



**ARISTOTLE UNIVERSITY OF THESSALONIKI
SCHOOL OF GEOLOGY**

M.SC. "HYDROCARBONS EXPLORATION AND EXPLOITATION"

"DEVELOPMENT OF A VIRTUAL FLOW METER FOR A SUBSEA FIELD USING MACHINE LEARNING"

**MASTER'S THESIS
CHARISOPOULOU IOANNA**

**SUPERVISOR: Dr. GAGANIS VASILEIOS, ASSOCIATE PROFESSOR,
NATIONAL TECHNICAL UNIVERSITY OF ATHENS (NTUA)**

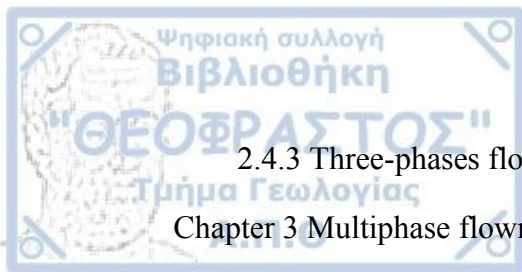
THESSALONIKI, 2024



Table of Contents

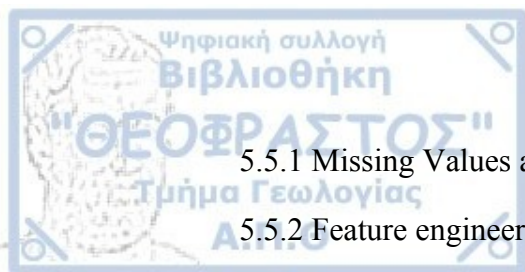
List of Figures.....	vi
List of Equations.....	x
List of Tables.....	xii
Abstract.....	xiii
Περίληψη.....	xiv
Acknowledgements.....	xvi
Chapter 1 Introduction.....	1
Chapter 2 Multiphase flow.....	3
2.1 Fundamentals of multiphase flows.....	3
2.1.1 Void fraction.....	3
2.1.2 Pressure gradient.....	4
2.2 Two-phase flow.....	5
2.2.1 Two-phase flow patterns in vertical pipes.....	7
2.2.1.1 Bubbly flow.....	8
2.2.1.2 Slug flow.....	8
2.2.1.3 Churn flow.....	9
2.2.1.4 Annular flow.....	10
2.2.1.5 Mist flow.....	11
2.2.2 Two-phase flow patterns in horizontal pipes.....	12
2.2.2.1 Bubble flow.....	13
2.2.2.2 Stratified flow.....	13
2.2.2.3 Wavy stratified flow.....	14
2.2.2.4 Intermittent flow.....	14
2.2.2.4.1 Plug flow.....	15
2.2.2.4.2 Slug flow.....	15
2.2.2.5 Annular flow.....	16

2.2.3 Flow regime maps.....	16
2.3 Two-phase flow pattern transitions and mechanisms.....	22
2.3.1 Transition from Bubbly to Slug flow.....	23
2.3.2 Transition from Slug to churn flow.....	25
2.3.3 Transition from Churn to Annular flow.....	26
2.4 Three-phase flow.....	27
2.4.1 Three-phase flow patterns in vertical pipes.....	29
2.4.1.1 Oil-Dominated Oil Annulus/Dispersed Annular.....	29
2.4.1.2 Oil-Dominated Broken Annulus.....	30
2.4.1.3 Oil-Dominated Dispersed Churn.....	31
2.4.1.4 Oil-Dominated Dispersed Slug.....	31
2.4.1.5 Water-Dominated Water Annulus/Oil Annular.....	31
2.4.1.6 Water-Dominated Dispersed Annulus/Oil Annular.....	31
2.4.1.7 Water-Dominated Dispersed Churn.....	31
2.4.1.8 Water-Dominated Dispersed Slug.....	32
2.4.1.9 Water-Dominated Oil Slug.....	32
2.4.2 Three-phase flow patterns in horizontal pipes.....	33
2.4.2.1 Oil-based dispersed plug flow.....	34
2.4.2.2 Oil-based dispersed slug flow.....	34
2.4.2.3 Oil-based dispersed stratified/wavy flow.....	35
2.4.2.4 Oil-based separated stratified/wavy flow.....	35
2.4.2.5 Oil-based separated wavy stratifying-annular flow.....	36
2.4.2.6 Oil-based separated/dispersed stratifying-annular flow.....	36
2.4.2.7 Water-based dispersed slug flow.....	37
2.4.2.8 Water-based dispersed stratified/wavy flow.....	37
2.4.2.9 Water-based separated/dispersed incipient stratifying-annular flow.....	38
2.4.2.10 Water-based dispersed stratifying-annular flow.....	38



2.4.3 Three-phases flow pattern transition and mechanisms.....	39
Chapter 3 Multiphase flowmeters.....	40
3.1 MPFMs -Test separators.....	40
3.2 History and Development of VFM.....	44
3.3 VFM Methods.....	47
3.4 First principles VFM systems.....	49
3.4.1 Commercial first principles VFM systems.....	51
3.4.2 Components of the first principles VFM systems.....	52
3.4.2.1 Fluid properties model.....	52
3.4.2.1.1 Black Oil model.....	53
3.4.2.1.2 Compositional model.....	54
3.4.2.2 Production system model.....	56
3.4.2.2.1 Reservoir inflow model.....	57
3.4.2.2.1.1 Linear.....	57
3.4.2.2.1.2 Vogel.....	58
3.4.2.2.1.3 Fetkovich's equation.....	58
3.4.2.2.1.4 Backpressure/Backpressure normalized.....	58
3.4.2.2.1.5 Forchheimer/Single Forchheimer.....	59
3.4.2.2.1.6 IPR table.....	59
3.4.2.2.2 Thermal-hydraulic model.....	60
3.4.2.2.2.1 Two-fluid model.....	60
3.4.2.2.2.2 Drift-flux model.....	63
3.4.2.2.2.3 Steady state mechanistic model.....	65
3.4.2.2.3 Choke model.....	66
3.4.2.2.3.1 Modified Bernoulli.....	67
3.4.2.2.3.2 Hydro (Long and Short).....	68
3.4.2.2.3.3 Perkins.....	68

3.4.2.2.4 Electric submersible pump (ESP) model.....	69
3.4.2.3 Data validation and reconciliation (DVR) algorithm.....	70
3.5 Data-driven VFM.....	72
3.6 Data-driven VFM components and applied methods.....	75
3.6.1 Data collection and pre-processing.....	75
3.6.2 Model development.....	76
3.7 Comparison of First principles with the Data driven VFMs.....	80
Chapter 4 Machine learning approaches for VFMs.....	81
4.1 Supervised and Unsupervised Machine Learning.....	81
4.2 Implemented techniques for data-driven VFM systems.....	84
4.2.1 Artificial neural network VFM solutions.....	84
4.2.1.1 Steady state artificial neural network VFM solutions.....	84
4.2.1.2 Dynamic artificial neural network VFM solutions.....	90
4.2.2 Support Vector Machine – Support Vector Regression.....	94
4.2.3 Decision trees.....	103
4.2.4 Random Forest.....	106
4.3 Parameters optimization.....	109
Chapter 5 Case of study.....	111
5.1 Introduction.....	111
5.2 Subsea network construction in Pipesim.....	111
5.3 Data collection.....	114
5.3.1 Dataset 1 ($GOR_1=GOR_2$).....	114
5.3.2 Dataset 2 ($GOR_1 \neq GOR_2$).....	114
5.3.3 Dataset 3 (Watercut).....	115
5.3.4 Dataset 4 (Combined).....	116
5.4 Data description.....	116
5.5 Pre-processing.....	118



5.5.1 Missing Values and Data Normalization.....	118
5.5.2 Feature engineering and Target engineering.....	120
5.5.2.1 Feature engineering.....	120
5.5.2.2 Target engineering.....	120
5.6 Feasibility Analysis and Data Feasibility Assessment.....	121
5.6.1 Dataset 1 ($GOR_1=GOR_2$) – Regression Learner App.....	123
5.6.2 Dataset 2 ($GOR_1\neq GOR_2$) – Regression Learner App.....	129
5.6.3 Dataset 3 (Watercut) – Regression Learner App.....	132
5.6.4 Dataset 4 (Combined) – Regression Learner App.....	136
5.7 Model development.....	140
5.7.1 Dataset 1 ($GOR_1=GOR_2$) – Neural Net Fitting App.....	140
5.7.2 Dataset 2 ($GOR_1\neq GOR_2$) – Neural Net Fitting App.....	142
5.7.3 Dataset 3 (Watercut) – Neural Net Fitting App.....	144
5.7.4 Dataset 4 (Combined) – Neural Net Fitting App.....	146
Chapter 6 Conclusions.....	148
References.....	149

List of Figures

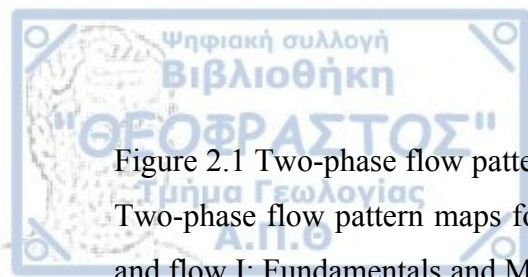


Figure 2.1 Two-phase flow patterns in vertical upflow Thome, J. R., & Cioncolini, A. (2016). Two-phase flow pattern maps for microchannels. in encyclopedia of two-phase heat transfer and flow I: Fundamentals and Methods (Pp. 47-84).....	10
Figure 2.2 Two-phase flow patterns in horizontal flow (from Silva Lima, 2011). Thome, J. R., & Cioncolini, A. (2016). Two-phase flow pattern maps for microchannels. in encyclopedia of two-phase heat transfer and flow I: Fundamentals and Methods (Pp. 47-84).	14
Figure 2.3 Two-phase horizontal flow map. Knotek, S., Fiebach, A., & Schmelter, S. (2016, September). Numerical simulation of multiphase flows in large horizontal pipes. In Proceedings of the 17th International Flow Measurement Conference FLOMEKO.....	19
Figure 2.4 Flow regime map for horizontal flow showing Mandhane et al. (1974) and Taitel et al. (1976) transitions (Brennen, 2005). Ali, S. F. (2009). Two-phase flow in a large diameter vertical riser (Doctoral dissertation, Cranfield University).....	20
Figure 2.5 Flow regimes map for vertical upflow showing Taitel et al. (1980) and Mishima and Ishii (1984) transitions (Mishima and Ishii, 1984). Ali, S. F. (2009). Two-phase flow in a large diameter vertical riser (Doctoral dissertation, Cranfield University).....	20
Figure 2.6 Reproduced gas-liquid flow pattern map for vertical upward flow.....	21
Figure 2.7 Flow patterns map adapted from reference (Mandhane et al., 1974).....	21
Figure 2.8 Oil dominated flow regimes in vertical three-phase flow Woods, G. S., Spedding, P. L., Watterson, J. K., & Raghunathan, R. S. (1998). Three-phase oil/water/air vertical flow. Chemical Engineering Research and Design, 76(5), 571-584.....	30
Figure 2.9 Water dominated flow regimes in vertical three-phase Woods, G. S., Spedding, P. L., Watterson, J. K., & Raghunathan, R. S. (1998). Three-phase oil/water/air vertical flow. Chemical Engineering Research and Design, 76(5), 571-584.....	32
Figure 2.10 Schematic diagram of oil-based dispersed plug flow (left) - Schematic diagram of oil-based dispersed slug flow (right), Açıkgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.....	34
Figure 2.11 Schematic diagram of oil-based dispersed stratified / wavy flow (left) - Schematic diagram of oil-based separated stratified/wavy flow (right), Açıkgöz, M., Franca,	

F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.....	35
Figure 2.12 Schematic diagram of oil-based separated wavy stratifying-annular flow (left) - Schematic diagram of oil-based separated / dispersed stratifying-annular flow (right), Açikgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.....	36
Figure 2.13 Schematic diagram of water-based dispersed slug flow (left) - Schematic diagram of water-based dispersed stratified/wavy flow (right), Açikgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.....	37
Figure 2.14 Schematic diagram of water-based separated dispersed incipient stratifying-annular flow (left) - Schematic diagram of water-based dispersed stratifying-annular flow (right), Açikgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.....	38
Figure 2.15 Schematic diagram of the boundary between water-based wavy flow and oil-based wavy flow., Açikgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.....	39
Figure 3.1 Physic based, first principle Virtual Flow Meter (VFM), Data Driven Versus Transient Multiphase Flow Simulator for Virtual Flow Meter Application, p.2, Mohd Azmin B Ishak, Idris B Ismail, Tareq Aziz Hasan Al-qutami.....	52
Figure 3.2 Data driven Virtual Flow Meter (VFM) Ishak, M. A. B., Ismail, I. B., & Al-Qutami, T. A. H. (2021, July). Data Driven Versus Transient Multiphase Flow Simulator for Virtual Flow Meter Application. In 2020 8th International Conference on Intelligent and Advanced Systems (ICIAS) (pp. 1-4). IEEE.....	74
Figure 3.3 Standard (left) and nested (right) K-fold cross-validation schemes for data-driven models. Timur Bismukhametov, Johannes Jäschke 2020, First Principles and Machine Learning Virtual Flow Metering: A Literature Review P.13.....	78
Figure 3.4 Workflow for data driven VFM Ishak, M. A., Hasan, A. Q., Aziz, T., Ellingsen, H., Ruden, T., & Khaledi, H. (2020, October). Evaluation of Data Driven Versus Multiphase Transient Flow Simulator for Virtual Flow Meter Application. In Offshore Technology Conference Asia. OnePetro.....	79

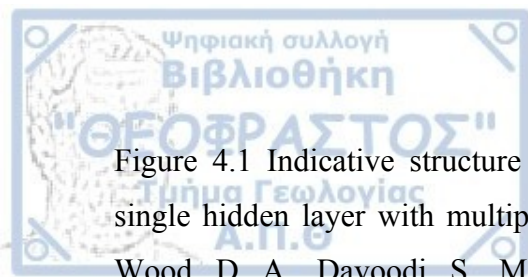


Figure 4.1 Indicative structure of a feed-forward artificial neural network (ANN) with a single hidden layer with multiple neurons Barjouei, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. Journal of Petroleum Exploration and Production, 11, 1233-1261.....	84
Figure 4.2 Flow diagram of single layer ANN Barjouei, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. Journal of Petroleum Exploration and Production, 11, 1233-1261.....	87
Figure 4.3 Ensemble learning structure Al-Qutami, T. A., Ibrahim, R., & Ismail, I. (2017, September). Hybrid neural network and regression tree ensemble pruned by simulated annealing for virtual flow metering application. In 2017 IEEE International Conference on Signal and Image Processing Applications (ICSIPA) (pp. 304-309). IEEE.....	88
Figure 4.4 Schematic diagram of SVM classification principle. (a) Class distribution in low dimensional space; (b) Class distribution in high dimensional space. Xu, Q., Wang, X., Luo, X., Tang, X., Yu, H., Li, W., & Guo, L. (2022). Machine learning identification of multiphase flow regimes in a long pipeline-riser system. Flow Measurement and Instrumentation, 88, 102233.....	96
Figure 4.5 The soft margin loss setting for a linear SVM. Liu, M., Kim, G., Bauckhage, K., & Geimer, M. (2022). Data-Driven Virtual Flow Rate Sensor Development for Leakage Monitoring at the Cradle Bearing in an Axial Piston Pump. Energies, 15(17), 6115.....	102
Figure 4.6 Typical structure of a decision-tree machine-learning model Barjouei, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. Journal of Petroleum Exploration and Production, 11, 1233-126.....	105
Figure 4.7 Schematic diagram of generic configuration of random forest algorithm Barjouei, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. Journal of Petroleum Exploration and Production, 11, 1233-126.....	109
Figure 5.1 Simplified representation of the geometry of the subsea well network.....	113

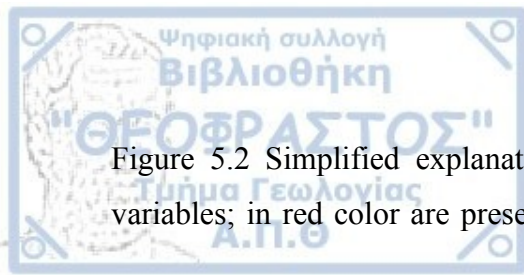
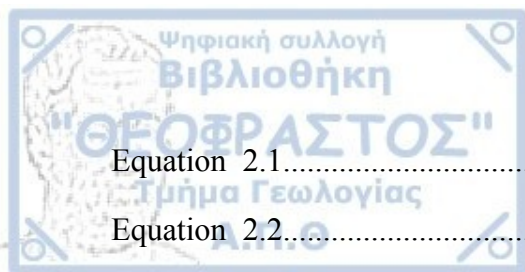


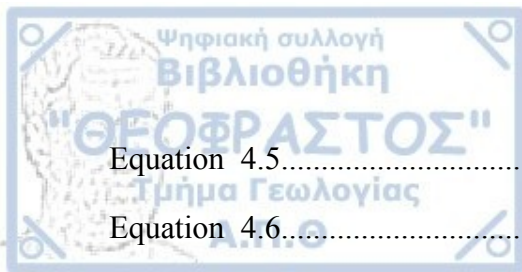
Figure 5.2 Simplified explanation of the distinction between dependent and independent variables; in red color are presented the inputs and in green color are presented the outputs

.....	118
Figure 5.3 Validation set for $GOR_1=GOR_2$ model (Medium Neural Network).....	127
Figure 5.4 Test set for $GOR_1=GOR_2$ model (Medium Neural Network).....	128
Figure 5.5 Validation set for $GOR_1\neq GOR_2$ model (Wide Neural Network).....	131
Figure 5.6 Test set for $GOR_1\neq GOR_2$ model (Wide Neural Network).....	132
Figure 5.7 Test set for Watercut model (Bilayered Neural Network).....	135
Figure 5.8 Validation set for Watercut model (Bilayered Neural Network).....	135
Figure 5.9 Validation set for Combined model (Medium Neural Network).....	139
Figure 5.10 Test set for Combined model (Medium Neural Network).....	140
Figure 5.11 Oil and gas fractions predictions for Well 1 ($GOR_1=GOR_2$ model).....	142
Figure 5.12 Oil and gas fractions predictions for Well 1 ($GOR_1\neq GOR_2$ model).....	144
Figure 5.13 Oil and gas fractions predictions for Well 1 (Watercut model).....	145
Figure 5.14 Oil and gas fractions predictions for Well 1 (Combined model).....	147

List of Equations



Equation 2.1.....	4
Equation 2.2.....	4
Equation 3.1.....	57
Equation 3.2.....	58
Equation 3.3.....	58
Equation 3.4.....	59
Equation 3.5.....	59
Equation 3.6.....	59
Equation 3.7.....	59
Equation 3.8.....	61
Equation 3.9.....	61
Equation 3.10.....	61
Equation 3.11.....	64
Equation 3.12.....	64
Equation 3.13.....	64
Equation 3.14.....	65
Equation 3.15.....	67
Equation 3.16.....	70
Equation 3.17.....	71
Equation 3.18.....	71
Equation 3.19.....	71
Equation 3.20.....	71
Equation 3.21.....	76
Equation 4.1.....	85
Equation 4.2.....	86
Equation 4.3.....	95
Equation 4.4.....	95



Equation 4.5.....	95
Equation 4.6.....	95
Equation 4.7.....	99
Equation 4.8.....	100
Equation 4.9.....	100
Equation 4.10.....	101
Equation 4.11.....	101
Equation 4.12.....	103
Equation 4.13.....	107
Equation 4.14.....	110
Equation 5.1.....	120
Equation 5.2.....	121
Equation 5.3.....	121
Equation 5.4.....	121

List of Tables

Table 3.1 Comparison of First principles with the Data driven VFMs, Timur Bismukhametov, Johannes Jäschke 2020, First Principles and Machine Learning Virtual Flow Metering: A Literature Review P.13.....	80
--	----

Table 4.1 Kernel functions commonly used with SVM algorithms, Barjouei, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. Journal of Petroleum Exploration and Production, 11, 1233-1261.....	97
Table 5.1 Performance evaluation metrics of different machine learning models for $GOR_1=GOR_2$ model.....	125
Table 5.2 Performance evaluation metrics of different machine learning models for $GOR_1=GOR_2$ model (normalized values).....	127
Table 5.3 Performance evaluation metrics of different machine learning models for $GOR_1 \neq GOR_2$ model.....	129
Table 5.4 Performance evaluation metrics of different machine learning models for $GOR_1 \neq GOR_2$ model (normalized values).....	131
Table 5.5 Performance evaluation metrics of different machine learning models for Watercut model.....	133
Table 5.6 Performance evaluation metrics of different machine learning models for Watercut model (normalized values).....	133
Table 5.7 Performance evaluation metrics of different machine learning models for Combined model.....	137
Table 5.8 Performance evaluation metrics of different machine learning models for Combined model (normalized values).....	138
Table 5.9 Training results for model $GOR_1=GOR_2$	141
Table 5.10 Training results for $GOR_1 \neq GOR_2$ model.....	143
Table 5.11 Training results for Watercut model.....	145
Table 5.12 Training results for Combined model.....	146

Abstract

The accurate measurement of individual well production is a critical challenge in hydrocarbon production, especially in subsea fields where traditional methods such as test separators are not feasible. In the demanding environment of subsea hydrocarbon fields,

employing test separators for precise measurement of individual well production may not always be a practical choice. This is primarily attributable to the logistical complexities involved in accessing deep-sea wellheads, potential flow disruption and corrosion issues associated with the introduction of test separators. Additionally, limited space availability on the seabed, along with the challenge of maintaining and calibrating the test separators at significant depths, render them a less favorable option. To address this issue, Multiphase Flowmeters (MPFMs) are commonly used, but their installation requires complex underwater operations and incurs significant costs. An innovative alternative is the implementation of Virtual Multiphase Flowmeters (VFMs), where flow rates at each wellhead are estimated based on available measurements. This master thesis investigates the development and application of VFMs to estimate the production of each phase from interconnected wells in a subsea field. The research focuses on creating a realistic well network model using a pipeline simulation software to mimic the complex flow dynamics in the field. Through iterative simulations under varying production conditions, relevant data is collected to train suitable machine learning models. The study showcases the potential of VFMs in accurately estimating well production by combining data from permanently installed manometers at the wellheads with total flow measurements obtained from a main separator. The proposed mathematical approach enables the determination of the individual well's contribution to the total production at the separator. The results demonstrate the efficacy of VFMs in providing accurate estimates of each well's production, facilitating optimal well management and enhancing overall hydrocarbon production efficiency. Moreover, the adoption of VFMs eliminates the need for physically separating the flow phases, reducing operational costs and complexity.

Key words: Virtual Multiphase Flowmeter, subsea field, PIPESIM, Machine Learning, MATLAB

Η ακριβής μέτρηση της παραγωγής μεμονωμένων γεωτρήσεων είναι μια κρίσιμη πρόκληση στην παραγωγή υδρογονανθράκων, ειδικά σε υποθαλάσσια πεδία όπου οι παραδοσιακές μέθοδοι όπως οι test separators δεν είναι εφικτές. Στο απαιτητικό περιβάλλον των υποθαλάσσιων πεδίων υδρογονανθράκων, η χρήση test separator για την ακριβή μέτρηση της παραγωγής μεμονωμένων γεωτρήσεων μπορεί να μην είναι πάντα πρακτική επιλογή. Αυτό αποδίδεται κυρίως στην υλικοτεχνική πολυπλοκότητα που σχετίζεται με την πρόσβαση σε κεφαλές γεωτρήσεων σε μεγάλα βάθη υδάτων, την πιθανή διακοπή ροής και τα προβλήματα διάβρωσης που σχετίζονται με τους test separators. Επιπλέον, η περιορισμένη διαθεσιμότητα χώρου στον πυθμένα της θάλασσας, μαζί με την πρόκληση της διατήρησης και της βαθμονόμησης των test separators σε σημαντικά βάθη, τους καθιστούν λιγότερο ευνοϊκή επιλογή. Για την αντιμετώπιση αυτού του ζητήματος, χρησιμοποιούνται συνήθως πολυφασικά ροόμετρα (MPFMs), αλλά η εγκατάστασή τους απαιτεί πολύπλοκες υποθαλάσσιες εργασίες και συνεπάγεται σημαντικό κόστος. Μια καινοτόμος εναλλακτική λύση είναι η εφαρμογή των εικονικών πολυφασικών ροόμετρων (VFM), όπου οι ρυθμοί ροής σε κάθε κεφαλή γεώτρησης υπολογίζονται με βάση τις ήδη διαθέσιμες μετρήσεις. Αυτή η μεταπτυχιακή εργασία διερευνά την ανάπτυξη και την εφαρμογή των VFM για την εκτίμηση της παραγωγής κάθε φάσης από διασυνδεδεμένες γεωτρήσεις σε υποθαλάσσιο πεδίο. Η έρευνα επικεντρώνεται στη δημιουργία ενός ρεαλιστικού μοντέλου δικτύου γεωτρήσεων χρησιμοποιώντας λογισμικό προσομοίωσης αγωγών για να μιμηθεί τη σύνθετη δυναμική ροής στο πεδίο. Μέσω επαναληπτικών προσομοιώσεων υπό διαφορετικές συνθήκες παραγωγής, συλλέγονται σχετικά δεδομένα για την εκπαίδευση κατάλληλων μοντέλων μηχανικής εκμάθησης. Η μελέτη δείχνει τη δυνατότητα των VFM στην ακριβή εκτίμηση της παραγωγής γεωτρήσεων συνδυάζοντας δεδομένα από μόνιμα εγκατεστημένα μανόμετρα στις κεφαλές των γεωτρήσεων με μετρήσεις συνολικής ροής που λαμβάνονται από έναν κύριο διαχωριστή. Η προτεινόμενη μαθηματική προσέγγιση επιτρέπει τον προσδιορισμό της συνεισφοράς της κάθε γεώτρησης ξεχωριστά στη συνολική παραγωγή στον διαχωριστή. Τα αποτελέσματα καταδεικνύουν την αποτελεσματικότητα των VFM στην παροχή ακριβών εκτιμήσεων της παραγωγής κάθε γεώτρησης, διευκολύνοντας τη βέλτιστη διαχείριση της και ενισχύοντας τη συνολική απόδοση παραγωγής υδρογονανθράκων. Επιπλέον, η υιοθέτηση των VFM εξαλείφει την ανάγκη για φυσικό διαχωρισμό των φάσεων ροής, μειώνοντας το λειτουργικό κόστος και την πολυπλοκότητα.



Λέξεις κλειδιά: Εικονικό πολυφασικό ροόμετρο, υποθαλάσσιο πεδίο, PIPESIM, Μηχανική Εκμάθηση, MATLAB

**This thesis is dedicated to my
father, Giorgos, my biggest
supporter.**



Acknowledgements

I would like to express my gratitude to my supervisor, Associate Professor Dr. Vasileios Gaganis, for his guidance and support throughout the entire process of completing my master's thesis. Dr. Gaganis played a pivotal role in helping me select the topic and managed to keep me motivated even during the most challenging moments. His invaluable advice and expertise were essential in shaping this work and I am truly grateful for his dedication and mentorship.

To my dear friends and fellow students, Dimitrios Karagiannis and Nikolaos Theologis, I want to express my deepest appreciation for their unwavering emotional support during these past years of postgraduate study. Through shared difficulties and triumphs, they have been my team and a constant source of strength. (We did it, guys!)

Last but not least, I would like to express my profound gratitude to my family for always being there for me, especially to my father, who has been an exceptional role model in every aspect of my life, both academically and personally. His support and encouragement have been the primary driving force behind my academic journey so far. I am truly indebted to him for exemplifying the value of perseverance and the importance of never giving up.



Chapter 1 Introduction

In contemporary oil and gas exploration and exploitation endeavors, a notable shift has been observed from onshore to offshore locations, primarily attributed to the dwindling reserves onshore and the escalating environmental concerns associated with terrestrial extraction activities. Consequently, offshore environments are now garnering increased attention, necessitating the deployment of equipment in marine environments. Nonetheless, this transition introduces a distinct array of challenges in comparison to onshore operations.

Offshore oil platforms are subject to spatial limitations that necessitate the development of innovative solutions for efficient equipment installation and operation. As a result, there has been a significant reliance on seabed deployment for various types of equipment, including crucial measuring instruments such as multiphase flowmeters (MPFMs). Nevertheless, leveraging the seabed for equipment deployment brings forth novel obstacles, particularly concerning cost escalation and logistical management. Subsea fields often span vast expanses, complicating the coordination and transportation of equipment to these remote locations.

An additional layer of complexity in offshore hydrocarbon exploitation arises from the equitable distribution of ownership stakes and corresponding production rights among companies/operators involved in the development of extensive subsea fields, contrasting with traditional block-based exploitation methods. Contemporary operations often entail interconnected subsea wells, with multiple companies sharing access to central production facilities and possessing rights over specific wells. Precisely determining each company's contribution to overall production is essential, considering the diverse origins of extracted hydrocarbons. Achieving equitable production distribution among these entities presents a multifaceted challenge, particularly when production comes from multiple reservoir zones within each well. Consequently, total production distribution must consider each operator's proportional contributions, accounting for both quantity and quality of extracted hydrocarbons.

The precise determination of individual well contributions necessitates sophisticated technological solutions. Traditional methodologies such as test separators have historically addressed this issue; however, the escalating intricacies of subsea operations demand more advanced approaches. Multiphase flow meters (MPFMs) have emerged as a consequential innovation in this domain, facilitating the precise measurement and allocation of production volumes. These systems utilize sensor arrays to concurrently gauge the flow rates of oil, gas and water, furnishing real-time data essential for accurate assessment.

Multiphase flowmeters offer precise measurement capabilities in subsea environments, yet their implementation presents notable challenges. Their substantial cost and intricate maintenance requirements in such settings pose significant operational hurdles. Furthermore, the necessity for individual flowmeters for each well exacerbates financial burdens, particularly in expansive offshore fields. Additionally, offshore pose threats to equipment durability and accuracy, with harsh environments potentially impacting longevity and environmental factors affecting measurement accuracy over time.

In response to the challenges posed by offshore environments and the limitations associated with traditional multiphase flowmeter deployment, the development of Virtual Multiphase Flowmeters (VFM) has emerged as a promising solution. VFMs have been introduced by advancements in computational fluid dynamics and machine learning algorithms, which enable the estimation of flow rates and phase fractions without reliance on physical equipment. VFMs offer a cost-effective and space-saving alternative by leveraging computational models to simulate multiphase flow behavior and infer flow characteristics based on input parameters such as fluid properties, pipe geometry and operating conditions.

The deployment of VFMs can be remotely conducted and seamlessly integrated into existing control and monitoring systems, providing real-time insights into flow dynamics. Moreover, VFMs offer the flexibility to adapt to evolving operational requirements and changing environmental conditions, facilitating dynamic adjustments and optimization of production processes.

By eliminating the requirement for sensors and related infrastructure, VFMs contribute to cost reduction and operational efficiency, thereby enhancing the viability of offshore exploration and exploitation endeavors. Even in scenarios involving multiple companies within the subsea network, where only a topside flowmeter/separator on the oil platform is utilized to measure overall production, a VFM based on machine learning algorithms can precisely ascertain the proportions of production from each individual well.

This thesis will focus on the development of a data-driven Virtual Multiphase Flowmeter. More specifically, Chapter 2 will detail all the correlations utilized to model multiphase flow in both horizontal and vertical pipelines. In Chapter 3, MPFMs and their virtual counterparts will be thoroughly analysed. In Chapter 4, the focus will be on detailing the machine learning models utilized in the creation of the VFMs. Following this, Chapter 5 will present a case study elucidating a simple network comprising of two wells, with a detailed examination of four variables. This thesis will close with its conclusions.

Chapter 2 Multiphase flow

This chapter provides a thorough exploration of fundamental principles governing multiphase flows, encompassing both two-phase and three-phase scenarios. A comprehension of these principles is vitally important for elucidating the intricate dynamics of multiphase flows, emphasizing critical factors such as flow regime transitions, pressure variations and fluid characteristics. This foundational knowledge serves as a basis for the subsequent development and assessment of the Virtual Multiphase Flowmeter (VFM) introduced in Chapter 4. By examining these pivotal elements, the chapter underscores the compelling necessity for a VFM system capable of overcoming limitations associated with conventional flow measurement methods. This need is particularly pronounced in the context of complex subsea fields, where accurately measuring individual well production poses challenges.

Furthermore, the comprehensive examination of two-phase and three-phase flow phenomena in both horizontal and vertical pipelines is of utmost significance within the framework of this thesis, given that the profound understanding of multiphase flow behavior in various pipe configurations is fundamental for the development of precise and reliable VFM systems. In two-phase flow scenarios, a mixture of two immiscible fluids, typically oil and gas, is conveyed, while three-phase flow introduces a third phase, often water. Variations in flow patterns and behaviors emerge in horizontal and vertical pipes, influenced by gravitational and hydrodynamic effects, exerting a substantial impact on the interaction and distribution of phases within the pipe. This knowledge has a pivotal role in the design of VFMs. It equips them with the capability to effectively address the complexities and uncertainties that are inherent in real-world applications. This is particularly crucial in subsea oil and gas production because ultimately influences the final production outcomes.

2.1 Fundamentals of multiphase flows

2.1.1 Void fraction

In the study of gas-liquid two-phase flow, the cross-sectional void fraction is an essential geometric flow parameter. It indicates the fraction of the total flow volume occupied by the gas phase. The void fraction is one of the most important flow parameters needed to accurately describe two-phase flows: it is required as an input for determining various other crucial flow parameters (such as two-phase viscosity and density or cross-sectional average velocities of the liquid and gas phases) and it has a significant role in modeling two-phase flow regime transitions, heat transfer and pressure drop. Void fraction is a parameter defined

as the ratio of the pipe's cross-sectional area A_g occupied by the gas or vapor phase to the total cross-sectional area A of the pipe as follows:

$$\varepsilon = \frac{A_g}{A} \quad \text{Equation 2.1}$$

As described in Equation 2.1, it is symbolized by the symbol (epsilon) and is stated as a fraction or decimal between 0 and 1. A void fraction of 0 ($A_g = 0$ corresponding to only liquid in the pipe cross section) indicates the absence of gas in the flow, while a void fraction of 1 ($A_g = A$ corresponding to only gas or vapor in the pipe cross section) indicates that the flow is fully comprised of gas.

Values of the void fraction close to 0+ are typical of bubbly two-phase flow where the large percentage of the pipe is occupied by the liquid phase and only a few gas or vapor bubbles are entrained in the continuous liquid phase, whereas values of the void fraction close to 1– are characteristic of dispersed mist flow, where the gas or vapor occupy the largest percentage of the pipe and only a few liquid droplets are entrained in the continuous gas phase.

A different term utilized in two phase flows to describe the ratio of each phase to the total volume of the pipe (Ali, S. F. 2009), is “hold-up or fraction”. However, it has prevailed to be used to describe the liquid fraction rather than the gas volume fraction hence, the term void fraction corresponds to gas volume fraction.

In practice, the void fraction is estimated using a range of experimental approaches, including direct measurement, void fraction correlations and empirical models. These approaches strive to reliably forecast the void fraction in various flow patterns and geometries, providing useful data for the design and optimization of many industrial processes involving two-phase flow.

2.1.2 Pressure gradient

The “pressure gradient” or sometimes referred as “total pressure gradient” describes the total two-phase flow pressure drop with distance along the pipeline and is mathematically defined as the sum of three components (Equation 2.2):

$$\left(\frac{dP_{tp}}{dL} \right) = \left(\frac{dP_f}{dL} + \frac{dP_h}{dL} + \frac{dP_a}{dL} \right) \quad \text{Equation 2.2}$$

where dp_f/dx is the frictional pressure gradient, dp_h/dy is the gravitational pressure gradient, and dp_a/dz is the acceleration pressure gradient.

The frictional pressure drop is caused by irreversible shear work at the pipe wall and the gas-liquid interface. The frictional pressure gradient in two phase flow is far more challenging to determine than in single phase flow. This is because it is affected by a variety of flow parameters, including pipe diameter, mass flux, pipe orientation, pipe surface roughness, fluid properties and the interfacial contact area between the phases. Since both phases are present in the pipe simultaneously, the cross-sectional area available for these phases to flow is consequently less. As a logical consequence of the friction between the phases, significant pressure drop is anticipated in two-phase flow.

The gravitational pressure gradient represents the pressure change due to the change in height along the pipe and is dependent on the orientation of the pipe (vertical or horizontal). The acceleration pressure gradient indicates the pressure change due to acceleration or deceleration of the fluid.

For adiabatic flows in both horizontal and vertical pipes (Al-Awadi, H. 2011), the acceleration pressure gradient typically constitutes less than 1% of the overall pressure gradient (ESDU, 2004). Consequently, the term dp_a/dz is disregarded and the evaluation focuses solely on the frictional and gravitational components (Ali, S. F. 2009).

To properly forecast the pressure gradient in a two-phase flow, both the frictional and gravitational pressure gradients must be considered, as well as any changes in fluid properties caused by the presence of two phases. This may be accomplished by employing models and correlations designed expressly for two-phase flow systems.

2.2 Two-phase flow

Multi-phase flow refers to the simultaneous flow of different phases. Multi-phase flow is widespread in various industries including the chemical, nuclear and geothermal industries. A two-phase flow of two immiscible fluids, such as oil/water (Al-Awadi, H. 2011) or gas/liquid develops in producing wells, related flow lines, separators, dehydration units, evaporators and other processing equipment in the oil industry (Al-Dogail, et al. 2021). Because of all these fields of occurrence of multiphase flow, it comprises an important area of study for engineers and scientists involved in the design, operation and optimization of production, transportation, and refining processes.

Two-phase flow is described as the simultaneous flow of two phases in a pipe, such as gas and liquid. A fluid compound mixture that flows through a pipeline can emerge as a

single-phase liquid, a single-phase gas, or a two-phase gas-liquid combination. The flow behavior of two-phase systems, on the other hand, is considerably more complicated than that of single-phase schemes due to variations in phase distribution in the pipeline; these flow features are defined as the flow pattern or flow regime. Numerous factors and forces impacting on fluids determine the sort of flow pattern that occurs at a specific place in the pipeline. These flow parameters include forces such as buoyancy, turbulence, inertia and surface tension (Ali, S. F. 2009), in addition to volume fluxes, pipe diameter, inclination angle and fluid properties including fluid density and viscosity (Al-Dogail, et al. 2021).

The ability to identify two-phase flow regimes is becoming necessary in oil and gas industry. An accurate evaluation of multiphase flow regimes will result in an accurate forecast of pipeline pressure drop. Furthermore, the mass, momentum and energy transfer rates and processes can be affected by the geometric distribution of the flow's components (Al-Dogail, et al. 2021). Moreover, flow regimes give essential information for several related flow issues, such as predicting erosion and corrosion, as well as predicting solids deposition and slug characteristics.

Nevertheless, there is currently no reliable technique for determining two-phase flow regimes. Two-phase flow regimes are often set based on subjective opinion of the researcher (Lowe and Rezkallah 1999). Due to relative movement and varying interfaces between the two phases, two-phase flow is, on the one hand, more complicated than single-phase flow. On the other hand, there is no reliable means for gathering precise data that can represent the real flow regimes.

Nowadays there are two different types of methods being utilized to determine two-phase flow regimes:

- **Direct methods:** This type of approach involves the direct observation of the flow patterns to determine the flow regime. Examples of direct methods include eyeballing, high-speed photography and radial attenuation. These methods provide visual information about the flow and can be useful for gaining qualitative understanding of the flow regime, but their accuracy can be limited by the resolution of the equipment and the subjectivity of the observer's interpretation.
- **Indirect methods:** These methods analyze recorded signals statistically in order to determine the flow regimes by reflecting the fluctuating characteristics of two-phase flows. To determine the flow regime, measurements and analyses may be conducted on variables including pressure drop and void fraction. These methods offer quantitative

data regarding the flow, but they may have limitations due to the precision of the measuring tools and the complexity of the data analysis.

To achieve accurate and trustworthy findings, it is critical to have a comprehensive grasp of the underlying physical principles and to employ proper measurement and analysis procedures in both direct and indirect approaches. The identification of two-phase flow regimes is an ongoing research topic, with new approaches and procedures being developed to increase the precision and validity of flow pattern identifying.

2.2.1 Two-phase flow patterns in vertical pipes

Diverse interpretations of flow behavior visualization by various scientists have resulted in a lack of unanimity in the description and classification of flow regimes, including those in small diameter pipes (Shoham, 2006; Pagan et al. 2017). Because of the highly complex nature of its interface, modeling of two-phase flows has always been limited to certain cases hence, the two-phase flows in upward vertical and near-vertical pipes were generally classified into four basic patterns: bubbly, slug, churn and annular flow.

When the flow conditions change, these regimes emerge. Bubbly flow emerges in examining vertical pipe flows while maintaining a constant liquid flow rate and increasing the gas flow rate. Bubble coalescence produces bigger bubbles as gas volume flow rates increases. The flow pattern is demonstrated to transition from bubble flow at low gas flow rates, to slug flow, then churn flow and eventually annular flow at higher gas rates (Montoya et al. 2016).

A suitable starting point for describing flow regime transitions is single phase liquid flow. Bubble flow emerges when liquid is pumped into a vertical pipe while gas is concurrently injected at the pipe's bottom (Perez, 2007). The pattern in the pipe changes from bubble to slug when the gas flow rate increases while the liquid flow rate remains constant. This is related to the coalescence of tiny bubbles. When the gas flow rate is increased further, the regime changes to churn and then to annular flow. Despite substantial efforts, robust and precise modeling techniques for such situations remain lacking.

In two-phase flow, the flow regime pertains to the pipe's design and structure (inclined, vertical, or horizontal pipeline), the local flow state and the fluid parameters of each phase. The fundamental challenge in analyzing two-phase flow is determining its continuity, momentum and energy, in addition to its reactivity to pipeline's geometry.

The intricacy of two-phase flows becomes more pronounced as the void fraction rises. This is due to opposing forces that cause coalescence of tiny bubbles into bigger ones, while

simultaneously causing the disintegration of the larger structures (Montoya et al. 2016). Moreover, as the liquid phase becomes more turbulent, the increase in bubble size leads to a greater complexity level at the interface. The greater deformability poses significant challenges in the theoretical modeling of large bubbles (Montoya et al. 2016). The difficulties in modeling high void fraction patterns are exacerbated by the restricted data available from experimental sources. While local and transient data for small and large bubble patterns has been collected (Prasser et al., 2007; Lucas et al., 2005; Montoya et al. 2016), churn-like bubbles are exceedingly unstable and hence not amenable to independent investigation.

The two-phase flow patterns observed in vertical pipes are described as follows:

2.2.1.1 Bubbly flow

Gas bubbles travel dispersed in a complicated motion via a dominant liquid flow (Perez, 2007). The vapor phase is evenly disseminated as bubbles of varying size and shape in the continuous liquid phase. They are, however, substantially smaller than the pipe diameter and in general, spherical. Unless at very low mass velocities, when buoyancy effects become obvious, the ratio of phase velocities in the bubble pattern is close to one and no significant slippage can be detected. This flow regime appears near zero vapor quality and is typically present prior to the fluid reaching a state of saturation, which is characterized by subcooled boiling (Schmid et al. 2022). Bubble flow regimes may be further subdivided into bubbly and dispersed-bubble flow. The first one is characterized by a lower number of larger bubbles traveling at a higher velocity than the liquid phase, whilst subsequent one is characterized by multiple smaller bubbles travelling at an equivalent velocity to the liquid phase (Brill and Mukherjee, 1999; Pagan et al. 2017).

2.2.1.2 Slug flow

As the void fraction increases, the bubbles from the bubble flow regime tend to coalesce, generating bigger bubbles approximately the size of the inner pipe diameter. These bubbles exhibit a bullet-like shape with a hemispherical front and a fluctuating tail in the direction of the flow. They maintain axial symmetry. (Schmid et al. 2022). They are usually referred to as "Taylor-Bubbles". A thin liquid layer separates the Taylor bubbles from the pipe wall (Pagan et al. 2017). The liquid film moves downwards, whereas the Taylor bubbles flow upwards. Liquid slugs carrying tiny bubbles bridge the pipe perimeters and create separation among the Taylor-Bubbles. Buoyancy influences the velocity of the phases and the net average velocity of the vapor phase is often greater than that of the liquid phase in the upward direction

(Schmid et al. 2022). As the void fraction grows, the Taylor-Bubbles become longer and liquid slugs emerge at lower frequencies. According to recent research, this pattern does not occur in larger diameter pipes; rather, a straight transition from bubble to churn flow occurs (Cheng et al. 1998; Perez, 2007).

