. We explained the theory, the details of each application and required calibrations, provide sample codes and experimental data for post-processing and discuss the sources of uncertainty and applications of fiber optic probes. Researchers can use the provided Matlab codes to process their experimental data. Sensitivity analyses show the sensitivity of the measure values to the calibration and data processing parameters.

Nomenclature

d_{bub}	Bubble size
d_{max}	Maximum bubble size
d_{spout}	Spout diameter
E	Voltage
f_s	Sampling frequency
G_s	Solids flux
l_e	Effective length of probe
m	Mass of particle
n	Data number
n_{ds}	Number of data segments
n_d	Delay in data number
N_e	Sample size
R	Spouted bed radius
r	Distance between probe tip projection and the disk center
R_{xy}	Correlation coefficient
R_{xy}^t	Acceptable cross correlation threshold value
S	Signal
t_{bubble}	Bubble time interval in front of the probe
V	Volume
v	Velocity
V_c	Volume of container
z	Axial co-ordinate (height)

Greek letters

α	Solids fraction
ϵ	Void fraction
ω	Rotational velocity
ρ	Density
σ	Standard deviation
σ^t	Standard deviation elimination threshold
τ	Time delay

Subscripts and superscripts

x, y	Sample signals
av	Average
b	Black
bub	Bubble
c	Container
p	Particle
s	Sample
w	White

References

[1] M. Menéndez, J. Herguido, A. Bérard, G. S. Patience, *Can. J. Chem. Eng*. **2019**, *97*, 2383.

[2] S. Golshan, R. Zarghami, N. Mostoufi, *Chem. Eng. Res. Des*. **2017**, *121*, 315.

[3] C. Moliner, F. Marchelli, B. Bosio, E. Arato, *Energies* **2017**, *10*, 1729.

[4] E. J. G. Sewalt, F. Zhang, V. van Steijn, J. R. van Ommen, G. M. Meesters, *KONA Powder Part. J*. **2021**, *38*, 26.

[5] M. Bartels, W. Lin, J. Nijenhuis, F. Kapteijn, J. R. van Ommen, *Prog. Energy Combust. Sci*. **2008**, *34*, 633.

[6] A. Bérard, G. S. Patience, B. Blais, *Can. J. Chem. Eng*. **2020**, *98*, 424.

[7] B. Blais, D. Vidal, F. Bertrand, G. S. Patience, J. Chaouki, *Can. J. Chem. Eng*. **2019**, *97*, 1964.

[8] C. Moliner, F. Marchelli, N. Spanachi, A. Martinez-Felipe, B. Bosio, E. Arato, *Chem. Eng. J*. **2019**, *377*, 120466.

[9] S. Golshan, O. Yaman, M. Koksal, G. Kulah, R. Zarghami, N. Mostoufi, *Exp. Therm. Fluid Sci*. **2018**, *96*, 358.

[10] J. R. Ommen, S. Sasic, J. der Schaaf, S. Gheorghiud, F. Johnsson, Marc-OlivierCoppens, *Int. J. Multiph. Flow* **2011**, *37*, 403.

[11] C. Savari, G. Kulah, M. Koksal, R. Sotudeh-Gharebagh, R. Zarghami, N. Mostoufi, *Powder Technol*. **2017**, *316*, 148.

[12] N. P. Franka, T. J. Heindel, *Powder Technol*. **2009**, *193*, 69.

[13] B. Esmaeili, J. Chaouki, C. Dubois, *Can. J. Chem. Eng*. **2008**, *86*, 543.

[14] G. Kulah, S. Sari, M. Koksal, *Ind. & Eng. Chem. Res*. **2016**, *55*, 3131.

[15] S. Maurer, E. C. Wagner, T. J. Schildhauer, J. R. van Ommen, S. M. Biollaz, R. F. Mudde, *Int. J. Multiph. Flow* **2015**, 143.

[16] T. Pugsley, H. Tanfara, S. Malcus, H. Cui, J. Chaouki, C. Winters, *Chem. Eng. Sci*. **2003**, *58*, 3923.

[17] J. Shabanian, M. Mokhtari, J. Chaouki, *Can. J. Chem. Eng*. **2021**.

[18] K. Dubrawski, S. Tebianian, H. Bi, J. Chaouki, N. Ellis, R. Gerspacher, R. Jafari, A. Kantzas, C. Lim, G. S. Patience, T. Pugsley, M. Z. Qi, J. X. Zhu, J. R. Grace, *Powder Technol*. **2013**, *235*, 203.