2.2.1.3 Churn flow

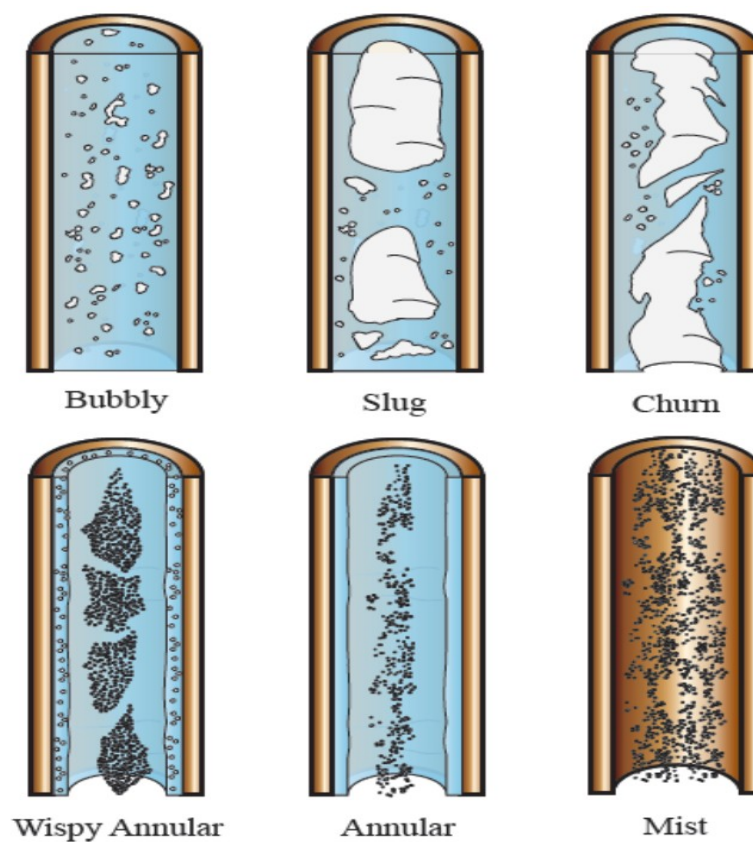
Many researchers consider churn flow to be the least comprehended flow pattern in vertical upward gas-liquid flows within pipes. Nonetheless, this flow pattern is widespread in various applications in the petroleum industry, including gas-lift operations, gas-condensate well production and two-phase flows with a high gas-liquid ratio in general. Furthermore, regardless of flow regime, the precision of two-phase flow modeling for larger pipe diameters remains questionable. Only a limited number of studies in publicly available literature validate two-phase flow models for pipe diameters exceeding 0.203 m (8 in.) (Pagan et al. 2017).

This regime is formed when the Taylor bubbles exhibit an erratic motion. It is an extremely chaotic flow pattern that involves the constant break up and dispersion of liquid into gas and gas into liquid (Perez, 2007). Prasser (2003) offered a nice illustration of an unstable "slug" flow for larger pipe diameters. Ohnuki and Akimoto (2000) offered various definitions for large diameter pipes (Perez, 2007). This flow pattern is distinguished by a chaotic mix of phases. It begins to form when the liquid slugs become unstable and start to collapse (Schmid et al. 2022). The liquid's local flow direction may fluctuate between upflow and downflow, with a net flow in the upward direction. The highest void fraction in the liquid slugs is achieved during the shift from slug to churn and the turbulent wake behind the preceding Taylor-Bubble begins to impact the nose of the subsequent one. The transition conditions to churn flow are determined by liquid slugs that are not retained (Schmid et al. 2022). As a result, their frequency is reduced to zero.

The churn flow regime is characterized by chaotic flow. Because of its intricacy, several researchers have stated that its physics is extremely difficult to understand and characterize experimentally (Ansari et al., 1994; Shoham, 2006; Pagan et al. 2017). The churn flow pattern has been observed between the slug and annular flow regimes, albeit at really high liquid flow rates, a direct transition from slug or even bubbly flow directly to annular flow may occur (Jayanti and Hewitt, 1992). Because of its similarities to annular flow, some scientists do not consider churn flow as a distinct flow regime. Many other scientists consider that this pattern is just a transition one. For a sufficiently long pipes, Taitel et al. (1980) and

Dukler & Taitel (1986) define churn flow pattern as an entry phenomenon that precedes the stable slug flow downstream (Pagan et al. 2017). Waltrich et al. (2013), on the other hand, recently claimed that the churn flow pattern exists and should be recognized as a distinct pattern. Their conclusion was drawn from experimental observations, including video recordings and liquid holdup measurements (Pagan et al. 2017) taken while utilizing a vertical tube system 42m long and 0.048m in diameter.

Figure 2.1 Two-phase flow patterns in vertical upflow Thome, J. R., & Cioncolini, A. (2016). Two-phase flow pattern maps for microchannels. in encyclopedia of two-phase heat transfer and flow I: Fundamentals and Methods (Pp. 47-84).



2.2.1.4 Annular flow

Annular flow in a vertical pipe is defined by a high-speed gas core including entrained liquid droplets (Pagan et al. 2017). A thin liquid film surrounds the pipe wall in this flow pattern, which may include entrained gas bubbles. Annular flow is characterized by the continuous vapor phase in the pipe's core, while the liquid phase is pushed from the center to the pipe's perimeter. The significant proportion of the liquid phase flows as an annulus at the inner pipe wall, resulting to the flow regime's name and remaining part of it as droplets

entrained in the vapor core (Schmid et al. 2022). The vapor phase's velocity is often significantly greater than the liquid annulus'.

The liquid film in vertical annular flow is thin and generally homogeneous across the inner pipe perimeter. The drag of the high-velocity gas core causes waves to develop on the liquid film. As the shear from the gas core against the liquid film breaks off the tip of the waves, it leads to the generation of entrained droplets within the gas core (Pagan et al. 2017). As the entrained droplets travel randomly in the gas core and are transported up by the gas stream, they will every once in a while, re-deposit in the liquid film downstream from where they were produced (Hewitt and Hall-Taylor, 1970). The interface within phases is frequently rough because of the generation of waves. The dispersed phase can transfer almost all of the liquid at some very high gas flow rates. The shear forces of the vapor phase core elevate the liquid film in vertical upflow. At low mass velocities, however, gravitational forces can dominate over shear, resulting in the liquid film flowing in the opposite direction (Schmid et al. 2022).

During the change from churn to annular flow, the liquid phase is no longer bridging the pipe perimeter, allowing the vapor in the pipe core to flow freely without being disturbed by the liquid. However, the transition from churn to annular flow lacks well-defined criteria. As a result, it is more of a continuous transition from a chaotic and turbulent condition to a steadier flow. In addition, oscillating between the two flow patterns may occur during the transition phase, when sequences from one pattern alternate with sequences of the other (Schmid et al. 2022).

2.2.1.5 Mist flow

Mist flow is a type of two-phase flow that arises when gas or vapor carries tiny droplets within it. These droplets are often too small to be observed without magnification or illumination. The onset of mist flow typically occurs at high vapor qualities when the shear forces from the gas core on the interface cause the annular film to become unstable (Pagan et al. 2017). As a result, the annular film becomes thinner until all of the liquid in the film is carried as droplets within the continuous gas phase. Another situation where mist flow may occur is during flow boiling when the critical heat flux is exceeded (Thome & Cioncolini, 2016). When the liquid droplets in the mist flow come into contact with the tube wall, they only intermittently wet it locally. Consequently, heat transfer is significantly reduced compared to annular flow because the droplets do not form a continuous film on the wall, which is necessary for efficient heat transfer.

2.2.2 Two-phase flow patterns in horizontal pipes

A precise forecast of pressure loss in pipelines may be achieved by accurately estimating multiphase flow regimes. They are susceptible to the rates and processes of mass, momentum, and energy transmission. Flow patterns also give information for various connected flow issues, such as erosion and corrosion prediction, in addition to solids deposition and slug characteristics prediction, as mentioned before.

Another significant consideration in two-phase flows pertains to horizontal pipelines. In fact, two-phase flow patterns in horizontal pipes closely resemble those in vertical flows, differing primarily in terms of flow stratification effects. In a horizontal two-phase flow, buoyancy forces lead to the separation of liquid and vapor, causing the liquid to accumulate at the bottom of the pipeline and the gas to rise towards the top. However, when the flow rates are sufficiently high, the impact of buoyancy becomes insignificant and the flows closely resemble those found in vertical configurations (Thome & Cioncolini, 2016).

Numerous researchers have established classifications for flow patterns to interpret the various flow arrangements leading to a wide range of flow pattern names (Al-Dogail, et al. 2021).

In horizontal flow regimes, where gravity acts perpendicular to the flow direction, there is a potential for flow separation to transpire. The various flow patterns include stratified flow, where gravitational separation is fully realized; stratified-wavy flow; bubble flow, characterized by the dispersion of bubbles within the liquid continuum; annular dispersed flow, akin to that in vertical flow but with asymmetry in film thickness caused by gravity; and various intermittent flow patterns. The intermittent flow category encompasses plug flow, where large bubbles flow near the top of the pipe; semi-slug flow, characterized by the presence of very large waves on the stratified layer; and slug flow, where these waves reach the top of the pipe and create a liquid slug that swiftly moves along the channel (Perez, 2007).

Predicting flow patterns is often a requirement and the conventional approach involves creating a flow regime map based on certain primary variables, such as the superficial velocity of the phases, mass flux and quality. However, substantial efforts have been devoted to developing generalized plots that can be utilized across various channel configurations and fluid characteristics (Perez, 2007). Several factors and forces exert influence on the fluids, leading to the emergence of specific flow regimes within a particular section of the pipeline. The inherent instability of these flow patterns contributes to the shifting boundaries of flow

regime regions on a flow regime map, subsequently leading to transitions between different flow patterns (Al-Dogail, et al. 2021). The two-phase flow patterns observed in horizontal pipes are described as follows:

2.2.2.1 Bubble flow

When the liquid flow rate in a pipeline is high, the bubble flow regime develops. Due to buoyancy force, many dispersed gas bubbles create a layer on the upper half of the pipeline, above the continuous liquid phase, in this regime (Al-Awadi, H. 2011). The bubbly flow regime is mostly seen in horizontal flows at high mass flow rates because at lower flow rates the numerous bubbles in the upper half swiftly amalgamate into a few large, elongated bubbles and it can quickly convert into plug flow (intermittent regime) (Thome & Cioncolini, 2016). In relation to the pipe diameter, the bubbles are rather minute. Shear forces dominate at high liquid velocities and bubbles are evenly distributed throughout the pipe. This flow regime may represent a different category. As the vapor velocity increases, the number of bubbles tends to increase as well to fill the whole pipe cross section.

2.2.2.2 Stratified flow

This flow is gravity dominated and appears at low gas and liquid flow rates when the two phases separate completely: the gas flows along the upper section of the pipe and the liquid travels along the lower section of the pipe with no notable interfacial waves (Thome & Cioncolini, 2016). The interface between the two phases is horizontal and flat. Typically, over half of the pipe's perimeter remains dry, leaving its remaining wet. This flow pattern has the simplest form of all horizontal flow regimes and is also described as the "stratified smooth" flow pattern. The dispersed phase creates a layer on top of the pipe, while the continuous phase lies underneath it. It can also be characterized by the coalescence of gas plugs to form a continuous gas flow along the top of the pipe, exhibiting a smooth gas-liquid interface characteristic of stratified flow at low flow rates (Ali, S. F. 2009).

2.2.2.3 Wavy stratified flow

The dispersed phase creates a layer at the top of the pipe in stratified wavy flow, while the continuous phase flows below. The interface between the two phases is wavy and moves in the flow direction. This happens as the gas velocity rises and generates interfacial shear on the liquid surface, leading to the generation of waves that travel in the direction of flow and wrap around the pipe's perimeter towards the top. The amplitude of the waves is noticeable

and it increases proportionately to the rate of gas flow. Their crests, however, do not contact the upper surface of the inner pipe. Although sometimes large-amplitude waves may wet it, a notable portion of the upper perimeter of the pipe remains dry (Thome & Cioncolini, 2016). The gas-liquid interface is seldom smooth and the waves generate ripples in the liquid surface (Al-Awadi, H. 2011).

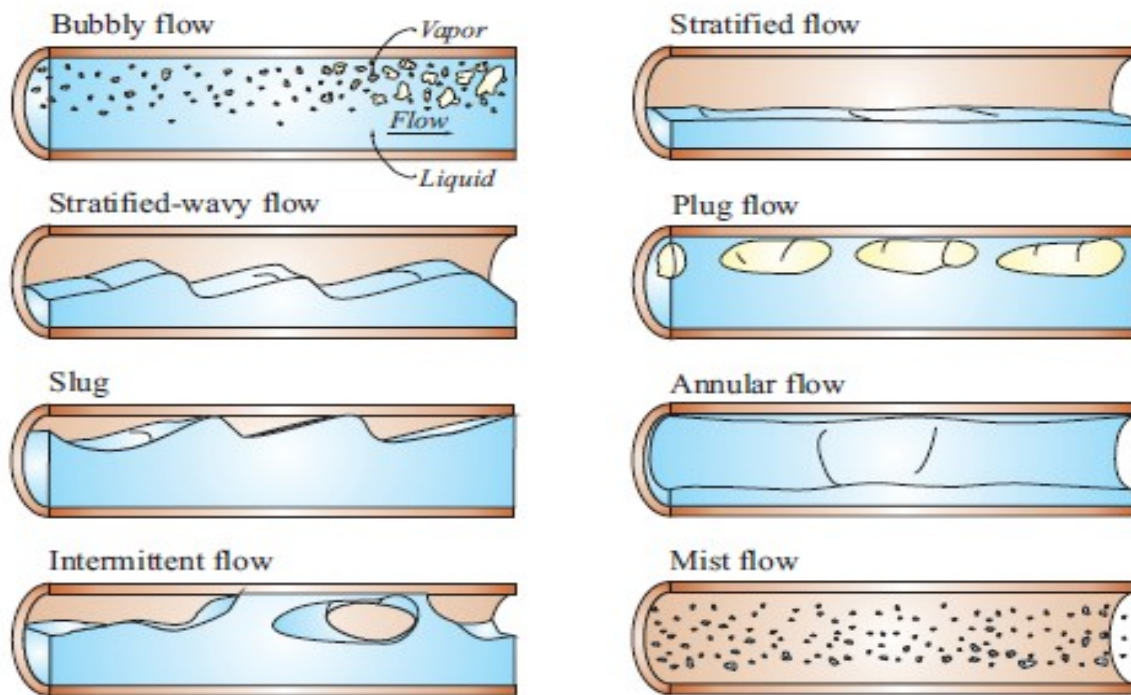


Figure 2.2 Two-phase flow patterns in horizontal flow (from Silva Lima, 2011). Thome, J. R., & Cioncolini, A. (2016). Two-phase flow pattern maps for microchannels. in encyclopedia of two-phase heat transfer and flow I: Fundamentals and Methods (Pp. 47-84).

2.2.2.4 Intermittent flow

With the increase in gas velocity, the interfacial waves attain sufficient height to consistently wet the upper perimeter, frequently leaving a liquid film in their wake. As a result, nearly the entire pipe's perimeter remains continuously wet. The high-amplitude waves are interspersed with lower-amplitude waves that do not attain the top of the channel. Due to the rolling motion of these waves, entrained bubbles are frequently observed within the larger waves. The liquid slugs that separate such big, elongated bubbles might alternatively be regarded as large periodic amplitude waves, giving rise to the regime's name: intermittent flow. Intermittent flow is occasionally further subdivided into two distinct patterns: plug flow and slug flow (Thome & Cioncolini, 2016):

2.2.2.4.1 Plug flow

This flow pattern is characterized by a continuous liquid flow with large, bullet-shaped, elongated gas bubbles (Thome & Cioncolini, 2016) and liquid plugs traveling along the pipeline's top. Many bubbles coalesce to generate intermittent gas plugs. The diameter of the elongated bubbles is smaller than the diameter of the pipe, allowing a continuous liquid phase to emerge on the pipe's bottom. This flow regime is also known as "elongated bubble" flow pattern. Collisions between the discrete bubbles happen more often at increased gas flow rate and they coalesce into elongated "plugs" (Ali, S. F. 2009). This is the reason it is termed plug flow.

2.2.2.4.2 Slug flow

The dispersed phase in this flow pattern is composed of big bubbles separated by slugs of the continuous phase, resulting in a periodic change in the cross-sectional area of the pipe (Perez, 2007). Large amplitude waves or splashes of liquid eventually build up and reach the upper wall of the pipe, forming liquid slugs. The slugs alternately travel with the dispersed phase, while the liquid slugs are carried by the faster-moving gas flow. In contrast to the plug flow pattern, wherein elongated bubbles are carried by the liquid phase, the fast-moving liquid slugs in the slug flow regime are frequently accompanied with sudden pressure pulses and severe pressure fluctuations that can cause damage to downstream components, making it very undesirable in practical applications. With an increase in gas flow rate, the diameters of the elongated bubbles get closer to that of the pipeline and the flow pattern starts to resemble slug flow in vertical pipelines. The liquid film between the bubble's top and the pipeline is (Thome & Cioncolini, 2016) often thinner in comparison to the liquid film underneath the bubble. The flow enters the slug flow regime when the amplitude of the waves flowing along the liquid surface becomes significant enough to span the top of the pipe (Al-Awadi, H. 2011).

2.2.2.5 Annular flow

When the gas flow rate is high enough to support the liquid film around the pipe walls (Ali, S. F. 2009), this flow is momentum-dominated and occurs at relatively high gas flow rates. When the gas flow rate is increased, the liquid forms a continuous annular film around the perimeter of the pipe wall and the gas flows as a central core, entrapping tiny droplets of liquid dispersed as mist. Small gas bubbles may be entrained in the liquid film as well. The

thickness of the liquid film is greater towards the bottom of the pipe due to gravity until the annular liquid ring achieves a uniform thickness when even higher gas flow rates prevail. Waves may disrupt the interface between the liquid annulus and the vapor core and droplets may be disseminated in the gas core. At high gas fractions, the top of the tube with its thinner film dries first, leaving the annular film to cover just the bottom pipe perimeter, resulting in a stratified-wavy flow (Thome & Cioncolini, 2016). This flow regime is linked with a substantially greater heat transfer coefficient in flow boiling or condensation.

2.2.3 Flow regime maps

The preceding chapter detailed typically observed flow regimes in vertical and horizontal gas-liquid flow. Organizing these frequently disorderly processes poses a considerable challenge. Usually, flow patterns, identified by analyzing trends in both quantitative and qualitative data, are systematically arranged (Perez, 2007). Mapping is a prominent method for analyzing distinctive flow regimes depending on different flow parameters. Within a spectrum of flow conditions, delineating the boundaries wherein a specific pattern prevails becomes feasible. Graphical classification of flow pattern maps is achievable for both vertical and horizontal multiphase flows. Upon mapping all observed pattern boundaries, a comprehensive understanding of behavior relative to flow conditions emerges (Perez, 2007).

A flow pattern map is a two-dimensional map depicting flow regime transition boundaries (Ali, S. F. 2009). Predicting regimes is frequently required and the standard technique is to plot the information in terms of a flow regime map. The selection of proper coordinates to represent the various flow regimes clearly and efficiently has long been a research topic. In practice, dimensional coordinates such as superficial velocities are commonly utilized as the one proposed by Taitel et al. (1976). Numerous maps are expressed in terms of fundamental variables, such as the superficial velocity of each phase, mass flux and quality. However, substantial efforts have been invested in generalizing these plots to enable their adaptation across a diverse spectrum of pipe geometries and fluid physical properties (Perez, 2007).

There are two different types of flow regime maps: empirical flow maps, that are typically fitted to data bases of observed flow patterns and theoretical or semi-theoretical flow maps, in which transitions are predicted from physical models of the flow phenomena (Montoya et al. 2016). Theoretical or semi-theoretical flow-pattern maps are designed with the flow structure in consideration and they are often related to heat transfer processes and

adiabatic characteristics. Typically, just two parameters, such as the superficial gas and liquid velocities, are utilized to construct a coordinate system on which the boundaries between distinct flow patterns are plotted. Similar to a conventional map, transition boundaries are presented to separate the various flow regimes.

Two-phase flow, that refers to the simultaneous flow of a liquid and a gas in a pipe, is a complicated process regulated by a number of competing forces including shear stress, capillarity, gravity and momentum. This contributes to making predicting two-phase flow behavior challenging. Flow-pattern maps or flow-regime maps are commonly employed to simplify the visualization of this complicated flow behavior. The relative magnitudes of the forces acting on the fluids are used to categorize the flow into different regimes (Al-Awadi, H. 2011).

These maps, however, have certain drawbacks. Flow regime maps have limitations considering that they are frequently dimensional and only applicable to certain pipe diameters and fluids. Furthermore, experimental data is necessary to construct these maps, yet evaluating the influence of each parameter on the flow regime map is not always possible. Attempts to establish generalized coordinates for flow maps have also been hampered by instabilities in the transition lines between flow regimes, which are regulated by distinct sets of fluid properties. Moreover, the impact of fluid parameters such as density, viscosity and surface tension on flow patterns and pressure drop in pipelines remains unclear.

Because of the non-axisymmetric structure of gravity-induced flow, horizontal two-phase flow is more challenging to analyze than vertical two-phase flow. Baker's flow map, which is frequently quoted for adiabatic gas-liquid flow, fails to account for the influence of pipe diameter since the majority of the data utilized was for a fixed pipe size.

Kosterin developed the first flow regime maps in 1949 and since then, other researchers have presented several flow regime maps based on experimental data acquired from various references. Baker's flow regime map is one of the most often utilized (Al-Awadi, H. 2011) flow regime maps, because it relies on experimental data gathered from several published references (Al-Dogail, et al. 2021).

Taitel and Dukler (1976) developed a widely used theoretical, mechanistic flow regime map, albeit with alterations to the computation of the interfacial friction factor (Perez, 2007). Taitel-Dukler's map is premised on a simplified theoretical model and several of their assumptions have been criticized by other scientists. The flow pattern map of Taitel et al. (1980) (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής) is the most frequently used for vertical flow. However, for vertical downward flow, different flow patterns have been devised, such

as those of Bilicki and Kestin (1987) and Barnea et al. (1982) (Perez, 2007). Mandhane's flow map (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής, Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής), on the other hand, is commonly accepted for horizontal flows since it is developed utilizing a broad range of parameter values.

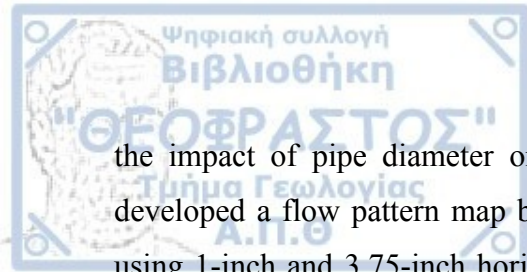
Flow regime maps, despite their limitations, are a useful tool for predicting the flow pattern and pressure drop in two-phase flow in pipes. They give a visual representation of the complicated behavior of two-phase flow, which can be difficult to comprehend based only on mathematical models. Engineers and scientists may better design and improve pipelines and other industrial processes requiring two-phase flow by employing flow regime maps. However, interpreting two-phase flow patterns is typically subjective and there is no precise quantitative approach for describing and categorizing the flow patterns.

A combination of parameters and dimensionless numbers were (Al-Dogail, et al. 2021) employed to broaden the applicability of the flow regime maps. A large amount of experimental data is necessary to construct these maps and it is typically not possible to examine the effect of each parameter (density, viscosity, surface tension, pipe size, or geometry) on the flow regime map. The presence of interfaces between phases and discontinuities in associated properties defines multi-phase flow. Surface tension is used to describe the presence of interphase and its impact was not evaluated for the air-water system, despite the fact that the air-water system is used in the majority of the research published in the literature to create the experimental data necessary for the flow regime map.

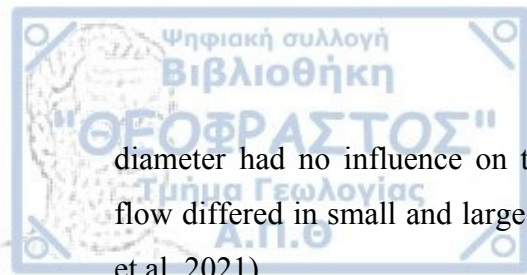
The flow regime maps provided in the literature are diverse since they were developed employing different parameters such as pipe diameter, fluid characteristics and flow rates. Bergelin and Gazley, for instance, introduced a flow pattern map for a two-phase flow system consisting of air and water flowing through a 1-inch diameter pipe, whereas Alves presented a flow pattern map for various fluid mixtures, some of which consisted of air-water and others of air-oil, also on a 1-inch diameter pipe (Al-Dogail, et al. 2021).

Mandhane et al. designed a flow pattern map for two-phase gas-liquid flow in horizontal (Al-Awadi, H. 2011) pipes utilizing around 6000 observations. The flow regime map's mapping parameters were liquid and gas superficial velocities (Perez, 2007). Their conclusion indicated that the transition boundaries of flow patterns are minimally influenced by fluid properties and pipe diameter when mapping parameters such as superficial velocities of liquid and gas are employed. (Al-Dogail, et al. 2021).

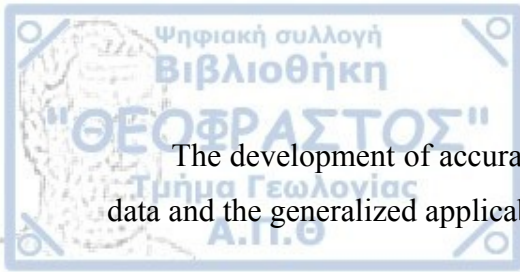
Other researchers have created flow regime maps based on experimental data collected at atmospheric pressure with various horizontal pipe diameters. Lin and Hanratty examined



the impact of pipe diameter on flow patterns for air-water flow in horizontal pipes and developed a flow pattern map based on experimental data recorded at atmospheric pressure using 1-inch and 3.75-inch horizontal pipes. They reported that at high gas flow rates, pipe



diameter had no influence on transition lines, and the transition from stratified to annular flow differed in small and large diameters owing to the effect of wave velocities (Al-Dogail, et al. 2021).



The development of accurate flow regime maps requires a large amount of experimental data and the generalized applicability of these maps remains a topic of ongoing research.

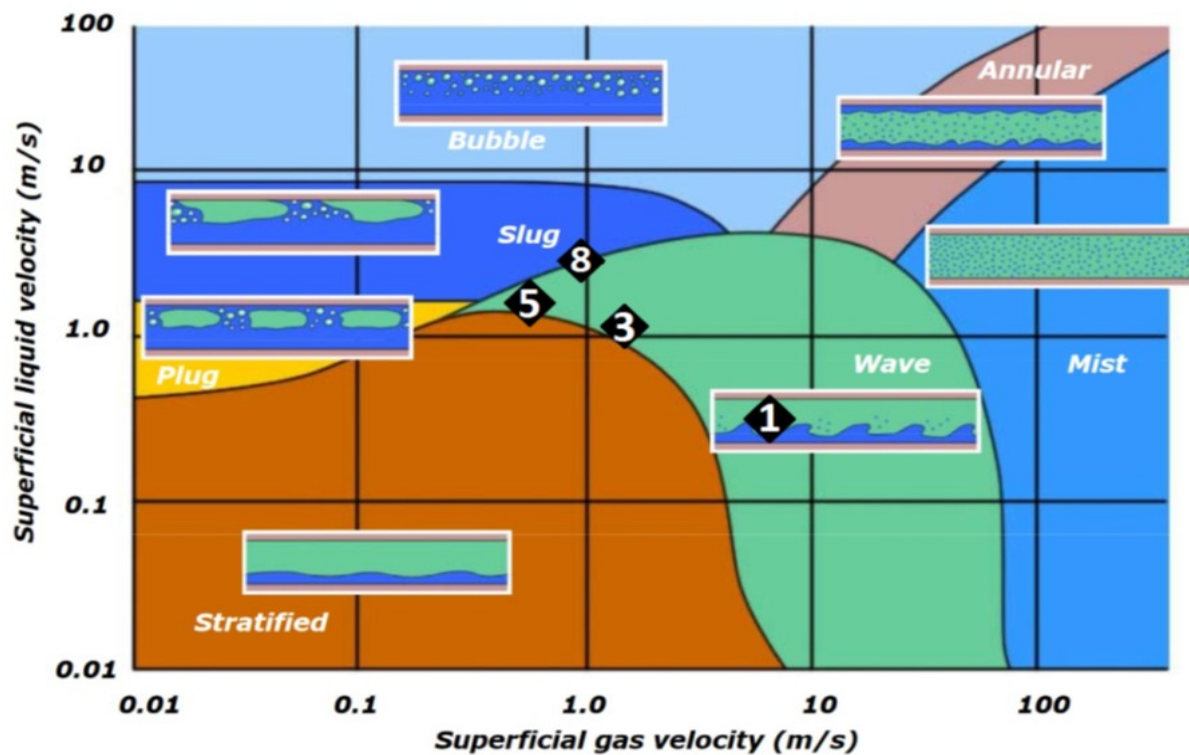
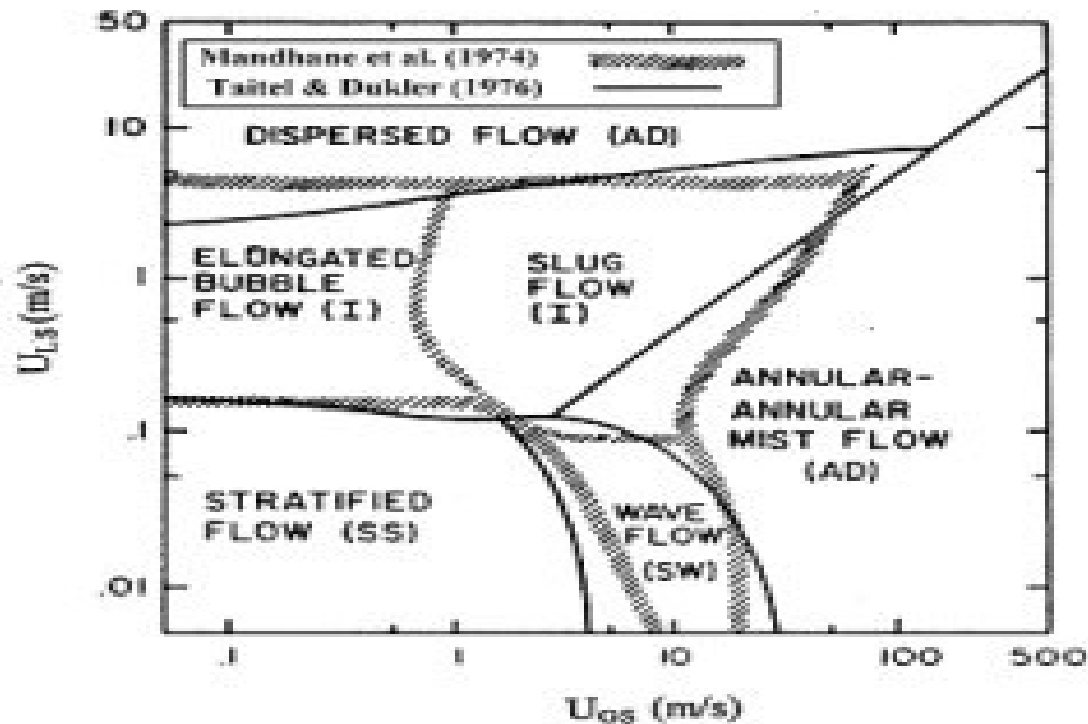


Figure 2.3 Two-phase horizontal flow map. Knotek, S., Fiebach, A., & Schmelter, S. (2016, September). Numerical simulation of multiphase flows in large horizontal pipes. In Proceedings of the 17th International Flow Measurement Conference, FLOMEKO.

Superficial gas velocity j_g (m/s)
Superficial Gas Velocity, v_{sg} , ft/sec

Figure 2.5 Flow regimes map for vertical upflow showing Taitel et al. (1980) and Mishima and Ishii (1984) transitions (Mishima and Ishii, 1984). Ali, S. F. (2009). Two-phase flow in a large diameter vertical riser (Doctoral dissertation, Cranfield University)

2.3 Two-phase flow pattern transitions and mechanisms

Changes in the type of flow regime which emerge in a pipe are referred to as flow pattern transitions. Changes in flow rate, fluid characteristics, or pipe geometry can induce these transitions, indicating that the behavior of gas-liquid flows is significantly influenced by both the physical properties and the geometrical features of the pipeline (Azzopardi, 2006; Brennen and Brennen, 2005; Crowe, 2005; Shen et al., 2014; Mohammed et al. 2019). Because two-phase flow behavior is very complicated and nonlinear, understanding the underlying mechanisms of flow pattern transitions constitutes a significant challenge, but it is also essential for forecasting and controlling multiphase flow in pipelines. Models and mechanisms described for small geometries are expected to exhibit substantial differences when applied to larger geometries and other physical attributes. It would be intriguing to examine how regime transition mechanisms, which are typically created for air-water systems in small diameter pipes, might convert in a larger diameter pipe at very high viscosity (Mohammed et al. 2019).

Despite controlled systems with simple geometries and regulated flow rates, the intricate nature of the transitions makes it challenging to fully understand the underlying physical phenomenon. The variations in local void and interfacial area distributions within a two-phase flow differ among patterns, highlighting the significance of momentum transfer between phases in controlling flow behavior. The strong dependence of the regime on flow characteristics underscores the need for understanding not only the behavior of the regime but also its emergence within the system. Theoretical studies of two-phase flow regime transitions emphasize on establishing correlations by modeling the dominant physical variables that induce regime transition. Mishima and Ishii (1984) proposed a hypothesis stating that the transition from bubbly to slug flow occurs at a void fraction of 30%. This transition point represents the threshold at which bubbly flow is transformed into large bubbles through coalescence. Additionally, it is theorized that the transition from slug to churn flow occurs when the overall void fraction surpasses the void fraction found merely in the bubble region. This criterion is equivalent to selecting the moment when the liquid slug becomes sufficiently aerated to the extent that it cannot be distinguished separately. In this condition, the void is distributed relatively evenly throughout the pipe and there are no longer distinct liquid slugs. Ultimately, the transition from churn to annular flow was hypothesized to take place when flow direction reverses in the liquid film section along elongated bubbles or when liquid slugs and large waves are eliminated due to entrainment or deformation (Zimmer & Bolotnov 2019).

The emergence of instabilities in the flow is one of the most typical mechanisms behind flow regime transitions. A variety of factors, including fluid viscosity, surface tension, gravity, and inertia, can generate these instabilities. The instabilities can become more severe when the flow rate or other parameters change, resulting in a transition to a different flow regime. The interaction of different phases in the flow is another mechanism that might induce flow pattern transitions. The existence of gas bubbles in a liquid flow, for instance, might create turbulence and result in a transition to slug or churn flow. Correspondingly, bubble or droplet coalescence can create a transition from dispersed to annular flow (Al-Awadi, H. 2011). Flow pattern transitions can also be influenced by the geometry of the pipe. Changes in the diameter or orientation of the pipe, for instance, might influence the flow regime by changing the relative significance of various forces and instabilities.

Researchers frequently employ flow visualization techniques such as high-speed video imaging and laser-based approaches to examine flow pattern transitions in multiphase flow. Researchers may determine the mechanisms underlying flow pattern transitions and establish models to predict and control the behavior of multiphase flow in different conditions by examining flow patterns and their behavior. To handle the challenging issue of comprehending two-phase flow regimes, cutting-edge experimental facilities have been established and different measuring techniques have been applied according to the research objectives. When the primary concern is the identification of the two-phase flow pattern, high-speed cameras have been employed to monitor the flow and categorize the pattern. In other research, such as Wojtan et al. (2005), the facility could add extra steam to the flow via boiling to examine its impact on the transition behavior. Researchers can additionally determine the flow pattern by utilizing an impedance void meter, that records the flow's instantaneous electrical impedance (Tsoukalas et al., 1997; Schlegel et al., 2013; Zimmer & Bolotnov 2019). Each flow pattern's impedance signal is different, enabling identification without visual observation of the flow (Zimmer & Bolotnov 2019).

Understanding flow pattern transitions is a critical area of study with significant consequences for a broad spectrum of industrial use cases, including nuclear reactors, chemical processes and oil and gas production (Perez, 2007).

2.3.1 Transition from Bubbly to Slug flow

The transition from bubbly flow to slug flow occurs when the gas volume fraction or superficial gas velocity increases, or when the liquid viscosity or superficial liquid velocity decreases. As the void fraction or superficial gas velocity increases, the gas bubbles begin to

coalesce, resulting in the formation of larger bubbles (Montoya et al. 2016). These larger bubbles may start to detach from the liquid phase, leading to the formation of slugs.

The transition from bubbly flow to slug flow is determined by the interplay between bubble coalescence and break-up. This transition occurs when coalescence prevails, leading to the growth of bubbles into Taylor bubbles. A bubble becomes a slug flow Taylor bubble only when its equivalent diameter exceeds that of the pipe (Taitel et al., 1980; Mohammed et al. 2019). This indicates that the pipe diameter has a direct impact on the transition. On the other hand, liquid viscosity has a direct effect on bubble rise velocity due to drag, transverse movement of bubbles and bubble-induced turbulence; all these parameters have a significant influence on the coalescence and break-up of bubbles, which determine the bubble-slug transition. According to Taitel et al. (1980), the transition to slug flow occurs when the void fraction reaches 0.2-0.3 (Mohammed et al. 2019). This represents the maximum number of bubbles that can be accommodated within a cube without significantly increasing the rate of coalescence. Radovcich and Moissis (1962) proposed a model to predict the transition to slug flow by considering the possibility of successful bubble coalescence in relation to pipe diameter, void fraction and velocities. However, it is worth noting that their model did not account for the physical properties of the fluids (Mohammed et al. 2019).

The mechanism behind the transition from bubbly flow to slug flow is related to the dynamics of the gas-liquid interface. The detachment of bubbles from the liquid phase is driven by buoyancy, which can cause the bubbles to rise to the top of the channel. The detachment can also be driven by shear forces, which can cause the bubbles to break away from the liquid phase. Once the bubbles detach, they can coalesce and form larger (Al-Awadi, H. 2011) gas pockets or slugs. The transition from bubbly flow to slug flow can also be affected by the channel geometry and the liquid properties. For example, in a horizontal channel, the transition is more likely to occur at lower gas velocities, while in a vertical channel, the transition is more likely to occur at higher gas velocities. The physical analysis of the transition from bubbly to slug flow, as described by Hasan and Kabir (1988), adheres to an approach analogous to that employed for vertical systems. In the context of vertical systems, the transition from bubbly flow is typically observed when the void fraction surpasses 0.25 (Perez, 2007). In inclined pipe configurations, the gas phase predominantly flows along the upper wall. Consequently, in proximity to the upper wall, the in-situ void fraction may surpass 0.25, leading to a substantial rise in bubble collisions and the subsequent formation of Taylor bubbles. In such cases, the transition to slug flow takes place at a cross-sectional average void fraction below 0.25 (Perez, 2007).

The transition can also be affected by the surface tension between the gas and liquid phases, which can influence the stability of the gas bubbles. In slug flow, the liquid slugs can cause significant pressure fluctuations and can impact the performance of heat exchangers, reactors and other process equipment. Thus, understanding the mechanism behind the transition and the factors that can affect it is crucial in optimizing the design and operation of such systems (Ali, S. F. 2009).

2.3.2 Transition from Slug to churn flow

The transition from slug flow to churn flow happens when the liquid film that separates the gas slug from the pipe wall gets unstable and breaks up, causing the generation of smaller bubbles or droplets that are spread in the liquid phase. This mechanism can occur for a variety of reasons, including an increase in gas flow rate or a decrease in liquid flow rate, resulting in a decrease in liquid film thickness (Ali, S. F. 2009). As the liquid film breaks up, the bubbles or droplets coalesce to produce bigger gas pockets, resulting in a bubbly flow. The development of these gas pockets, as well as the break-up of the liquid film, increases turbulence and mixing between the gas and liquid phases, culminating in the establishment of a churn flow regime. Alteration in pipe geometry, for instance a quick increase in pipe diameter or the presence of an abrupt constriction in the pipe, can also cause the transition from slug to churn flow, resulting in an increase in the velocity of the gas phase and a decrease in the thickness of the liquid film.

Churn flow is recognized by the presence of a froth consisting of small bubbles and cap-sized bubbles, as well as a large, unstable, incoherent structure of bubbles that undergo frequent coalescence and breakup. The most distinguishing characteristic of churn flow is the intermittent downward motion of the liquid film when massive gas formations ascend the vertical column. Mohammed et.al 's work (2019), is recommended for a more in-depth research of churn flow, particularly at high viscosities and in large diameter pipes. One of the prominent transition mechanisms proposed by Mishima and Ishii (1984) and Chen and Brill (1997) suggests that the transition occurs when Taylor bubbles reach a sufficient length for the wake of the leading bubble to meet the nose of the following bubble. This interaction forces the slug to collapse and results in the formation of substantial liquid structures. This scenario is believed to manifest when the average void fraction within the liquid slug matches or surpasses that of the Taylor bubble region. This mechanism is commonly referred to as the "wake effect" mechanism. Nicklin et al. (1962) postulated a transition mechanism involving the flooding of the liquid film surrounding the Taylor bubble. Hewitt and Hall-Taylor (1970)

presented a comprehensive experimental method for identifying the transition point. Some researchers, including Taitel et al. (1980), proposed that churn flow constitutes an emerging flow pattern for slug flow, driven by the backflow of the liquid slug when two unstable Taylor bubbles merge and transform into slugs. Brauner and Barnea (1986) postulated an alternative mechanism centered around bubble coalescence (Mohammed et al. 2019).

2.3.3 Transition from Churn to Annular flow

The transition to annular flow begins with the formation of a continuous gas core in which the liquid mainly flows as waves and film on the pipe wall. Complete transition happens when the gas superficial velocity reaches a level at which it can lift the largest stable droplet within the gas core upwards (Mohammed et al. 2019). This transition is caused by the coalescence and breakup of liquid droplets in the churn flow area (Perez, 2007), resulting in a progressive increase in the thickness of the liquid film on the pipe wall. The liquid film becomes continuous when the liquid film thickness reaches a critical point and the churn flow pattern transitions to annular flow. Hinze (1955) hypothesized a transition mechanism focusing on the balance between drag and gravity forces within the gas core. Only a limited number of researchers have conducted studies on the dynamics of gas-very viscous liquid flows and even fewer have explored the nature of flow pattern transitions and expounded on the probable transition mechanisms (Mohammed et al. 2019).

The interfacial shear stress between the liquid film and the gas core also influences the transition from churn to annular flow. The interfacial shear stress increases as the liquid film thickness increases, which improves liquid film stability and accelerates the transition to annular flow. The fluid characteristics, such as surface tension and viscosity, along with the flow conditions, including gas and liquid superficial velocities and pipe diameter, all impact the transition from churn to annular flow. These parameters can impact the coalescence and breakage of liquid droplets, as well as the stability of the liquid film, and hence the transition from churn to annular flow.

2.4 Three-phase flow

Multiphase flow is quite commonplace in the oil and gas industry. Gas-liquid-liquid three-phase flow is particularly essential in offshore petroleum recovery well lines, which may contain a three-phase combination of crude oil, sea water (naturally occurring or injected to preserve reservoir pressure) and natural gas. The pressure drop and stability characteristics

of the flow in these lines, which are inextricably linked to the flow regimes that emerge, are critical to the proper functioning of offshore oil well platforms (Montoya et al.2016).

Water and hydrocarbon condensate are frequently found in gas pipelines and water and vapor are found in oil pipelines. The flow pattern and pressure gradient along the horizontal pipeline are the most essential characteristics of multiphase flow. It is crucial to investigate the flow rates, flow regimes, liquid-hold-up/water cut, pressure gradients, and volume fractions of gas, oil and water in pipelines during petroleum product transportation. The water cut describes the quantity of water at the pipe inlet expressed as a percentage of the total inlet volumetric flow rate. During the transportation of the multiphase flow, water in the system begins to separate from oil and accumulates at the pipe bottom, which is known as the local water contents, local water, or water hold-up. The water cut always constitutes one of the starting points regarding the designing of pipelines and equipment.

Because of densities differences, three-phase flow occurs in horizontal or near horizontal pipelines, with air flowing at the top, oil flowing in the center and water flowing at the bottom. In the latter stages of a well's operation, there is a substantial rise in water production, with water being co-produced alongside oil and natural gas. Pumping water or steam during secondary and tertiary recovery phases in oil fields is also standard practice, resulting in a three-phase combination of water, oil and natural gas. It is essential to understand the flow distribution across various flow conditions (Mukherjee et al. 2014) in order to design optimal transport pipelines.

Water, salts and carbon dioxide gas included in petroleum products are the primary causes of carbon steel pipeline corrosion in oil transportation and storage. When water is fully dispersed in oil at low water cut, corrosive water does not pose significant issues. Most oil wells operate with various water cuts, up to 90%, resulting in different flow patterns. Water droplets begin to coalesce as the water cut increases and oil and water phase separation occurs. Each phase wets different segments of the pipe. As the water phase comes into contact with the pipe wall, the potential of corrosion increases. It is therefore vital to comprehend the three-phase air-oil-water dynamics in production pipelines, as well as forecast flow patterns, pressure gradient and as a consequence, minimize pipe corrosion.

There have been multiple studies of two-phase flow patterns, but three-phase flow patterns have been given less emphasis. A deeper insight of this complicated flow phenomena is required due to the availability of three-phase flow applicability in the petroleum and chemical industries (Montoya et al. 2016). In the petroleum industry, precise investigation and prediction of flow configurations and pressure drop is required. A lot of research on the

characteristics of oil-water-gas, as well as the wetting characteristics of oil and water in the presence of gas, has been conducted. Ensuring the prolonged lifespan of the pipeline is critical, as the interaction between the pipe material and the fluid, with its impact on the protective wall layer, can lead to corrosion in the pipeline and its associated equipment (Mukherjee et al. 2014).

Aside from oil and gas recovery, concurrent gas-liquid-liquid flow takes place when two immiscible liquids flow together with one undergoing evaporation/condensation, or when two immiscible liquids react to release a gaseous component. All three constituents subsequently flow through the reactor in a simultaneous manner. Information about the existing flow pattern is required for every multiphase flow system in order to determine the hydrodynamics and heat and mass transfer properties of the mixture (Mukherjee et al. 2014).

The conventional methodology to modeling three-phase flow typically entails one of two methods: either simplifying the flow as a two-phase system, focusing on oil and gas while disregarding the water phase, or combining the oil and water phases into a single phase. In the latter case, the system is treated as a two-phase mixture comprising gas and a pseudo-single-phase liquid. (Tek,1961; Mukherjee et al. 2014). These techniques are constrained by specific flow conditions and cannot be applied across the entire operational spectrum. Following these studies, Galyamov and Karpushin (1971) investigated the separation of liquid phases, finding that the effective viscosity of the mixture was contingent upon the liquid volume fraction. Subsequently, Foreman and Woods (1975) conducted research that involved separating the gas phase from the liquids. Shean (1976) then formulated flow regime maps and employed drift-flux techniques to analyze the vertical three-phase flow within pipelines (Montoya et al. 2016).

Additionally, in the realm of research, Govier and Aziz (1972) published flow regime maps specifically addressing horizontal liquid/liquid flows; however, these maps did not include a gas phase. In a separate study, Bhaga and Weber (1972) employed one drift-flux theory to examine vertical gas/liquid/solid flows, focusing on average volumetric fractions. While they substantiated their analytical findings with experimental data, they did not provide flow regime maps as part of their research. Subsequently, Giot (1982) delved into research on vertical gas/liquid/solid flows and his models accurately predicted the volumetric fraction of the phases and the frictional pressure drop across various pipe diameters, ranging from 40 to 300 millimeters, as documented by Montoya et al. (2016).

2.4.1 Three-phase flow patterns in vertical pipes

The majority of the research has focused on horizontal pipes. Little is known about three-phase gas-liquid-liquid flows in vertical pipes. Woods et al. performed one of the early experiments on vertical simultaneous oil/water/air upflow in a 0.026m inner diameter (ID) pipe. They assessed the impact of input liquid and gas flow rates as well as the flow pattern on holdup. They presented correlations to identify the transition between regimes dominated by oil and water (Mukherjee et al. 2014). Descamps et al. examined the impact of gas injection on the phase inversion of vertical oil-water flow. Although the dispersion phase (oil-water) significantly affects bubble size, their findings indicated that gas injection has a minimal impact on the critical concentration of oil and water. Wang et al. utilized two signal types from a small conductance probe array and a vertical multielectrode array to identify oil-water-gas three-phase flow patterns in a vertical pipe. They generated six flow pattern maps corresponding to four different total mixture liquid flow rates. Their analysis revealed that an increased ratio of oil fraction in the liquid (f_o) led to the emergence of oil-in-water slug flow at lower superficial gas velocity. Furthermore, they observed that with higher superficial gas velocity, the phase inversion of liquids occurred at lower (f_o) values (Mukherjee et al. 2014).

Woods et al. conducted a comprehensive study involving vertical concurrent upflow of oil, water and air in a 0.026 m inner diameter (ID) pipe. Their investigation included an in-depth analysis of visual/video data, holdup measurements and pressure loss data. Additionally, they considered existing flow pattern descriptions provided by other researchers for both horizontal and vertical three-phase oil/water/gas flow. Through this analysis, they successfully identified nine distinct flow regimes. These flow regimes were further categorized into two types: oil-dominated (OD) and water-dominated (WD) (Woods et al. 1998; Mukherjee et al. 2014). The three-phase flow patterns observed in vertical pipes are described as follows:

2.4.1.1 Oil-Dominated Oil Annulus/Dispersed Annular

On the inside wall of the pipe, the oil created an annulus film, which was covered by an annular dispersion of water droplets in an oil continuum that interfaced with the gas core. Increasing gas and/or water flow induced a gradual thinning of the oil annulus until total water dispersion in oil was developed, for instance the OD dispersed annular regime. Such flow seemed opaque/white, posing issues with optical detection of detail. The OD dispersed annular regime might be semi-annular in nature, with surface waves in each (Woods et al. 1998). (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής – 1)

2.4.1.2 Oil-Dominated Broken Annulus

When the water rate neared the inversion point (when the system transitioned from OD to WD), the OD annulus next to the wall began to break down and was progressively replaced by a WD annulus. Small strips of the OD annulus delaminated from the pipe wall and were replaced by a WD phase before being covered by the OD dispersed film (Woods et al. 1998). The pipe wall appeared marbled, with a significant degree of liquid entrainment in the gas phase. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής – 2)

2.4.1.3 Oil-Dominated Dispersed Churn

The pattern resembled two-phase churn flow in which the liquid phase exhibited a vertical oscillatory dispersion of water droplets in oil (Woods et al. 1998). (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής – 3)

2.4.1.4 Oil-Dominated Dispersed Slug

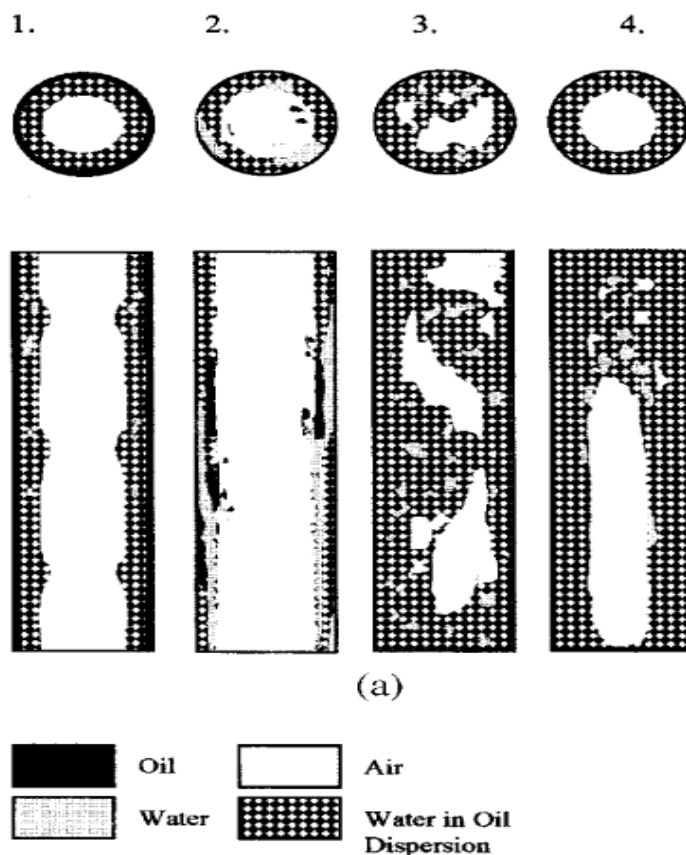


Figure 2.8 Oil dominated flow regimes in vertical three-phase flow Woods, G. S., Spedding, P. L., Watterson, J. K., & Raghunathan, R. S. (1998). Three-phase oil/water/air vertical flow. Chemical Engineering Research and Design, 76(5), 571-584.

The pattern resembled two-phase slug flow, with the liquid phase consisting of water droplets dispersed in oil. Yet, there was barely any liquid flow against the pipe wall in the annulus that surrounded the rising gas bubble. The thickness of the dispersed liquid annulus surrounding the gas bubble was significantly higher than in the two-phase system. The dispersed annulus was, in fact, a characteristic of the OD dispersed slug pattern (Woods et al. 1998). (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής – 4)

2.4.1.5 Water-Dominated Water Annulus/Oil Annular

As the water rate was raised past the inversion point, the conversion to WD flow was accompanied by the creation of a water annulus adjacent to the pipe wall, which was overlaid by an annular oil film. The flow regime was transparent. Depending on the gas flow, the annular oil film may flow as an annular or semi-annular film with roll waves or ripples on the surface (Woods et al. 1998). (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής - 5)

2.4.1.6 Water-Dominated Dispersed Annulus/Oil Annular

Under this pattern, increased mixing in the liquid phases occurred, leading to a partial dispersion of oil droplets in the water annulus. Therefore, the water film could take on a semi-annulus shape. Depending on the gas flow, the annular film interacting with the gas might flow as an annular film or as roll waves or ripples on the surface (Woods et al. 1998). (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής - 6)

2.4.1.7 Water-Dominated Dispersed Churn

The pattern resembled two-phase churn flow, with the liquid phase consisting of an oil droplet dispersion in water exhibiting a vertical oscillatory motion (Woods et al. 1998). (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής – 7)

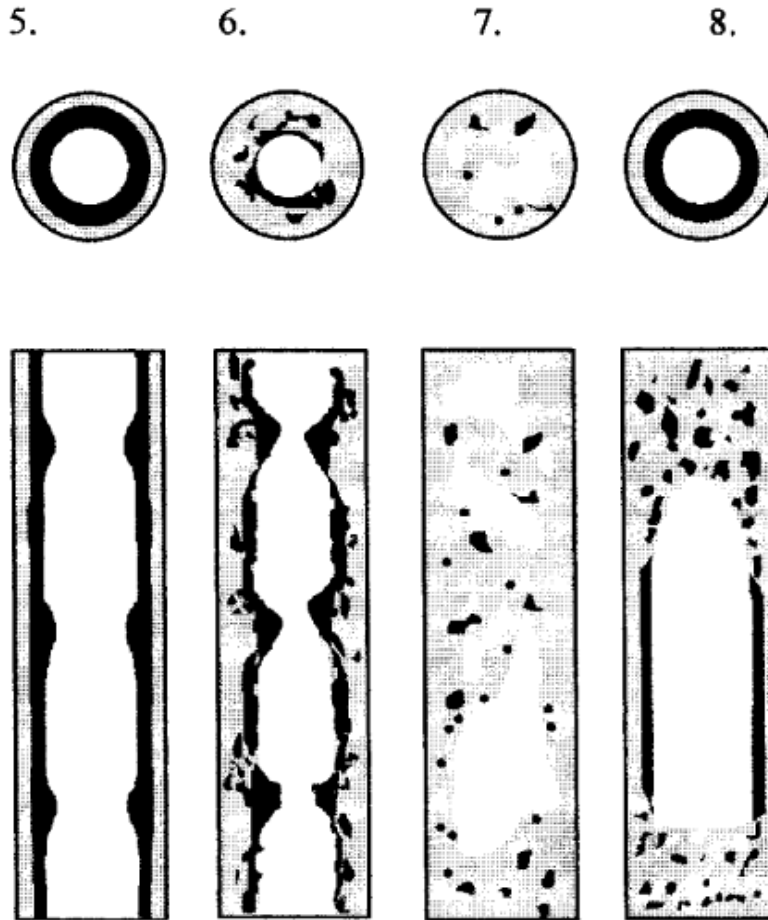


Figure 2.9 Water dominated flow regimes in vertical three-phase Woods, G. S., Spedding, P. L., Watterson, J. K., & Raghunathan, R. S. (1998). Three-phase oil/water/air vertical flow. Chemical Engineering Research and Design, 76(5), 571-584.

2.4.1.8 Water-Dominated Dispersed Slug

The pattern was comparable to two-phase slug flow, with the liquid phase consisting of oil droplets dispersed in water. A dispersed annulus was, in fact, a characteristic of the WD dispersed slug regime at times (Woods et al. 1998). (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής - 8)

2.4.1.9 Water-Dominated Oil Slug

Inside a water annulus, the oil was carried in fast-moving oil slugs. At low slug frequencies, distinct layers of water and oil were identified in the film area around the gas slug. Yet, at high slug frequencies, the distinction was not visible (Woods et al. 1998).

2.4.2 Three-phase flow patterns in horizontal pipes

It is challenging to develop a common terminology for horizontal two-phase flows since gravitational forces typically induce asymmetric horizontal two-phase flows. Various researchers have described horizontal two-phase flow regimes differently. Consequently, consistency across two-phase flow regime maps created by various researchers is sometimes difficult to observe. The same or even more complicated problem is posed by the maps for horizontal three-phase flows.

To resolve this issue, the researchers sought to characterize horizontal three-phase flow patterns by discriminating between plug and slug flows. They classified the flow regime as plug flow when the liquid phases drove the gas phase and as slug flow when the gas phase was the dominant driving force (Montoya et al. 2016). The authors also classified the flow regime as either oil dominated (OD) or water dominated (WD), with the dominant liquid phase constituting the continuum in which the second liquid phase was dispersed in some way. Moreover, classifications were established for intermittent, stratified and annular regimes.

There are numerous flow regimes for three-phase oil flows. Furthermore, oil dominated flow regimes introduce a new variety of flow regimes that do not emerge in water-based three-phase flow. Because of this, new terminology for various flow regimes is required. Acikgoz et al. (1992), Lahey et al. (1992) and Pan et al. (1995) all indicated a demarcation between separated and dispersed stratified flows. In three-phase stratified flows, the immiscible liquid phases can be dispersed. Small amplitude waves on the oil/water interface may be observed in stratified flow regimes. The liquid film was either annular separated or annular dispersed in the annular flows and the film adhering to the inner pipe wall had a marbled pattern. All annular flow observations are just for stratifying-annular flow conditions, in which the liquid film at the bottom of the pipe is thicker than at the top (Montoya et al. 2016) due to stratification.

In water dominated intermittent flows, while in the dispersed slug regime, the gas bubbles exhibited extremely defined tails around which oil droplets congregated. The slug regime maintained the basic characteristics of this type of pattern, except that the liquid was cleaner than in the analogous oil dominated regime.

The water-dominated stratified regimes resembled the corresponding two-phase gas/water regimes in appearance, except for a visible dispersion of oil droplets within the water continuum (Al-Awadi, H. 2011). The water-dominated annular regimes might be separated or dispersed.

The three-phase flow patterns observed in horizontal pipes are described as follows:

2.4.2.1 Oil-based dispersed plug flow

Oil-based dispersed plug flow (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής left) is seen at relatively low water and air superficial velocities. At these flow rates, water combine with oil, resulting in a foamy liquid solution. The flow regime was deemed plug flow as long as the liquid phases drive the air phase and the flow (Montoya et al. 2016) is more continuous and uniform. The existence of elongated liquid plugs separated by short air bubbles characterizes the mixture in plug flow.

2.4.2.2 Oil-based dispersed slug flow

In oil-based dispersed flow, increasing the air superficial velocity (Ali, S. F. 2009) can result in the establishment of a slug flow regime, in which the air phase begins to drive the liquid phases. This happens when the momentum of the gas phase gets sufficient enough to drive the liquid phases in the form of slugs. Slugs of varying lengths can be separated by relatively short or long gas bubbles. As previously, the oil-based liquid phase appears foamy within the slug flow region. However, in contrast to plug flow, the trailing edge of the large air bubbles was less distinct (Al-Awadi, H. 2011). This is because the gas bubbles in slug flow are not as well-defined as those in plug flow and tend to break up more quickly owing to the liquid's churning motion. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής right)

2.4.2.3 Oil-based dispersed stratified/wavy flow

Stratification and gravitational phase separation are seen in this flow regime (Açıkgöz et al. 1992). A mixture consisting of oil with relatively large water droplets forms an upper layer over a continuous layer of water (Al-Awadi, H. 2011). Oil and water exist in distinct layers separated by a wavy interface. Small-amplitude surface waves on the oil/water layer are noticed in this region of the three-phase flow regime map. The difference in density and

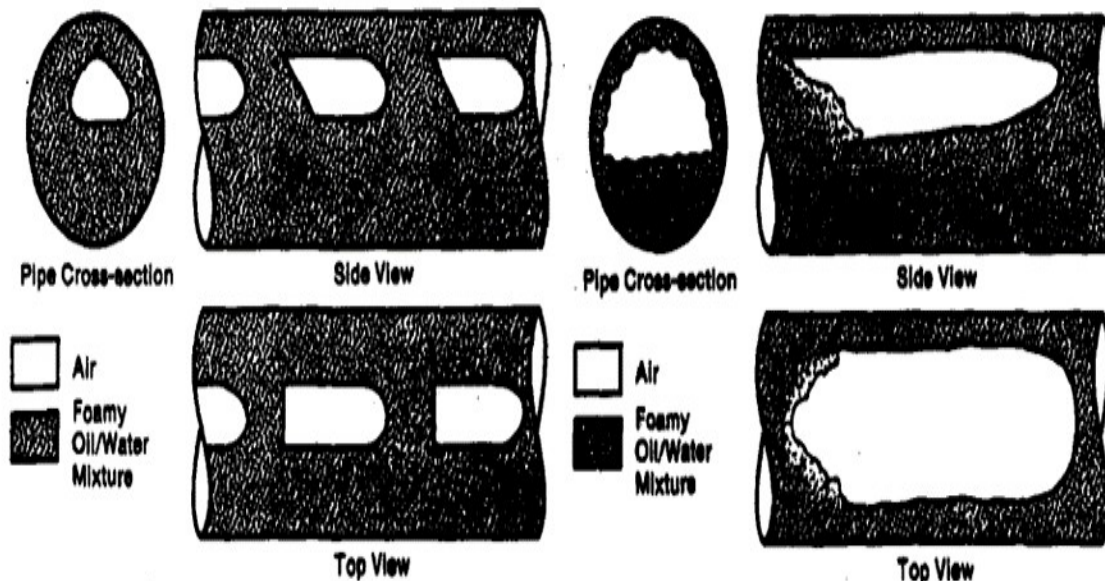


Figure 2.10 Schematic diagram of oil-based dispersed plug flow (left) - Schematic diagram of oil-based dispersed slug flow (right), Açıkgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.

viscosity of the two fluids causes the wavy interface between the oil and water layers. Many factors, such as flow rate, pipe diameter and fluid characteristics, can create waves at the interaction. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής left)

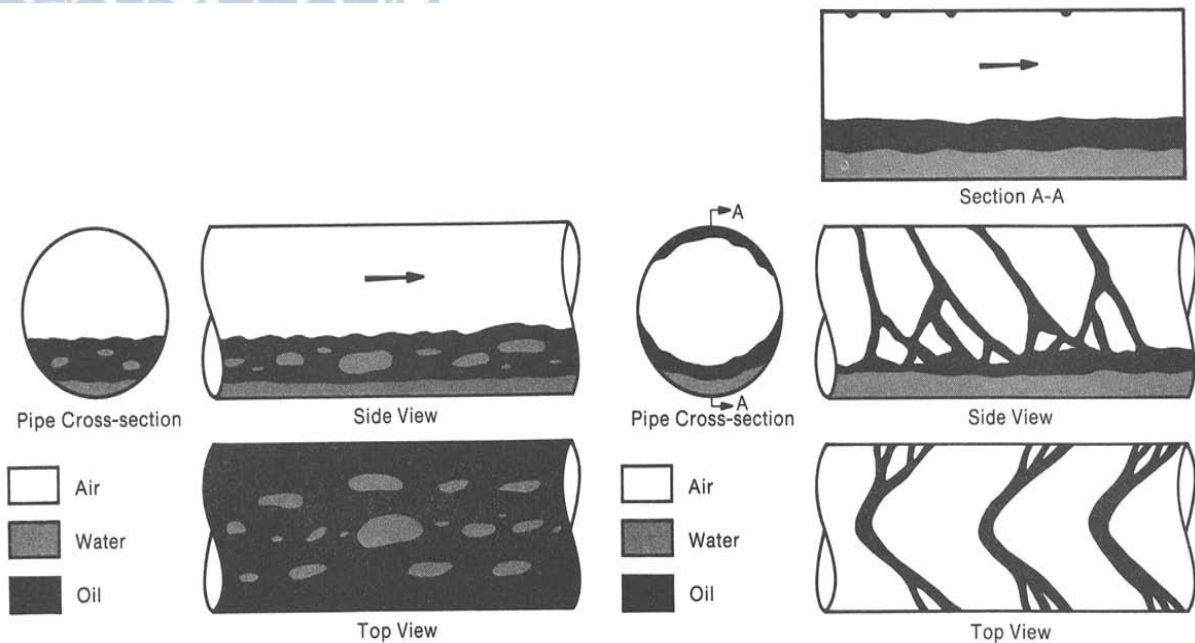


Figure 2.11 Schematic diagram of oil-based dispersed stratified / wavy flow (left) - Schematic diagram of oil-based separated stratified/wavy flow (right), Açıkgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.

2.4.2.4 Oil-based separated stratified/wavy flow

The oil and water phases were entirely separated for this flow regime. Because of gravitational stratification, the oil phase flowed above the water phase, resulting in the formation of a complex wave structure on the upper section of the pipe (Al-Awadi, H. 2011). In the interface between the oil and water phases, ripple waves are also seen. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής right)

2.4.2.5 Oil-based separated wavy stratifying-annular flow

In this flow pattern, the upper oil structures observed in separated stratified/wavy flow become denser in this flow regime and are linked with a thinner oil film, leading to continuous wetting of the upper pipe wall (Al-Awadi, H. 2011). Remarkably, stratification continues to play a major role in this flow regime. This results in a wavy annular flow pattern with a distinct oil film on the pipe wall and an oil layer distinct from the water phase. Despite the presence of the annular flow regime, stratification remains prevalent in this flow regime. Because of gravitational stratification, the oil phase is still separated from the water phase (Ali, S. F. 2009), with the heavier water phase moving below the lighter oil phase. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής left)

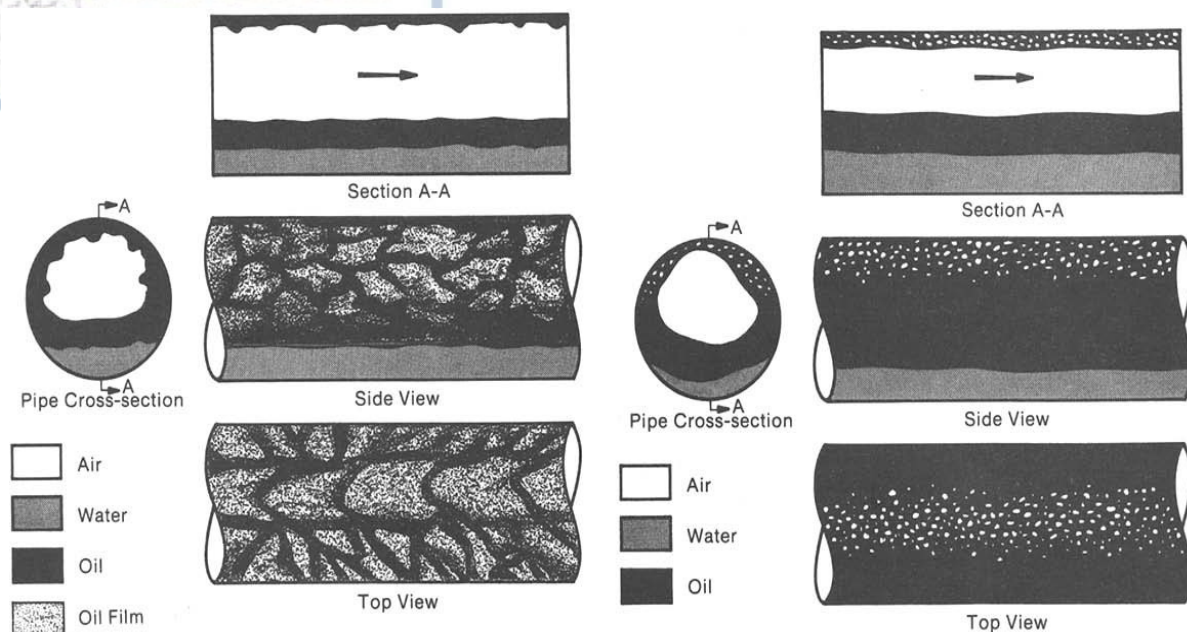


Figure 2.12 Schematic diagram of oil-based separated wavy stratifying-annular flow (left) - Schematic diagram of oil-based separated / dispersed stratifying-annular flow (right), Açıkgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.

2.4.2.6 Oil-based separated/dispersed stratifying-annular flow

Variations in the oil film thickness on the top pipe wall, a distinctive feature of the preceding flow regime (Montoya et al. 2016), are eliminated by raising the air flow rate. Small air bubbles may be seen in the oil film on the upper part of the pipe. The existence of these bubbles, generated by the higher air flow rate, might change the flow behavior and heat transfer properties of the system. Stratification effects are still visible in this flow regime, despite the existence of tiny air bubbles, as in the preceding flow regime. Because of gravitational stratification, the oil phase remains separated from the water phase. This causes the annular flow pattern to emerge, where the oil film wets the top pipe wall. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής right)

2.4.2.7 Water-based dispersed slug flow

In this flow regime, the air phase is the driving phase and the water phase is dispersed into slug-like structures that follow the air bubbles. Air bubbles with highly pronounced tails were noted under conditions of relatively low air and high-water flow rates (Montoya et al. 2016). Additionally, a relatively high concentration of oil droplets was observed in the areas subsequent to the air bubbles. An increased air flow rate resulted in the replacement of

distinct boundaries between the air plug tails and the water phase by a frothy appearance at the back of the air plugs (Al-Awadi, H. 2011), indicating that the oil droplets are being dispersed more evenly throughout the water phase. The frothy appearance of the water phase in this flow regime is caused by the turbulence generated by the air bubbles. The air bubbles break up into smaller bubbles, which then coalesce to form larger bubbles. The resulting frothy mixture of water and air bubbles has a higher interfacial area, which promotes efficient mass transfer and heat transfer between the two phases. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής left)

2.4.2.8 Water-based dispersed stratified/wavy flow

Excluding the dispersed oil droplets (Al-Awadi, H. 2011), this flow regime resembles a two-phase stratified/wavy flow, with the water phase flowing below the air phase due to gravitational stratification. The water and air phases create separate layers or waves in this

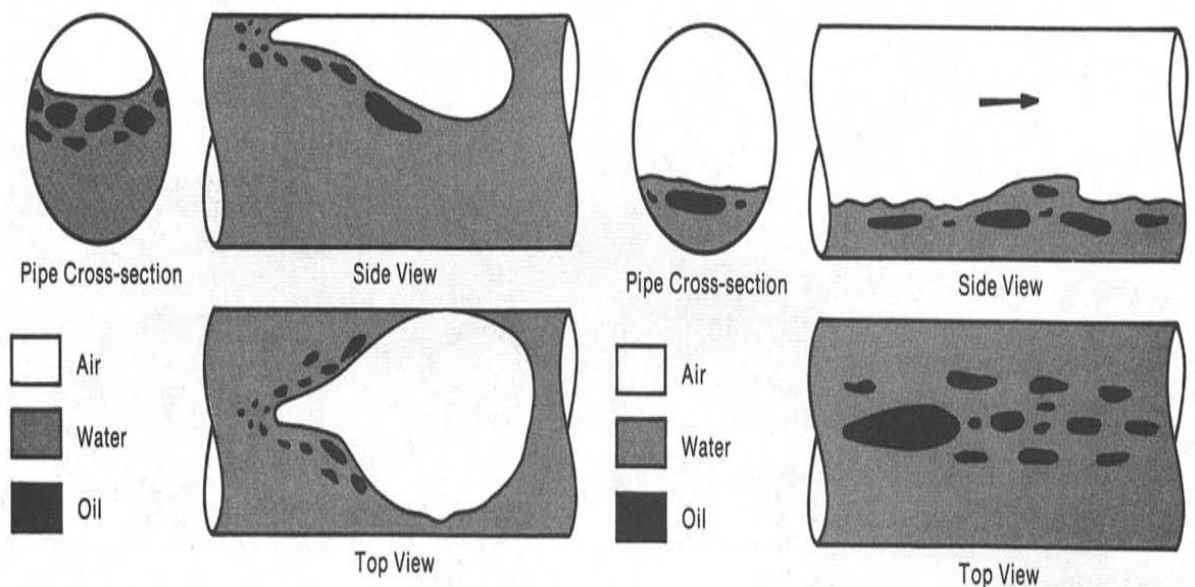


Figure 2.13 Schematic diagram of water-based dispersed slug flow (left) - Schematic diagram of water-based dispersed stratified/wavy flow (right), Açikgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.

flow regime, while minute oil droplets are dispersed throughout the water phase. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής right)

2.4.2.9 Water-based separated/dispersed incipient stratifying-annular flow

The relatively small waves are substituted by a new structure that incorporates roll waves as the air flow increases (Al-Awadi, H. 2011). Also, liquid "phase" separation occurs, probably as a result of gravity and shear effects, with oil droplets dispersed all through the

water phase. The flow regime is close to that of stratifying-annular flow, which is distinguished by a continuous liquid film on the pipe wall and a gas core in the pipe's center. The increased air flow rate allows the air phase to become the driving force in the system, causing the change from dispersed to separated flow. This causes bigger and more complex waves to develop, as well as liquid phase separation owing to gravity and shear forces. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής left)

2.4.2.10 Water-based dispersed stratifying-annular flow

During this flow pattern, the pipe periphery is continually wetted by a water-based film

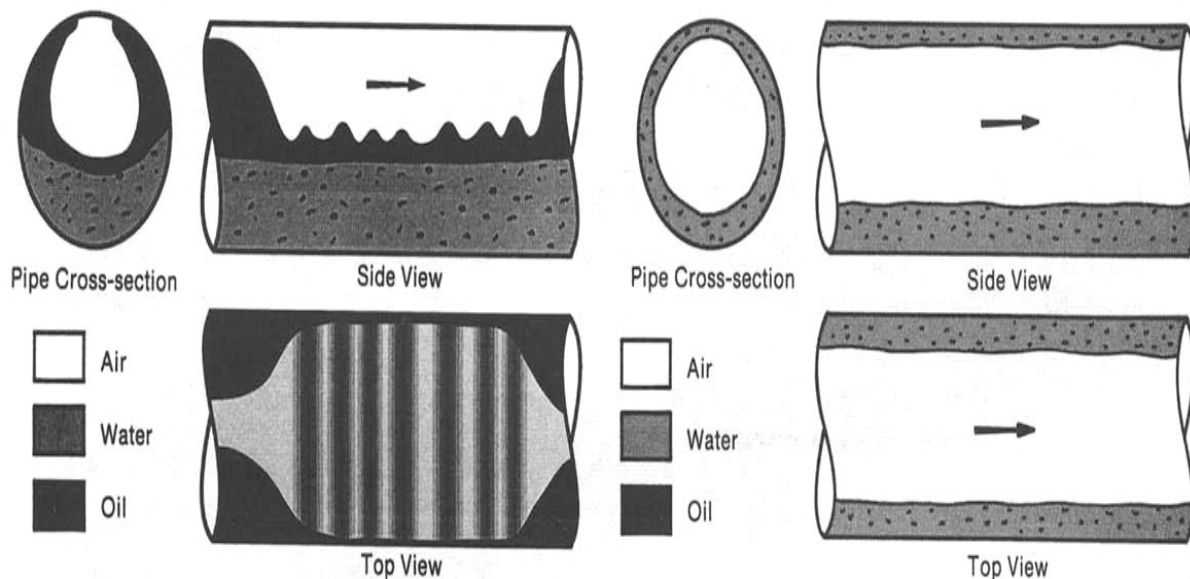


Figure 2.14 Schematic diagram of water-based separated dispersed incipient stratifying-annular flow (left) - Schematic diagram of water-based dispersed stratifying-annular flow (right), Açıkgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. International Journal of Multiphase Flow, 18(3), 327-336.

with minute oil droplets dispersed in it. The differences in the water film thickness between the upper and lower sections of the pipe were most evident at lower superficial air velocities and become less as the air flow rate increases. With the exception of the dispersed oil droplets, this flow regime resembled two-phase stratifying-annular flow. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής right)

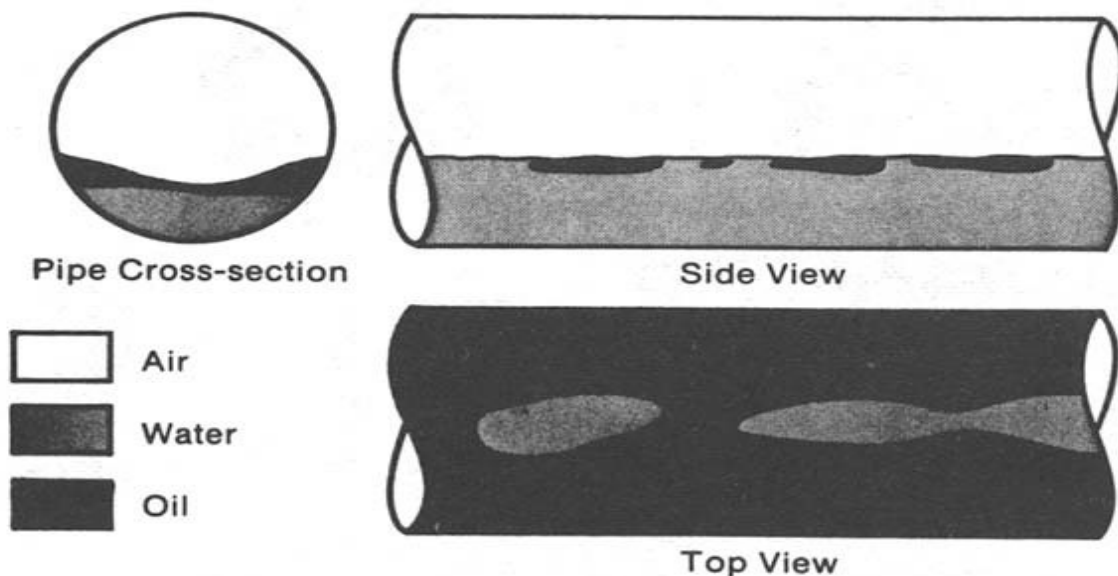


Figure 2.15 Schematic diagram of the boundary between water-based wavy flow and oil-based wavy flow., Açıkgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. *International Journal of Multiphase Flow*, 18(3), 327-336.

2.4.3 Three-phases flow pattern transition and mechanisms

Transition boundaries in three-phase flows (Ali, S. F. 2009), such as in two-phase flows, constitute essential regions of the flow regime map. Yet, in contrast to two-phase flows, three-phase flows exhibit new transition boundaries when the base fluid transitions. Such a boundary emerges when the flow is on the verge of transitioning from water-based stratified/wavy flow to oil-based dispersed stratified/wavy flow (Montoya et al. 2016). Quantifying the boundaries of three-phase flow regimes (Al-Awadi, H. 2011) poses a difficult problem due to the various transport properties of three-phase fluid mixtures.

Chapter 3 Multiphase flowmeters

Chapter 2 delves into Virtual Multiphase Flowmeters, designed to establish a solid foundation for the subsequent development and evaluation of the Virtual Multiphase Flowmeter in this master's thesis. The chapter begins with a comprehensive examination of test separators and physical multiphase flowmeters, emphasizing the challenges and limitations inherent in conventional methods and highlighting a shift towards more sophisticated and accurate measurement technologies. The historical overview and development of VFMs provide valuable insights into the evolution of production measurement techniques, underscoring the critical necessity and significance of VFMs in today's oil and gas industry.

The chapter proceeds by categorizing virtual multiphase flowmeters into two subgroups: first-principles or physics-driven VFM systems and data-driven VFMs. An extensive analysis follows, exploring the commercial applications, components and methodologies of these VFMs to clarify the complexities in characterizing and modeling multiphase flow. The comparison between first-principles and data-driven VFMs within this analysis aims to offer a comprehensive understanding, highlighting the strengths and limitations of each approach.

3.1 MPFMs -Test separators

A typical oil and gas production setup involves several wells linked to a flowline that transports the extracted liquid between the wellheads and an inlet separator located at a treatment plant. In subsea fields, a riser connects the flowline to the inlet separator. Choke valves, positioned at the wellheads, regulate the flow rate of the extracted liquid. Most fields have multiple wells and the extracted liquid typically comprises a complex blend of oil, gas, water and solids like sand or asphaltenes (Falcone et al., 2009; Bikmukhametov, T., & Jäschke, J., 2020). Upon reaching the inlet separator, the extracted liquid mixture undergoes a phase separation process, wherein its individual components are segregated. Subsequently, further processing is undertaken at the treatment plant (Falcone et al., 2009; Bikmukhametov, T., & Jäschke, J., 2020).

To ensure the efficient operation of oil and gas production systems, it is crucial to have accurate knowledge of the flowrates of oil, gas and water from each well. This information is essential for effective production optimization, reservoir management, rate allocation and future field operation prognosis (Falcone et al., 2001; Retnanto et al., 2001; Morra et al., 2014; Bikmukhametov, T., & Jäschke, J., 2020).

Traditionally, the main approach used for estimating well flowrates involved well testing. This process entailed directing the well's stream into a test separator, where it underwent separation into its fundamental components: oil, gas and water. Subsequently, the flowrates of these individual streams were quantified utilizing single-phase meters located at the separator outlet. (Corneliussen et al., 2005; Bikmukhametov, T., & Jäschke, J., 2020). Conventional test separators are large and expensive pieces of equipment demanding substantial monitoring time for assessing each well's performance due to the time it takes to achieve stabilized flow conditions. Furthermore, in deepwater developments, the exceptional length of the flowlines makes it even more challenging to monitor individual wells' performance. Test separators necessitated an independent flowline, which meant that every well had to be connected to the test separator and evaluated without the need to halt operations across the entire system. (Bikmukhametov, T., & Jäschke, J., 2020).

An additional approach for estimating flowrates is through the use of an inlet separator. This involves either shutting down all wells except the one being tested, which results in a significant production loss, or the temporary closure of the well under evaluation, followed by the measurement of the production rates of the remaining operating wells recorded at the separator conditions. The flowrates of the well under evaluation are subsequently determined through the subtraction of the obtained flowrates from the pre-tested measurements, a technique known as deduction well testing. (Idso et al., 2014; Bikmukhametov, T., & Jäschke, J., 2020). However, it is necessary to establish a steady operational environment to obtain accurate flowrate measurements, which can be time-consuming and the required duration may vary based on the well-separator distance. Additionally, the closure of a single well can affect the functionality of the remaining wells, potentially resulting in less accurate flowrate assessments (Falcone et al., 2001; Idso et al., 2014; Bikmukhametov, T., & Jäschke, J., 2020).

In order to prevent the necessity of shutting down all wells and conducting individual tests, a dedicated test line can be employed to oversee production from each well linked to the same manifold. Nevertheless, the expense associated with implementing a separate flowline can be a significant barrier. As an alternative, multiphase flow meters (MPFMs) can be installed in the subsea manifold, which provides several advantages over conventional test separators. MPFMs do not require separation of the phases. Conventional separators offer an accuracy within the range of about 5 to 10%, a level that can presently be attained using MPFMs. Moreover, MPFMs can provide continuous well monitoring without the need for regular intervention by trained personnel. (Falcone et al., 2001), Retnanto, et al., (2001),

Corneliussen, S., Dahl, B., & Hognestad, A. (2005), Morra et al., (2014), Bikmukhametov, T., & Jäschke, J., (2020))

Physical multiphase flow meters (MPFMs) were first commercialized in the early 1990s as a substitute for well testing for measuring well multiphase flowrates (Falcone et al., 2001; Bikmukhametov, T., & Jäschke, J., 2020). They were created in the 1980s in response to the anticipated decline in production from the major North Sea fields, which required future smaller discoveries to be tied back to existing infrastructure. In a mature producing province, higher gas and water proportions, which are inherent, can lead to increased instability in flow conditions within existing production facilities. This situation necessitates the implementation of more adaptable multiphase solutions. MPFMs have gained acceptance in the field and are becoming a primary metering solution for latest field developments. MPFMs are designed to gauge the flowrates of oil, gas and water without the need for phase separation. Flowmeters of this kind are typically placed at the wellhead to oversee the real-time multiphase flowrates from a particular well. Flowrates are indirectly determined by measuring fluid phase characteristics including velocities and phase compositions within the device (Falcone et al., 2001; Gryzlov, 2011; Bikmukhametov, T., & Jäschke, J., 2020). Considerable endeavors were invested in the development of precise multiphase flow measurement devices, employing various technologies such as acoustic attenuation, impedance and gamma densitometry to achieve this objective (Falcone et al., 2001; Bikmukhametov, T., & Jäschke, J., 2020). Numerous review articles delve extensively into the methods employed, the underlying principles, governing equations, and measurement strategies applied in this field. (Corneliussen et al., 2005; Falcone et al., 2009; Thorn et al., 2013; Bikmukhametov, T., & Jäschke, J., 2020).

MPFMs are recognized by the oil and gas industry as having significant advantages in terms of production facility layout, well testing, reservoir management, production allocation, production monitoring and capital and operating expenses. MPFMs minimize the amount of equipment required for onshore, offshore topside and offshore subsea applications. The elimination of a dedicated test separator for well-testing applications is crucial. The use of MPFMs for topside applications reduces the amount of platform space and load required for well-testing operations. High - cost well-testing lines can be removed from manufacturing facilities, which could be essential for unmanned locations, deepwater developments and satellite fields.

Both well testing and MPFMs have advantages and disadvantages for measuring flowrates. Well testing, for instance, requires the installation of an independent flowline and

separator, leading in significant initial investment expenditures for field development (Falcone et al., 2001; Bikmukhametov, T., & Jäschke, J., 2020). Closing the well of interest may lead to major production losses if the inlet separator serves as a test separator. Furthermore, due to potential flow assurance issues, deduction testing may be impossible (Melbo et al., 2003; Bikmukhametov, T., & Jäschke, J., 2020). Nonetheless, even with the installation of multiphase flow meters in the field, the practice of well testing remains prevalent in the monitoring of oil and gas production. This enduring reliance on well testing can be attributed to its role in providing a benchmark for calibrating multiphase flow meters and determining essential fluid properties (Corneliussen et al., 2005; Bikmukhametov, T., & Jäschke, J., 2020).

Another negative aspect of traditional well testing is that well performance deteriorates after shutdown cycles. Wells subjected to testing regularly, typically necessitate additional workover operations to preserve production levels. MPFMs, on the contrary, do not require shutting down the wells for testing, which means that the shutdown cycles affiliated with well testing have no effect on well performance. MPFMs can also be utilized to oversee the well as it undergoes the clean-up flow, which often results in lost flow data since the well stream does not pass through the test separator. This contributes to added benefits by providing enhanced management of formation drawdown, pressure transients and shorter flow periods.