[19] J. Liu, J. R. Grace, X. Bi, *AIChE J*. **2003**, *49*, 1405.

[20] G. Xu, C. Liang, X. Chen, D. Liu, P. Xu, L. Shen, C. Zhao, *Sensors* **2013**, *13*, 9201.

[21] C. Van Lare, H. Piepers, J. Schoonderbeek, D. Thoenes, *Chem. Eng. Sci*. **1997**, *52*, 829.

[22] H. Taofeeq, S. Aradhya, J. Shao, M. Al-Dahhan, *Flow Meas. Instrumentation* **2018**, *63*, 18.

[23] Z. Wang, H. T. Bi, C. J. Lim, *Can. J. chemical engineering* **2009**, *87*, 264.

[24] M. Olazar, M. J. San Jose, R. LLamosas, S. Alvarez, J. Bilbao, *Ind. & Eng. Chem. Res*. **1995**, *34*, 4033.

[25] M. Olazar, M. J. San José, S. Alvarez, A. Morales, J. Bilbao, *Ind. & Eng. Chem. Res*. **1998**, *37*, 4520.

[26] M. J. San José, M. Olazar, S. Alvarez, M. A. Izquierdo, J. Bilbao, *Chem. Eng. Sci*. **1998**, *53*, 3561.

[27] M. Olazar, M. J. San José, M. A. Izquierdo, A. O. de Salazar, J. Bilbao, *Chem. Eng. Sci*. **2001**, *56*, 3585.

[28] M. San José, S. Alvarez, A. Morales, M. Olazar, J. Bilbao, *Chem. Eng. Res. Des*. **2006**, *84*, 487.

[29] M. J. San José, M. Olazar, S. Alvarez, J. Bilbao, *Ind. & Eng. Chem. Res*. **1998**, *37*, 2553.

[30] M. J. San José, M. Olazar, S. Alvarez, A. Morales, J. Bilbao, *Chem. Eng. Sci*. **2005**, *60*, 2017.

[31] X. Wei, J. Zhu, *Chem. Eng. Sci*. **2019**, *207*, 713.

[32] X. Wei, J. Yang, J. Zhu, *Chem. Eng. J*. **2020**, *379*, 122301.

[33] X. Wei, J. Zhu, *Chem. Eng. Sci*. **2020**, *228*, 115946.

[34] M. J. San José, M. Olazar, S. Alvarez, A. Morales, J. Bilbao, *Ind. & Eng. Chem. Res*. **2005**, *44*, 193.

[35] Y. Xu, T. Li, L. Lu, S. Tebianian, J. Chaouki, T. W. Leadbeater, R. Jafari, D. J. Parker, J. Seville, N. Ellis, et al., *Chem. Eng. Sci*. **2019**, *195*, 356.

[36] X. Ma, Z. Sun, H. Luo, X. Li, *Appl. Sci*. **2020**, *10*, 4829.

[37] X. Li, C. Yang, S. Yang, G. Li, *Sensors* **2012**, *12*, 12519.

[38] M. O. Guerrero-Pérez, G. S. Patience, M. A. Bañares, *Can. J. Chem. Eng*. **2021**, *99*, 97.

[39] H. Abramczyk, Introduction to laser spectroscopy, Elsevier **2005**.

[40] G. S. Patience, *Can. J. Chem. Eng*. **2018**, *96*, 2312.

[41] H. Cui, N. Mostoufi, J. Chaouki, *Fluid. X* **2001**, 779.

[42] J.-X. Zhu, G.-Z. Li, S.-Z. Qin, F.-Y. Li, H. Zhang, Y.-L. Yang, *Powder technology* **2001**, *115*, 184.

[43] J.-P. Laviolette, G. S. Patience, J. Chaouki, *AIChE J*. **2010**, *56*, 2850.

[44] Z. Wang, Experimental studies and CFD simulations of conical spouted bed hydrodynamics, Ph.D. thesis, University of British Columbia **2006**.

[45] G. Kirbas, Solids motion and mixing in high-density circulating fluidized beds, Ph.D. thesis, University of British Columbia **2004**.

[46] G. S. Patience, Experimental Methods and Instrumentation for Chemical Engineers, 2nd edition, Elsevier B.V., Amsterdam, Netherlands **2017**.

[47] X. Wei, J. Liu, J. Zhu, *Powder Technol*. **2021**, *379*, 223.

[48] S. Okhovat-Alavian, J. Behin, N. Mostoufi, *Powder Technol*. **2020**.

[49] F. Mostafaei, S. Golshan, R. Zarghami, R. S. Gharebagh, N. Mostoufi, *Powder Technol*. **2020**, *366*, 938.