MPFMs can identify and gauge changes in the gas/oil ratio or water cut in real time, whereas conventional test separators only provide data on cumulative volumes at discrete points in time. By providing real-time data on changes in gas and liquid flow rates, MPFMs allow operators to detect well-slugging effects or gas-lift issues as they arise. This enables production optimization as well as field life extension.

Operators should compare the capital and operating expenses of each solution before deciding between a traditional approach to manufacturing facilities and one that includes MPFMs. Although hardware costs are relatively easy to estimate, operating expenses are remarkably challenging to predict. While it is widely recognized that the capital cost of MPFMs is significantly lower than that of traditional metering hardware, operators often fail to appreciate the operating expenses involved with MPFMs, especially during the commissioning phase and the first year of operation. The real-time data on well flowrates provided by MPFMs undeniably offer advantages for the functioning of the system. Nevertheless, they come with a significant expense and in case of a malfunction, they require intervention, resulting in substantial system costs (Falcone et al., 2001; Patel et al., 2014; Bikmukhametov, T., & Jäschke, J., 2020). Besides that, physical multiphase flow meters

exhibit a particular operating range. After this point, the precision of flowrate estimations can significantly diminish. In addition, the flowmeters may degrade due to sand erosion or partial blockage, affecting measurement accuracy (Marshall and Thomas, 2015; Bikmukhametov, T., & Jäschke, J., 2020).

It is anticipated that as more field experience is gained, MPFMs will demand less maintenance after installation, resulting in lower operating costs. Furthermore, MPFMs could improve system management by increasing the value of recovered oil by 6 to 9%. However, existing MPFMs are not (and may never be) precise enough to meet fiscal-metering requirements, despite the fact that such an application would ensure MPFMs' future. While their accuracy has some limitations, the advantages of using MPFMs end up making them a promising solution for the oil and gas industry.

3.2 History and Development of VFM

Multiphase flow meters expanded the potential of monitoring individual wells in real time whereas minimizing production losses resulting from well test shutdowns. These flow meters play a pivotal role in optimizing well operation and control by providing real-time data (Góes et al. 2021). Because the expense of repairing or replacing damaged subsea equipment is prohibitively expensive, oil and gas companies prioritize the reliability of multiphase flow meters deployed in subsea fields. Furthermore, multiphase flow meters are either expensive, inaccurate, or unable to use downhole owing to harsh conditions (Leskens et al., 2008; Graham, 2015).

Virtual sensors, soft sensors and inferential models constitute mathematical models that can be employed to substitute measuring devices. These virtual sensors have the capacity to predict multiple system variables by employing mathematical models, replacing physical sensors and utilizing data obtained from other available variables (Fortuna et al. (2007); Góes et al. (2021)).

Typical sensors are effective in detecting straightforward process variables such as temperature, pressure, differential pressure and single-phase flow rate. However, for deriving online variables for which hardware sensors are either unavailable or too costly, the utilization of virtual sensors becomes a viable option, as advocated in Chu et al. (1998) and Góes et al. (2021).

Recently, the oil and gas industry has placed significant emphasis on virtual flow metering (VFM) as an alternative to conventional hardware-based flow measurement methods. This shift is prompted by the challenges and costs associated with both flow

measurement approaches. VFM operates by gathering existing field data and utilizing it within a numerical model to predict flowrates (Rasmussen, 2004; Toskey, 2011; Bikmukhametov, T., & Jäschke, J., 2020). The typical measurement data includes:

- Bottomhole pressure and temperature (PBH and TBH).
- Wellhead pressure and temperature upstream of the choke (PWHCU and TWHCU).
- Wellhead pressure and temperature downstream of the choke (PWHCD and TWHCD).
- Choke opening (Cop).

As opposed to well testing and physical multiphase flow meters, virtual flow meters can be incorporated into existing infrastructure and do not need the installation of extra equipment, allowing them to save field development capital and operational expenses. This approach enables real-time modeling and optimization of production while also limiting the need for additional equipment. Moreover, VFM systems provide capacities for estimating the flowrates in real time and adapt to alterations in flow conditions, providing a significant benefit over the well testing technique, where steady well flowrates are presumed between tests (Marshall and Thomas, 2015; Bikmukhametov, T., & Jäschke, J., 2020). Furthermore, Virtual Flow Meters may serve either as a standalone approach or as a backup system in conjunction with a MPFM, utilizing data from a physical multiphase flow meter to enhance flowrate estimations (Holmas et al., 2013). Despite the ongoing research endeavors in the realm of VFM and the various methods and models applicable to it, there is still a notable absence of a comprehensive overview, as pointed out by Bikmukhametov, T., & Jäschke, J. (2020).

Flow rate estimation can be derived from various inputs, including factors such as pressure drop across a choke, wellhead temperature and downhole pressure, as discussed in Graham (2015) and Góes et al. (2021). Carbone (2007) utilized temperature and pressure measurements obtained from surface sensors on a platform as input variables for virtual sensors to estimate individual well flow rates. However, Leskens et al. (2008) investigated the capabilities and constraints of multiphase soft sensors and stated that relying solely on downhole pressure and temperature values is insufficient for real-time well flow rate estimation (Góes et al. 2021). It is important to note that their conclusions were premised on simulations and did not take into account actual well data.

An approach was proposed by Garca et al. (2010) to estimate individual well oil production using data from sensors, well tests and simulations. They utilized a neural network and online correlation logic, training the neural network based on prediction errors

assessed using data from the most recent well test. This method achieved results with errors of less than 4% when compared to fiscal meters. Amim (2015) conducted an evaluation of the performance of commercial VFMs using actual production data from a subsea well. Despite encountering challenges in predicting variations in the gas-oil ratio (GOR) with the defined PVT model for the virtual sensor, these virtual sensors still delivered reliable flow rate estimations over extended periods (Góes et al., 2021).

Qutami et al. (2017) presented a soft sensor relying on a neural network that could estimate multiphase flow rates in oil and gas production pipes. This soft sensor was designed to provide flow rate estimations between well tests and could also function as a backup in case of failures in conventional multiphase flow meters, particularly when a common flow meter or test separator was in place. (Góes et al., 2021) Ursini et al. (2019) similarly created a real-time VFM by combining commercial software packages and optimization algorithms. This virtual flow meter was effectively implemented in an offshore gas field, offering real-time estimations exclusively for gas production flow rates from individual wells. It relied on upstream choke and bottomhole pressure and temperature data for its calculations (Góes et al., 2021).

Sanzo et al. (2020) also suggested a neural network-based machine learning algorithm for estimating multiphase flow rates. This algorithm took into account a range of process parameters and the functioning of installed equipment. The VFM developed in this work could function as a backup in situations where MPFMs experienced failures. Its flow rate predictions relied on process parameters located both upstream and downstream of the choke, incorporating factors such as pressure, temperature, and choke opening (Góes et al., 2021).

Bikmukhametov and Jaschke's (2020) recent research offered an extensive overview of the current state of VFM methods, incorporating numerical techniques, field experience and recent research activities. According to the conclusions drawn from this research, future studies should give priority to the development of auto-tuning and calibration techniques. Additionally, the review highlighted that the integration of machine learning into VFM has the potential to enhance precision, yet further research is essential to guarantee their robustness (Góes et al., 2021).

The total flow rate generated by a Production and Exploration Unit (PEU), encompassing all its wells, can be continuously monitored in real time. However, individual liquid and gas flow rates from each well are usually confirmed through a production test, which is conducted approximately bimonthly in some PEUs. Between these tests, it is presumed that every well is operating in accordance with the most recent Well Test Report.

This can result in delays in identifying issues and it is not practical to oversee the production of each well during this interim period (Góes et al., 2021).

In the research mentioned earlier, Leskens et al. (2008) provided instance analyses derived from simulations rather than real well data. García et al. (2010) utilized data from well tests for neural network training. Amim (2015) did not specify the utilization of well test data in their study. Additionally, Bikhmukhametov and Jaschke (2020) analyzed VFM approaches and noted the assumption that individual flow rates are expected to remain steady between well testing (Góes et al., 2021).

3.3 VFM Methods

Over the past two decades, various techniques have arisen for estimating multiphase flowrates through the utilization of accessible field data within the Virtual Flow Metering (VFM) concept. Several global firms have established commercial VFM systems for deployment by oil and gas operators worldwide. While some methods are currently undergoing evolution with the aim of enhancing the precision of flowrate forecasts, others are not presently employed in the industry but hold the capacity to drive advancements in VFM development in the future. Bikhmukhametov, T., & Jäschke, J. (2020) have outlined that two main Virtual Flow Metering methods can be distinguished based on modeling frameworks:

- First principles VFM
- Data-driven VFM

First principles VFM systems depend on mechanistic modeling of multiphase flows near the well, wells, pipelines and production chokes (Holms and Lvli, 2011; Bikhmukhametov, T., & Jäschke, J., 2020). These models are employed in conjunction with observations and data recordings, including pressure and temperature, for the evaluation of precise flowrate predictions. An optimization algorithm adjusts the flowrates and other tuning parameters to minimize the disparities between model estimations and real measurements (Holms and Lvli, 2011; Bikhmukhametov, T., & Jäschke, J., 2020). The production system can either be modeled in its entirety, covering everything from the reservoir to the processing facility, or be dissected into sub-models, contingent on the accessible data. Most contemporary commercial VFM systems predominantly rely on first principles models. (Bikhmukhametov, T., & Jäschke, J., 2020).

The data-driven VFM method depends on gathering field data and adjusting it to a mathematical model without specifying the physical characteristics of the production system, such as wellbore and choke geometry or flowline wall thickness. This approach, commonly

referred to as "machine learning" modeling, has gained widespread popularity across various applications, extending beyond the realm of oil and gas (Bikmukhametov, T., & Jäschke, J., 2020). The fitting process of the model is conventionally denoted as "training" in the context of data-driven modeling (Hastie et al., 2009). When the data-driven model is well-trained and the operational conditions fall within the range used for training, it demonstrates the capability to perform rapid and precise real-time metering. This method does not require as extensive specialized knowledge of production engineering as first principles models do and the model can be established at a relatively more cost-effective level. Beyond classifying methods based on modeling principles, another approach is to categorize them according to their treatment of the time-dependent aspects in the model. Following this criterion, the implementation of the following sub-classification is possible (Bikmukhametov, T., & Jäschke, J., 2020):

- First principles VFM – steady state and dynamic models
- Data-driven VFM – steady state and dynamic models

While the conservation equations in first principles VFM typically have a dynamic form, it is important to mention that the optimization problem is usually formulated in a steady-state or quasi-steady-state. In these scenarios, an optimization solver is responsible for finding the solution for a single time point or using the solution from the previous step as an initial estimate to predict the current time step. In specific situations, such as those concerning choke models (Perkins, 1993; Bikmukhametov, T., & Jäschke, J., 2020), the conservation equations may adopt a steady-state form or omit the consideration of time. Notably, while it is feasible to frame the VFM optimization problem dynamically, the existing body of literature on first principles VFM does not extensively delve into this method. This omission can be attributed to the fact that dynamic optimization for first principles VFM systems is often associated with significant computational cost (Lew and Mauch, 2006). It is worth noting that while such methods may have been employed in practice, they might not have been comprehensively documented in the literature (Bikmukhametov, T., & Jäschke, J., 2020).

State estimation methods, including approaches like the Kalman filter, can be employed to establish a dynamic VFM, as demonstrated by De Kruif et al. (2008) and Bikmukhametov, T., & Jäschke, J. (2020) in the context of dynamic optimization. The key reason for this selection might be the requirement for advanced knowledge in both setup and implementation, as well as the complexity of achieving reliable tuning for real field data.

The model formulation for most data-driven algorithms employed in VFM is typically established in a steady state. This indicates that, at a single time point, pressure and temperature data recordings are utilized by these algorithms to make predictions regarding flowrates for the same time step. However, there exist data-driven algorithm structures with dynamic formulations, allowing the use of past data recordings to predict flowrates at the current time step. Recently, various dynamic algorithm structures for VFM applications have been under investigation by several researchers (Bikmukhametov, T., & Jäschke, J., 2020).

3.4 First principles VFM systems

First principles Virtual Flow Meter systems, also known as physics-driven VFMs, are the most extensively utilized VFMs in the oil and gas industry. To determine multiphase flow rates, these models employ first-principles modeling, which is based on physics and chemistry laws. The models can be further classified as steady-state or transient. During the past 50 years, tremendous work has been expended to define each component of the approach, resulting in a thorough comprehension of production system behavior, fluid characteristics and optimization techniques. The Physics-driven VFM was initially designed as an add-on to SPT Group's (later acquired by Schlumberger) OLGA transient multiphase flow simulator for deepwater subsea development design and flow assurance monitoring (Canon et al., 2015). Kongsberg Ledaflow is a popular physics-driven VFM. These VFMs have undergone performance evaluations (Belt et al., 2011). Turbulent Flux's Flux Simulator, developed for real-time monitoring applications, is the latest transient multiphase flow simulator offered to the industry. According to Bikmukhametov, T., & Jäschke, J. (2020), a cutting-edge physics-driven VFM system today comprises the following essential elements:

- Fluid properties model
- Production system model including
 - Reservoir inflow model
 - Thermal-hydraulic model
 - Choke model
 - Electric submersible pump (ESP) model
- Data validation and reconciliation (DVR) algorithm

In a physics-driven VFM system, the fundamental concept is to create model outputs, including pressures and temperatures throughout the production system using thermal-hydraulic, choke, ESP and reservoir inflow models. These models demand pre-existing pressure-volume-temperature (PVT) data, describe fluid characteristics under specific

conditions and are produced utilizing fluid properties models like Equations of State (EoS) and Black Oil models (BOM) (Bikmukhametov, T., & Jäschke, J., 2020).

The Data Validation and Reconciliation (DVR) algorithm serves the purpose of validating the measured field data by eliminating outliers and filtering noise. In the reconciliation step, adjustments are made to model parameters, including flowrates, choke discharge coefficient, gas and water fractions, friction and heat transfer coefficients and slip relation. These adjustments aim to match the model outputs with data recordings from the physical system and guarantee an overall material balance. Creating a virtual flow meter with physics-driven VFM systems entails numerous crucial steps as follows (Bikmukhametov, T., & Jäschke, J., 2020):

- i. The first step is to develop a fluid properties model that precisely describes the fluid data. This model is crucial for precisely simulating the flow of fluid through the production system.
- ii. The subsequent phase involves the selection of suitable production system models contingent upon the available measurements. These models should be able to accurately simulate the flow of the fluid through the system and should take into account factors such as pressure, temperature and flowrate.
- iii. Once the models are chosen, the measurement data must be read and validated. Outliers and noise in the data must be eliminated to ensure that the resultant model is as precise as feasible.
- iv. When the data has been validated, tuning parameters must be chosen. These parameters, which include flow rates, choke discharge and heat transfer coefficients, will be utilized to fine-tune the model to match the real flow data. To begin the tuning procedure, initial parameter values must be estimated by a guess.
- v. The production system models chosen in step 2 must then be simulated utilizing the fluid characteristics from step 1 and the initial tuning parameter values from step 4.
- vi. The simulation outputs must be chosen based on which measurements are available such as pressures and temperatures at the bottomhole and wellhead,
- vii. After that, to reduce the mismatch between the model outputs from step 6 and the validated measurement data from step 3 the data reconciliation algorithm is run. This algorithm adjusts the tuning parameters chosen in step 4 to get the model output as close to the measurement data as possible.

- viii. Finally, depending on the solution from step 7, the oil, gas and water flowrates for each well are provided. This provides a virtual flow meter which can be utilized to monitor and optimize production in real-time.

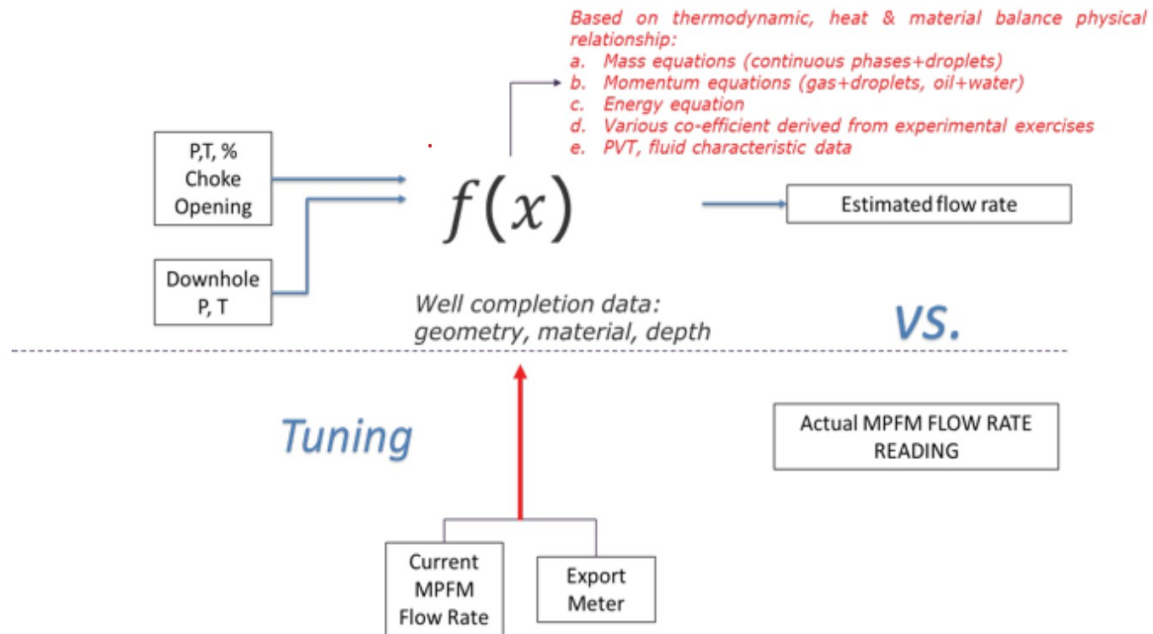
3.4.1 Commercial first principles VFM systems

The physics-driven VFM approach is extensively employed in industry because of its reliability in describing behavior of production systems and phenomena related to multiphase flow (Bikmukhametov, T., & Jäschke, J., 2020). Commercial physics-driven VFM systems employ advanced methodologies and models, making them an accurate representation of contemporary VFM technology. By examining these systems in depth, a further insight into the variations in models and capabilities among different software products on the market can be gained. Fully understanding the models and capabilities of commercial VFM systems is crucial since it allows for the industry to evaluate the effectiveness of VFM technology. These systems are continually changing and improving and by examining their strengths and limits, areas for further development may be discovered. Furthermore, by analyzing the differences between different software products, it is possible to determine which system is best suited to a particular application.

OLGA Online by Schlumberger, K-Spice Meter (K-Spice + LedaFlow) by KONGSBERG, FlowManager by FMC, Rate&Phase by BP, FieldWatch + METTE by Roxar, Well Monitoring System (WMS) by ABB, Virtuoso by WoodGroup and ValiPerformance by Belsim are among the commercial VFM systems available. These products have a similar structure in that they estimate multiphase flowrates using fluid characteristics and production system models in conjunction with a DVR algorithm (Bikmukhametov, T., & Jäschke, J., 2020). It should also be noted that there are additional VFM vendors omitted from the aforementioned, including Ensys Yocum, which provides the VMSS3 Virtual Flow Meter but with less literature available and TurbulentFlux, an upstart company cutting-edge VFM solution that is not yet publicly accessible.

Aside from the software mentioned, there are several more software packages that might be regarded as virtual flow meters. For instance, in Amin's (2015) study, Prosper (PETEX, 2017) was employed as a VFM system. Prosper is a widely used petroleum industry software that simulates the behavior of wells and production systems under varied conditions. It incorporates numerous reservoir inflow, choke and hydrodynamic models that are coupled to PVT data to assess well performance (PETEX, 2017). Nevertheless, because Prosper lacks the DVR algorithm to calibrate the models for specific field measurement circumstances, this

article does not consider it a fully integrated VFM system. Additionally, no other research or literature exists that evaluates Prosper as a VFM system. Mokhtari and Waltrich (2016) employed Pipesim as a VFM system to analyze several models for VFM purposes in another



study. Schlumberger's Pipesim is a steady-state multiphase flow simulator that also provides OLGA Online as a VFM product. As a result, while Pipesim does not serve as Schlumberger's VFM, it may be used for VFM purposes (Bikmukhametov, T., & Jäschke, J., 2020).

Figure 3.1 Physic based, first principle Virtual Flow Meter (VFM), Data Driven Versus Transient Multiphase Flow Simulator for Virtual Flow Meter Application, p.2, Mohd Azmin B Ishak, Idris B Ismail, Tareq Aziz Hasan Al-qutami

3.4.2 Components of the first principles VFM systems

3.4.2.1 Fluid properties model

Hydrocarbon mixtures are intricate compounds with varying characteristics under varying pressure and temperature scenarios throughout the production system. To identify the alterations, fluid characterization must be performed utilizing fluid samples acquired from various locations, including the downhole or separator (Whitson and Brule, 2000; Bikmukhametov, T., & Jäschke, J., 2020). From this characterization process, pressure-volume-temperature (PVT) data is generated and VFM systems subsequently utilize this data to accomplish two primary objectives outlined by Falcone et al. (2009):

The initial utilization of PVT data by VFM systems is to determine the local attributes of hydrocarbon mixtures including density, viscosity and thermal conductivity, which are required for physics-driven VFM models. These properties are directly engaged in thermal-hydraulic conservation equations in addition to choke, ESP, and reservoir inflow models, influencing VFM flowrate predictions. Consequently, providing inaccurate phase densities or enthalpies to the VFM system for certain pressures and temperatures might cause deviations between real and predicted flowrates. As a result, this can pose challenges in finding an efficient solution using the DVR algorithm when adjusting flow measurements and other modeling variables (Bikmukhametov, T., & Jäschke, J., 2020).

The other reason for incorporating PVT data into VFM systems is to reconcile flowrate measurements collected at various locations, such as at a separator under standard conditions and at the wellhead under local conditions, with local flowrate measurements or estimations. VFM systems use data validation and reconciliation (DVR) algorithms to adjust flowrate estimations and guarantee they match reference flowrate measurements despite changes in the production system's conditions. This is crucial for precise flowrate estimations, which are typically given under reference conditions for reasons of reconciliation, production and sales reporting (Pinguet et al., 2005; Bikmukhametov, T., & Jäschke, J., 2020). Reconciliation is essential in fields with interconnected wells to back-allocate overall collective measured output rates at separator conditions to individual wells (Bikmukhametov, T., & Jäschke, J., 2020).

There are two commonly used methods for fluid characterization (Whitson and Brule, 2000; Falcone et al., 2009; Bikmukhametov, T., & Jäschke, J., 2020):

3.4.2.1.1 Black Oil model

The term "black oil" is used to describe any liquid phase containing dissolved gas, which is often found in hydrocarbons extracted from oil reservoirs. These oils tend to have dark colors, gravities below 40°API and experience minor changes in composition within the two-phase envelope. The Black Oil model is a straightforward method to characterize fluid behavior, assuming that the oil component does not interact with gas or water phases. This method utilizes correlations to calculate properties of oil and gas separately (Whitson and Brule, 2000), taking into account three primary PVT properties: the oil formation volume factor (B_o), the gas formation volume factor (B_g) and the solution gas-oil ratio (R_s).

B_o can also be used to indicate how the oil phase shrinks or expands. Changes in dissolved gas as well as the oil's compressibility and thermal expansion create volume

changes in the oil. By far the most significant factor that affects volume change is dissolved gas. Oil formation volume factor can be evaluated in a laboratory or predicted using empirical correlations. A simpler parameter has been proposed for black oils with associated gas to take into account gas that dissolves (condenses) or evolves (boils) from solution in the oil. R_s can be evaluated in a laboratory or identified through empirical correlations. The black-oil model shouldn't be utilized at temperatures close to the critical-point temperature since it cannot predict retrograde condensation phenomena. The volume factor for gas formation is commonly referred to as B_g . It is referred to as the amount of gas at reservoir conditions that is required to produce one standard cubic foot of gas at standard conditions. B_g is a crucial parameter in reservoir engineering that is used to calculate how much gas can be produced from a specific reservoir volume. It can be calculated experimentally employing correlations based on the reservoir fluid parameters or determined experimentally in the lab. Once the black-oil-model parameters have been determined, oil density and other physical properties of the two phases can be estimated.

Modified Black Oil models (MBOM) for volatile hydrocarbon mixtures, on the other hand, have been established to add another key variable known as the solution oil-gas ratio. Whenever water is part of the fluid product, it becomes necessary to establish added attributes, including the gas-to-water ratio (R_{sw}) and the water formation volume factor (B_w). It is noteworthy that the amount of gas that can dissolve in water and the resulting changes in water volume are notably smaller in comparison to gas/oil systems. Additional comprehensive information regarding traditional and modified Black Oil models for volatile oils and water/hydrocarbon systems can be found in Whitson and Brule's SPE monograph (Bikmukhametov, T., & Jäschke, J., 2020).

3.4.2.1.2 Compositional model

A Compositional Model is a more accurate method for determining mass transfer of volatile oils and condensate fluids than black-oil-model parameters. If the composition of a fluid mixture, also known as "feed," is provided, a VLE calculation can be used to establish the distribution of the feed between the vapor and liquid phases and identify the composition of each phase. This information allows for the determination of the quality or mass fraction of gas in the mixture. Additionally, the knowledge of the composition of each phase enables the computation of the interfacial tension, densities, enthalpies and viscosities associated with each phase.

VLE calculations are more precise but more challenging to execute than black-oil-model parameters. If a detailed composition is available for a gas/oil system, black-oil parameters can be generated from VLE calculations. However, the black-oil model is more practical for non-volatile oils due to the constant composition that results for the liquid phase and the increased computational requirements of the Compositional Model.

In contrast to the Black Oil model, the compositional fluid model takes into account interactions between different hydrocarbon phases. A compositional fluid model relies on Equations of State (EoS), which describes the pressure, volume and temperature connections to determine the phase and volumetric behavior of the produced fluid (Whitson and Brule, 2000; Bikmukhametov, T., & Jäschke, J., 2020). The development of EoS can be traced back to Van der Waals' fundamental work in 1870, which has since been modified and improved. Peng-Robinson (PR) (Peng and Robinson, 1976), Redlich-Kwong (RK) (Redlich and Kwong, 1949) and Soave-Redlich-Kwong (SRK) (Soave, 1972) equations are the most widely utilized versions for oil and gas applications (Whitson and Brule, 2000; Falcone et al., 2009; Bikmukhametov, T., & Jäschke, J., 2020).

Both compositional and BOM methods for PVT modeling are supported by modern VFM systems, as described in various sources (Bendiksen et al., 1991; Melb et al., 2003; Haldipur and Metcalf, 2008; Kongsberg, 2016; Lvli and Amaya, 2016; Bikmukhametov, T., & Jäschke, J., 2020). While the compositional model is currently favored in first principles VFM, as evidenced by the Virtual Flow Metering assessment study conducted by Letton-Hall Group (Toskey, 2011), basic VFM solutions that rely solely on a single model (such as a choke/orifice model) typically opt for the Black Oil model due to its ease of use, as noted by Da Paz et al. (2010), Campos et al. (2014) and Bikmukhametov, T., & Jäschke, J., (2020).

Fluid property models, such as BOM or compositional models, are usually saved in PVT tables for usage by VFM models to simplify the simulation process. Employing the models directly in VFM systems would be computationally costly, necessitating the use of PVT tables to store the data and interpolate between data points during simulations.

Nevertheless, before constructing PVT tables, the fluid property models must be tuned to the specific petroleum fluid, as default model parameters may not reliably predict fluid characteristics from a particular field (Coats and Smart, 1986). Furthermore, fluid properties change throughout the field's life cycle, necessitating model adjustment (Falcone et al., 2009; Bikmukhametov, T., & Jäschke, J., 2020). Non-linear regression (Agarwal et al., 1990; Coats and Smart, 1986) or reiterative calibration of EoS parameters (Pedersen et al., 1988; Whitson and Torp, 1983) can be employed to do this. Calibration for EoS models is based on

laboratory test data, which includes compositional analysis (gas chromatography), constant composition expansion, multistage surface separation, constant volume depletion and differential liberation expansion (Whitson and Brule, 2000; Bikmukhametov, T., & Jäschke, J., 2020). Laboratory tests are carried out to determine the primary model variables whenever utilizing BOM. An instance of BOM parameter evaluation utilizing laboratory results is exemplified by the Whitson-Torp approach (Whitson and Torp, 1983; Bikmukhametov, T., & Jäschke, J., 2020).

Following the adjustment of fluid models to match particular fluid characteristics, PVT tables are constructed to cover a variety of anticipated pressure and temperature scenarios. Subsequently, these tables are imported into the VFM system (Bikmukhametov, T., & Jäschke, J., 2020). Utilizing the data available in the tables, the system can perform interpolations to determine properties such as phase density and viscosity to match the specific local pressure and temperature, for instance those at the wellhead (Bendiksen et al., 1991; Bikmukhametov, T., & Jäschke, J., 2020).

It is critical to ensure the precision of the fluid model utilized to characterize PVT characteristics (Bikmukhametov, T., & Jäschke, J., 2020). With MPFMs, which are highly sensitive to these properties, inaccuracies in PVT data have been recorded (AlDabbous et al., 2015). These deviations may be induced by inaccurate estimate of phase properties or improper use of EoS. (Abro et al., 2017; Bikmukhametov, T., & Jäschke, J., 2020). Predictions of fluid characteristics using EoS are only accurate inside the pressure and temperature boundaries employed during model tuning. Extrapolating outside of these boundaries may lead to severe estimation mistakes (Joshi and Joshi, 2007; Bikmukhametov, T., & Jäschke, J., 2020).

The precision of physics-driven VFM models, similar to MPFMs, heavily relies on the quality of PVT data. Therefore, it is crucial to have well-characterized fluid properties data for accurate flowrate estimations (Petukhov et al., 2011; Zhang et al., 2017; Bikmukhametov, T., & Jäschke, J., 2020). Whenever VFM is utilized in preliminary studies or fields, the importance of PVT data cannot be overstated (Haouche et al., 2012a). Erroneous PVT description can increase the unreliability in VFM estimations (Ausen et al., 2017; Bikmukhametov, T., & Jäschke, J., 2020). This is because PVT data determines local fluid attributes and differences in fluid attributes immediately impact local flowrate estimations. Furthermore, the reconciliation algorithm results are also influenced since the fluid attributes play a role in the conversion of rates from local to reference conditions, which are subsequently utilized in the algorithm (Bikmukhametov, T., & Jäschke, J., 2020).

3.4.2.2 Production system model

The production system model generally comprises of various elements that are determined by the field's available measurements and the equipment installed. Presented below are the primary models that can be incorporated into the VFM system based on first principles.

3.4.2.2.1 Reservoir inflow model

A commonly utilized model for representing reservoir inflow is the Inflow Performance Relationship (IPR) model. In this model, the production rate of a well is expressed based on the differential pressure between the reservoir and bottomhole conditions. They are defined as the well static or reservoir pressure P_R , the well flowing or bottom hole pressure P_{BH} and the flowrate Q . IPR curves are developed through multi-rate well testing (Golan and Whitson, 1991; Bikmukhametov, T., & Jäschke, J., 2020) and give essential information for understanding the behavior of fluids flowing from the reservoir through the formation and into the well. IPRs are widely used in the industry, with various models available to be implemented in current VFM systems. The IPR quantifies production potential by analyzing the well's flowing pressure and production rate. It also provides key insight for well completion design, production optimization, nodal analysis and artificial lift design. Several IPR correlations exist, including Vogel's and Fetkovitch's models, in addition to a few analytical correlations that have limited applicability. The following models are some of the more popular ones:

3.4.2.2.1.1 Linear

The linear model, commonly employed in undersaturated oil wells (Cholet, 2008; Schlumberger Limited., 2014; Bikmukhametov, T., & Jäschke, J., 2020), posits that the well's rate is in direct proportion to the pressure differential between the reservoir and bottomhole conditions, as proposed by Bradley (1987). This linear model underpins the productivity index relationship for liquid reservoirs, which is the simplest and most widely used IPR equation. This straight-line equation indicates that the rate of production is directly proportional to the pressure drawdown between the bottomhole and the reservoir. The productivity index (PI) is typically estimated during a well test and once the PI is estimated, it can be used in subsequent calculations, such as nodal analysis, production optimization and

to artificial lift systems design, such as gas lift or electric submersible pumps. The subsequent Equation 3.3 describes this model.

$$q_l = PI(P_R - P_{BH}) \quad \text{Equation 3.3}$$

Where q_l represents the liquid flowrate, PI indicates the productivity index, P_R represents the reservoir pressure and P_{BH} stands for bottomhole pressure.

3.4.2.2.1.2 Vogel

Vogel's (1968) equation is among the various methods available for specifying the Inflow Performance Relationship (IPR) for a completion. It was generated to simulate oil wells that are saturated. A number of simulated well performance calculations were used to create Vogel's equation, which is the best-fit approximation. Vogel's study exclusively examines the impact of reservoir and fluid characteristics on systems under maximum saturation conditions. The impacts of high-velocity flow that might occur in high-rate wells are not taken into consideration by the Vogel relation. Vogel's equation is (Equation 3.4):

$$Q_l = Q_{max} \left(1 - 0.2 \left(\frac{P_{BH}}{P_R} \right) - \left(\frac{P_{BH}}{P_R} \right)^2 \right) \quad \text{Equation 3.4}$$

Q_l represents the liquid flow rate (STB/D or m³/d), Q_{max} represents the absolute open hole flow potential, that is the liquid flow rate when the bottom hole pressure equals zero, P_{BH} refers to the well flowing (or bottom hole) pressure (psia or bara), P_R refers to the well static (or reservoir) pressure (psia or bara).

3.4.2.2.1.3 Fetkovich's equation

Fetkovich's equation, an extension of the Vogel equation, is among several methods available for determining the Inflow Performance Relationship (IPR) for a completion. This equation incorporates the impact of high velocity, as outlined in Equation 3.5 (Schlumberger, 2017).

$$Q_l = Q_{max} \left(1 - \left(\frac{P_{BH}}{P_R} \right)^2 \right)^n \quad \text{Equation 3.5}$$

where Q_l signifies the liquid flow rate (measured in STB/D or m³/d), Q_{\max} denotes the absolute open hole flow potential, indicating the liquid flow rate at zero bottom hole pressure, P_{BH} pertains to the well's flowing (or bottom hole) pressure (expressed in psia or bara), while P_R corresponds to the well's static (or reservoir) pressure (given in psia or bara).

3.4.2.2.1.4 Backpressure/Backpressure normalized

The Back Pressure Equation was formulated by Rawlins and Schellhardt (1935) following tests on 582 wells. Although it is primarily employed for gas wells, its relevance to oil wells has been verified. In cases where correlations are already available for oil wells, the backpressure equation is exclusively utilized for gas wells. The equation is written as follows (Equation 3.6):

$$q_g = C_b (P_R^2 - P_{BH}^2)^n \quad \text{Equation 3.6}$$

where q_g represents the gas flowrate, C_b and n stand for the tuning coefficients, both determined through well tests, P_R signifies the reservoir pressure and P_{BH} indicates the bottomhole pressure.

For saturated oil wells, a normalized form of the preceding equation is utilized, and it can be written as (Equation 3.7):

$$q_l = q_{l,\max} \left[1 - \left(\frac{P_{BH}}{P_R} \right)^n \right] \quad \text{Equation 3.7}$$

where $q_{l,\max}$ represents the maximum liquid flowrate.

3.4.2.2.1.5 Forchheimer/Single Forchheimer

Forchheimer developed an equation for non-Darcy flow in reservoirs in 1901. It is critical to address inertial effects in gas reservoirs with high flowrates. To model these effects, the Forchheimer model, a non-law Darcy's model, is applied (Bradley, 1987). The tuning coefficients B_f and C_f in the Forchheimer model are determined by well testing. (Equation 3.8)

$$P_R^2 - P_{BH}^2 = B_f q_g + C_f q_g^2 \quad \text{Equation 3.8}$$

A single Forchheimer model is utilized for high-pressure gas wells. The single Forchheimer model has a linear equation where the pressure difference is proportional to the flowrate. (Schlumberger Limited, 2014; Bikmukhametov, T., & Jäschke, J., 2020) (Equation 3.9)

$$P_R - P_{BH} = B_f q_g + C_f q_g^2 \quad \text{Equation 3.9}$$

3.4.2.2.1.6 IPR table

An Inflow Performance Relationship (IPR) table is a collection of data that demonstrates the relationship between a well's flowrate and pressure differential. This table is produced using user-input information and an estimated pressure delta. The table data is then used to estimate the flowrate through a linear or polynomial interpolation technique (Schlumberger Limited, 2014; Bikmukhametov, T., & Jäschke, J., 2020).

Depending on the production conditions of the well, the models used to represent Inflow Performance Relationship (IPR) might take several forms. IPR models can serve several purposes regarding VFM. Firstly, they can be utilized as standalone models to predict a well's productivity capacity. Additionally, the incorporation of IPR models along with thermal-hydraulic and choke models allows for the representation of the system's boundary conditions and the simulation of reservoir inflow to the well. This approach introduces additional parameters into the VFM calibration procedure, which can be employed to fine-tune the model to historical data. Lansagan (2012) has demonstrated an approach where the IPR equation is employed in conjunction with a thermal-hydraulic model to calculate multiphase flowrate under steady-state conditions, as outlined in Bikmukhametov, T., & Jäschke, J. (2020).

3.4.2.2.2 Thermal-hydraulic model.

Multiphase flow in oil and gas fields, both in wells and pipelines, has been present for over a century (Shippen and Bailey, 2012). Lockhart and Martinelli made the first attempt at modeling multiphase flow in 1949, relying on empirical correlations from experiments and field data. This method was eventually replaced by a more fundamental modeling strategy that took into account the underlying physics of multiphase flow phenomena. Shippen and Bailey provide a thorough overview of the history of multiphase flow model development (2012). In accordance with the literature, numerous thermal-hydraulic multiphase models are

presently employed in physics-driven Virtual Flow Metering (Bikmukhametov, T., & Jäschke, J. (2020).

3.4.2.2.1 Two-fluid model

The two-fluid model is a numerical technique used for simulating gas-liquid two-phase flow, focusing on the thermo-fluid behavior. This model considers conservation of mass, momentum and energy for each phase, including interfacial transfer terms that describe the exchange of mass, momentum and energy between the phases. These transfer terms account for thermal and kinematic non-equilibrium, making the two-fluid model more accurate. In one-dimensional thermo-fluid simulation codes using the two-fluid model, the interfacial drag force term is critical for interfacial momentum transfer and it is formulated as the product of the drag coefficient and the average relative velocity between the gas and liquid phases, squared. To derive this formulation, the drift-flux model concept is utilized.

The two-phase flow model comprises of three sub-models: the homogenous flow, drift flow and two-fluid flow models. The homogenous flow model treats the two-phase flow as a single-phase flow, lacking the ability to capture phase interactions. The drift flow model represents the two-phase flow as a mixture and introduces drift velocity to account for inter-phase slip, but it does not account for interphase heat exchange. The two-fluid model is the most precise and reasonable model among the three, as it employs distinct physical equations to model the flow and heat transfer mechanisms of each phase. However, due to the absence of empirical correlations, the auxiliary model of the two-fluid model remains incomplete. Nevertheless, the two-fluid model is expected to be the primary model for two-phase flow analysis, owing to the advancement of numerical simulations.

The multi-fluid model, commonly known as the two-fluid model, formulates distinct conservation equations for each phase, regardless of being in a continuous or dispersed state. The general form of mass, momentum and energy equations for each phase has been expressed in previous studies conducted by Goldszal et al. (2007), Nydal (2012) and Bikmukhametov, T., & Jäschke, J., (2020):

Mass equation:

$$\frac{\partial \alpha_k \rho_k}{\partial t} + \frac{\partial \alpha_k \rho_k u_k}{\partial x} = \Psi \quad \text{Equation 3.10}$$

Momentum equation:

$$\frac{\partial \alpha_k \rho_k u_k}{\partial t} + \frac{\partial \alpha_k \rho_k u_k u_k}{\partial x} = -\frac{\partial \alpha_k p_k}{\partial x} - \alpha_k \rho_k g \sin \theta - F_{kw} \pm F_{ki} + O_k \quad \text{Equation 3.11}$$

Energy equation:

$$\frac{\partial \alpha_k \rho_k h_k}{\partial t} + \frac{\partial \alpha_k \rho_k u_k h_k}{\partial x} = \frac{\partial}{\partial x} \alpha_k k_k \frac{\partial T_k}{\partial x} + a_k \frac{Dp}{Dx} + Q_{kw} + \sum_{i \neq k} O_{ki} + Q_{ext} \quad \text{Equation 3.12}$$

where k stands for phase density, α_k for phase volume fraction and t for time. u_k stands for the phase velocity. The pipe's axial dimension is indicated by x . Ψ identifies the sources of mass transfer (e.g. phase change and mixing), Phase pressure is denoted by p_k , while pipe inclination is denoted by θ . F_{kw} stands for wall friction. The interphase friction symbol is F_{ki} . The letters O_k represent additional momentum exchange terms, encompassing phase change, droplet exchange and the level gradient term, along with p representing system pressure and k_k indicating the effective phase thermal conductivity. The rate of phase transfer at the pipe wall is denoted by Q_{kw} . The rate of heat transfer between the k phase and other fields is indicated by the letters Q_{ki} . The remaining net external heat transfer sources are referred to as Q_{ext} .

The mass conservation equation for a specific phase in a multiphase flow system is represented by Equation 3.10. It indicates that the quantity on the right-hand side of the equation, which stands for the net rate of mass transfer into or out of the phase, is equal to the rate of change of the volume fraction of the phase with time, or the left-hand side of the equation. The term "mass transfer" includes all forms of reactions, including phase changes, mixing and others, as sources of mass transfer.

Equation 3.11, which provides the conservation of momentum for each phase in a multiphase flow, is the momentum equation for the two-fluid model. The phase volume fraction α_k , density ρ_k and velocity u_k are combined to form the left-hand side of the equation, which reflects the time and space rate of change of momentum for the k phase. The pressure gradient force acting on the k -th phase is represented by the first term on the right-hand side of the equation, $-\frac{\partial \alpha_k p_k}{\partial x}$. It describes the rate of pressure change in the pipe's axial direction,

which causes a force to accelerate the phase in the direction of decreasing pressure. The negative sign indicates that the force acts in the opposite direction to the pressure gradient, which is a fundamental principle of fluid mechanics. The gravitational force per unit volume is represented by the second term on the right, $\alpha_k \rho_k g \sin \theta$.

Equation 3.12 is a representation of the energy conservation principle in the multi-fluid model, which considers various factors affecting the energy balance of each phase. This equation takes into account the contributions of heat transport through both convective and conductive mechanisms, pressure work, wall friction, interphase heat transfer and external sources of heat transfer. It describes how the enthalpy of a particular phase changes over time and space.

3.4.2.2.2 Drift-flux model

The drift-flux model introduces the term of drift velocity, which stands for the disparity between the volume of the mixture and the gas's velocity. This drift velocity reveals the relative velocity or kinematic non-equilibrium that exists between the gas and liquid phases. The local drift-flux model is averaged over the flow channel when the drift-flux model is used in one-dimensional analysis, which results in an uneven distribution of the gas, liquid, and void fraction and a covariance term. The crucial parameter that defines the phase distribution in the flow channel is the distribution parameter, which is the covariance term. The drift-flux model offers precise numerical simulations of the thermo-fluid behavior of the gas-liquid two-phase flow by incorporating local information about flow distribution into one-dimensional analyses via the distribution parameter. Reliable results are ensured by providing the distribution parameter and drift velocity through constitutive equations.

By multiplying the liquid fraction by the relative velocity between the gas and liquid phases, the drift velocity can be determined. The bubble-shape regime affects how the relative velocity in drift velocity is modelled. In bubble shapes such infinite media, viscous, distorted-fluid-particle and cap bubbles, for example, the drift velocity is described by a drag law. In contrast, potential flow theory is used to predict the drift velocity in a slug flow regime where the bubble shape is confined by a channel wall. The drift velocity in a severely deformed bubble-shape regime, also known as a churn-turbulent-flow regime, is modeled using the crucial Weber number. When used with real two-phase flows, empirical coefficients and correction factors are typically added to the predicted drift velocity to account for differences between the model and experimental data. Although it can be difficult to

characterize the drift velocity at flow regime transitions, the drift velocity model for each bubble-shape regime can serve as a starting point for explaining the drift velocity's behavior at these transitions.

On the contrary, the determination of the distribution parameter relies on the distributions of mixture volumetric flux and void fraction, which are not extensively available in experimental databases. Hence, in the modeling of the distribution parameter, it is typically determined empirically as a slope in the drift-flux plot of mixture volumetric flux vs gas velocity.

Within the drift-flux model, mass conservation equations are expressed individually for every component, whereas the energy and momentum equations are solely developed for the mixture (Bikmukhametov, T., & Jäschke, J., 2020). This aspect of the model is elucidated in the research conducted by Holmas and Lovli (2011).

Mass Conservation Equation (same to Equation 3.10)

$$\frac{\partial \alpha_k \rho_k}{\partial t} + \frac{\partial \alpha_k \rho_k u_k}{\partial x} = \Psi \quad \text{Equation 3.13}$$

Momentum Equation

$$\frac{\partial}{\partial t} \sum_k \alpha_k \rho_k u_k + \frac{\partial}{\partial x} \sum_k \alpha_k \rho_k u_k u_k + \sum_k \alpha_k \rho_k g + \frac{\partial p}{\partial x} = -F_{tot} - O_{tot} \quad \text{Equation 3.14}$$

Energy Equation

$$\frac{\partial}{\partial t} \sum_k \alpha_k \rho_k E_k + \frac{\partial}{\partial x} \sum_k \alpha_k \rho_k u_k E_k + \frac{\partial}{\partial x} \sum_k \alpha_k \rho_k p + U_{tot} = 0 \quad \text{Equation 3.15}$$

where F_{tot} stands for total wall friction, O_{tot} for source term, E_k for total energy and U_{tot} for total source term, which includes sources, wall heat transfer and mass transfer.

The mass conservation equation, Equation 3.13, explains how the volume fraction of each phase changes over time and along the axial dimension of the pipe. It states that the net rate of mass transfer into or out of the system owing to multiple factors, including phase change and mixing, is equal to the rate of change of the volume fraction of phase k with

respect to time. The right-hand side of the equation denotes the sources of mass transfer, whereas the left-hand side denotes the change in the mass of phase k through distance and time.

Equation 3.14 which is written in terms of the mixture, represents the momentum conservation equation for a two-phase flow system. It represents the overall fluid mixture's momentum changes through time and space. The two terms on the left-hand side of the equation are the time and space rates of change of the momentum of each phase weighted by their respective volume fractions. This term refers to how much each phase contributes to the mixture's momentum. The third term, $\sum_K \alpha_K \rho_K g$, denotes the gravitational force per volume of the mixture as a result of the acceleration of gravity. The first component in the equation's right-hand side F_{tot} , reflects the total force of wall friction acting on the mixture. Included in this are the frictional losses brought on by the mixture's flowing along the pipe or channel's walls. The source term is denoted by the second term on the right-hand side of the equation, O_{tot} . This term incorporates contributions from a variety of sources, such as internal forces (for instance gravity) or external forces (such as interfacial forces between the phases). The pressure gradient term $\frac{\partial p}{\partial x}$, which indicates how the pressure difference affects the momentum of the mixture, is also a part of the equation. The spatial rate of change of the pressure is correlated with the pressure gradient. The total source term, U_{tot} , which is included in the equation as a final component, accounts for all the sources of energy transfer in the system, including heat transfer to and from the pipe or channel walls.

Equation 3.15 explains the drift-flux model's two-phase flow energy conservation. According to the equation, the time derivative of the summation of the energy densities of the two phases, the spatial derivative of the summation of the energy flux densities of the two phases, the spatial derivative of the summation of the pressure and the spatial derivative of the summation of the total source term all have to equal zero.

In order to account for the difference in phase velocities, the drift-flux model necessitates a slip relation. Zuber and Findlay (1965) established the most well-known and widely used form (Equation 3.16):

$$u_g = C_o u_m + u_d \quad \text{Equation 3.16}$$

where the abbreviations u_g , u_m , u_d , and C_o stand for "gas velocity," "mixed velocity," "drift velocity," and "profile parameter," respectively.

3.4.2.2.3 Steady state mechanistic model

The multiphase flow models discussed above are time-dependent, meaning they represent the flow behavior in both space and time. The model enters steady state and is only resolved in space if the time derivative term $\partial/\partial t$ has a value of zero. The model assumes a constant flow rate throughout time. This kind of model is helpful in situations when identifying the spatial distribution of flow parameters including temperature, pressure and velocity is the main goal and the flow behavior does not vary considerably over time. Instabilities in well behavior, such as liquid loading or severe slugging, cannot be well described by a steady state model since they are transient in nature (Waltrich and Barbosa, 2011; Bikmukhametov, T., & Jäschke, J., 2020).

Different VFM systems may employ different thermal-hydraulic models, with some utilizing only one specific model formulation while others use a combination of them. For example, OLGA incorporates a mixture energy equation with a two-fluid momentum equation formulation, resulting in a total of five mass equations, three momentum equations and one mixture energy equation (Nydal, 2012; Shippen and Bailey, 2012). On the other hand, K-Spice Meter employs LedaFlow to resolve multiphase flows in wells and utilizes nine mass equations, three momentum equations, and three energy equations, making it a two-fluid model with nine fields, including three continuous and six dispersed phases (Kongsberg, 2016; Shippen and Bailey, 2012; Bikmukhametov, T., & Jäschke, J., 2020). In contrast, FlowManager utilizes a transient drift-flux model with one mixture momentum equation and one energy equation, solving the mass balances for each phase separately (Holmas and Lovli, 2011; Bikmukhametov, T., & Jäschke, J., 2020). A similar approach is employed by METTE, a multiphase flow solver in FieldWatch that uses the transient drift-flux model with mixture momentum and energy equations, with an option to incorporate or omit slip effects between phases (Roxar, 2015; Bikmukhametov, T., & Jäschke, J., 2020).

3.4.2.2.3 Choke model

In order to control production rates and avoid problems like gas or water coning, sand production, or excessive erosion velocities, surface chokes are frequently installed close to the wellhead of producing wells. They may be necessary to fulfil certain production requirements or restrictions. In some circumstances, flow rates are estimated using pressure measurements taken at the choke.

Critical flow occurs when fluids flowing through the choke throat in compressible flow reach velocities that are greater than the velocity of sound. This occurs in both single-phase gas and multiphase flow and indicates that flow behavior becomes independent of downstream conditions. Subcritical flow is what occurs when the highest velocity of the fluids in the choke is lower than the sonic velocity. Predicting the sonic velocity or the boundary between critical and subcritical flow is crucial for effectively describing the behavior of compressible fluids via chokes. The flow rate through the choke becomes constant when the pressure ratio is at the critical value. Predicting the flow behavior of multiphase fluids through chokes requires a thorough understanding of single-phase flow through restrictions.

Choke valves have been utilized for security and management reasons in the hydrocarbons industry for several decades (Buffa and Balio, 2017; Bikhmukhametov, T., & Jäschke, J., 2020). Due to the proportional relationship between pressure decrease across the choke and flowrate, the choke valve represents a viable option for flow estimation. Therefore, deeming a choke valve model as a basic form of a VFM is suitable because it approximates the flow instead of precisely calculating it. Measuring flow across the choke presents challenges due to the complexity of multiphase flow (Bikhmukhametov, T., & Jäschke, J., 2020).

Similar to numerous other applications of fluid dynamics, empirical correlations were initially utilized for flow estimation through a choke. Gilbert (1954) proposed one such empirical model. Mechanistic models, however, were incorporated as research progressed on and are nowadays employed in VFM (Bikhmukhametov, T., & Jäschke, J., 2020).

Four models employed in industrial VFM solutions have been identified in the currently available published information. Nevertheless, it is probable that more models are in use but are not publicly disclosed by VFM suppliers. Modified Bernoulli, Hydro (Long and Short) and Perkins are some of these models (Bikhmukhametov, T., & Jäschke, J., 2020). Yet, there are not a lot of publications from VFM suppliers on this subject.

3.4.2.2.3.1 Modified Bernoulli

In the oil and gas sector, the Modified Bernoulli model is a widely used model for evaluating the flowrate of multiphase fluids flowing through a choke valve. One such system that uses this model is FieldWatch, a commercial Virtual Flow Meter system from Roxar.

The Modified Bernoulli model is a modification of the Bernoulli equation, which describes the relationship between pressure, velocity and height of an incompressible fluid

and is a fundamental equation in fluid mechanics. Nevertheless, because the Bernoulli equation was first established for single-phase flow, modifications are necessary to take many phases into account.

The Modified Bernoulli model adds two new parameters—the mixture density and the choke discharge coefficient—to account for the multiphase flow. The mixture density is the density of the multiphase fluid flowing through the choke, whereas the choke discharge coefficient is a dimensionless factor that quantifies the efficiency of the choke in converting pressure drop to flowrate. The Modified Bernoulli model is expressed as follows (Equation 3.17):

$$\dot{m} = A_1 C_D \left[\frac{2 \Delta p \rho_m}{\frac{A_1}{A_2} - 1} \right]^{\frac{1}{2}} \quad \text{Equation 3.17}$$

where C_D stands for the choke discharge coefficient and \dot{m} stands for the mass flowrate. A_1 denotes the inlet choke area, A_2 denotes the choke throat area, ρ_m signifies the fluid mixture density and Δp denotes the pressure drop over the choke (Bikmukhametov, T., & Jäschke, J., 2020).

3.4.2.2.3.2 Hydro (Long and Short)

The Selmer-Olsen-created Hydro model, a choke model used in the oil and gas industry's OLGA Online Virtual Flow Meter system, was created in 1995. There are two variations of the Hydro model, Long and Short, which differ in their assumptions regarding the location of the vena contracta, which is the point of maximum constriction in the flow. The critical flow is the largest possible flowrate, hence both versions of the Hydro model compute sub-critical and critical flows before choosing the smaller of the two. The Hydro model does not necessitate the use of a discharge coefficient in its calculations, in contrast to many other choke models, and takes into account permanent choke losses mechanistically. The works of Schüller et al. (2003) and Sampath et al. (2006) provide a thorough explanation of the whole Hydro model, including its derivation, an improved slip relation and testing outcomes using experimental data (Bikmukhametov, T., & Jäschke, J., 2020).

In their 2006 study, Schüller et al. observed that the Hydro models demonstrate outstanding accuracy in the estimation of mass flow rates in both subcritical and critical flow conditions. These models achieved an average absolute error of only 6.2% and a standard

deviation of 8.9%. In contrast, models developed by Sachdeva et al. in 1986 and Perkins in 1993 exhibited inferior performance, with average errors exceeding 22% and standard deviations surpassing 25% (Schüller et al., 2006). Furthermore, the Hydro models can anticipate the transition from subcritical to critical flow conditions with high accuracy, indicating their effectiveness in capturing complex flow behavior (Schüller et al., 2006).

3.4.2.2.3.3 Perkins

The Rate&Phase software uses the Perkins model (Perkins, 1993) for predicting mass flowrate through chokes. This model is based on the energy equation for a control volume of fluid and computes mass flowrate for both sub-critical and critical flow conditions. However, the actual flowrate is obtained by multiplying the calculated value with a discharge coefficient (Perkins, 1993; Bikmukhametov, T., & Jäschke, J., 2020). Unlike the Hydro model, the Perkins model does not take into account slip effect between phases and frictional losses in the throat. Sampath et al. (2006) noted that these limitations are a drawback of the Perkins model and that the Hydro model surpasses it by considering the slip effect. K-Spice Meter software offers the Hydro as well as the Perkins model, giving users the flexibility to select their desired alternative (Kongsberg, 2016; Bikmukhametov, T., & Jäschke, J., 2020).

Moreover, a multitude of non-commercial simulation models are available for estimating mass flowrates through choke systems. Ashford's model, developed in 1974, relies on fluid properties, choke size and the discharge coefficient to determine the total mass flowrate. This, in turn, allows for the estimation of the oil flowrate based on Black Oil properties (Bikmukhametov, T., & Jäschke, 2020).

In 1986, Sachdeva et al. introduced a two-phase flow model, which assumed a no-slip frozen flow condition and utilized the discharge coefficient for flowrate adjustment. However, a study conducted by Sampath et al. in 2006 revealed that this model exhibited lower accuracy compared to the Hydro model, primarily due to its reliance on the no-slip assumption. Nevertheless, Sachdeva et al.'s (1986) model continues to be widely employed as an early pioneer among mechanistic choke models, serving the purpose of enabling analysis and comparison with more recent models (Bikmukhametov, T., & Jäschke, 2020).

In 2009, Al-Safran and Kelkar proposed a choke model that considers slip between phases. Their objective was to achieve a balance between simplicity, reminiscent of the Sachdeva and Perkins models and accuracy on par with the Hydro model. This model is rooted in the foundational concepts of Sachdeva and Perkins models, incorporating the slip model originally proposed by Schüller et al. in 2003 into a modified version of the Hydro

model. The model's performance was evaluated by Sampath et al. in 2006 and experimental results demonstrate that the Al-Safran and Kelkar model outperforms the Sachdeva and Perkins models, leading to an average percent error reduction of 5-10% (Bikmukhametov, T., & Jäschke, 2020).

Since there are numerous choke models available for estimating flowrate, aside from the models discussed above, there are other general models that have been employed for real field cases but are not as well-known.

3.4.2.2.4 Electric submersible pump (ESP) model.

In the oil and gas sector, electric submersible pumps (ESPs) are often employed artificial lift systems to produce oil when natural production is not feasible because of factors including low bottomhole pressure, liquid loading, or the presence of heavy oil. As described in a review by Lea and Bearden, ESPs have been used extensively both onshore and offshore (1999). Given to its widespread use, there have been various attempts to develop a first-principles model that explains how ESP functions. Establishing a correlation between the rise in pump pressure, pump inlet pressure, flow rate and pump speed is the basic idea underlying these models (as described by Schlumberger Limited, 2014 and Bikmukhametov, T., & Jäschke, 2020) (Equation 3.18).

$$\Delta P = f(q, \xi, \alpha_1, P_{inlet}) \quad \text{Equation 3.18}$$

where q stands for flowrate, ξ for pump speed, α_1 for liquid fraction, and P_{inlet} for pump inlet pressure (Bikmukhametov, T., & Jäschke, 2020).

ESP models can be used to estimate the flowrate of a multiphase flow mixture pumped by an electric submersible pump (ESP) by assessing pressure pre and post pump and applying a model similar to the one described above. However, grouping ESP models can be challenging since some VFM suppliers withhold information on the precise models they employ. Yet, it is worth mentioning that such setups employ ESP models and can provide a helpful tool for estimating multiphase flowrates eliminating the requirement for extra equipment placement (Bikmukhametov, T., & Jäschke, 2020).

3.4.2.3 Data validation and reconciliation (DVR) algorithm

The DVR algorithm is a critical component of contemporary first-principles VFM systems. Some literature and software refer to this algorithm as the "optimization algorithm" (Melb et al., 2003; Heddle et al., 2012; Holmås and Løvli, 2011; Bikhmukhametov, T., & Jäschke, J., 2020), but other sources may refer to it as the DVR (van der Geest et al., 2001; Haouche et al., 2012a; Patel et al., 2014). In essence, the DVR process involves changing the VFM's variables to ensure that its outputs align with the recorded field data (Bikhmukhametov, T., & Jäschke, J., 2020).

Implying its name, the data validation and reconciliation (DVR) algorithm is composed of two main stages: validation and reconciliation. During the data validation stage, the primary objective is to eliminate data that is inaccurate data and data with interference. To accomplish this objective, analytical methods and data filtering techniques are employed. These techniques may include the use of exponential filters or moving averages (Stanley, 1982; Bikhmukhametov, T., & Jäschke, J., 2020).

After the data validation stage, the reconciliation phase is executed. During this phase, an optimization algorithm is utilized to fine-tune various model parameters, including flow rates, choke discharge coefficients, gas and water percentages, as well as friction and heat transfer coefficients. Its objective is to ensure that the VFM's results are consistent with the validated recorded data, while also adhering to the specified process conditions, such as mass balances (Câmara et al., 2017; Bikhmukhametov, T., & Jäschke, J., 2020). Typically, the reconciliation algorithm in VFM systems is formulated in the form of constrained least-squares, as described in Petukhov et al. (2011).

$$\min_x \sum_i^N \left(\frac{y_{meas_i} - y_{predicted_i}}{\sigma_i} \right)^2 \quad \text{Equation 3.19}$$

Constrained by the conditions below:

$$F(s, y) = 0 \quad \text{Equation 3.20}$$

$$y_{min} \leq y_{predicted_i} \leq y_{max} \quad \text{Equation 3.21}$$

$$s_{min} \leq s_i \leq s_{max} \quad \text{Equation 3.22}$$

The equation put forth by Petukhov et al. (2011) (Equation 3.19) involves several variables, where i represents the index of a specific measurement, y_{measi} represents the corresponding measured value and $y_{\text{predicted}i}$ denotes the predicted (reconciled) value. Additionally, the measurement uncertainty is represented by σ_i , while s_i refers to any unmeasured variables that may be affecting the process. Finally, the process equality constraints, which are typically related to factors such as mass and energy balances, are represented by $F(s, y) = 0$ in the equation.

When dealing with VFM applications, the problem at hand often involves non-linear formulations because of the system's intricate nature. As a result, a variety of methods may be utilized in order to obtain a solution to the non-linear data reconciliation problem. In cases where there are no inequality constraints, one commonly used approach is the method of Lagrangian multipliers, which can yield the solution (Camara et al., 2017; Bismukhametov, T., & Jäschke, J., 2020). However, when inequality constraints are present, gradient-based optimization techniques like Levenberg-Marquardt, SQP or Gauss-Newton are typically employed to obtain a solution (Holmas and Lovli, 2011; Camara et al., 2017; Bismukhametov, T., & Jäschke, J., 2020). Ultimately, the algorithm is able to estimate the flowrates that lead to either a local or global minimum error.

After the completion of the reconciliation process, the outcomes are subjected to a validation step. In this stage, statistical tests are carried out to identify any potential measurement or estimation errors and assess the likelihood of gross errors. To achieve this, individual tests such as penalty tests and global tests like chi-squared tests are conducted (Petukhov et al., 2011; Bismukhametov, T., & Jäschke, J., 2020). These tests help to additionally identify untrustworthy measurements and estimations and ensure the final results' accuracy.

3.5 Data-driven VFM

Data-driven modeling involves the examination of the system's data and establishing connections among its input and output parameters, all while not necessitating a comprehensive understanding of its underlying physical behavior (Solomatine et al., 2009; Bismukhametov and Jäschke, 2020). The primary benefit of the method is that the necessity for a comprehensive physical representation of a system's mechanisms or workflows, which can pose mathematical challenges, is circumvented. This is exemplified in the case of multiphase flows in pipelines. Data-driven approaches operate under the assumption that experimental or industrial data accurately describe the system and strive to deduce its

underlying physical relationships directly from this information (Bikmukhametov, T., & Jäschke, J., 2020). Machine learning models such as Neural Networks (Ahmadi et al., 2013; Andrianov, 2018; Grimstad et al., 2021) and ensemble learning (AL-Qutami et al., 2018; Bikmukhametov and Jäschke, 2019) have been tested in data-driven VFMs, which use historical production data to enable machine learning and regression model generation. Shell has developed a data-driven VFM called 'Production Universe' (PU) (Poullisse et al., 2006).

Disregarding the physical interconnections among the data points, the data-driven VFM directly establishes a mathematical relationship between key parameters, such as temperature and pressure within the pipeline and the flow rate (Andrianov, 2018; Song et al., 2017; Bikmukhametov & Jäschke, 2020; Song et al. 2022). Garcia et al. (Garca et al., 2010) commercialized data-driven VFMs in oil fields in 2010, with a maximum relative error of 5%. Scientists have preferred neural networks in the evolution of data-driven VFMs (Qiu and Toral, 1993; Ahmadi et al., 2013; Song et al., 2017; Song et al. 2022). In 1993, Qiu and Toral (Qiu and Toral, 1993) established a VFM method by training Back Propagation (BP) neural networks (Song et al., 2022). An ICA-ANN model was suggested by Ahmadi et al. (Ahmadi et al., 2013). In this model, an imperialist competitive algorithm was used to optimize neural network connection weights. As per the research conducted by Al-Qutami et al. (2018), the gas flow rate estimation is highly affected by the parameters related to the oil nozzle, while the crude oil flow rate prediction is most influenced by the well bottom pressure parameter. Furthermore, in data-driven VFMs, the Long Short-Term Memory (LSTM) network is frequently utilized (Andrianov, 2018; Sun et al., 2018; Song et al., 2022).

Despite the enhanced precision of data-driven VFMs when compared to first principles VFMs, their adaptability and robustness still fall short for a wide range of industrial implementations. Consequently, existing data-driven VFMs are seldom implemented in real-world scenarios. (Al-Qutami et al., 2017; Omrani et al., 2018; Song et al., 2022). Limited data amount and diversity as well as substandard data quality are commonly blamed for degrading the efficacy of data-driven VFMs (Qiu and Toral, 1993; Ahmadi et al., 2013; Song et al., 2022). Moreover, conventional data-driven VFMs often lack broad applicability to other wells, which is commonly the primary cause for their underutilization.

To begin the modeling process, the initial step involves the collection of data. This encompasses various aspects of the process, including the system's past records, its present state, or even information from a related system with past documentation. Following this, the subsequent step involves the preliminary preparation of the information or data pre-processing. This could include a variety of operations. To begin, researchers ensure the

appropriateness of the collected data for modeling purposes through procedures that involve eliminating irregular data points, handling data gaps and minimizing interference (Bikmukhametov, T., & Jäschke, J., 2020). Data transformation and feature engineering can also provide enhanced understanding into the information contained in data. The term "feature engineering" is derived from the practice of often referring to input data as "features" within the machine learning community, so feature engineering involves the manipulation of input data with the aim of revealing valuable insights that can enhance the model training phase (Bikmukhametov, T., & Jäschke, J., 2020).

After pre-processing the data, the model is developed. This phase involves the creation and training of a data-driven model using the pre-processed data. The training procedure involves adjusting a mathematical function to precisely represent the data. In certain instances, this function may take on an analytical form, as in the case of a linear regression model. In other situations, it can be represented as a black-box model, such as a neural network (NN). In order to verify the model's capacity to produce precise forecasts for unseen data, it must undergo validation using an independent dataset. (Bikmukhametov, T., & Jäschke, J., 2020). Only after the validation process has been successfully completed, the model can be employed for predictions with recently collected data.

Collected data within a data-driven VFM system typically encompasses information about bottomhole and wellhead pressures and temperatures, specifications for choke opening, ESP parameter configurations and related records of oil, gas and water flow rates. (Bikmukhametov, T., & Jäschke, J., 2020). Flowrate data can be collected through various methods. One option is to utilize well testing data, while another approach is to utilize data obtained from MPFM. When physical multiphase flow meters are in place at every wellhead, the data-driven model serves as a backup metering system for the well (Bikmukhametov, T., & Jäschke, J., 2020). However, when a single physical multiphase flow meter is implemented to monitor a group of wells, its data is utilized in a manner similar to how well test and separator data are used. This facilitates the acquisition of flowrate measurements from each well following the established well testing schedule. In this scenario, the data-driven model functions as an independent VFM system after training and validation (Bikmukhametov, T., & Jäschke, J., 2020).

3.6 Data-driven VFM components and applied methods

3.6.1 Data collection and pre-processing

In order to develop a data-driven model, the initial step involves collecting and pre-processing the relevant data. For Virtual Flow Metering systems, this could include sensor readings obtained from various sources like wells and processing facilities. Furthermore, past measurements pertaining to akin wells or fields can contribute to model refinement. After collecting the data, the following procedure is to pre-process it before beginning the training process. Usually, the data collected is noisy, can be corrupted and may contain data gaps, anomalies, or unnecessary inputs (Famili et al., 1997; Bismukhametov, T., & Jäschke, J., 2020). Therefore, it is crucial to clean and validate the data before putting it to further use. Essentially, this validation process is similar to the one carried out by the DVR algorithm, that is typically utilized in the physics-driven VFM systems, as previously stated.

During data pre-processing, it is possible to transform the data and extract additional insights. This is often known as feature engineering. In the case of VFM, the raw features typically include pressure and temperature readings at different locations within the production system, choke openings and ESP data. Various techniques can be applied in

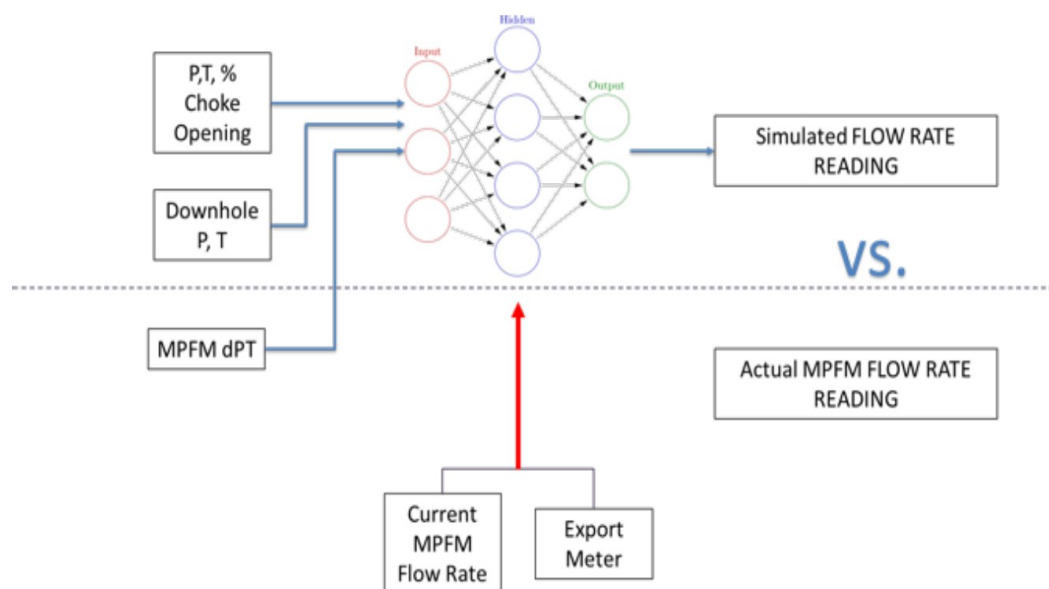


Figure 3.2 Data driven Virtual Flow Meter (VFM) Ishak, M. A. B., Ismail, I. B., & Al-Qutami, T. A. H. (2021, July). Data Driven Versus Transient Multiphase Flow Simulator for Virtual Flow Meter Application. In 2020 8th International Conference on Intelligent and Advanced Systems (ICIAS) (pp. 1-4). IEEE.

feature engineering, such as Principal Component Analysis (PCA) for dimensionality reduction, feature selection techniques and combinations of original features using both linear

and non-linear methods. (Bikmukhametov, T., & Jäschke, J., 2020). Feature engineering is a significant step in data pre-processing because it can assist the data-driven algorithm in discovering intricate interdependencies between the input data and the output parameters while also removing unnecessary features. This may result in more cost-efficient computing throughout the learning and prediction phases. Creating informative features for training algorithms requires a deep understanding of the specific field and VFM is no different in this regard. Constructing useful features that characterize the procedure of transporting multiphase flows from the input data has the potential to enhance prediction accuracy. (Bikmukhametov, T., & Jäschke, J., 2020). Nevertheless, in most cases, sensor data from the production system is used without further feature engineering and the opportunities for fully exploring effective feature engineering in Virtual Flow Metering implementations has not been realized in the literature (Bikmukhametov, T., & Jäschke, J., 2020).

3.6.2 Model development

The procedure of model development entails crafting an algorithm with the ability to accurately link input features to output (target) parameters. This procedure, commonly referred to as training or learning, entails the adjustment of the algorithm's parameters to enhance the accuracy of its target variable estimations. The choice of which parameters to adjust is determined by the algorithm used. In neural networks, for instance, it is typically the weights connecting the neurons that are adjusted, while in regression trees, the adjustable factor is the tree depth. The training procedure is designed to minimize a cost function, characterized as the disparity between the algorithm's forecasts and the actual values as outlined by Bikmukhametov, T., & Jäschke, J. (2020). The cost function commonly employed for Virtual Flow Metering regression problems is the mean squared error (MSE), which is expressed in the following form (Equation 3.23):

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_{measi} - y_{predicted_i})^2 \quad \text{Equation 3.23}$$

where the cost function is represented by MSE, the measured (true) value of the i-th training example is denoted as y_{measi} , the predicted value of the i-th training example is represented as $y_{predicted_i}$, the number of training examples is denoted as N and the training example index is indicated as i (Bikmukhametov, T., & Jäschke, J., 2020).

This equation bears a resemblance to the data reconciliation algorithm employed in physics-driven VFM systems, as both aim to reduce the disparity between estimated and measured values. While data-driven VFM models typically involve unconstrained optimization, data reconciliation incorporates constraints and accounts for uncertainty in the cost function. Both physics-driven and data-driven VFM systems share the underlying concept of adjusting model variables to minimize the disparity in predicted and measured values. The key distinction lies in their mathematical methodology: physics-driven models utilize physical principles to describe the procedure of transporting multiphase flows, while data-driven models derive insights about multiphase flow behavior straight from data (Bikmukhametov, T., & Jäschke, J., 2020).

After a model has undergone training, there arises a necessity to conduct validation and testing on a separate dataset to verify its capability for effective generalization to new unseen data. Generalization performance refers to the model's ability to have accurately predict outcomes on new data (Abrahart et al. (2008); Bikmukhametov, T., & Jäschke, J., (2020)). Validation also encompasses a process for the selection of suitable hyperparameters that enhance the model's alignment with the data. These hyperparameters constitute pre-training settings that are not acquired throughout training phase. Neural network hyperparameters include elements such as the number of layers, the quantity of nodes in the hidden layers and regularization parameters, among others. Notably, regularization parameters stand out as essential hyperparameters capable of mitigating the influence of interference and outliers in model predictions, thereby preventing overfitting (Bikmukhametov, T., & Jäschke, J., 2020).

There are several validation methods available, with standard K-fold cross-validation standing out as one of the most frequently employed techniques (Hastie et al., 2009). This method involves the division of the existing data into training and test sets. Subsequently, the training set is further partitioned into K subsets or K-folds. A predetermined set of hyperparameters is chosen prior to the training phase and the model is trained on K-1 folds using these hyperparameters. The evaluation of the model's performance is accomplished by assessing the error between the actual and predicted values on the remaining fold (Bikmukhametov, T., & Jäschke, J., 2020).

This procedure is iterated K times, with each iteration involving training on K-1 folds and evaluating on the one left out, yielding K distinct evaluations. The resulting error from these evaluations is calculated as an average over the K folds, providing a representative measure of model performance with the selected hyperparameters. Following this, the hyperparameters can be modified and the process of computing the average error over K

folds is repeated. The optimal hyperparameters are determined as those that minimize the error across the K folds. Upon identifying the optimal hyperparameters, the model can be retrained using these settings on the entire training set (Bikmukhametov, T., & Jäschke, J., 2020). This final training step ensures that the model makes the best possible use of the available data, taking into account the selected hyperparameters.

The K-fold cross-validation method [Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής \(left\)](#) assumes that data points are independent of one another, which is not always true in Virtual Flow Metering. Bottomhole pressures, for instance, may be affected by conditions at a previous time, unless there are significant time intervals or steady-state operations with no pressure fluctuations. Despite this limitation, most data-driven VFM models have used standard K-fold cross-validation without taking this into account (Al-Qutami et al., 2017c, 2017a; Bikmukhametov, T., & Jäschke, J., 2020).

Nested K-fold cross-validation [Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής \(right\)](#) can be used instead of standard K-fold cross-validation. The training set is divided into K-folds in this approach and the model undergoes training and validation in a nested form. As an illustration, the model might undergo training using data from fold 1 and validation using data from fold 2, followed by training on data from folds 1 and 2 combined and validation using data from fold 3. This methodology ensures that the model is not exposed to future data to predict past results, thereby avoiding the possibility of biases and ensuring accurate and trustworthy assessments about its efficiency (Bikmukhametov, T., & Jäschke, J., 2020).

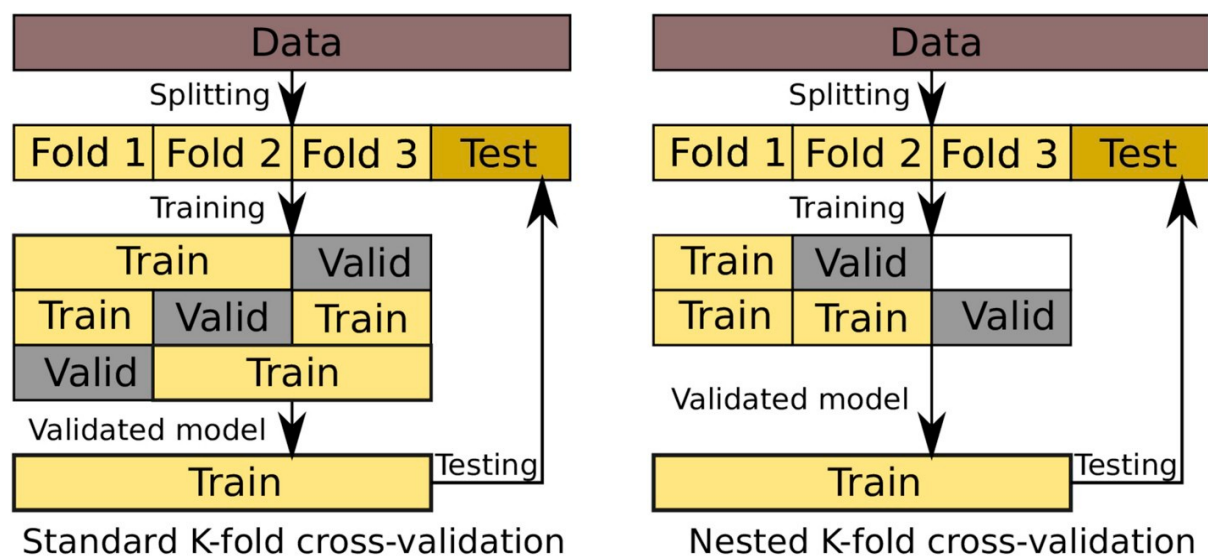


Figure 3.3 Standard (left) and nested (right) K-fold cross-validation schemes for data-driven models. Timur Bikmukhametov, Johannes Jäschke 2020, First Principles and Machine Learning Virtual Flow Metering: A Literature Review P.13



To reach conclusions about model generalization, the model's efficiency after validation should ideally be tested on a separate test dataset. When testing the model's performance, one of two things can happen:

- The training and test sets exhibit notable errors
- The training set error is minimal, whereas the test set error is significant

The initial condition is commonly termed underfitting and it occurs when the trained model displays significant bias. Conversely, the second scenario is referred to as overfitting, which arises when the trained model exhibits a substantial amount of variance. In practice, the main purpose of data-driven model training is to find the right equilibrium between bias and variance, with the aim of achieving the optimal point known as the bias-variance trade-off (Hastie et al., 2009; Bikhmukhametov, T., & Jäschke, J., 2020). The goal of validation and testing is to find the best set of hyperparameters for achieving the best bias-variance trade-off. A well-generalized algorithm with a good bias-variance trade-off can make accurate and reliable predictions on new data.



Figure 3.4 Workflow for data driven VFM Ishak, M. A., Hasan, A. Q., Aziz, T., Ellingsen, H., Ruden, T., & Khaledi, H. (2020, October). Evaluation of Data Driven Versus Multiphase Transient Flow Simulator for Virtual Flow Meter Application. In Offshore Technology Conference Asia. OnePetro

The early stopping method, widely employed in data-driven training scenarios, offers an alternative technique for enhancing model performance, even in VFM applications (Prechelt, 2012; Al-Qutami et al., 2017b; Bikhmukhametov and Jaschke, 2019; Bikhmukhametov, T., & Jäschke, J., 2020). The method involves partitioning the data into training, validation and test sets. Throughout the training process, the model's performance on both the training and validation sets is closely monitored. Training continues to the point where there is a noticeable sign of the model's performance deteriorating on the validation set, typically due to overfitting. When this happens, the model is retrained using a combination of the training and validation datasets to improve its generalization capabilities. Finally, the model is assessed on the test set to ensure that it can perform well on entirely new, unseen data. (Bikhmukhametov, T., & Jäschke, J., 2020).

3.7 Comparison of First principles with the Data driven VFMs

First-principles VFMs	Data-driven VFMs
ADVANTAGES	
Tested, tried and proven across industries Operational experience is relatively long	It is not necessary to have a thorough understanding of the of the system's physical principles

Reliable over the lifetime of an asset, can predict future events and be utilized to simulate other challenges encountered during well operation such as slugging and erosion	Can make use of all available sensor data
Suitable for predicting unobserved variables beyond the scope of available data	Cost-effective computing
Several suppliers are accessible	Easy to update continuously with newly obtained data
The uncertainty in the models has been extensively studied and can be taken into account during design and operation	Convenient amalgamation of diverse features from separate sections of the production system without the use of a complicated physical system
DISADVANTAGES	
May experience a lag in capturing dynamic scenarios	The majority of approaches are in steady state
Is significantly influenced by PVT data precision	Inadequate for scenarios with a shortage of past data
Substantial computational expense	Limited operational experience
Commercially available simulation software typically entails considerable expenditure	Functions efficiently with data falling within or near the parameters of the training dataset; if not, it must be fine-tuned and retrained
Involves a comprehensive understanding of the system's physical principles	In-depth feature engineering relies on a deep understanding of the procedure

Table 3.1 Comparison of First principles with the Data driven VFMs, Timur Bismukhametov, Johannes Jäschke 2020, First Principles and Machine Learning Virtual Flow Metering: A Literature Review P.13

Chapter 4 Machine learning approaches for VFMs

Chapter 3 lays the foundation for the subsequent development of the data-driven Virtual Multiphase Flowmeter of this thesis detailed in chapter 4. It comprehensively examines various machine learning approaches tailored to achieve the overarching objective of developing VFM systems. The chapter begins with an assessment of supervised and unsupervised machine learning methods for VFMs and then delves into supervised learning models, specifically emphasizing artificial neural networks (ANNs). ANNs are extensively discussed for their popularity in virtual multiphase flowmeters, showcasing their ability to

comprehend intricate relationships within multiphase flow data. The examination considers both steady-state and dynamic conditions, recognizing their significance in industrial processes and anticipating diverse operational scenarios confronted by VFMs.

The comprehensive analysis of diverse machine learning algorithms underscores the importance of identifying the most suitable approach for accurate and reliable virtual multiphase flow measurements in various operating conditions. Subsequent chapters will leverage this foundational study to systematically develop and evaluate the envisioned data-driven Virtual Multiphase Flowmeter.

4.1 Supervised and Unsupervised Machine Learning

To further explain the machine learning methods used in virtual multiphase flowmeters a brief reference to their basic definitions should be given. To begin, a comparison between supervised and unsupervised machine learning should be made.

Machine learning is a subset of artificial intelligence that aims to address problems by analyzing past or historical data (Libbrecht et al., 2015). Machine learning, as opposed to conventional AI applications, entails finding hidden patterns in data (data mining) and utilizing those patterns to classify or forecast future instances of the problem (Alpaydin, 2014). In essence, machine learning algorithms provide intelligent machines the information they need to function effectively. These algorithms are integrated into machines, which are then fed with data streams, allowing them to extract and use information more efficiently. Although all artificial intelligence approaches are machine learning algorithms, not all artificial intelligence techniques can be categorized as machine learning algorithms (Berry et al., 2019).

Machine learning algorithms are broadly classified into supervised and unsupervised types, although some experts also consider reinforcement learning as a distinct type. However, the majority of literature recognizes supervised and unsupervised machine learning. The primary difference between these two classes is the presence or absence of labeled data in the training subset (Berry et al., 2019).

Supervised learning is a machine learning technique that relies on labeled datasets to train algorithms. These datasets are used to supervise the algorithm's classification or prediction accuracy. The model can learn from these labeled inputs and outputs over time. According to Kotsiantis (2007), supervised learning involves predetermined output attributes along with input attributes (Berry et al., 2019). The algorithms predict and classify the predetermined attribute and their accuracy and misclassification rates are dependent on the

correct prediction or classification of the predetermined attribute. The learning process terminates when the algorithm achieves an acceptable performance level (MathWorks, 2016).

As per Libbrecht and Noble (2015), supervised algorithms first perform analytical tasks using the training data and then create contingency functions for mapping new instances of the attribute. It is essential to pre-specify the maximum settings for desired outcomes and performance levels (Libbrecht et al., 2015, MathWorks. 2016). The supervised learning algorithms are further divided into classification and regression algorithms (Alpaydin, 2014; Kotsiantis, 2007).

- **Classification** algorithms are used to accurately categorize test data into specific groups or categories. Common types of classification algorithms include linear classifiers, decision trees, support vector machines, and random forests.
- **Regression** is another supervised learning technique that helps to understand the relationship between dependent and independent variables. Regression models can be used to predict numerical values based on different data points. Regression algorithms that are often used are logistic, polynomial, and linear.

Artificial neural networks (ANNs) and SVMs can be used for both regression and classification tasks in machine learning.

On the other hand, unsupervised learning is a type of machine learning that involves the use of algorithms to analyze and cluster data sets that are not labeled. These algorithms can find hidden patterns in data without the need for human interaction. Unsupervised machine learning involves pattern recognition without the use of a target attribute. In other words, all variables in the analysis are treated as inputs, making the techniques well-suited for clustering and association mining. According to Hofmann (2001), unsupervised algorithms can be used to create labels in the data, which can then be utilized in supervised learning tasks. There are three main tasks that unsupervised learning models are commonly used for: clustering, association and dimensionality reduction.

- **Clustering** is a method employed in data mining that involves categorizing unlabeled data based on their similarities or differences. The K-means clustering algorithm is an example of a technique that can group data points with similar features into clusters, where the K value sets the size and precision of the clusters. Clustering algorithms discover natural patterns within unlabeled data and then allocate labels to each data point. This approach is documented in studies such as Dougherty et al. (1995) and Marshland (2015).

- **Association** refers to the use of various rules to discover connections between variables within a dataset. These algorithms search for rules that provide an accurate representation of the relationships between attributes.
- **Dimensionality reduction** is a machine learning method employed when a dataset has too many features (or dimensions) to manage. It compresses the data inputs to a manageable size while maintaining data integrity. This technique is commonly used in the data pre-processing stage, such as when autoencoders remove visual noise to enhance picture quality.

Several machine learning approaches are often utilized to train the VFM model for the data-driven VFM using previous well test data. Regarding the virtual multiphase flowmeters, supervised learning techniques can be applied to them to improve accuracy in predicting flow rates in oil and gas production. By training algorithms on labeled data, such as historical flow rate measurements, the model precisely forecasts the flow rates of different phases in real-time (Góes et al., 2021). This approach allows for better monitoring of production and can aid in decision-making processes. Common supervised learning techniques used in virtual multiphase flowmeter applications include artificial neural networks, support vector machines and decision trees.

Unsupervised learning can also be applied in virtual multiphase flow metering. By analyzing large amounts of unlabeled data, unsupervised machine learning algorithms can discover hidden patterns in the data and identify similarities or differences in the multiphase flow behavior. This can be used to cluster different flow regimes, which can then be used to predict flow rates of oil, gas and water in virtual multiphase flow metering. Additionally, unsupervised learning can also be used for dimensionality reduction, which reduces the number of features in the data and improves the efficiency and accuracy of the virtual multiphase flow metering process.

For the purposes of this thesis, supervised machine learning methods for virtual multiphase flowmeters will be further explained.

4.2 Implemented techniques for data-driven VFM systems

In this section, there will be examined the various data-driven techniques that have been utilized in VFM systems. Given that VFM entails non-linear regression, the majority of data-driven methods in Virtual Flow Metering rely on ANN, either individually or in combination with adjustments such as ensemble algorithms (Bikmukhametov, T., & Jäschke, J., 2020). As a result, ANN-based VFM approaches will be examined in greater detail compared to other

machine learning methods. Additionally, recurrent neural networks capable of representing dynamic problems will be analyzed separately.