APPENDIX

A SOLIDS HOLDUP

Listing 1: Sample Matlab code for calculation of solids holdup

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%              Author: Shahab Golshan, 2021                  %%
%% A sample Matlab code for processing solids holdup %%
%% measurements.                                               %%
%%                                                             %%
%% Before running the code, data from two probe               %%
%% bundles should be saved as experiment1.mat and             %%
%% experiment2.mat in the same folder as the main             %%
%% code.                                                       %%
%%                                                             %%
%% Inputs: data (time-series) from two probe bundles %%
%% (experiment1.mat and experiment2.mat)                      %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc
clear all
close all

% Loading experimental data. Here it is assumed that data are saved in .mat format and data from bundle 1 and 2 are stored as experiment1
% and experiment2
load('experiment1.mat');
load('experiment2.mat');

% Segmentation of signals
n_segments = input('Please enter number of data segments ','s');
n_segments = str2num(n_segments);

% Standard deviation elimination threshold
std_elimination_threshold = input('Please enter standard deviation elimination threshold ','s');
std_elimination_threshold = str2num(std_elimination_threshold);

data_number_in_segment = floor(length(voltage_signal_bundle1) / n_segments);
for i = 1 : n_segments
    bundle1_segment(:,i) = voltage_signal_bundle1(((i - 1) * data_number_in_segment + 1) : i * data_number_in_segment);
    bundle2_segment(:,i) = voltage_signal_bundle2(((i - 1) * data_number_in_segment + 1) : i * data_number_in_segment);
end

% Average voltages of bundle1 and bundle2
for i = 1 : n_segments
    average_voltage_bundle1(i) = mean(bundle1_segment(:,i));
    average_voltage_bundle2(i) = mean(bundle2_segment(:,i));
end

% Using calibration equataion to obtain local holdups from bundle1 and bundle2
for i = 1 : n_segments
    local_holdup_bundle1(i) = 0.133 * average_voltage_bundle1(i) + 0.177;
    local_holdup_bundle2(i) = 0.079 * average_voltage_bundle2(i) + 0.457;
end

% Elimination of outliers
local_holdup_bundle1 = local_holdup_bundle1(find(abs(local_holdup_bundle1 - mean(local_holdup_bundle1)) < std_elimination_threshold * std(local_holdup_bundle1)));
local_holdup_bundle2 = local_holdup_bundle2(find(abs(local_holdup_bundle2 - mean(local_holdup_bundle2)) < std_elimination_threshold * std(local_holdup_bundle2)));

% Local holdup is the average of these two values (bundles 1 and 2)
bundle_local_holdup1 = mean(local_holdup_bundle1);
bundle_local_holdup2 = mean(local_holdup_bundle2);
local_holdup = (bundle_local_holdup1 + bundle_local_holdup2) / 2;

disp('The local holdup is: ');
disp(local_holdup);
```

B PARTICLE VELOCITY

Listing 2: Sample Matlab code for calculation of particle velocity

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%              Author: Shahab Golshan, 2021                  %%
%% A sample Matlab code for processing particles'            %%
%% velocity measurements.                                     %%
%%                                                             %%
%% Before running the code, data from two probe               %%
%% bundles should be saved as experiment1.mat and             %%
%% experiment2.mat in the same folder as the main             %%
%% code.                                                       %%
%%                                                             %%
%% Inputs: data (time-series) from two probe bundles %%
%% (experiment1.mat and experiment2.mat)                      %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc
clear all
close all

% Sampling freuqnecy in experiment
sampling_frequency = input('Please enter sampling frequency ','s');
sampling_frequency = str2num(sampling_frequency);