4.2.1 Artificial neural network VFM solutions

4.2.1.1 Steady state artificial neural network VFM solutions.

Artificial Neural Networks (ANNs) are a popular type of machine learning algorithm that is widely utilized in engineering to address nonlinear problems (Shahbaz et al. 2019). However, they can be considered black boxes because of the hidden layers of regression-like computations (Wood 2018; Barjoui et al., 2021). Two of the most common ANN algorithms are the feed-forward neural network (FFNN), which has a single hidden layer and the multiple hidden layers perceptron's (MLP) (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής). A single-hidden-layer ANN was developed by Barjoui et al. (2021) to predict the two-phase flow rate (Q_1) through a wellhead choke.

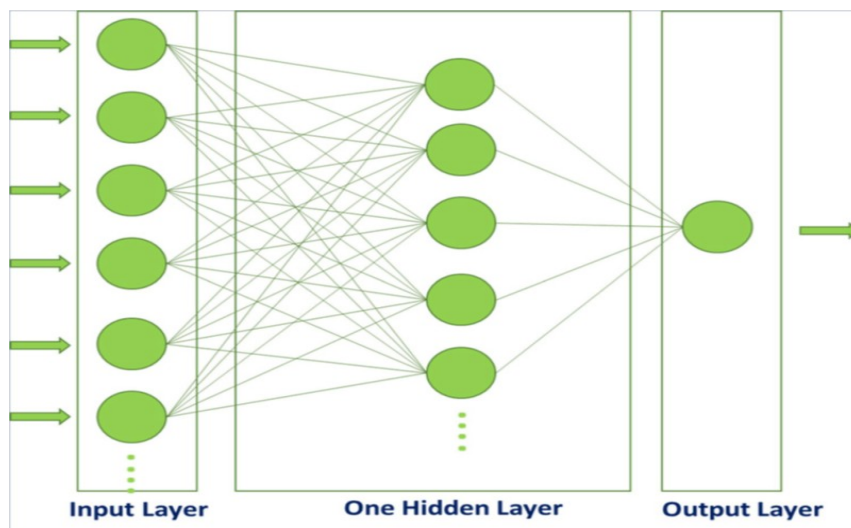


Figure 4.1 Indicative structure of a feed-forward artificial neural network (ANN) with a single hidden layer with multiple neurons Barjoui, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. Journal of Petroleum Exploration and Production, 11, 1233-1261.

Multilayer Perceptrons (MLPs) are a form of artificial neural network (ANN) that can approximate complex functions using a fixed amount of input data. They do not have feedback connections from inputs to outputs (Bikmukhametov, T., & Jäschke, J., 2020). They are modeled after the structure and functionality of biological neural networks, with interconnected nodes organized into layers. Typically, MLPs consist of an input layer, one or more hidden layers and an output layer (Goodfellow et al., 2016). To train and make

predictions using MLPs, feature values must be inputted into the input layer. In a VFM system, these features may include pressure and temperature measurements, choke opening, or other production system characteristics (Bikmukhametov, T., & Jäschke, J., 2020). The hidden layers are responsible for transforming the input features through non-linear operations and approximating the function that describes the system behavior. Finally, the output layer predicts output variables, such as flow rates, by applying an activation function to the data obtained in the hidden layer. MLPs are considered universal approximators because they can estimate connections and patterns between variables, making them a popular choice for VFM applications (Hornik et al., 1989). It's worth noting that MLPs are more suitable for steady-state solutions than transient flow behavior (Omrani et al., 2018).

In an ANN, the data transmitted from the neurons in a given layer to the neurons in the following layer is modified using weight and bias vectors. The neurons located in the hidden layer process the data and transfer processed signals forward to the output layer (Barjouei et al., 2021). These signals are adapted using an activation function. The equation below demonstrates the signal modifications that take place as data passes through the Artificial Neural Network.

$$y_j = f \left(\sum_{i=1} W_{ij} x_i + b_j \right) \quad \text{Equation 4.24}$$

Equation 4.24 represents the output signal generated by the j^{th} neuron in the hidden layer. The input signal x_i is multiplied by the weight W_{ij} and the hidden layer bias b_j is added to the result. The sum of these values is then passed through the activation function f , which generates the output signal y_j .

In an artificial neural network (ANN), the weights and bias values of the hidden layer (Barjouei et al., 2021) are initially assigned random values. To enhance its predictive performance, the ANN is trained using a backpropagation algorithm, which involves repeatedly presenting input data and comparing the output to the actual target values. The algorithm adjusts the weights and biases assigned to the hidden layer by minimizing a loss function that quantifies the discrepancy between the predicted and actual outputs.

The backpropagation algorithm computes the error gradient, which represents the rate of change of the loss function with respect to the weights and biases in the network. This gradient indicates the direction and magnitude of the adjustment needed to reduce the error in

the output of the ANN. Using the error gradient, the backpropagation algorithm updates the weights and biases in the network with the objective of minimizing the loss function.

Typically, the loss function used in the backpropagation algorithm is the mean squared error (MSE), which computes the average of the squared differences between the predicted and actual values for all data records in the training subset. The backpropagation algorithm minimizes the MSE by iteratively adjusting the weights and biases until the error is minimized. As a result, the ANN is trained to make accurate predictions on new, unseen data.

$$E_{MSE} = \frac{1}{m} \sum_i^m (\hat{y}_i - y_i)^2 \quad \text{Equation 4.25}$$

The Equation 4.25 for calculating the MSE (also mentioned in chapter 3.6.2), is shown above, where \hat{y}_i represents the measured value pertaining to the dependent variable for the i^{th} data record, y_i is the forecasted value of the dependent variable for the i^{th} data record and m represents the overall data record count in the training subset (Barjouei et al., 2021). The goal is to minimize the MSE, which reflects the average squared difference between the predicted and actual values of the dependent variable. By minimizing the MSE, the algorithm seeks to achieve the best possible fit between the predicted and actual values of the dependent variable.

Different optimization algorithms can be used in place of backpropagation in Artificial Neural Networks (ANN) to improve prediction accuracy and hasten convergence. Adam, RMSprop, Adagrad, Adadelta, Momentum and Nesterov Accelerated Gradient (Barjouei et al., 2021) are some of these algorithms. A flowchart illustrating the series of steps the ANN takes to produce its predictions for the dependent variable is shown in [Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής.](#)

Qiu and Toral (1993) were among the first to estimate multiphase flowrates using neural network models based on pressure sensor data. They utilized laboratory pressure transducers as inputs and estimated gas-liquid rates as outputs (Bikmukhametov, T., & Jäschke, J., 2020). Subsequently, there have been multiple reports of neural networks being employed for Virtual Flow Metering.



Figure 4.2 Flow diagram of single layer ANN Barjouei, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. *Journal of Petroleum Exploration and Production*, 11, 1233-1261.

Al-Qutami et al. (2018, 2017a, 2017c, 2017b) have significantly contributed to this field, as noted in Bikmukhametov and Jäschke (2020). In their 2017b research, Al-Qutami et al. employed a NN fine-tuned with the Levenberg-Marquardt optimization algorithm and applied the k-fold cross-validation approach to determine the optimal number of neurons for their model (Bikmukhametov, T., & Jäschke, J., 2020). The model's validation process utilized well test data spanning a 1.5-year period and they employed the early stopping technique to minimize overfitting, as discussed in Bikmukhametov and Jäschke (2020). Regarding the evaluation of the trained model on the test dataset, they performed a sensitivity analysis, revealing that the predicted gas flow rate was most influenced by the choke position, whereas

bottomhole pressure played a pivotal role in predicting oil flow rates within the context of VFM, as outlined by Bikhmukhametov and Jäschke (2020).

By integrating the outputs of many models or learners, ensemble learning is a prominent approach used in machine learning and data mining to increase the accuracy of predictions. Ensemble learning includes creating a group of models that are separately trained on the same dataset rather than depending on a single model to generate predictions. Each model in the ensemble has its own advantages and disadvantages and is meant to capture particular aspects of the data. Once the individual models have been trained, their predictions are integrated to provide an overall forecast that is more accurate. This may be accomplished using a variety of strategies, such as weighted averaging, in which each model's predictions are weighted based on their performance on a validation set, or by employing a voting mechanism, in which the ensemble output is the most often predicted class by the individual models.

Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής depicts the fundamental principle, with the ensemble consisting of numerous models whose predictions are merged to obtain greater performance.

In their study from 2017a, Al-Qutami and their research team introduced an innovative ensemble learning technique known as NN-RTE. This approach integrates the capabilities of neural networks and regression trees. The concept behind this strategy is to generate several learners, employ multiple learning algorithms (specifically neural networks and regression trees), subject them to a pruning process via simulated annealing and subsequently combine their predictions through straightforward averaging to yield the final output. Their research discussed by Bikhmukhametov and Jäschke in 2020 found strong evidence supporting the superiority of this hybrid approach over conventional ensemble methods that solely rely on neural networks and regression trees.

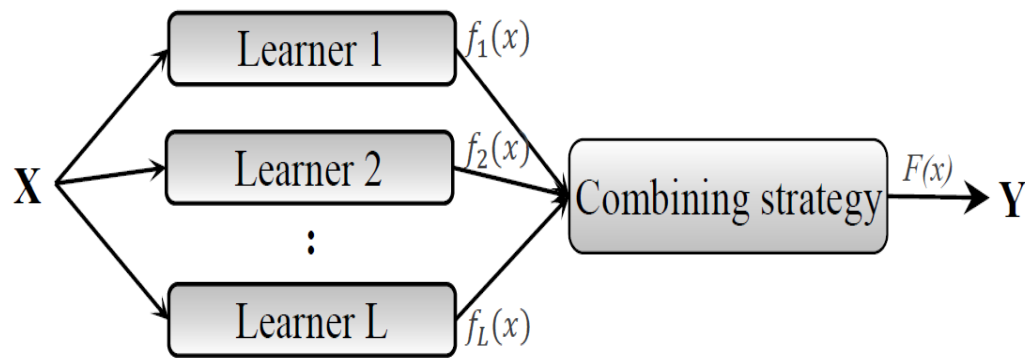


Figure 4.3 Ensemble learning structure Al-Qutami, T. A., Ibrahim, R., & Ismail, I. (2017, September). Hybrid neural network and regression tree ensemble pruned by simulated annealing for virtual flow metering application. In 2017 IEEE International Conference on Signal and Image Processing Applications (ICSIPA) (pp. 304-309). IEEE.

In a subsequent study in 2017c, Al-Qutami et al. delved into the optimization of a radial basis function network (RBFN) by introducing a Gaussian transfer function in the hidden layer, which deviated from the conventional sigmoid function (Bikmukhametov, T., & Jäschke, J., 2020). This alteration notably accelerated the network's training process. To train the network, they utilized the Orthogonal Least Squares algorithm, a method typically associated with RBFNs. By conducting a sensitivity analysis that excluded bottomhole pressure and choke opening as model inputs, the researchers made a significant discovery. Their analysis showed that bottomhole pressure did not impact the predictions, while the degree of choke opening played a pivotal role in the network's accuracy. This finding was in alignment with the results observed by Al-Qutami et al. (2017b) regarding gas rate (Bikmukhametov, T., & Jäschke, J., 2020). The researchers emphasized the necessity for further investigations to establish the reliability of sensitivity analysis in NN.

In their study, Al-Qutami and colleagues (2018) examined adjusted version of ensemble learning in contrast to their earlier work (2017b). This time, they utilized a neural network ensemble to introduce diversity by implementing several regularization criteria such as scaled conjugate gradient and Bayesian regularization. Moreover, they applied weighted averaging and neural network meta-learner combination techniques (Bikmukhametov, T., & Jäschke, J., 2020).

Omran and colleagues (2018) examined the usage of feedforward neural networks on both model-generated and factual field measurements. They initially applied a neural network to predict oil and gas flowrates during steady state conditions (Bikmukhametov, T., &

Jäschke, J., 2020), which demonstrated a high level of accuracy. However, the NN's performance during transient operation was proved to be less reliable. The researchers also performed sensitivity analyses on the target variables, which revealed that NNs can produce accurate flowrate predictions even in the presence of noisy input data, unless the uncertainty levels become very high. Lastly, the study suggested a back-allocation method for well flowrates utilizing flow measurements obtained from a separator (Bikmukhametov, T., & Jäschke, J., 2020), which exhibited acceptable performance and merits further examination.

In research by Alajmi and colleagues (2015), a NN was employed to forecast the oil flowrate through a choke. As compared to flowrate estimations based on empirical choke correlations, the NN performed much better. The NN utilized various input parameters, including pressure, temperature, choke size, water cut data and an empirical correlation for critical choke flow. In comparison to Flow predictions based on empirical choke correlations, the NN demonstrated significantly improved performance. Nevertheless, it is important to acknowledge, that the choke models utilized in this research were solely empirical lacking the fundamental mechanical principles that typically result in more precise forecasts, as discussed by Bikmukhametov and Jäschke (2020).

Berneti and Shahbazian (2011) and Ahmadi et al. (2013) conducted a study comparing the performance of conventional NN training with a hybrid method that involved optimizing the initial weights of the network using Imperialist Competitive Algorithm (ICA). Ahmadi et al. (2013) further investigated the use of Particle Swarm Optimization (PSO) and Genetic Algorithm in addition to ICA and also introduced Fuzzy Logic flow estimation. The results indicated that the NN with ICA had superior performance in comparison with the other hybrid methods and the conventional Neural Network training method (Bikmukhametov, T., & Jäschke, J., 2020).

In the study mentioned by Bikmukhametov and Jäschke in 2020, Zangl and colleagues (2014) utilized a NN to predict oil and water rates through multi-rate well tests. The network was trained utilizing gradient descent and it provided accurate predictions on a test dataset. In related research, Hasanvand and Berneti (2015) employed a three-layer FFNN to estimate oil flowrates. They utilized actual well testing measurements from 31 wells collected over an 8-year production period, as also mentioned in Bikmukhametov and Jäschke's study in 2020. Additionally, Xu et al. (2011) and Shaban and Tavoularis (2014) employed Principal Component Analysis (PCA) to derive input features from experimental datasets for their neural networks (Bikmukhametov, T., & Jäschke, J., 2020). This approach helped optimize

the network's performance, resulting in flow rate estimations that were consistent with the measured values.

Apart from academic research, Baker Hughes has created NeuraFlow software that employs a NN model for forecasting flowrates in systems employing electric submersible pumps (Denney et al., 2013; Baker Hughes, 2014; Bikmukhametov, T., & Jäschke, J., 2020). NeuraFlow employs the same neural network approach as previously stated, taking in input parameters including pump intake and discharge pressures, in addition to pump frequency, to provide flowrate estimations.

4.2.1.2 Dynamic artificial neural network VFM solutions

There are several neural network modifications that can effectively model dynamic transient phenomena, unlike the steady-state feedforward neural networks. Despite the emergence of sophisticated deep learning methods, ANN remains a prevalent baseline model for many forecasting tasks. However, the primary limitation of ANNs is that they only transfer information from input to output, leading to suboptimal performance in highly dynamic systems. Feedforward neural networks are generally limited to steady-state mapping since they rely exclusively on current data. In contrast, recurrent neural networks (RNNs) are better suited for transient data and have shown effectiveness in tasks such as voice identification and machine translation tasks (Graves et al., 2013; Bikmukhametov, T., & Jäschke, J., 2020). RNNs utilize past data to forecast the present target parameter. In virtual flow metering, RNNs leverage past pressure and temperature measurements to determine the current flow rate (Bikmukhametov, T., & Jäschke, J., 2020), while feedforward neural networks are constrained to steady-state mapping based solely on current data.

While it is possible to integrate historical data into feedforward neural networks, limited research has been conducted on this approach. In contrast, RNN-based methods have proven to be effective in estimating transient flow rates in VFM and are therefore a promising area for further investigation.

Among RNNs, LSTM and GRU are notable examples capable of capturing short-term and long-term dynamics in data and have demonstrated impressive results in time series prediction tasks. By iteratively providing the network with the estimated current moment's output as input for generating the next prediction, recurrent neural networks can preserve the series' historical information and produce forecasts with enhanced precision (Mercante & Netto, 2022).

LSTM (Long-Short Term Memory), a deep learning technique introduced by Hochreiter and Schmidhuber in 1997, is widely known for its capability to effectively capture short-term and long-term dynamics in data. On the other hand, GRU (Gated Recurrent Unit), another type of recurrent neural network, can also handle these dynamics similarly to LSTM but is characterized by a less complex architecture with fewer parameters. This makes GRU more computationally efficient and faster to train than LSTM, as indicated by Cho et al. in 2014. Ultimately, the choice of which model to use depends on the specific task and available computational resources, with LSTMs being preferred for complex long-term dependency tasks and GRUs being ideal for simpler models that require faster training times.

One of the primary advantages of RNNs is their ability to retain information from previous time steps and use it to make predictions for future time steps. This makes them particularly useful for time series prediction tasks. To make a prediction at time step $t+1$, an RNN takes as input the previous prediction at time step t , along with any other relevant inputs and updates its internal state or memory. At time $t + 1$, the recurrent neural network receives the prediction for time t , which is then multiplied by a weight just like any other input. The RNN then uses this updated state to make a prediction for the next time step.

Training an RNN involves adjusting the weights of the network to minimize the error between its predictions and the true values in the training data. This is typically done using backpropagation through time, which involves computing gradients of the error with respect to the weights at each time step and updating the weights accordingly. This approach enables the recurrent network to identify patterns and behaviors specific to the time series and adapt its predictions accordingly (Mercante & Netto, 2022).

Recurrent neural networks (RNNs) frequently encounter a computational issue known as the exploding or disappearing gradient problem (Haykin, 2007; Mercante & Netto, 2022). This drawback happens because the back-propagation of the training over time adjusts the weights of neurons for each state by multiplying an error gradient (Mercante & Netto, 2022). This process can result in the gradient values becoming very large or very small, making it difficult for the network to learn effectively.

The problem of the exploding gradient occurs when the weights are greater than one. In this case, the gradient values become very large and can cause the weights to change dramatically, leading to unstable training. On the other hand, the problem of the disappearing gradient occurs when the weights are less than one. In this case, the gradient values become very small and the network cannot learn effectively. The LSTM is a specific type of recurrent

network designed to solve the problem of the exploding or disappearing gradients (Hochreiter and Schmidhuber, 1997; Mercante & Netto, 2022).

LSTM architecture comprises three distinct neural components: a forget gate, an input gate and an output gate. The primary function of the forget gate is to enable the network to adapt and selectively disregard input from the previous state when required, effectively addressing the gradient-related challenges. This feature emulates the human brain's aptitude for discarding insignificant data, mirroring the cognitive process of our brains. The GRU, with its two neurons per cell, specifically the reset gate and the update gate, offers a distinct architecture. A significant benefit of LSTM and GRU networks lies in their capacity to incorporate historical values from the series, regardless of how old they are, while intelligently discarding less crucial information. The solution of the gradient problem simplifies the training process and these models typically exhibit a robust ability to predict intricate time series data. (Mercante & Netto, 2022).

LSTM and GRU networks are two types of recurrent neural networks that were developed to address the problem of vanishing or exploding gradients that can occur in traditional RNNs. LSTM networks have three types of neurons: the input gate, the forget gate and the output gate. The input gate controls which values from the current time step are passed on to the cell state, while the forget gate controls which values from the previous cell state are retained or forgotten. The output gate controls which values from the cell state are output at the current time step. This allows the network to selectively remember or forget information from previous time steps, improving its ability to learn long-term dependencies. On the other hand, GRU networks have two types of neurons: the reset gate and the update gate. The reset gate determines how much of the previous hidden state is combined with the current input to produce a new candidate state. The update gate determines how much of the previous hidden state is retained and how much of the new candidate state is added to it. GRUs simplify the architecture of the network compared to LSTMs, making it easier to train and reducing the risk of overfitting.

Andrianov's research demonstrates the use of recurrent neural networks (RNNs) for VFM (2018). In this work, Long-Short Term Memory was employed. The LSTM model's capabilities for not just flowrate estimation but also predicting future flowrates were demonstrated utilizing simulated well test data. The LSTM model was also tested for severe slugging forecasting, a markedly dynamic multiphase flow phenomenon that usually tends to happen in risers. The findings indicated that the model properly forecasted the volumetric flowrate of the periodic slugging flow in the riser (Bikmukhametov, T., & Jäschke, J., 2020).

The work of Loh et al. is an additional instance of a recurrent neural network VFM system (2018). The authors employed an LSTM model to predict gas rates for one couple of natural gas wells. The system underwent training using data from the first well before making predictions on new data from both wells. The findings indicated that, overall, the model exhibited capability of reliably forecasting gas flowrates (Bikmukhametov, T., & Jäschke, J., 2020), although the predictions were occasionally wrong for the well whose data was not included in training. The LSTM model was also combined with an ensemble Kalman filter. The combination of these two methods yielded more accurate flowrate estimations for each of the two wells.

Omrani et al. (2018) conducted comparative research between an LSTM NN and a FFNN, demonstrating that the LSTM model performs better under dynamic conditions such as well shut-in and start-up. Furthermore, the LSTM model accurately tracked changes in the liquid-gas ratio during production (Bikmukhametov, T., & Jäschke, J., 2020).

Sun et al. (2018) employed a LSTM NN to forecast oil, gas and water flowrates from shale wells, which exhibit extremely transient behavior that is challenging to model with typical FFNNs and other steady-state data-driven algorithms. The authors proved that the LSTM model can estimate flowrates not just for a well whose historical data was used for training, but also for a new well utilizing historical data from nearby wells (Bikmukhametov, T., & Jäschke, J., 2020). This application shows the promising potential of LSTM NNs in VFM.

4.2.2 Support Vector Machine – Support Vector Regression

Machine learning algorithms have gained popularity for classification tasks due to their accuracy and the advancement of computer science [Wang, Y.; Yang, P.; Zhao, S.; Chevallier, J.; Xiao, Q., 2023, Wang, Y.; Yang, P.; Song, Z.; Chevallier, J.; Xiao, Q., 2023]. Previous research by Nnabuife et al. utilized spectral features and a support vector machine (SVM) for objective flow-regime identification [Nnabuife, G.S.; Pilario, S.E.K.; Lao, L.; Cao, Y.; Shafiee, M., 2019]. In 2023, Kai Yang and colleagues conducted a study on identifying flow patterns in a narrow channel using SVM. Originally, SVM was a binary classification algorithm that could handle both linear and nonlinear classifications [Xu, Q.; Wang, X.; Chang, L.; Wang, J.; Li, Y.; Li, W.; Guo, L., 2022], but it has since evolved to support multiple classification problems.

The support vector machine (SVM) algorithm was created in 1991 on the premise of statistical learning theory by Cortes and Vapnik (1995). A variety of classification,

regression, and time series prediction problems have been resolved using SVM (Cortez and Vapnik 1995; Drucker et al. 1997; Cao and Tay 2003; Smola and Scholkopf 2004; Kuo et al. 2013; Vapnik 2013; Rui et al. 2019; Ahmad et al. 2020; Shao et al. 2020; Barjouei et al., 2021).

The aim of SVM in classification is to determine a hyperplane that divides the data into two or more classes. The hyperplane constitutes a decision boundary that divides data into different classes. The hyperplane may take the form of a line in two dimensions, a plane in three dimensions, or a hyperplane in a higher-dimensional space. The Support Vector Machine algorithm aims to find a hyperplane that can best separate the data points of different classes. It does so by maximizing the margin, which refers to the maximum distance between the hyperplane and the closest data points from each class. The support vectors, that correspond to the data points closest to the hyperplane, are established by SVM in order to maximize the boundary separation between the classes and the hyperplane for the training subset (Barjouei et al., 2021). The hyperplane is created using these support vectors. If the training data is linearly separable, meaning that it is possible to draw a hyperplane that separates the different classes perfectly, then a pair (w , b) exists such that the following two conditions are satisfied (Equation 4.26, Equation 4.27):

$$w^T x_i + b \geq 1, \text{ for all } x_i \in P \quad \text{Equation 4.26}$$

$$w^T x_i + b \leq -1, \text{ for all } x_i \in N \quad \text{Equation 4.27}$$

Where x_i is a data point, P is the set of data points in the positive class and N is the set of data points in the negative class. The bias or threshold is denoted by the symbol b and the vector w is known as the weight vector.

The sign of the expression $w^T x_i + b$, which establishes the class of a new data point x , provides the SVM decision rule. If $w^T x_i + b$ is positive, then the class of x is positive; if it is negative, then the class of x is negative. In other words, the equation $w^T x_i + b = 0$ determines the decision boundary.

To find an optimum separating hyperplane that linearly separates two classes, the squared norm of the separating hyperplane can be minimized. This optimization problem can be formulated as a convex quadratic programming (QP) problem. (Equation 4.28)

$$\underset{w,b}{\text{minimize}} \Phi(w) = \frac{1}{2} \|w\|^2 \quad \text{Equation 4.28}$$

Where $\|w\|^2$ is the squared norm of the weight vector w , which represents the distance of the hyperplane from the origin. The objective function $\Phi(w)$ is convex, which means that it has a unique global minimum. The QP problem is subject to the following constraints (Equation 4.29):

$$\text{subject to } y_i(w^T x_i + b) \geq 1, i=1, \dots, l. \quad \text{Equation 4.29}$$

where y_i is the class label of the i -th data point ($y_i=1$ for the positive class and $y_i=-1$ for the negative class), x_i is the feature vector of the i -th data point, b is the bias or intercept term, and l is the total number of data points. These constraints ensure that all data points lie on the correct side of the hyperplane. If a data point is on the wrong side of the hyperplane, the product $y_i(w^T x_i + b) \geq 1$ will be negative, violating the constraint.

The solution to the QP problem can be represented as a linear combination of the support vector points. Other data points that are not support vectors are ignored since they do not contribute to the hyperplane's definition.

The data points might not always be able to be separated by a straight line or a hyperplane. By using kernel functions, SVM can solve this problem by transforming the input data into a higher-dimensional space that makes it linearly separable. The input data is mapped by the kernel function into a higher-dimensional feature space, where a hyperplane can be used to divide it. A linear hyperplane, which corresponds to a nonlinear boundary in the original feature space, divides the transformed data points in the higher-dimensional space after that. This renders SVM a more effective data separator than conventional linear classifiers. The polynomial kernel, the radial basis function (RBF) kernel and the sigmoid kernel are some of the frequently used kernel functions in SVM.

The data points might not always be able to be separated by a straight line or a hyperplane. By using kernel functions, SVM can solve this problem by transforming the input data into a higher-dimensional space that makes it linearly separable. The input data is mapped by the kernel function into a higher-dimensional feature space, where a hyperplane can be used to divide it (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής). A linear hyperplane,

which corresponds to a nonlinear boundary in the original feature space, divides the transformed data points in the higher-dimensional space after that. This renders SVM a more effective data separator than conventional linear classifiers. The polynomial kernel, the radial basis function (RBF) kernel and the sigmoid kernel are some of the frequently used kernel functions in SVM.

The Support Vector Machine (SVM) technique relies on the kernel function to compute the degree of similarity between pairs of data points in a high-dimensional space. To handle the computational complexity brought on by high-dimensional space, a suitable kernel function must be used. In determining how similar the data points are, the kernel function computes their inner products. This value is used for establishing the hyperplane's position. The data may be transformed using a variety of kernel functions, each of which has special characteristics. The exact issue that has to be solved determines which kernel function should be used. The term "kernel function" refers to any function that meets Mercer's condition (Vapnik 2013). The four kernel functions that are often employed in SVM are given in Table 4.1. (Smola and Scholkopf 2004; Vapnik 2013).

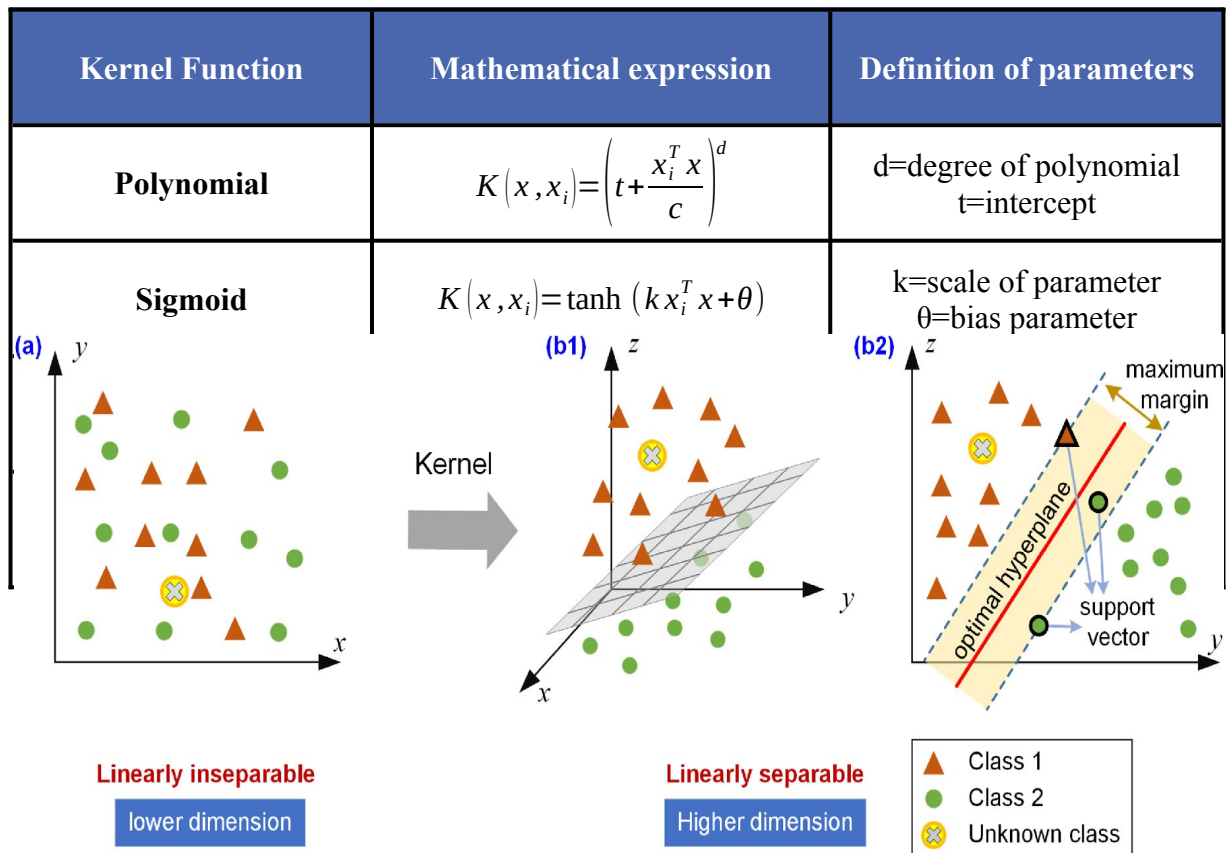


Figure 4.4 Schematic diagram of SVM classification principle. (a) Class distribution in low dimensional space; (b) Class distribution in high dimensional space. Xu, Q., Wang, X., Luo, X., Tang, X., Yu, H., Li, W., & Guo, L. (2022). Machine learning identification of multiphase flow regimes in a long pipeline-riser system. Flow Measurement and Instrumentation, 88, 102233.

Table 4.1 Kernel functions commonly used with SVM algorithms, Barjouei, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. *Journal of Petroleum Exploration and Production*, 11, 1233-1261.

The polynomial kernel function uses polynomial functions to map the input data to a high-dimensional space, making it appropriate for processing nonlinearly separable data. The decision boundary's complexity may be managed by adjusting the polynomial's degree. The RBF kernel function, which effectively separates complicated nonlinear boundaries, maps the input data to an infinite-dimensional space utilizing Gaussian functions. The sigmoid kernel function is a powerful tool for addressing binary classification problems because it uses sigmoid functions to map the input data to a high-dimensional space. The radial basis function (RBF) kernel, sometimes referred to as the Gaussian kernel, is a widely used kernel with SVM and has the capacity to reduce interference from noise in data (Vapnik et al. 1996; Kuo et al. 2013; Liu and Xu 2014; Wu et al. 2018; Hashemitaheri et al. 2020; Barjouei et al., 2021).

Overfitting is a challenge with SVM, when the model is overly complicated and fits the training data too closely, leading to subpar generalization on new data. SVM employs regularization parameters to manage the model's complexity in order to avoid overfitting. The balance between fitting the data well and preventing overfitting may be achieved by adjusting these parameters.

The shape and position of the optimal hyperplane in the SVM algorithm are determined by the support vectors, while the kernel scale and box constraint hyperparameters control the model's complexity. Accurate results depend on the number and weight of support vectors, which need to be carefully controlled for optimal performance. The support vector number and positions impact the complexity of the hyperplane. The kernel scale and box constraint hyperparameters govern the number and weight of support vectors. The kernel scale defines the similarity threshold for considering data points in the same class. A smaller kernel scale produces more support vectors, leading to a more intricate hyperplane shape. The trade-off between margin size (distance between the decision boundary and the nearest data points) and the number of incorrectly categorized points are controlled by the box constraint hyperparameter. It penalizes data points that violate the ideal hyperplane's boundary condition. Larger box constraints make the model of the hyperplane and support vectors more complicated and take longer to train since they are less likely to remove outliers in the sample.

SVR (Support Vector Regression) is a variant of SVM used for predicting the value of a dependent variable based on input variables. In SVR, a line or plane is sought that can fit the data with minimal error. The algorithm uses methods similar to those employed in classification to generate the line or plane. However, instead of maximizing the margin, the algorithm minimizes the error between the predicted and actual values. The relationship between the dependent and independent variables is studied in the context of multiphase flowmeters when predicting two-phase flow rates across wellhead chokes (Barjouei et al., 2021; Liu et al., 2022). In their research, Brereton et al. (2010) and Pan et al. (2009) developed an SVR model for predicting the flow rate of two-phase fluids through wellhead chokes based on regression. Similarly, Barjouei et al. (2021) utilized the SVR method and RBF kernel to forecast the two-phase flow rate (Q_1) through a wellhead choke.

For SVR models, the input variables are represented by a vector $[x_i] \in X = R^n$ and the dependent variable, or prediction target, is represented by a value $y_i \in Y = R$, where $i = 1, 2, 3, \dots, N$. N stands for the total number of records in the dataset. The main objective of SVR is to accurately fit a regression function $y = f(x)$ that can predict the dependent variable's value based on the given input variables. The model approximates the target values of the dependent variable by utilizing a learning function, which can be expressed as follows (Equation 4.30):

$$f(x, w) = w^T \phi(x) + b \quad \text{Equation 4.30}$$

The objective of Support Vector Regression (SVR) is to estimate a regression function $f(x)$ that can accurately predict the value of a dependent variable y based on a set of input variables x . Similar to Support Vector Machines (SVM), SVR is also used for regression tasks, where the goal is to find a regression function that can approximate the mapping from input variables x to the output or dependent variable y . The objective of SVR is to minimize the structural risk, which is a combination of the empirical risk and a regularization term, by expressing the regression function in a feature space induced by a kernel function. The kernel function is used to define the mapping function $\phi(x)$, which maps the input vector x into the feature space.

The SVR predicted target values are denoted by $f(x)$ and are obtained by a linear combination of the feature space mapping of the input variables x , represented by $\phi(x)$ and the weight vector w , with the addition of a bias term b . SVR aims to find a regression

function that maps the input variables x to the corresponding output or dependent variable y . The mapping function $\phi(x)$ is used to map the input vector from the low-dimensional input space x to a high-dimensional feature space where linear regression can be performed.

Once the input variables are mapped to the feature space, the predicted target values, denoted by $f(x)$, are obtained by a linear combination of the feature space mapping of the input variables x represented by $\phi(x)$, the weight vector w and a bias term b . The weight vector $w \in \mathbb{R}^n$ is a set of n coefficients that are learned during the training phase of the model and the bias term $b \in \mathbb{R}$ is a threshold that represents the distance between the predicted target values and the origin in the feature space.

The objective of the SVR model is to minimize the structural risk, which is the sum of the empirical risk and a regularization term. The empirical risk is the difference between the predicted target values and the actual target values, while the regularization term is used to prevent overfitting.

To derive the coefficients w and b in the support vector regression model the aim is to minimize the regularized risk function, which is composed of three components and expressed in the equation below. By minimizing this regularized risk function, the optimal values of w and b can be obtained, which define the regression function $f(x)$ for the support vector regression model.

$$\frac{1}{2}w^2 + C \frac{1}{l} \sum_{i=1}^n L_{\epsilon}(y_i, f(x_i, w)) \quad \text{Equation 4.31}$$

- The first component is $\|w\|^2$ or the smoothness penalty term encourages a flat or smooth function, which helps to avoid overfitting and improves the generalization ability of the model.
- The second component of the regularized risk function is the regularization factor C , which controls the trade-off between the smoothness penalty and the empirical error. The regularization factor allows to balance the complexity of the model and its accuracy. A larger value of C results in a less smooth function but with higher accuracy on the training data, while a smaller value of C leads to a smoother function but with lower accuracy on the training data.
- The third component of the regularized risk function is the empirical error term

$\frac{1}{l} \sum_{i=1}^n L_{\epsilon}(y_i, f(x_i))$, which measures the difference between the predicted target value

$f(x)$ and the true target value y_i for each training example. The empirical error is measured using an ε -insensitive loss function, denoted by L_ε , which is defined as the absolute difference between the predicted target value $f(x)$ and the true target value y_i , with an offset of ε to account for small errors. The empirical error is calculated as the average of the ε -insensitive loss function over all training examples. This term ensures that the model fits the training data well while also being able to generalize to unseen data. The ε -insensitive loss function over all training examples is expressed by Equation 4.32 below:

$$L_\varepsilon(Y_i, f(x_i, w)) = \begin{cases} 0, & y_i - f(x_i, w) \leq \varepsilon \\ y_i - f(x_i, w) - \varepsilon, & \text{otherwise} \end{cases} \quad \text{Equation 4.32}$$

where the range of ε values is determined in a way that ensures the loss is zero when the (Barjoui et al., 2021) absolute difference between the predicted target value and the actual target value is less than or equal to ε , which means that the predicted value falls within the range. On the other hand, when the absolute difference between the predicted value and the true value is greater than ε , the loss is equal to the difference between ε and the absolute difference between the predicted value and the true value. This means that the forecasted value lies outside the specified range (Barjoui et al., 2021) and the difference between the predicted value and the true value is penalized by ε . The ε -insensitive loss function encourages the SVM model to fit the data within a certain range and allows for some tolerance for errors. This helps to improve the robustness of the model and its ability to generalize to new data.

To estimate the coefficients w and b , a common approach is to transform the regularized risk function ($\frac{1}{2}w^2 + C \frac{1}{l} \sum_{i=1}^n L_\varepsilon(y_i, f(x_i, w))$) into the original objective function equation below by introducing two positive constants known as slack variables, denoted by ξ and ξ^* . The slack variables allow for some deviation between the predicted and true values and the optimization problem can be formulated as minimizing the sum of a regularization term and a loss term, subject to constraints that ensure the model stays close to the training data points. Specifically, the regularization term penalizes the complexity of the model by encouraging small weights, while the loss term penalizes the deviation between the predicted and true values. By solving the optimization problem, the values of w and b that minimize the error and satisfy the constraints can be found.

$$\text{Min} \frac{1}{2} w^2 + C \frac{1}{l} \sum_{i=1}^l (\xi_i - \xi_i^*) \quad \text{Equation 4.33}$$

$$\text{subjected to} \begin{cases} y_i - f(x_i, w) - b \leq \varepsilon + \xi_i \\ f(x_i, w) + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* > 0 \end{cases} \quad \text{Equation 4.34}$$

The optimization problem for Support Vector Regression (SVR) with a linear kernel, has two components (Equation 4.33): the first term $\frac{1}{2} w^2$ is the regularization term that penalizes the complexity of the model and encourages the model to have small weights. The second term $C \frac{1}{l} \sum_{i=1}^l (\xi_i - \xi_i^*)$ is the loss term, which penalizes the deviation of the predicted values from the true values, while allowing for some slack variables ξ and ξ^* to account for noise or outliers in the data. The constraints (Equation 4.34) in the problem ensure that the model stays close to the training data points. The first constraint $y_i - f(x_i, w) - b \leq \varepsilon + \xi_i$ requires that the predicted value for each training point y_i is within a distance of ε from the true value, plus an additional slack variable ξ_i . The second constraint $f(x_i, w) + b - y_i \leq \varepsilon + \xi_i^*$ requires the same for the opposite direction. The slack variables ξ_i and ξ_i^* are non-negative and represent the amount of deviation allowed for each data point. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής)

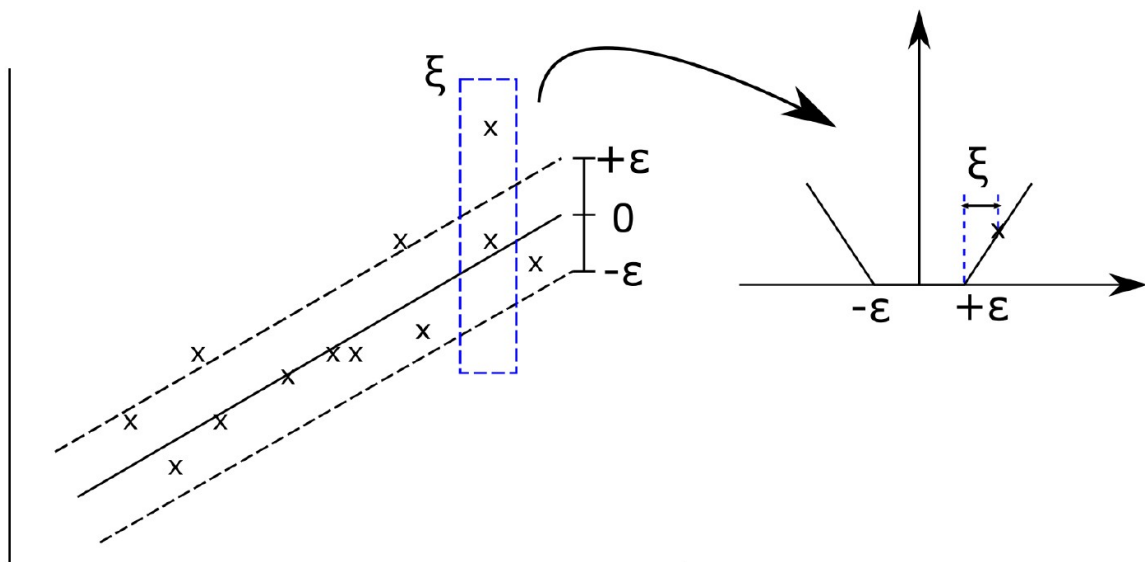


Figure 4.5 The soft margin loss setting for a linear SVM. Liu, M., Kim, G., Bauckhage, K., & Geimer, M. (2022). Data-Driven Virtual Flow Rate Sensor Development for Leakage Monitoring at the Cradle Bearing in an Axial Piston Pump. Energies, 15(17), 6115.

The regression error in SVR is defined as the difference between the predicted output value and the true output value for each training data point. The goal of SVR is to minimize this error while fitting a function to the training data. The regression error is controlled using Lagrangian multipliers, which are introduced as constraints in the optimization problem. The margin in SVR is the distance between the regression hyperplane and the closest support vectors. The regression hyperplane is the linear function that best fits the training data, and the support vectors are the most informative data points that define the decision boundary. The margin is used to control the complexity of the SVR model and prevent overfitting. The margin is also controlled using Lagrangian multipliers, which are introduced as constraints in the optimization problem.

The optimization problem in SVR is a trade-off between minimizing the regression error and maximizing the margin. This convex optimization expression can be addressed by employing a Lagrangian multiplier strategy (Smola and Scholkopf 2004; Vapnik 2013; Rui et al. 2019; Ahmad et al. 2020; Shao et al. 2020; Barjouei et al., 2021). The Lagrangian multipliers are introduced as constraints to enforce this trade-off and find the best solution that fits the training data while preventing overfitting. Equation 4.35 defines the transformed equation with these multipliers:

$$f(x, \alpha_i, \alpha_i^*) \sum_{i=1}^N (\alpha_i - \alpha_i^*) (\varphi^T(x_i) \varphi(x)) + b \quad \text{Equation 4.35}$$

where the α_i and α_i^* represent the Lagrangian multipliers.

In the transformed equation, $f(x, \alpha_i, \alpha_i^*)$ is the regression function, which is a linear combination of the transformed input vectors $\varphi(x_i)$. The term $\sum_{i=1}^N (\alpha_i - \alpha_i^*) (\varphi^T(x_i) \varphi(x))$ represents the difference between the Lagrangian multipliers associated with positive and negative errors, multiplied by the transformed input vectors. The parameter b is the bias term.