```

```
% Probe equivalent bundle distance
probe_equivalent_length = input('Please enter probe equivalent length ','s');
probe_equivalent_length = str2num(probe_equivalent_length);

% Cross-correlation elimination threshold
cross_correlation_threshold = input('Please enter acceptable cross correlation threshold value ','s');
cross_correlation_threshold = str2num(cross_correlation_threshold);

% Standard deviation elimination threshold
std_elimination_threshold = input('Please enter standard deviation elimination threshold ','s');
std_elimination_threshold = str2num(std_elimination_threshold);

% Number of data segments
n_segments = input('Please enter number of data segments ','s');
n_segments = str2num(n_segments);

% Loading experimental data. Here it is assumed that data are saved in .mat format and data from bundle 1 and 2 are stored as experiment1
% and experiment2
load('experiment1.mat');
load('experiment2.mat');

% Segmentation of signals
data_number_in_segment = floor(length(voltage_signal_bundle1) / n_segments);
for i = 1 : n_segments
    bundle1_segment(:,i) = voltage_signal_bundle1(((i - 1) * data_number_in_segment + 1) : i * data_number_in_segment);
    bundle2_segment(:,i) = voltage_signal_bundle2(((i - 1) * data_number_in_segment + 1) : i * data_number_in_segment);
end

% Obtaining cross correlation curves for each data segment
for i = 1 : n_segments
    [c,lags] = xcorr(bundle2_segment(:,i) , bundle1_segment(:,i));
    maximum_number(i) = find(c == max(c(:)));
    maximum_time(i) = lags(maximum_number(i)) ./ sampling_frequency;
    cross_correlation_value(i) = c(maximum_number(i));

    if maximum_time(i)  == 0
        error('Maximum cross correlation occured at time 0, inappropriate data');
    end
end

% Converting time lag to local average velocity of particles
for i = 1 : n_segments
    raw_particle_velocity(i) = probe_equivalent_length ./ maximum_time(i);
end

% Elimination of inappropriate data because of data segment size
j = 1;
for i = 1 : length(raw_particle_velocity)
    if abs(lags(maximum_number(i))) == length(bundle1_segment(:,i))
        fprintf('data segment size is too short, you can consider decreasing the number of data segments \n');
    else
        raw_particle_velocity_elimination1(j) = raw_particle_velocity(i);
        j = j + 1;
    end
end

% Elimination of weak cross-correlation values
j = 1;
for i = 1 : length(raw_particle_velocity_elimination1)
    if cross_correlation_value(i) > cross_correlation_threshold
        raw_particle_velocity_elimination2(j) = raw_particle_velocity_elimination1(i);
        j = j + 1;
    end
end

% Elimination of inappropriate data because of sampling frequency
j = 1;
for i = 1 : length(raw_particle_velocity_elimination2)
    if abs(lags(maximum_number(i))) == 1
        fprintf('sampling frequency is not acceptable, you can consider increasing the sampling frequency \n')
    else
        raw_particle_velocity_elimination3(j) = raw_particle_velocity_elimination2(i);
        j = j + 1;
    end
end

% Elimination of outliers
raw_particle_velocity_elimination3 = raw_particle_velocity_elimination3(find(abs(raw_particle_velocity_elimination3 - mean(raw_particle_velocity_elimination3)) < std_elimination_threshold * std(raw_particle_velocity_elimination3)));

% Local particle velocity is the average of different segment values
particle_velocity = mean(raw_particle_velocity_elimination3);

disp('The local velocity is: ');
disp(particle_velocity);
disp(' m/s');
```

C BUBBLE SIZE AND RISE VELOCITY

Listing 3: Sample Matlab code for calculation of bubble size and rise velocity

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%              Author: Shahab Golshan, 2021                    %%
%% A sample Matlab code for calculation of bubble               %%
%% size and rise velocity.                                      %%
%%                                                              %%
%% Before running the code, data from two probe                 %%
%% bundles should be saved as experiment1.mat and               %%
%% experiment2.mat in the same folder as the main               %%
%% code.                                                        %%
%%                                                              %%
%% Inputs: data (time-series) from two probe bundles            %%
%% (experiment1.mat and experiment2.mat)                        %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc
```

```
clear all
close all

% Loading experimental data. Here it is assumed that data are saved in .mat
% format and data from bundles 1 and 2 are stored in columns 1 and 2 of the
% experiment.mat file, respectively
mydata = load('experiment.mat');

% Probe equivalent bundle distance
probe_equivalent_length = input('Please enter probe equivalent length ','s');
probe_equivalent_length = str2num(probe_equivalent_length);

% Minimum acceptable bubble size
minimum_bubble_size = input('Please enter minimum acceptable bubble size ', 's');
minimum_bubble_size = str2num(minimum_bubble_size);

% Sampling freuqnecy in experiment
sampling_frequency = input('Please enter sampling frequency ', 's');
sampling_frequency = str2num(sampling_frequency);

% Maximum number of data for searching maximum cross-correlation
bubble_velocity_search_domain = input('Please enter maximum number of lags for cross-correlation function ', 's');
bubble_velocity_search_domain = str2num(bubble_velocity_search_domain);

% Signal threshold for detection of bubbles for bundle1
bubble_signal_threshold1 = input('Please enter bubble signal threshold for bundle1. Signals below this threshold are considered as a bubble ', 's');
bubble_signal_threshold1 = str2num(bubble_signal_threshold1);