The input vectors x_i that correspond to $\alpha_i - \alpha_i^* = 0$ are called support vectors. These are the most informative data points that lie closest to the regression hyperplane and define the decision boundary. The support vectors play a crucial role in the SVR algorithm, as they determine the shape of the regression function.

4.2.3 Decision trees

Decision trees are a well-known and commonly used machine learning technique that can be applied to analyze various types of datasets, including both regression and classification problems (Lorena et al. 2007; Tsai and Chiou. 2009; Nie et al. 201; Barjouei et al., 2021). These trees are built by dividing the dataset into groups based on a set of learned rules, creating nodes and branches to form a hierarchical structure. Although decision trees can be used for both classification and numerical (regression) datasets, they are most often applied to classification problems (Osei-Bryson 2004; Lorena et al. 2007; Ortuno et al. 2015; Barjouei et al., 2021). To solve classification problems, each leaf of the tree is assigned a class label, with the rules used to discriminate between the different classes and allocate the data to specific leaves. The process of constructing a decision tree for machine learning typically involves three steps:

- When developing decision trees for machine learning tasks, it is necessary to distinguish input (attribute) variables from dependent (target) variables (Barjouei et al., 2021). Input variables refer to the features or predictors used to make predictions, while dependent variables are the variables being predicted.
- Splitting the data records at "child" nodes according to certain criteria requires the application of a splitting algorithm that evaluates the input variables (Barjouei et al., 2021). The decision tree algorithm identifies the most informative variable that can divide the dataset into distinct groups or classes. This is achieved by computing the impurity or entropy of the dataset with each input variable. The splitting variable that yields the largest reduction in impurity is chosen and the dataset is split into subsets based on the values of that variable.
- Continuing to split the data at each child node to generate further nodes and decision-tree layers. At each child node, the dataset is further divided into subsets by selecting the most informative variable and dividing the dataset. This process is repeated until a stopping criterion is met, which can be determined by factors such as the maximum depth of the tree, the minimum number of records in a node, or when further splitting no longer improves classification accuracy.

A decision tree is a structure that arranges nodes and branches in a hierarchical manner to provide a set of rules for predicting an outcome based on input variables. Nodes within the tree represent decision points where the dataset is divided into subsets based on a specific attribute, while branches represent the potential outcomes of each decision. The first node in the decision tree is called the root node (top layer). At the top layer, the dataset is initially divided into two subsets, that subsequently evolve into child nodes, constituting the second

layer of the tree. Subsequently, these child nodes undergo further division into sub-nodes, creating additional layers of child nodes, eventually resulting in the last layer of terminal nodes, also known as leaves (Barjouei et al., 2021).

To ensure consistency-homogeneity among the data records assigned to each child node, a splitting algorithm is utilized in the decision tree. The algorithm identifies the attribute that provides the highest information gain or reduction in impurity for the dataset and applies it to determine the data splits at each node. The aim is to attain the utmost level of consistency for each sub-node. Through this process, the algorithm establishes clearly differentiated (homogeneous) data records at the final layer of terminal nodes (Barjouei et al., 2021). Each terminal node represents a specific decision outcome, such as a classification label or numerical value in regression analysis.

The decision tree is developed through the iterative division of the dataset into subsets at each node, until a predetermined stopping criterion is achieved, such as a maximum depth, minimum number of records in a node, or no further improvement in classification accuracy. A set of rules is utilized to determine which variables should be split at each node of the tree. In practice, decision trees are constructed by continuously splitting the data until every training data point is accurately classified, resulting in a decision tree with 100% accuracy on the training data. However, this accuracy may not extend to new and previously unseen data points, as the decision tree may have been overly adjusted to the training data.

A key issue with decision trees is their tendency to overfit the training data, whereby the model becomes too intricate and too finely-tuned to the training data, leading to deficient performance on new and previously unseen data points. The risk of overfitting rises with an increase in the number of layers and nodes within a decision tree. (Fakhari and Moghadam 2013; Czajkowski and Kretowski 2016; Liu et al. 2016; Barjouei et al., 2021). This is because, as the tree grows more complex, it is more likely to capture noise in the training data rather than the underlying patterns.

To prevent overfitting in machine learning models, there are several methods that can be employed. One frequently used technique involves pruning the decision tree, which entails eliminating redundant nodes or branches that do not contribute to accurate classification on the testing data. Another strategy is to use an ensemble of decision trees, such as Random Forest, to minimize the risk of overfitting by aggregating the predictions from multiple decision trees.

Regarding the VFMs, in their research, Barjouei et al. (2021) utilized the decision tree module of scikit-learn (sklearn) library, which was implemented using Python. Barjouei and

colleagues (2021) selected the "gini" criterion to evaluate the significance of features and employed the "best" splitter approach to determine both the feature and the threshold value for each split. The main objective of their study was to predict the two-phase flow rate (Q_i) passing through a wellhead choke.

4.2.4 Random Forest

Ensemble learning is a highly effective approach in machine learning that utilizes multiple models to enhance the overall performance of the model. One of the most widely used ensemble learning methods is bagging, short for bootstrap aggregating. This technique involves training several models on various subsets of the training data. The subsets are created by bootstrapping, a process of randomly sampling data points from the training set with replacement to form each subset. This ensures that each model in the ensemble is trained on a distinct set of data, making them independent and reducing the variance in the ensemble's predictions. In the case of the Random Forest algorithm, decision trees serve as the weak learners. The ensemble of decision trees generates a Random Forest (Breiman, 2001). Decision trees are models that learn a tree-like structure of decision rules to classify data. Each node in the tree represents a condition on one of the data features and the tree branches out into two paths depending on whether the condition is met or not. The leaves of

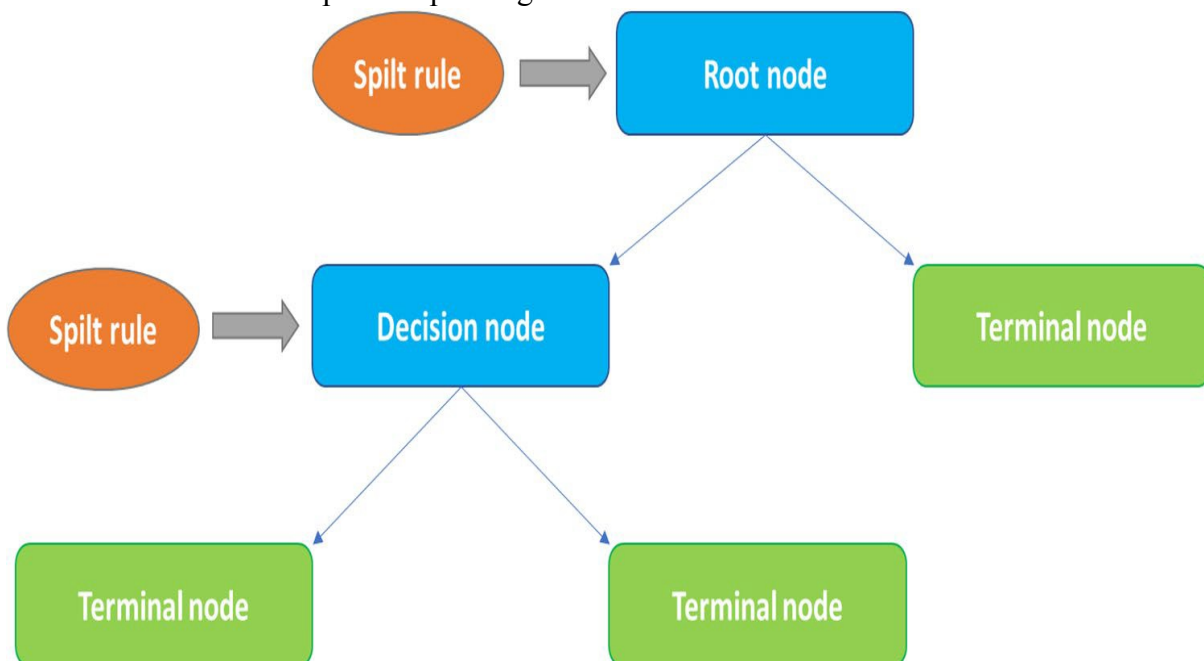


Figure 4.6 Typical structure of a decision-tree machine-learning model Barjouei, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. *Journal of Petroleum Exploration and Production*, 11, 1233-126

the tree represent the predicted class for a given data point, as mentioned in the previous paragraph 4.2.3.

The Random Forest algorithm is created by training multiple decision trees using bagging. Each decision tree is trained on a different subset of the training data, and hyperparameters such as the minimum leaf size, maximum number of splits and the number of variables to sample in each split are optimized to create diverse models within the ensemble. By aggregating the predictions of these decision trees, the Random Forest algorithm uses a majority vote to predict the final class of a given data point.

It is worth highlighting that although bagging and Random Forest are sometimes used interchangeably, Random Forest is a specific variation of bagging that employs decision trees as the weak learners. In contrast, other types of bagging algorithms may use different models as the weak learners. The Random Forest approach has demonstrated high performance across diverse applications and it remains one of the most widely utilized ensemble learning methods in the field of machine learning.

The random forest algorithm is a supervised machine learning technique that extends the decision tree algorithm by constructing multiple decision trees to evaluate input data. It is commonly used for classification and regression tasks and involves training and testing subsets of data sets displaying distinct input and dependent parameters (Zhou et al. 2020; Grape et al. 2020; Barjoui et al., 2021). Unlike individual decision trees, each decision tree in the random forest ensemble is constructed in parallel and utilizes only a few layers or nodes. This mitigates the risk of overfitting and reduces the variance and bias of prediction results. The collective assessment of all decision trees ensures accurate predictions without compromising overall decision accuracy (Breiman 2001; Ahmad 2018; Barjoui et al., 2021).

To facilitate the training of the random forest model, the process involves the random sampling of subsets of data records from the complete dataset, utilizing a method commonly referred to as bootstrapping. (Barjoui et al., 2021). After generating the bootstrapped data subsets, each subset can be utilized to construct an unpruned decision tree for classification or regression purposes. In constructing each decision tree, a random subset of the available input variables (G) is selected for splitting, rather than using all input variables (M) (Barjoui et al., 2021). This selection of only a few input variables for each tree ensures that each tree is independent of the others and reduces the correlation between the individual trees in the ensemble. Several decision trees are constructed sequentially until a predetermined number of trees (K) is reached. In the context of regression tasks, predictions for the dependent variable are generated by combining predictions (bagging) obtained from all the individual regression

trees that have been constructed (Barjoui et al., 2021). Bagging is a technique that simplifies each decision tree and reduces the likelihood of the algorithm overfitting the training data. Bagging reduces the risk of the algorithm fitting too closely to the training data. This decreases the chance of overfitting and ultimately leads to improved algorithm performance. Equation 4.36 represents the prediction function for the random forest algorithm (Barjoui et al., 2021):

$$\hat{f}_{RF}^K(x) = \frac{1}{K} \sum_{k=1}^K T_i(x) \quad \text{Equation 4.36}$$

The equation calculates the predicted value for a given input variable vector (x) by aggregating the predictions of individual regression trees. The variable K in the equation represents the number of individual regression trees that are built in the random forest model. Each regression tree is trained on a different subset of the bootstrapped data records and uses a limited, randomly chosen set of input variables (G) for splitting (Barjoui et al., 2021). The prediction for each individual tree is denoted by $T_i(x)$, where i represents the index of the data record. The prediction from each tree is combined by taking the average of all the predictions (i.e., the sum of all the predictions divided by K). This final aggregated prediction is denoted by $\hat{f}_{RF}^K(x)$. So, when the random forest model is given a new input variable vector (x), each individual regression tree will predict a value for the dependent variable based on the input variable vector, and the predicted values from all the trees are then averaged to obtain the final prediction of the random forest model.

To evaluate the performance of each tree of the random forest model, a technique called out-of-bag (OOB) error estimation is used. OOB error is an important feature of the random forest algorithm as it allows for a more accurate estimation of the generalization error of the model compared to just relying on the training set.

The random forest algorithm utilizes the OOB error to assess the accuracy of its predictions. The OOB error is determined by evaluating the predictions of each tree on data points that were not used for its training, which is achieved by randomly selecting a subset of data during the bootstrap process. This approach allows for an unbiased estimation of the generalization error of the model and helps to prevent overfitting or underfitting of the data. By comparing the OOB error to the accuracy of the forest on an independent testing subset, the algorithm can determine whether the model is overfitting or underfitting the data. There is an overfitting of the model to the data if the OOB error is significantly greater than the testing

error. Overall, the OOB error provides a valuable tool for evaluating the accuracy of the random forest algorithm's predictions.

Random forest can be used to assess the importance of input variables in predicting the dependent variable. One approach is to randomly permute the values of a single input variable in the OOB subset and observe the resulting decrease in prediction accuracy. This process is repeated for all input variables to rank them based on their relative importance to the prediction accuracy. By determining how much the prediction accuracy decreases when a particular input variable is removed from the model, it becomes possible to identify the variables that are most significant in predicting the dependent variable. This information can then be used to eliminate low-contribution variables, reducing the dimensionality of the model and enhancing its performance on datasets with high dimensionality (Ahmad et al. 2017; Barjouei et al., 2021). Ranking the relative importance of input variables in a random forest can be accomplished through several methods. One approach entails interchanging two input variables within the tree solutions while keeping the other variables unchanged and subsequently calculating the average decrease in prediction accuracy. This procedure enables the assessment of the relative importance of each input variable concerning the accuracy attained for the dependent variable (Ahmad 2018; Barjouei et al., 2021). The random forest algorithm is visually represented in [Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής.](#)

The study conducted by Barjouei et al. (2021) utilized the Scikit Learn Random Forest Regressor to establish a regression random forest model for predicting the two-phase flow rate (Q_i) through a wellhead choke.

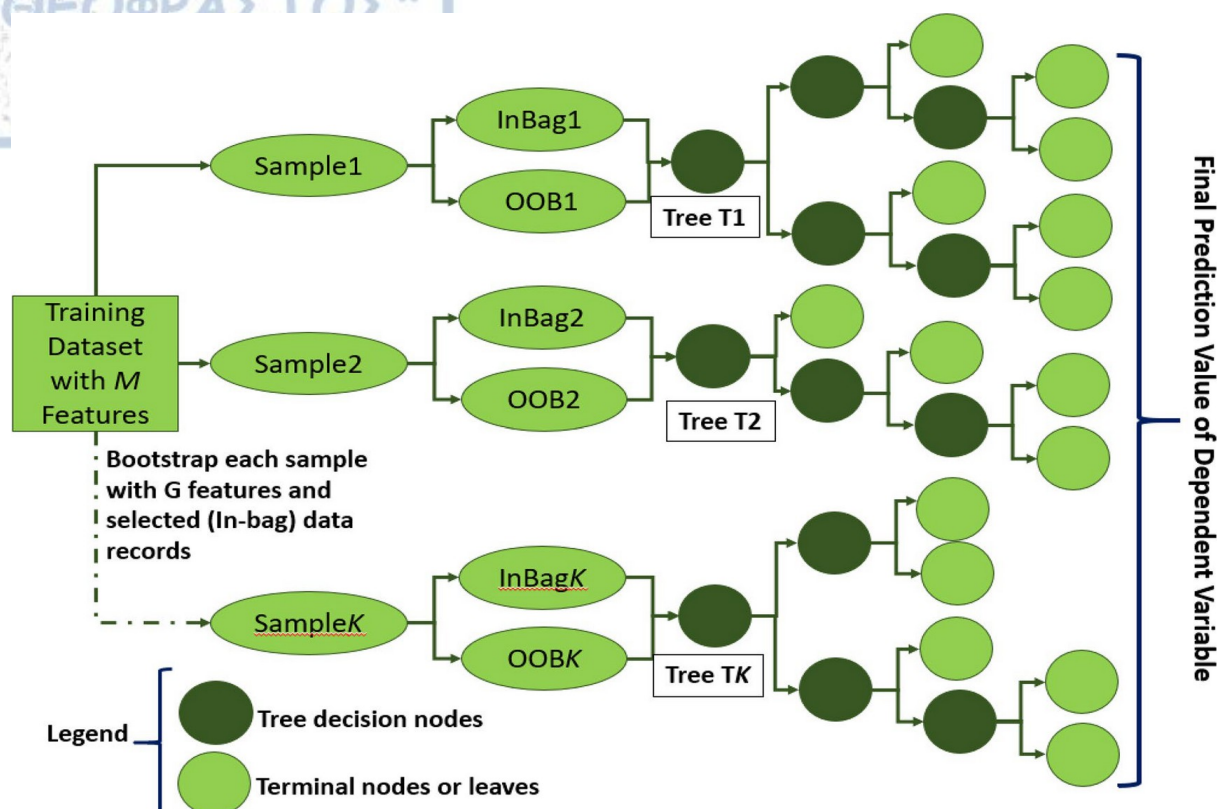


Figure 4.7 Schematic diagram of generic configuration of random forest algorithm Barjouei, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. *Journal of Petroleum Exploration and Production*, 11, 1233-126

4.3 Parameters optimization

Hyperparameters are values that are set before the training process begins, and they determine how the model will learn from the data. They are not learned by the model during training, unlike the parameters which are learned from the data. The selection of hyperparameters can have a significant impact on the performance of a machine learning algorithm. If the hyperparameters are not set correctly, the model may overfit or underfit the data, leading to poor performance on new, unseen data.

Each machine learning algorithm involves a multitude of hyperparameters that play a crucial role in determining the accuracy of the resulting model. These hyperparameters encompass various aspects such as the number of hidden layers and neurons in a neural network, the number of trees in a Random Forest, the kernel function in SVM, the learning rate and the choice of optimization algorithm, among others.

Choosing the right hyperparameters is essential, yet there are no established principles or fixed guidelines that can ensure the finest performance for a particular machine learning

algorithm. The pursuit of locally optimal hyperparameters typically relies on a blend of individual experience-derived insights and search algorithms (Song et al., 2022). Some hyperparameters, for instance the number of hidden layers in a neural network, can be determined empirically.

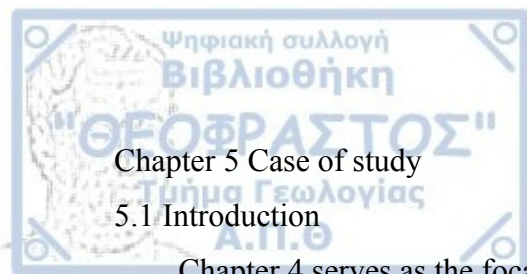
Hornik et al. (1989) demonstrated that a single hidden layer neural network with an appropriate number of neurons can approximate highly complex functions. However, using more than one hidden layer in a neural network can increase the risk of getting stuck in local minima and lead to slower convergence (Liu et al., 2022). Similarly, the structure of LSTM networks resembles that of BP networks, and increasing the number of hidden layers can lead to higher computational costs and slower convergence. As a result, a single hidden layer LSTM network is preferred.

Other hyperparameters such as the quantity of trees in a Random Forest, the learning rate and the number of training iterations, are established through personal experience. The grid search algorithm is then employed to pinpoint locally optimal hyperparameters within a specified range. While training the model, the utilization of a suitable loss function can expedite model convergence and minimize errors (Culotta et al., 2007; Song et al., 2022).

The Mean Squared Error (MSE) (Equation 4.25) is used as the evaluation function during model training, whereas the Mean Absolute Error (MAE) (Equation 4.37) is typically employed for practical applications.

$$E_{MAE} = \frac{1}{m} \sum_i^m |\hat{y}_i - y_i| \quad \text{Equation 4.37}$$

where \hat{y}_i represents the real value of the dependent variable for the i^{th} data record, y_i is the predicted value of the dependent variable for the i^{th} data record and m represents the total number of data records in the training subset (Barjouei et al., 2021).



Chapter 5 Case of study

5.1 Introduction

Chapter 4 serves as the focal point of this thesis, delineating the procedural workflow for the development of a data-driven Virtual Multiphase Flowmeter. The objective of this thesis is to present the development of a VFM tailored for a subsea network, leveraging the capabilities of machine learning techniques and the Pipesim software. The generation of training data for the VFM is facilitated by the steady-state simulator and the machine learning techniques elucidated in Chapter 3 are subsequently applied to train the VFM.

The forthcoming chapters will expound upon the proposed methodology and its practical application. The primary aim of this VFM is to accurately estimate hydrocarbon flowrates at the wellheads of each well within the subsea network. This stands in contrast to other VFMs previously discussed, which primarily focus on identifying flow regimes.

Building upon this contextual foundation, the section dedicated to data collection, pre-processing and feasibility analysis assumes a pivotal role. It delineates the process of acquiring and preparing datasets, each representing specific conditions such as GOR equality, GOR inequality, water cut and pressure variations. The intentional diversification of datasets enables a comprehensive study of the VFM's performance under varied operational scenarios, contributing to a thorough understanding of its adaptability and efficacy across a spectrum of operational conditions.

A pivotal aspect of the pre-processing phase involves the application of feature engineering techniques, enhancing datasets to ensure the effective capture of pertinent information. The use of tools such as the Regression Learner App for each dataset showcases the workflow employed during the pre-processing phase. The subsequent section on model development delineates the application of the Neural Net Fitting App, marking the realization of the data-driven VFMs.

5.2 Subsea network construction in Pipesim

The datasets utilized in this thesis were generated using the renowned Pipesim software by Schlumberger. Pipesim stands as Schlumberger's flagship software for steady-state multiphase flow simulations, specifically designed to model the behavior of wells and networks. Its initial introduction dates back to 1984, primarily addressing the challenges faced in designing production systems for demanding environments like the North Sea. Throughout its existence, Pipesim has remained committed to incorporating the latest scientific breakthroughs to ensure the utmost accuracy in predictions.

Over the years, Pipesim has undergone continuous enhancement and expansion, aligning itself with the advancements in scientific knowledge and technological progress. By utilizing Pipesim, users gain the capability to optimize production processes and guarantee flow assurance across the entire lifecycle of an oil and gas field. This encompassing functionality covers diverse aspects ranging from field development to production operations.

In order to construct the datasets, a deliberate decision was made to create a simplified subsea network consisting of two wells. This selection was intentional, with the purpose of minimizing complexity within the resulting model and subsequently mitigating potential errors associated with intricate configurations. By opting for a simpler system, the focus could be directed towards accurately capturing the essential dynamics and behaviors of multiphase flow, facilitating a more precise analysis and modeling process.

The dataset was designed to simulate a subsea network comprising of two wells draining the same reservoir and positioned at same depths. Both wells share identical geometries, with their bottomhole extending to a depth of 12,000 feet. The wellheads were located on the seafloor at a depth of 1,600 feet. The borehole diameter for both wells in the datasets was specified as 9.75 inches, and the drainage radius was defined as 2,000 feet. In this configuration, each well was associated with an individual flowline that connected its respective wellhead to the topside processing facilities via a riser.

The length of the flowline for well 1 was set to 6,336 feet, while that of the flowline for well 2 to 7,920 feet. These specific values were chosen to reflect realistic scenarios and operational conditions typically encountered in subsea networks. By maintaining consistent geometries and comparable flowline lengths, the datasets aimed to investigate the impact of specific parameters and variations in operational conditions.

The flowlines from each well were connected to a riser, which extended to a height of 1,600 feet, reaching the seafloor. The riser served as a conduit for the produced fluids, facilitating their transportation from the subsea wells to the platform. Upon reaching the platform, the riser was connected to a three-phase separator system. For simulation purposes Pipesim used a separator system comprised of three separate tanks, each designated for a specific phase of the produced fluids: a gas tank, an oil tank and a water tank. In real-world applications, however, it corresponds to a three-phase separator that effectively separates gas, oil and water components from the extracted fluids.

The reservoir fluid utilized in the simulations is a light black oil. It exhibited properties indicative of a relatively light crude oil, with an API gravity of 31. The specific gravity of water was determined to be 1.02 and additionally, the specific gravity of the gas present

within the reservoir was 0.74. The gas-oil ratio (GOR) played a significant role as a parameter in the datasets, with two distinct values considered depending on the specific scenarios under investigation. The GOR values selected for the dataset were either 300 scf/stb or 500 scf/stb. The choice of these GOR values was based on the particular objectives being explored within the datasets, which would be elaborated upon in subsequent subchapters. It is important to highlight that the fluid did not incorporate any contaminants.

The reservoir featured in the datasets is characterized by a thickness of 180 feet. The permeability of the reservoir was set at 100 millidarcies (md). A permeability value of 100 md suggests a moderately permeable reservoir, allowing for a satisfactory flow of fluids through the rock matrix. This permeability is crucial in determining the ease with which hydrocarbons can flow within the reservoir. The temperature of the reservoir was equal to 180 °F.

5.3 Data collection

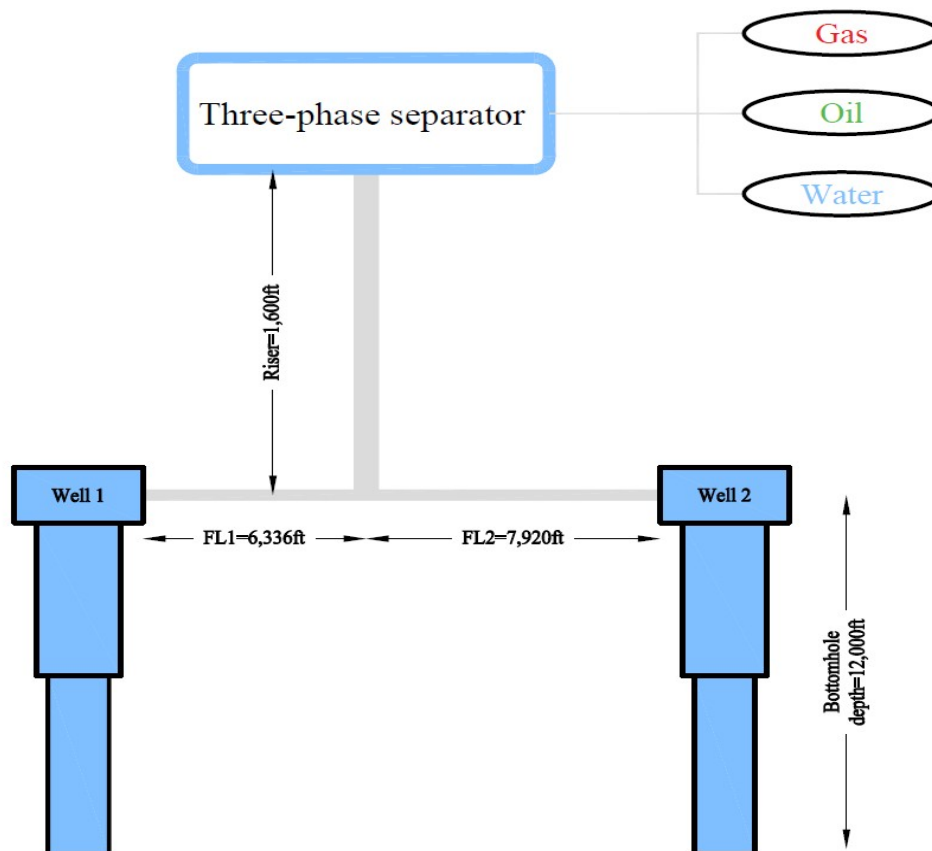


Figure 5.1 Simplified representation of the geometry of the subsea well network

Data points were generated utilizing the Pipesim software through the execution of multiple simulation runs. The outcomes of these simulations were systematically collected manually and categorized into four distinct data sets, delineated as follows.

5.3.1 Dataset 1 ($GOR_1=GOR_2$)

The first data set comprises of 362 data points. In its creation, a hypothetical scenario was considered involving the two identical wells with the geometry described in [Chapter 5.2](#). The GOR is identical for both wells, set at 500 scf/stb. For the purpose of this study, this particular data set will henceforth also be denoted as " $GOR_1=GOR_2$."

To generate each data point, the bottomhole pressures of the wells underwent systematic manual adjustments. Initially, a uniform grid of data points was established, with variations in the bottomhole pressures of both wells by 500 psi within the range of 4,500-9,000 psi. By assigning 10 different values to each pressure, a total of 100 data points were created.

However, noticeable fluctuations were observed in the recorded values of hydrocarbon flowrates, intended for prediction. To address this, the resolution of the data point grid was refined by introducing additional bottomhole pressure values for well 2. This adjustment resulted in the creation of an irregular grid, where bottomhole pressure 1 exhibited variations every 500 psi and bottomhole pressure 2 demonstrated variations every 125 psi, all confined within the range of 4,500-9,000 psi. Consequently, a grid comprising 362 points, representative and sufficiently dense, was established, forming a foundational basis upon which a machine learning model can be developed and trained.

5.3.2 Dataset 2 ($GOR_1 \neq GOR_2$)

In the second dataset, a total of 421 data points were generated, employing a methodology similar to that of the previous dataset. This time, the two wells are positioned in the same reservoir but are separated by a geological fault. This geological structure introduces variations in the composition of the fluids on either side of the fault, thus varying their GOR values. Alternatively, the fluid differentiation between the two wells can be attributed to an anticlinic shape of the reservoir. In this scenario, the second well receives an additional influx of gas from the gas cap, resulting in a higher GOR for this particular well. Conversely, the first well withdraws fluid from the base of the reservoir, leading to a comparatively lower GOR that corresponds to the solution GOR only. Consequently, in this new scenario, the GOR exhibits variability between the two wells, with GOR_1 set at 300 scf/stb, while GOR_2 is set at to be 500 scf/stb. All other fluid properties employed remain

same to the previous scenario in Dataset 1. Moreover, the specific gravity of the gas within the reservoir remains at 0.74. For the purpose of this study, this data set will be denoted as " $GOR_1 \neq GOR_2$."

To generate each data point, manual adjustments were systematically applied to the bottomhole pressures of the wells. Initially, a uniform grid of data points was established, featuring variations in the bottomhole pressures of both wells by 500 psi within the range of 4,500-9,000 psi, mirroring the approach employed in the previous dataset. By assigning 10 different values to each pressure, a total of 100 data points were created in the initial phase.

However, once again, noticeable fluctuations were observed in the hydrocarbon flowrates, constituting the output variables. To address this challenge, it was deemed necessary to enhance the data grid resolution beyond the scope of the previous scenario. Accordingly, additional values were introduced for well 2 bottomhole pressure, varying at intervals of 125 psi within the extended range of 4,125-9,750 psi, while pressure 1 maintained variations every 500 psi within the range of 4,500-9,000 psi. Consequently, a grid of 421 points was meticulously constructed, providing a representative dataset for the development and training of a machine learning model.

5.3.3 Dataset 3 (Watercut)

The third data set consists of 498 data points. In this scenario, the two wells are located in the same reservoir but are separated by a fault. The fault acts as a barrier, creating a physical flow separation between the two wells. Such geological structures are common features in reservoir systems and can significantly influence fluid flow and distribution. The GOR for the first well GOR_1 , was established at 300 scf/stb, while the GOR for the second well, GOR_2 was set at 500 scf/stb as was the case with Dataset 2.

This time, two additional variables were varied, specifically the water cut of each well is now also altered. For the purpose of this study, this data set will be denoted as "Dataset watercut." The fluid properties used in both wells were same as the previous scenarios. To account for the increased complexity resulting from the inclusion of four variables, a more randomized selection of pressures and water cuts values was made.

In this dataset, the water cut values ranged from 10% to 90%, while the bottomhole pressures were within the range of 5000–9000 psi. The generation of a comprehensive dataset, totaling more than 6,500 data points, is considered impractical due to the manual nature of data entry. Consequently, a more pragmatic approach was employed to ensure the creation of a representative data set of manageable size.

To achieve this, a systematic selection process was implemented from the potential combinations of the aforementioned variables. A carefully chosen subset of 498 data points was strategically curated, with each selected point designed to adequately represent the broader parameter space. The selection criteria involved assessing points that exhibited substantial hydrocarbon production, while simultaneously discarding combinations with no production or insufficient production. This judicious and selective approach aimed to capture the essential characteristics of the system while providing a dataset size reduced yet with a sufficiently dense grid to effectively train a machine learning model.

By incorporating these scientifically plausible scenarios and ensuring the specified ranges for bottomhole pressures and water cuts, the data set provides a comprehensive representation of the various conditions and their potential influence on reservoir behavior and production dynamics.

5.3.4 Dataset 4 (Combined)

In this case, the data sets 2 and 3 were combined without generating new data points. The hypothetical scenario involves a network of two subsea wells situated in the same reservoir but separated by a fault. Initially, the wells commence production with zero water cut, indicating the extraction of dry oil. However, as time progresses, water production starts to manifest in the sinks of the separator. In the subsequent sections, the data set resulting from the combination of data sets 2 and 3 will be denoted as the "combined" data set.

This scenario closely reflects real-world situations encountered in subsea reservoir engineering. Over time, it is common for water to migrate from adjacent aquifers within the reservoir due to pressure depletion or influx from surrounding formations. As a result, water production gradually emerges in the separator sinks. This could also be the case when breakthrough appears after water flooding.

5.4 Data description

After each simulation, the following parameters were recorded for each data point and stored in an Excel sheet:

- **Two pressures at the wellheads ($P_{\text{Wellhead1}}$, $P_{\text{Wellhead2}}$):** These pressures represent the measured pressure values at the respective wellheads and are expressed in pounds per square inch (psi). They are obtained from the manometers that are permanently installed at these wellheads for accurate pressure monitoring.

- **Gas flow rates (Q_{gas}):** The gas flow rates, denoted as Q_{gasWell1} and Q_{gasWell2} , are expressed in million standard cubic feet per day (scf/d) and measured at the respective wellheads of the two wells. These values represent the amount of gas produced by each well. Due to the prevailing pressure in the separator, there is a small amount of gas that only appears in the oil sink at standard conditions. Therefore, the gas flow rate in the gas sink is denoted as $Q_{\text{gasSinkGas}}$, while the gas appearing in the oil sink is denoted as $Q_{\text{gasSinkOil}}$.
- **Oil flow rates (Q_{oil}):** The oil flow rates, represented as Q_{oilWell1} and Q_{oilWell2} , are expressed in stock tank barrels per day (STB/d) and measured at the respective wellheads of the two wells. These values signify the volume of oil being produced by each well. Additionally, the oil flow rate at the oil sink is denoted as $Q_{\text{oilSinkOil}}$. This value indicates the overall oil production and represents the volume of oil collected after the oil-gas separation process is complete. Note that no oil appears in the gas sink since the produced gas is dry.
- **Water flow rates (Q_{water}):** Water flow rates are recorded when there is a presence of water cut in the produced fluid. The water flow rates are measured in stock tank barrels per day (STB/d) at the wellheads ($Q_{\text{waterWell1}}$, $Q_{\text{waterWell2}}$) and at the water sink which was a constant fraction of 100%.

At this point, it is imperative to elucidate the distinction between dependent and independent variables in the context of the case study. The independent (predictor) variables consist of the pressures at wellhead 1 ($P_{\text{wellhead1}}$) and wellhead 2 ($P_{\text{wellhead2}}$), along with the flow rates of each phase in the sinks post three-phase separation. Notably, these inputs are initially extracted directly from Pipesim but undergo subsequent modification during the feature engineering process, as detailed in the subsequent [Chapter 5.5.2](#). Consequently, the raw variables from the software are not directly employed to train the machine learning model but instead, the feature-engineered variables are utilized. All selected variables serving as inputs are measurable in real-life scenarios. Simultaneously, the dependent (response) variables in this study comprise the flow rates of each phase (oil, gas, and water) at wellhead 1 and wellhead 2. Unlike the input variables, the flowrates of each phase at the wellheads require prediction, given their inherent inability to be directly measured in real-life scenarios. Nevertheless, their economic significance remains paramount and this represents the challenge that a VFM aims to address. The following figure visually depicts inputs in green and outputs in red in a simplified representation of the subsea network.

5.5 Pre-processing

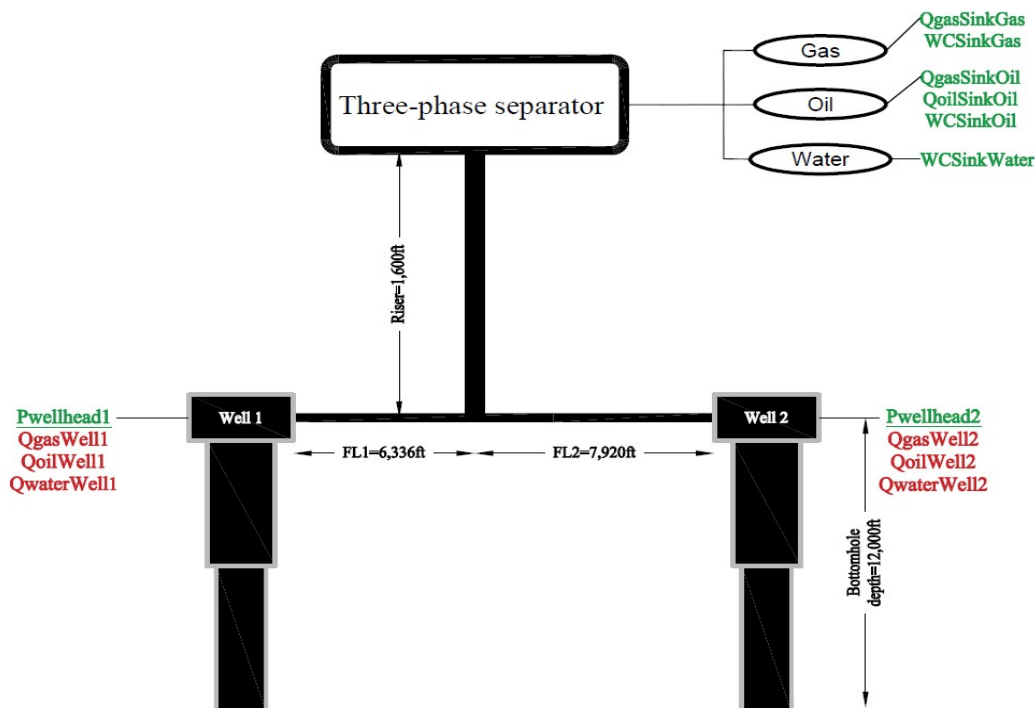


Figure 5.2 Simplified explanation of the distinction between dependent and independent variables; in red color are presented the inputs and in green color are presented the outputs

Preprocessing in machine learning is the preparation and transformation of raw data to improve its quality, structure and compatibility for optimal performance when training and evaluating machine learning models. In this work, the following two operations were implemented:

5.5.1 Missing Values and Data Normalization

In this study, the hydrocarbon phases fractions of well1 were predicted since the fractions of well2 could be calculated by subtracting the hydrocarbon fractions of well1 from 100%. This approach is based on the assumption that the sum of hydrocarbon fractions over all wells equals to 100% and no gas mass is lost.

However, during the data collection process, certain data points were identified where the bottomhole pressures were set in a way that caused one of the wells to inject rather than produce. This happened because one well exhibited significantly higher potential compared to the other well. In such cases, the wellhead pressure for the weaker well was set to zero,

while the other well maintained a positive pressure value. Consequently, the hydrocarbon production fraction for such wells with zero wellhead pressure was recorded as 0, signifying the complete absence of hydrocarbon production from that specific well. Conversely, the well with a positive wellhead pressure registered a hydrocarbon production fraction of 1, indicating that all the hydrocarbons produced in that particular scenario originated solely from this active well.

It is important to acknowledge that machine learning models are typically designed to operate effectively with datasets that conform to smooth distributions, devoid of abrupt and extreme deviations among individual data points. The presence of scenarios where one well exhibits a hydrocarbon production fraction of 0 and the other well demonstrates a fraction of 1 poses challenges for machine learning algorithms, primarily due to the necessity of learning two distinct and contrasting behaviors.

During the training phase, the algorithm learns to predict updates for oil and gas fractions by considering both wellhead pressures as inputs. However, when encountering a situation where one well's production fraction is exclusively 1, the algorithm is suddenly required to make predictions using only the wellhead pressure of that particular well. This shift in behavior can perplex the algorithm, as its training was predicated on considering both fractions and wellhead pressures simultaneously. As a result, the algorithm may encounter difficulties in effectively generalizing its learned patterns to accommodate this novel scenario, potentially leading to reduced prediction accuracy or uncertainty in its outputs.

In this context, missing values in the dataset, which correspond to data points where the wellhead pressures and consequently the hydrocarbon fractions were 0 for well1, were appropriately removed. This step was taken to ensure that all required data is available for analysis. Additionally, in cases where one well is not producing any oil, it is scientifically justified to consider that the production fraction for that well is 0%. Therefore, the information related to the non-producing well can be disregarded during the prediction process and these data points may be excluded from the dataset.

Moreover, in certain cases, it was observed that after the simulations, some data points resulted in gas and oil fractions that very slightly exceeded unity (values greater than 1). This occurrence can be attributed to the utilization of numerical methods by the Pipesim software during each simulation, which introduces a degree of numerical error on the order of decimal thousands or tens of thousands. Simulation software often incorporates simplifying assumptions to expedite calculations. While these assumptions facilitate the simulation process, they may also introduce limitations and potential inaccuracies. Therefore, it is

crucial to have a comprehensive understanding of the assumptions implemented in the software and assess their impact on the accuracy of the obtained results. To address instances where the oil or gas fraction exceeded unity, manual corrections were applied to obtain precise values. These corrections were made to rectify the inaccuracies introduced by the simulation software. By manually adjusting the values, the dataset was refined to ensure that the gas and oil fractions adhered to their physical limits (between 0 and 1) and provided more accurate representations of the actual data.

5.5.2 Feature engineering and Target engineering

Feature engineering involves crafting or modifying input variables to enhance their informativeness and improve machine learning model performance. It includes tasks like selecting relevant features, creating new ones, or transforming existing variables. On the other hand, target engineering, often known as target creation, is the process of deliberately constructing or defining the target variable in a way that optimizes the model's predictive accuracy. This can involve transforming the target variable to better align with the modeling objectives or creating composite targets that capture specific aspects of the prediction task. In the context of the recorded data, both processes were employed:

5.5.2.1 Feature engineering

Acquiring knowledge of the ultimate flowrates of each phase after the three-phase separation was essential for facilitating the creation of a new input variable, Gas-Oil Ratio (GOR). As substantiated in subsequent chapters, the GOR variable played a pivotal role in shaping the machine learning development of the datasets.

- **Gas-Oil Ratio (GOR):** GOR is a measure of the amount of gas produced in relation to the amount of oil produced. It is calculated by dividing the total gas production ($Q_{\text{gasSinkGas}} + Q_{\text{gasSinkOil}}$) by the total oil production ($Q_{\text{oilSinkOil}}$) at the oil sink. (Equation 5.38)

$$GOR = \frac{Q_{\text{gasSinkGas}} + Q_{\text{gasSinkOil}}}{Q_{\text{oilSinkOil}}} \quad \text{Equation 5.38}$$

5.5.2.2 Target engineering

The primary objective of the virtual multiphase flowmeter is to anticipate the flowrates of individual phases (oil, gas and water) at each wellhead. Rather than relying on absolute values, the model leverages the known ultimate flowrates of each phase derived from data

points in Pipesim. The prediction process is streamlined by estimating the proportion or fraction of each phase between the two wells, guided by the following equations (Equation 5.39, Equation 5.40, Equation 5.41). This approach utilizes the established final flowrates in the sinks to infer the fractional contributions of each phase, enhancing the efficiency of the predictive modeling process. Accurate predictions of these fractions enable the determination of the flowrate of each phase for each well by multiplying the fraction with the total flowrate.

- **Gas fraction well 1:** This parameter represents the proportion of gas production from well 1 compared to the total gas production from both well 1 and well 2. It can be calculated by dividing the gas flow rate from well 1 ($Q_{gasWell1}$) by the sum of the gas flow rate at the gas sink ($Q_{gasSinkGas}$) and the gas flow rate at the oil sink ($Q_{gasSinkOil}$). (Equation 5.39)

$$Gas\ fraction\ well\ 1 = \frac{Q_{gasWell1}}{Q_{gasSinkGas} + Q_{gasSinkOil}} \quad \text{Equation 5.39}$$

- **Oil fraction well 1:** This parameter represents the proportion of oil production from well 1 compared to the total oil production at the oil sink. It can be calculated by dividing the oil flow rate from well 1 ($Q_{oilWell1}$) by the oil flow rate at the oil sink ($Q_{oilSinkOil}$). (Equation 5.40)

$$Oil\ fraction\ well\ 1 = \frac{Q_{oilWell1}}{Q_{oilSinkOil}} \quad \text{Equation 5.40}$$

- **Water fraction well1:** This parameter represents the proportion of water production from well 1 compared to the total water production from both well 1 and well 2. It can be calculated by dividing the water flow rate from well 1 ($Q_{waterWell1}$) by the sum of the water flow rates from both wells ($Q_{waterWell1} + Q_{waterWell2}$). (Equation 5.41)

$$Water\ fraction\ well = \frac{Q_{waterWell1}}{Q_{waterWell1} + Q_{waterWell2}} \quad \text{Equation 5.41}$$

5.6 Feasibility Analysis and Data Feasibility Assessment

Prior to commencing a machine learning project, a feasibility analysis becomes imperative to ascertain the overall viability of the proposed solution. This evaluation constitutes the fundamental basis for the development of a robust machine learning solution.