% Signal threshold for detection of bubbles for bundle2
bubble_signal_threshold2 = input('Please enter bubble signal threshold for bundle2. Signals below this threshold are considered as a bubble ', 's');
bubble_signal_threshold2 = str2num(bubble_signal_threshold2);

% Defining minimum acceptable cross-correlation value for elimination
cross_correlation_threshold = input('Please enter acceptable cross-correlation threshold value ', 's');
cross_correlation_threshold = str2num(cross_correlation_threshold);

% Converting minimum acceptable bubble size to data number using sampling frequency
bubble_length_minimum_data_number = minimum_bubble_size * sampling_frequency;

% Defining a few counters for calculations
j = 1;
n = 0;
l = 1;
p = 0;

% Finding bubbles
for i = 1 : length(mydata)
    if mydata(i,1) < bubble_signal_threshold1
        n = n + 1;
    else
        rawdata1(j) = n;
        n = 0;
        j = j + 1;
    end

    if mydata(i,2) < bubble_signal_threshold2
        p = p + 1;
    else
        rawdata2(l) = p;
        p = 0;
        l = l + 1;
    end
end

% Finding bubbles larger than minimum acceptable bubble size
k = 1;
for i = 1 : length(rawdata1)
    if rawdata1(i) >= bubble_length_minimum_data_number
        bubble1(k) = rawdata1(i);
        k = k + 1;
    end
end

k = 1;
for i = 1 : length(rawdata2)
    if rawdata2(i) >= bubble_length_minimum_data_number
        bubble2(k) = rawdata2(i);
        k = k + 1;
    end
end

% Converting to time
t1 = bubble1 ./ sampling_frequency;
t2 = bubble2 ./ sampling_frequency;

% Finding data positions attributed to detected bubbles
accepted_data_locations1 = find(rawdata1 >= bubble_length_minimum_data_number);
accepted_data_locations2 = find(rawdata2 >= bubble_length_minimum_data_number);

if length(t1) < length(t2)
    time = t1;
    accepted_data_locations = accepted_data_locations1;
    signal1 = mydata(:,1);
    signal2 = mydata(:,2);
    rawdata = rawdata1 + 1;
    cumulative_rawdata = cumsum(rawdata);
else
    time = t2;
    accepted_data_locations = accepted_data_locations2;
    signal1 = mydata(:,1);
    signal2 = mydata(:,2);
    rawdata = rawdata1 + 1;
    cumulative_rawdata = cumsum(rawdata);
end

% Calculation of cross-correlation for bublle rise velocity
for i = 1 : length(accepted_data_locations)
    signalp1 = signal1(cumulative_rawdata(accepted_data_locations(i)-1)-bubble_velocity_search_domain:cumulative_rawdata(accepted_data_locations(i))+bubble_velocity_search_domain);
    signalp2 = signal2(cumulative_rawdata(accepted_data_locations(i)-1)-bubble_velocity_search_domain:cumulative_rawdata(accepted_data_locations(i))+bubble_velocity_search_domain);


    [c, lags] = xcorr(signalp2 , signalp1);
    maximum_number(i) = find( c == max (c (:)));
```

```
    maximum_time(i) = lags(maximum_number( i ) ) ./
        sampling_frequency;
    cross_correlation_value(i) = c (maximum_number(i));
end

data_before_elimination = [maximum_time',
    cross_correlation_value'];

% Elimination of weak cross-correlation values
    j = 1;
    for i = 1 : length(data_before_elimination)
        if data_before_elimination(i,2) >
            cross_correlation_threshold
            data_after_elimination1(j,:) =
                data_before_elimination(i,:);
            time_after_elimination1(j) = time(i);
            j = j + 1;
        end
    end

    % Elimination of inappropriate data because of
        sampling frequency
    k = 1;
    for i = 1 : length(data_after_elimination1)
        if data_after_elimination1(i,1) > 0
            data_after_elimination2(k,:) =
                data_after_elimination1(i,:);
            time_after_elimination2(k) =
                time_after_elimination1(i);
            k = k + 1;
        end
    end

maximum_time    = data_after_elimination2(:,1);

% Calculation of bubble rise velocity and bubble size
bubble_velocity  = probe_equivalent_length ./
    maximum_time;
bubble_size = bubble_velocity .*
    time_after_elimination2';

% Visualization and printing information
hist(bubble_size)
xlabel('Bubble size (m)');
ylabel('Number of bubbles');
mean(bubble_size)
std(bubble_size)
```