A critical facet of this analysis involves the scrutiny of data feasibility, which entails a thorough examination of the quality, quantity and relevance of the available data. The objective is to ensure that the data is suitable for training a machine learning model and accurately represents the intricacies of the problem domain. By addressing both feasibility analysis and data feasibility, informed decisions can be made regarding potential successes and challenges.

In the context of the case study, subsequent to the theoretical analysis, it was established that specific inputs are necessary for predicting the outputs of the VFM. To corroborate the correlation beyond theoretical predictions, a numerical test involving various machine learning models was conducted, with a focus on feasibility over accuracy. A feasibility analysis was executed to assess the presence of the theoretically presumed correlations within the data. The aim was to verify the accurate association of inputs and outputs with their corresponding variables at the data level before the finalization of the machine learning model development. This involved the exclusion of data considered uncorrelated or lacking essential information, ensuring the existence of a functional relationship.

In this study, the Regression Learner App available in the MATLAB software was employed for these purposes. The app provides a user-friendly interface for implementing regression analysis, allowing control over a single output variable while incorporating multiple input variables for analysis. Through the utilization of the Regression Learner App, the reliability of the datasets was evaluated by examining either the oil fraction or the gas fraction of well 1. This analytical decision was motivated by the objective to elucidate that the modeling of either gas or oil as a variable can be extrapolated to the other, emphasizing their intrinsic interdependence as variables.

Feasibility analysis was executed for all four datasets, revealing that, to varying extents, the data can be effectively captured by a model. A successful feasibility demonstration is characterized by data points clustering around the diagonal, signifying their amenability to being learned by a machine learning model. The presence of an off-diagonal cluster indicates potential issues not with the machine learning models but with the datasets themselves. Additionally, considering the inherent potential for typographical errors in manually generated data, a meticulous data validation process was implemented to detect and rectify any erroneous data points resulting from such errors. Incorrect data points were identified and corrected by comparing the data against expected values or established patterns.

Each dataset will be presented separately to ensure the appropriate focus and analysis of the respective variables.

5.6.1 Dataset 1 ($GOR_1=GOR_2$) – Regression Learner App

In the initial data set, the experimental conditions were relatively simpler compared to the subsequent sets. The wells used in this data set were completely identical and they produced the same type of fluid. Since the data was obtained from simulations, it was expected that these simulations would provide perfect predictions due to the lack of noise. Consequently, it was anticipated that the actual and predicted values would be identical, resulting in the data points falling precisely on the diagonal line $((0,0), (1,1))$. However, it is important to note that the primary objective of the analysis in the Regression Learner App is not solely focused on aligning the data points with the diagonal line. Rather, it aims to investigate the presence of correlation between the variables and determine the machine learning algorithm that best describes the data.

Specifically, the chosen input variables for this analysis are the two wellhead pressures. These variables were selected because they are the only ones that can potentially influence hydrocarbon production in the given scenario.

Regarding the reservoir conditions in this data set, it is indicated that the reservoir lies above the bubble point, which signifies that the pressure exceeds the threshold at which gas begins to separate from the oil phase. In this state, the reservoir exists in phase equilibrium, meaning that oil and gas coexist as a homogeneous mixture without physical separation.

In previous chapters, two fundamental metrics frequently employed in statistical analysis and Machine learning were introduced. To provide a brief summary, those metrics will be revisited (MSE, MAE) and the two new metrics will be introduced: (R^2 , RMSE).

- **Mean Squared Error (MSE):** Mean Squared Error (MSE) is a commonly used metric in statistical studies and machine learning to assess predictive model performance. It calculates the average of the squared differences between predicted and actual values. Squaring the differences ensures both positive and negative deviations are treated equivalently. A lower MSE indicates better model performance, as it reflects a smaller average discrepancy between predictions and observations. However, MSE does not consider the scale of the data and reports the error in the original units squared. To address this, the square root of MSE, known as Root Mean Squared Error (RMSE), is often used for interpretability, as it is in the same units as the dependent variable.

- **Root Mean Squared Error (RMSE):** Root Mean Squared Error (RMSE) is a widely used metric in machine learning to evaluate the performance of predictive models. It provides a measure of the average difference between the predicted values of a model and the corresponding actual values in a dataset. Mathematically, RMSE is computed as the square root of the mean of the squared residuals. Residuals represent the vertical distances between the observed data points and the corresponding predicted values based on a regression line or the model's predictions. A lower RMSE value indicates that the model has smaller prediction errors and is considered to be more accurate. In an ideal scenario, where a hypothetical model could consistently predict the exact expected values, the RMSE would be 0. As the observed data points align more closely with the regression line or the model's predictions, the model's error decreases, leading to a lower RMSE. This would indicate a perfect fit between the model's predictions and the actual values.
- **R-squared (R^2):** also known as the coefficient of determination, measures how well a regression model fits the data by quantifying the proportion of the dependent variable's variance explained by the independent variables. It ranges from 0 to 1, where 0 signifies no explanation of variance and 1 represents a perfect fit. Higher R^2 values indicate better model performance in capturing underlying patterns. However, R^2 does not measure predictive power or accuracy and should be used alongside other metrics for a comprehensive evaluation of the model's performance.
- **Mean Absolute Error (MAE):** Mean Absolute Error is a statistical metric used to quantify the average magnitude of errors in a set of predictions, regardless of their direction. It is computed as the average absolute difference between the predicted values and the corresponding actual values. MAE is commonly employed to evaluate the performance and effectiveness of a regression model. It provides a measure of the average absolute discrepancy between predicted and actual values. Lower MAE values indicate higher accuracy, as they reflect reduced overall error magnitude in the model's predictions.

After running the Regression Learner App, the obtained results are presented in the [Table 5.1](#) below. These metrics are evaluated both on the validation and test sets. The validation dataset is used during model development to tune hyperparameters and evaluate the model's

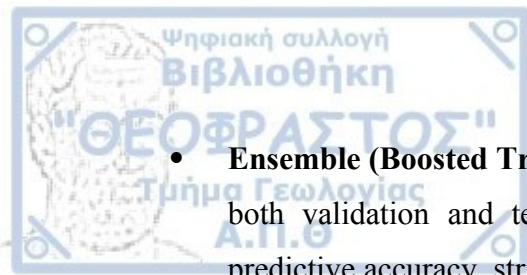
performance. The test dataset is kept separate and is used to assess the model's generalization ability on unseen data.

Model Type	RMSE (Validation)	MSE (Validation)	R ² (Validation)	MAE (Validation)	MAE (Test)	MSE (Test)	RMSE (Test)	R ² (Test)
Linear Regression	0.057	0.003	0.902	0.039	0.037	0.003	0.055	0.916
Stepwise Linear Regression	0.057	0.003	0.901	0.040	0.037	0.003	0.055	0.916
Fine Tree	0.038	0.001	0.956	0.025	0.023	0.001	0.031	0.974
Medium Gaussian SVM	0.024	0.001	0.982	0.015	0.017	0.001	0.028	0.978
Ensemble (Boosted Trees)	0.038	0.001	0.956	0.026	0.023	0.001	0.028	0.979
Medium Neural Network	0.005	0.00002	0.999	0.003	0.002	0.00002	0.004	1.000
Least Squares Regression Kernel	0.167	0.028	0.897	0.125	0.167	0.043	0.207	0.899

Table 5.1 Performance evaluation metrics of different machine learning models for GOR₁=GOR₂ model

According to Table 5.1, the following observations about the different machine learning models arise:

- **Linear Regression and Stepwise Linear Regression:** Both linear regression and stepwise linear regression models exhibit similar metrics. However, the linear regression-based models may not be the optimal choices for this task, as indicated by their moderate RMSE and R² values, suggesting potential limitations in capturing data patterns.
- **Fine Tree:** The decision tree model demonstrates promising performance, surpassing linear regression-based models with a notably higher R² value.
- **Medium Gaussian SVM:** The medium Gaussian SVM model showcases robust predictive capabilities, reflected in its low RMSE and impressive R² value. The model effectively captures the data patterns, making it a competitive choice for accurate predictions.



- **Ensemble (Boosted Trees):** The ensemble model delivers competitive results across both validation and test sets. Its ability to combine multiple models enhances predictive accuracy, striking a favorable balance between RMSE and R^2 values.
- **Medium Neural Network:** The medium neural network stands out as the top-performing model, consistently producing the lowest RMSE and the highest R^2 value among all models. Its adeptness at capturing intricate patterns and nonlinear relationships positions it as a compelling choice for further exploration.
- **Least Squares Regression Kernel:** The least squares regression kernel model exhibits the least favorable performance among all models. Its significantly higher RMSE, MSE and MAE values, coupled with a low R^2 value, indicate a poor fit to the data.

For a more comprehensive evaluation of the models' performances, normalization was applied to [Table 5.1](#) by dividing the metrics of each model by the corresponding metrics of the Linear Regression model presented in the first row, followed by expressing the results as percentages.

This normalization method facilitates a systematic and thorough comparative analysis, assessing each model's deviation or improvement relative to the linear baseline across metrics for both validation and test sets. In the context of normalized performance metrics, lower percentages for metrics such as RMSE, MSE and MAE indicate superior performance, while higher percentages suggest a relative decrease in performance compared to the linear regression baseline. Conversely, for the R^2 metric, higher percentages indicate improved performance, whereas lower percentages suggest a relative decrease in performance compared to the linear regression baseline.

In [Table 5.2](#) with the normalized values, the model with the lowest percentage value in each column, considering RMSE, MSE and MAE, and the highest for R^2 , is considered the best for interpreting the dataset. Once again, the Neural Network model evidently emerges as the best-performing model.

Model Type	RMSE (Validation)	MSE (Validation)	R^2 (Validation)	MAE (Validation)	MAE (Test)	MSE (Test)	RMSE (Test)	R^2 (Test)
------------	-------------------	------------------	--------------------	------------------	------------	------------	-------------	--------------

Linear Regression	100.000%	100.000%	100.000%	100.000%	100.000%	100.000%	100.000%	100.000%
Stepwise Linear Regression	100.940%	101.889%	99.796%	100.890%	100.000%	100.000%	100.000%	100.000%
Fine Tree	67.387%	45.409%	105.902%	62.945%	63.100%	31.099%	55.767%	106.328%
Medium Gaussian SVM	42.458%	18.027%	108.863%	37.886%	46.663%	26.107%	51.095%	106.786%
Ensemble (Boosted Trees)	67.368%	45.384%	105.905%	66.093%	61.144%	24.831%	49.831%	106.904%
Medium Neural Network	8.375%	0.701%	110.736%	7.555%	6.359%	0.557%	7.461%	109.133%
Least Squares Regression Kernel	295.687%	874.308%	99.391%	320.438%	450.209%	1402.002%	374.433%	98.190%

Table 5.2 Performance evaluation metrics of different machine learning models for $GOR_1=GOR_2$ model (normalized values)

In Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής and Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής, the graphs depicting the validation and test sets of the first data set are presented.

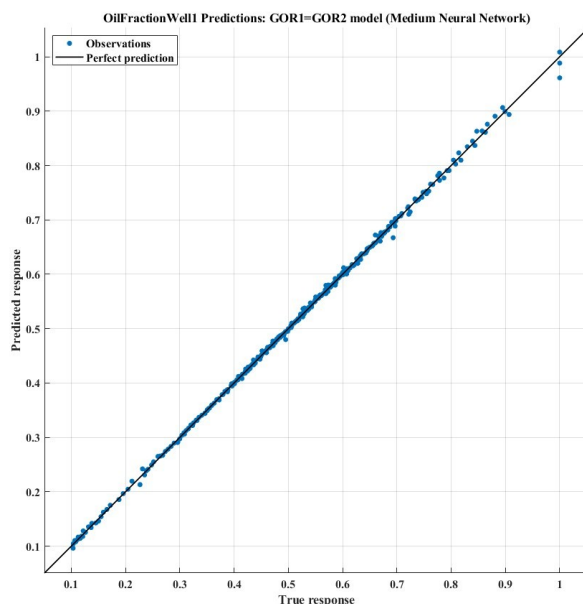


Figure 5.3 Validation set for $GOR_1=GOR_2$ model (Medium Neural Network)

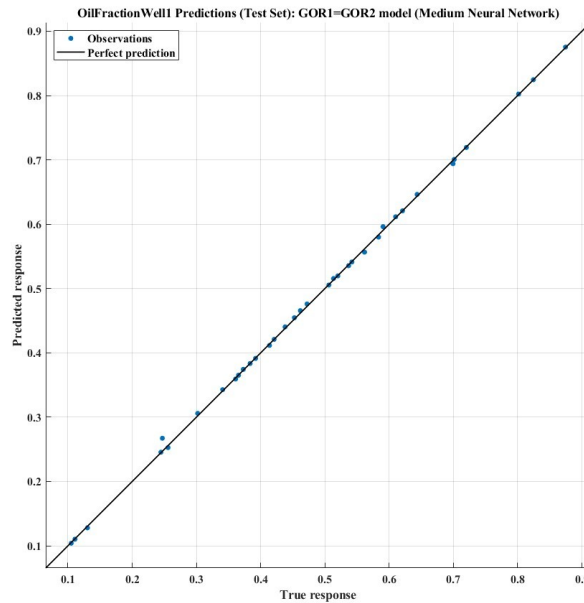


Figure 5.4 Test set for $GOR_1=GOR_2$ model (Medium Neural Network)

As observed in the validation set [Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής](#), three points are identified where the prediction of the oil fraction appears to be slightly inaccurate. These instances correspond to scenarios where production solely originates from well 1, resulting in an oil fraction of unity. While these three points could have been excluded from the dataset under the assumption that well 2 has zero production, they were intentionally retained for analysis. Interestingly, these points do not significantly impact the neural network model's performance. Instead, they contribute valuable information to the model, as will be elucidated in subsequent chapters. Within the dataset, numerous data points exhibit hydrocarbon fractions in the range of 0.9-1. Consequently, instances where the fraction is exactly 1 do not deviate significantly from these values, contributing to a smooth distribution. Following this rationale, a machine learning model has the capacity to effectively learn from these data points, considering them as additional information.

It is crucial to note that, at this stage, the primary focus is on the feasibility analysis. The chosen inputs, namely the two pressures at the wellheads, are deemed appropriate for predicting the oil fraction as an output. Following this analysis, it becomes evident that there are no extensively off-diagonal data points, indicating that a model can effectively learn these data points. Thus, the feasibility analysis is deemed successful.

5.6.2 Dataset 2 ($GOR_1 \neq GOR_2$) – Regression Learner App

In this dataset, the wells remain unchanged compared to Dataset1 however, the fluid they produce exhibits different GORs as a consequence of a geological fault that has caused reservoir compartmentalization. A thorough examination was undertaken to identify the most suitable machine learning model for accurately characterizing the dataset and exploring potential correlations among the variables using the Regression Learner App.

The selected input variables for the analysis are the two wellhead pressures, considered the primary factors influencing hydrocarbon production in the specified scenario. After multiple tests in the Regression Learner app, other variables present in the dataset were excluded from this analysis, as they were determined to have either negligible or no direct impact on the chosen output variable, which was the oil fraction for well 1. The feasibility analysis for Dataset 2 produced outcomes that are tabulated in [Table 5.3](#).

Model Type	RMSE (Validation)	MSE (Validation)	R ² (Validation)	MAE (Validation)	MAE (Test)	MSE (Test)	RMSE (Test)	R ² (Test)
Linear Regression	0.062	0.004	0.930	0.046	0.053	0.005	0.071	0.927
Stepwise Linear Regression	0.062	0.004	0.930	0.046	0.053	0.005	0.071	0.926
Fine Tree	0.030	0.001	0.983	0.019	0.013	0.0003	0.018	0.995
Fine Gaussian SVM	0.038	0.001	0.973	0.021	0.017	0.0004	0.021	0.993
Ensemble (Boosted Trees)	0.029	0.001	0.985	0.023	0.023	0.001	0.028	0.989
Wide Neural Network	0.005	0.00003	0.999	0.002	0.001	0.000002	0.002	1.000
SVM Kernel	0.210	0.044	0.909	0.163	0.137	0.036	0.189	0.918

Table 5.3 Performance evaluation metrics of different machine learning models for GOR1≠GOR2 model

- **Linear Regression and Stepwise Linear Regression:** The linear regression models, both basic and stepwise, exhibit commendable performance on the validation set, as indicated by low RMSE and high R-squared values. However, their efficacy in capturing intricate and complex patterns is called into question, given the moderate RMSE observed on the test set. This suggests potential limitations in generalization to new data.



- **Fine Tree:** The decision tree model stands out with exceptional performance, outperforming the linear regression-based models. Its low RMSE and high R-squared values on both validation and test sets indicate its ability to effectively capture a substantial portion of the variance in the data, positioning it as a promising choice for this predictive task.
- **Fine Gaussian SVM and Ensemble (Boosted Trees):** The SVM and the Ensemble models demonstrate competitive performance across both validation and test sets. The SVM model showcases robust predictive capabilities, reflected in low RMSE and an impressive R-squared value. Meanwhile, the ensemble method leverages the strengths of multiple models to achieve a favorable balance between RMSE and R-squared, making it a solid and accurate prediction option.
- **Wide Neural Network:** The Neural Network emerges as the top-performing model, consistently delivering the lowest RMSE and the highest R-squared value among all models. Its adeptness at capturing intricate patterns and nonlinear relationships positions it as a compelling candidate for further exploration and consideration as the primary model for this predictive task.
- **SVM Kernel:** The Kernel model displays the poorest performance, marked by significantly higher RMSE, MSE and MAE values. The presence of a low R-squared value indicates its unsuitability for predicting the target variable. Given these limitations, prudent consideration should be given to alternative models demonstrating superior predictive capabilities tailored to the specific requirements of this task.

For a thorough evaluation of model performances, the metrics in [Table 5.3](#) were normalized by dividing them with the corresponding metrics of the Linear Regression model and expressed as percentages in [Table 5.4](#). The model with the lowest percentage value for RMSE, MSE and MAE and the highest for R^2 , is deemed the best for interpreting the dataset. The Neural Network model consistently emerges as the top-performing model.

Model Type	RMSE (Validation)	MSE (Validation)	R^2 (Validation)	MAE (Validation)	MAE (Test)	MSE (Test)	RMSE (Test)	R^2 (Test)
------------	-------------------	------------------	--------------------	------------------	------------	------------	-------------	--------------

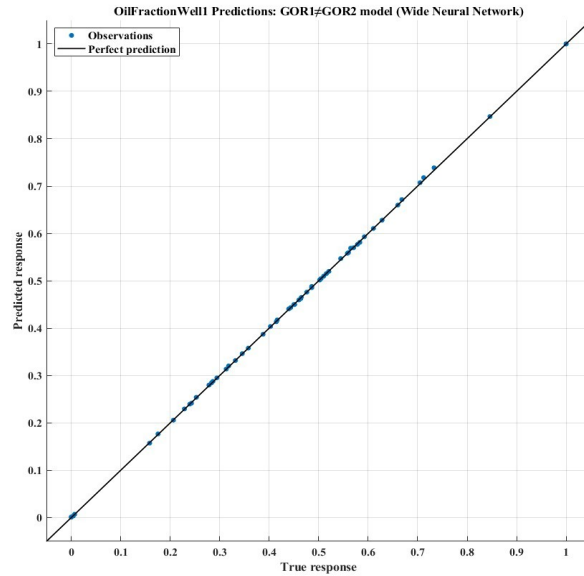


Figure 5.6 Test set for $GOR_1 \neq GOR_2$ model (Wide Neural Network)

Table 5.4 Performance evaluation metrics of different machine learning models for $GOR_1 \neq GOR_2$ model (normalized values)

In Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής and Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής, the graphs illustrate the performance and evaluation results of the validation and test sets for data set 2, respectively. The feasibility analysis is deemed successful as the data points cluster around the diagonal.

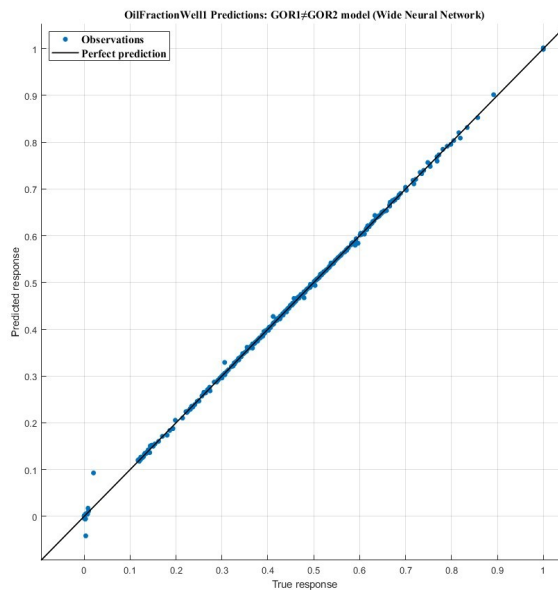


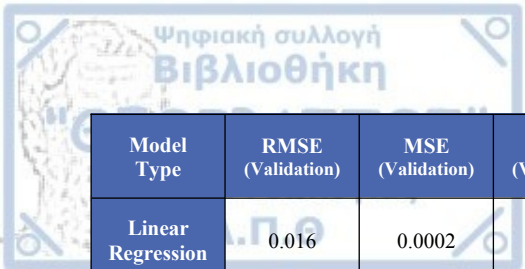
Figure 5.5 Validation set for $GOR_1 \neq GOR_2$ model (Wide Neural Network)

5.6.3 Dataset 3 (Watercut) – Regression Learner App

The third dataset comprises a total of 498 data points, exhibiting a notable variation in complexity. The model under consideration involves two identical wells, physically separated by a fault, with each well producing distinct fluids. The fluid from the first well is characterized by a GOR of 300 scf/stb, while the second produces fluid with a GOR of 500 scf/stb. Each data point is associated not only with varying bottomhole pressures but also with different water cuts. The selection of diverse pressures and water cuts to generate these data points was elaborated upon in [Chapter 5.3.3](#) and the method could be described as relatively random.

In the Regression Learner App, the input variables encompass wellhead pressures and GOR measured in scf/stb. Notably, the GOR utilized as an input is the one measured on the platform post the separation process. The output variable denotes the gas fraction of the first well. Initially, the correlation between GOR and wellhead pressures with the dataset was uncertain. However, following rigorous testing and feasibility analysis in the Regression Learner app, it was revealed that GOR serves as a highly suitable and influential parameter, as substantiated by the findings presented in [Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής](#) and [Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής](#).

After employing the Regression Learner App, the obtained metrics are detailed in [Table 5.5](#). The normalized values, expressed as percentages and derived by dividing each model's metrics by the corresponding metrics of the Linear Regression model in the first row, are presented in [Table 5.6](#).



Model Type	RMSE (Validation)	MSE (Validation)	R ² (Validation)	MAE (Validation)	MAE (Test)	MSE (Test)	RMSE (Test)	R ² (Test)
Linear Regression	0.016	0.0002	0.987	0.010	0.009	0.0003	0.016	0.986
Stepwise Linear Regression	0.011	0.0001	0.994	0.006	0.006	0.0001	0.009	0.995
Fine Tree	0.010	0.0001	0.995	0.005	0.004	0.0001	0.012	0.993
Quadratic SVM	0.006	0.00004	0.998	0.005	0.005	0.00003	0.006	0.998
Ensemble (Boosted Trees)	0.025	0.001	0.969	0.019	0.019	0.001	0.025	0.966
Bilayered Neural Network	0.002	0.000005	1.000	0.001	0.001	0.000005	0.002	1.000
SVM Kernel	0.026	0.001	0.965	0.012	0.012	0.0005	0.022	0.974

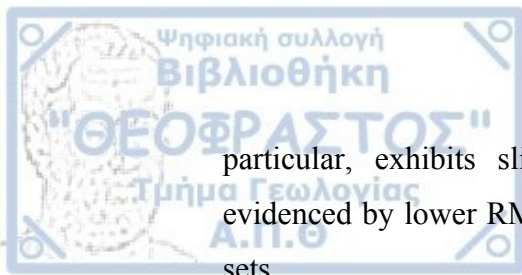
Table 5.5 Performance evaluation metrics of different machine learning models for Watercut model

Model Type	RMSE (Validation)	MSE (Validation)	R ² (Validation)	MAE (Validation)	MAE (Test)	MSE (Test)	RMSE (Test)	R ² (Test)
Linear Regression	100.000%	100.000%	100.000%	100.000%	100.000%	100.000%	100.000%	100.000%
Stepwise Linear Regression	69.079%	47.720%	100.669%	62.997%	60.852%	34.565%	58.792%	100.924%
Fine Tree	64.604%	41.737%	100.745%	47.275%	47.506%	52.575%	72.508%	100.670%
Quadratic SVM	38.582%	14.886%	101.089%	51.865%	49.234%	13.640%	36.933%	101.219%
Ensemble (Boosted Trees)	157.600%	248.378%	98.102%	197.890%	204.099%	244.637%	156.409%	97.958%
Bilayered Neural Network	15.021%	2.256%	101.250%	14.150%	14.682%	1.796%	13.401%	101.386%
SVM Kernel	167.582%	280.839%	97.687%	127.477%	123.715%	187.124%	136.793%	98.770%

Table 5.6 Performance evaluation metrics of different machine learning models for Watercut model (normalized values)

Upon reviewing the metrics for the various machine learning models, several observations can be made:

- **Linear Regression and Stepwise Linear Regression:** Both linear models demonstrate strong predictive performance, with low RMSE and high R-squared values on both the validation and test sets. The Stepwise Linear Regression, in

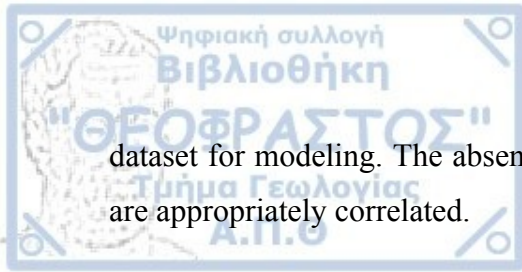


particular, exhibits slight improvement over the basic Linear Regression, as evidenced by lower RMSE and higher R-squared values on both validation and test sets.

- **Fine Tree:** The Tree model performs exceptionally well, outperforming linear regression-based models with minimal RMSE and high R-squared values on both sets.
- **Quadratic SVM and Ensemble (Boosted Trees):** The SVM model showcases exceptional performance, with extremely low RMSE and high R-squared values on both validation and test sets. The Ensemble model provides competitive results, incorporating the strengths of multiple models for accurate predictions, albeit with slightly higher RMSE and lower R-squared values compared to individual models.
- **Bilayered Neural Network:** The Neural Network stands out as the top-performing model, achieving near-perfect performance with the lowest RMSE and highest R-squared values among all models on both validation and test sets. Its capacity to capture intricate patterns and nonlinear relationships makes it an excellent choice for further consideration and exploration.
- **SVM Kernel:** The Kernel model exhibits reasonable performance with moderate RMSE and R-squared values on both validation and test sets. While not the top performer, it provides a balanced performance and could be considered as a suitable option for certain predictive tasks.

In conclusion, based on the quantitative analysis of the provided metrics, the Neural Network emerges as the top-performing model among the listed options. It demonstrates high accuracy, strong fit to the data and excellent RMSE, R-squared and MAE values.

In Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής and Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής, the graphs illustrate the performance and evaluation results of the validation and test sets for Dataset 3, respectively. Feasibility test is successful as the observed data points exhibit a notable clustering around the diagonal. This clustering indicates that the NN model can effectively learn and predict the associated data, affirming the viability of the



dataset for modeling. The absence of off-diagonal clusters suggests that the chosen variables are appropriately correlated.

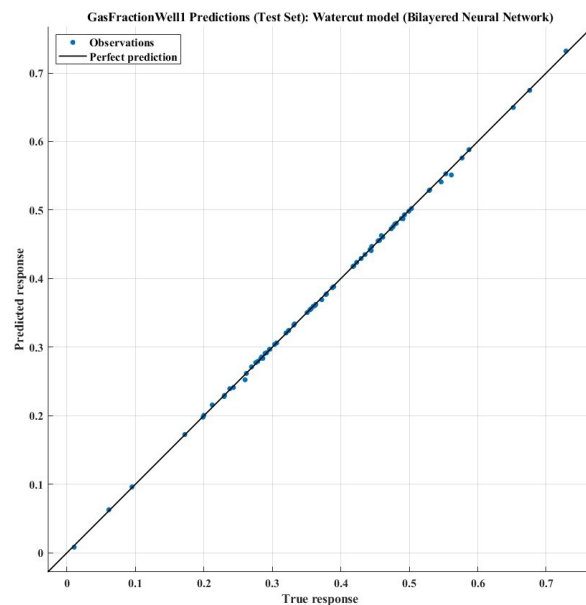


Figure 5.7 Test set for Watercut model (Bilayered Neural Network)

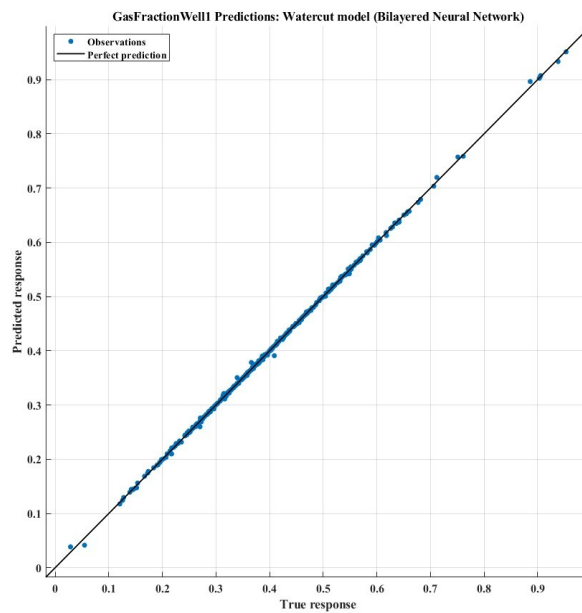


Figure 5.8 Validation set for Watercut model (Bilayered Neural Network)

5.6.4 Dataset 4 (Combined) – Regression Learner App

Scenario 4 involves the examination of data originating from two distinct wells characterized by different GORs due to their physical separation caused by a fault. The dataset for Model 4 spans the entire production period, starting from the initiation of well operations until the commencement of water co-production alongside hydrocarbons. This dataset results from the merger of datasets 2 and 3, with the anticipated total number of data points being $421 + 498 = 919$. However, the actual combined dataset did not attain this anticipated size due to the removal of certain data points, resulting in a final count of 894 data points, as elaborated subsequently.

Machine learning models significantly benefit from datasets characterized by a smooth distribution, as it facilitates a more comprehensive understanding of underlying patterns and relationships. A smooth distribution implies minimal fluctuations in data values, aiding the model in effectively learning the data. In Dataset 2, where water production was absent, certain data points exhibited notably lower values for wellhead pressures, consequently leading to diminished hydrocarbon fractions. Specifically, these occurrences were observed

when the bottomhole pressure for well 1, the well with a lower Gas-Oil Ratio (GOR) of 300 scf/stb compared to the second well with GOR=500 scf/stb, ranged between 4500-5000 psi. This led to proportionally diminished oil and gas fractions. The rationale lies in the well with a higher liquid density, attributed to a lower dissolved gas content, where the pressure was inadequate to generate higher flowrates. As a result, hydrocarbon fractions were lower compared to other data points in Dataset 2 and significantly lower in relation to data points in Dataset 3.

On the contrary, Dataset 3, predominantly dedicated to water cut modeling, incorporated higher bottomhole pressures. The inclusion of small bottomhole pressures (ranging from 4500 to 5500 psi) combined with water cut would either result in zero production or extremely low values of flowrates.

In scenario 2, where only bottomhole pressures were modified while retaining data points with small hydrocarbon fractions, the machine learning model demonstrated effective training. The relatively smaller dataset, emphasizing a higher proportion of data points with small hydrocarbon fractions, positively contributed to the efficiency of the machine learning algorithm during the training process. However, the integration of data sets 2 and 3 presented challenges during the training of the machine learning model, mainly attributed to the significant contrast between the data points in Dataset 2 (characterized by low bottomhole pressures and hydrocarbon fractions) and the Dataset in set 3.

To overcome this challenge and facilitate effective training, data points in Dataset 2 with bottomhole pressures ranging between 4500-5000 psi for well 1 were excluded. As a result, Dataset 4 was formed, consisting of 894 data points. The exclusion of data points that might hinder the training of the combined dataset due to their contrasting nature allowed for a thorough feasibility analysis in the Regression Learner App. This approach enhances the machine learning model's ability to understand and capture the underlying relationships between variables, thereby leading to more effective and accurate predictions.

The Regression Learner App was employed to investigate the correlation between the gas fraction for well 1 and specific input variables, namely bottomhole pressures and GOR measured at the sinks on the platform. These particular variables were deliberately chosen due to their identified significant influence on the output. Following the execution of the models, the evaluation of the performance of each model was conducted. The resulting metrics are recorded in [Table 5.7](#). After the normalization process, [Table 5.8](#) was generated.

Model Type	RMSE (Validation)	MSE (Validation)	R ² (Validation)	MAE (Validation)	MAE (Test)	MSE (Test)	RMSE (Test)	R ² (Test)
Linear Regression	0.026	0.001	0.981	0.019	0.017	0.001	0.024	0.978
Stepwise Linear Regression	0.009	0.0001	0.997	0.005	0.005	0.0001	0.008	0.998
Fine Tree	0.006	0.00003	0.999	0.003	0.002	0.00001	0.003	1.000
Quadratic SVM	0.006	0.00004	0.999	0.005	0.004	0.00003	0.005	0.999
Ensemble (Boosted Trees)	0.019	0.0004	0.990	0.017	0.017	0.0003	0.018	0.988
Medium Neural Network	0.002	0.000005	1.000	0.001	0.002	0.00001	0.003	1.000
SVM Kernel	0.018	0.0003	0.991	0.009	0.008	0.0001	0.011	0.996

Table 5.7 Performance evaluation metrics of different machine learning models for Combined model

Model Type	RMSE (Validation)	MSE (Validation)	R ² (Validation)	MAE (Validation)	MAE (Test)	MSE (Test)	RMSE (Test)	R ² (Test)
Linear Regression	100.000%	100.000%	100.000%	100.000%	100.000%	100.000%	100.000%	100.000%
Stepwise Linear Regression	36.196%	13.101%	101.706%	28.881%	27.946%	9.718%	31.174%	102.013%
Fine Tree	22.865%	5.228%	101.861%	15.485%	13.006%	1.904%	13.800%	102.187%
Quadratic SVM	23.270%	5.415%	101.857%	24.517%	23.353%	4.653%	21.571%	102.126%
Ensemble (Boosted Trees)	73.562%	54.114%	100.901%	90.001%	94.634%	55.084%	74.219%	101.001%
Medium Neural Network	8.643%	0.747%	101.949%	7.511%	10.999%	1.408%	11.864%	102.198%
SVM Kernel	69.821%	48.750%	101.006%	49.631%	47.413%	18.666%	43.204%	101.813%



Table 5.8 Performance evaluation metrics of different machine learning models for Combined model (normalized values)

- **Linear Regression Stepwise Linear Regression:** The Linear Regression model demonstrates consistent and reliable performance with low RMSE, high R-squared, and low MAE on both the validation and test sets. The Stepwise Linear Regression model showcases improvements over the basic Linear Regression. With lower RMSE, higher R-squared and low MAE on both sets, this model offers increased precision in capturing the underlying patterns in the data.
- **Fine Tree:** The Tree model stands out with exceptional performance, displaying remarkably low RMSE and high R-squared values on both validation and test sets. Its ability to effectively capture intricate patterns makes it a promising choice for tasks that demand high predictive accuracy.
- **Quadratic SVM and Ensemble (Boosted Trees):** The SVM model exhibits robust predictive capabilities, demonstrating low RMSE, high R-squared and minimal MAE values. On the other hand, the Ensemble model delivers competitive performance, albeit with slightly elevated RMSE and lower R-squared compared to individual models.
- **Medium Neural Network:** The Neural Network emerges as the top-performing model, consistently delivering the lowest RMSE, highest R-squared, and lowest MAE values among all models. It is crucial to note the Neural Network's exceptional ability to capture intricate patterns and nonlinear relationships, making it the choice for further exploration.
- **SVM Kernel:** The Kernel model exhibits reasonable performance with moderately low RMSE and high R-squared values on both validation and test sets.

In [Table 5.7](#) and [Table 5.8](#), the presented performance metrics highlight the Neural Network as the optimal model for this dataset. Consistently exhibiting the lowest RMSE and MAE values, the Neural Network model signifies superior accuracy in predictions. Moreover,

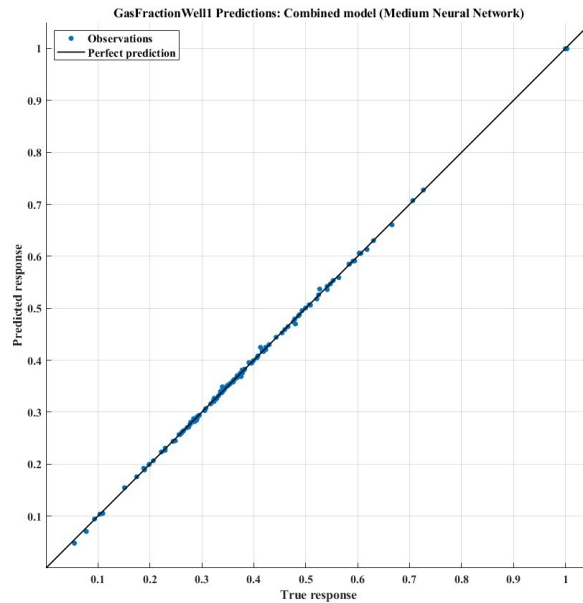


Figure 5.10 Test set for Combined model (Medium Neural Network)
for R^2 , the Neural Network model showcases the highest value, emphasizing its capability to capture the variance in the dataset effectively.

In Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής and Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής, the graphs illustrate the performance and evaluation results of the validation and test sets for data set 4, respectively.

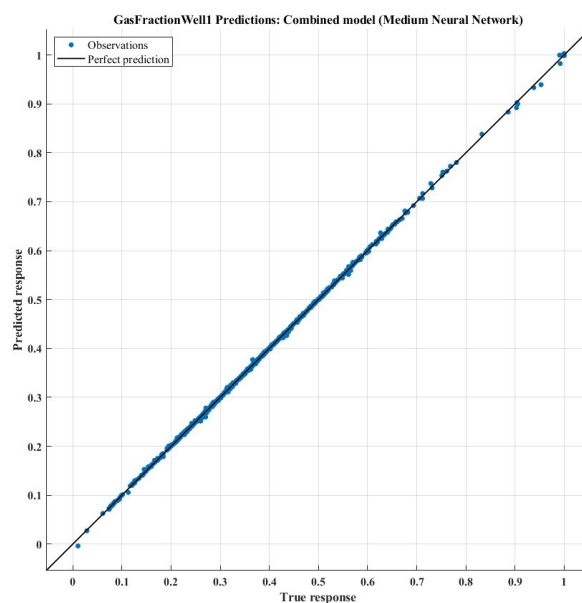


Figure 5.9 Validation set for Combined model (Medium Neural Network)

5.7 Model development

After concluding the preprocessing stage and conducting a feasibility analysis, it has been determined that neural networks stand out as the most appropriate machine learning model for all considered scenarios. The models exhibit the capability to effectively learn the datasets. The current chapter provides a detailed account of the process involved in constructing the finalized models of virtual multiphase flowmeters.

For this purpose, MATLAB's Neural Net Fitting App will be employed, serving as a potent tool for creating, visualizing, and training two-layer feed-forward networks tailored specifically for data fitting challenges. A noteworthy advantage of the Neural Net Fitting App's implementation is its flexibility, allowing the incorporation of the required number of multiple output and input variables during the modeling process. The Virtual Multiphase Flowmeters will be deployed to predict both oil and gas fractions as simultaneous outputs. To address the four distinct scenarios related to gas and oil production under varied conditions, the development of four distinct algorithms will be undertaken to represent the VFMs.

5.7.1 Dataset 1 ($GOR_1=GOR_2$) – Neural Net Fitting App

In the first scenario for Dataset 1, the goal was to develop a final neural network model and evaluate its performance in predicting hydrocarbon fractions of Well 1 based on wellhead pressures. The neural network utilized the Levenberg-Marquardt algorithm for training and underwent evaluation using a random data division approach. The model architecture comprised a two-layer feed-forward neural network with a hidden layer consisting of 20 neurons. The input variables for the model were the wellhead pressures ($P_{Wellhead1}$ and $P_{Wellhead2}$), and it aimed to predict the corresponding combined outputs of Gas Fraction Well₁ and Oil Fraction Well₁.

To evaluate the neural network model's performance, three distinct datasets were utilized: the training, validation, and test sets. The mean squared error (MSE) metric served as a measure of the accuracy of the predictions. The neural network demonstrated exceptional performance across all datasets, as indicated by an "R" value of 1.0000. This perfect positive linear relationship between the predicted values and the actual values implies that as the predicted values increase, the actual values also increase proportionally. The perfect correlation coefficient across all datasets suggests that the model's predictions align perfectly with the actual values in both the training and validation sets. While this alignment is logical considering that the data points are generated from simulations, it remains essential to

evaluate the model's generalization to new data and assess its performance in real-world scenarios.

The results indicate that the neural network model performed exceptionally well on all three datasets: training, validation, and test sets. For the training set with 290 observations, the achieved MSE was 1.0831e-06, and the correlation coefficient (R) was 1.0000, denoting a strong positive linear correlation. Similarly, on the validation set comprising 36 observations, the model attained an MSE of 4.7054e-07, along with an R value of 1.0000. These results affirm the model's capability to generalize effectively to previously unseen data. The reliability and accuracy of the neural network model are further demonstrated by its performance on the test set, encompassing 36 observations. On this dataset, the model achieved an MSE of 3.0972e-06 and an R value of 1.0000. These outcomes underscore the model's ability to accurately predict gas and oil fractions based on wellhead pressures.

(Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής)

	Observations	MSE	R
Training	290	1.0831e-06	1.0000
Validation	36	4.7054e-07	1.0000
Test	36	3.0972e-06	1.0000

Table 5.9 Training results for model $GOR_1=GOR_2$

Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής illustrates the performance of the neural network model across diverse datasets, including training, test, validation, and the overall dataset. The $Y=T$ line signifies the ideal alignment, where observed values coincide with predicted values. The fit line, representing the model's predictions, closely mirrors the line of perfect match ($Y=T$) for each dataset. In the training set diagram with 290 observations, data points align nearly perfectly along the $Y=T$ line, showcasing the model's precision in capturing the relationship between wellhead pressures and corresponding gas and oil fractions. This alignment persists in the validation and test set diagrams, where data points also closely follow the $Y=T$ line. The diagram for the entire dataset, combining observations from training, test, and validation sets, consistently demonstrates alignment with the $Y=T$ line. This analysis emphasizes the model's efficacy in predicting gas and oil fractions based on wellhead pressures across the dataset.

5.7.2 Dataset 2 ($GOR_1 \neq GOR_2$) – Neural Net Fitting App

In the second scenario for Dataset 2, the primary objective was to formulate a definitive neural network model and evaluate its efficacy in predicting hydrocarbon fractions of well 1 based on wellhead pressures. The training of the neural network employed the Levenberg-Marquardt algorithm. The model's performance underwent assessment through a random data division approach, ensuring an unbiased evaluation. The architecture of the two-layer FFNN featured a hidden layer comprising 20 neurons. This configuration utilized the wellhead

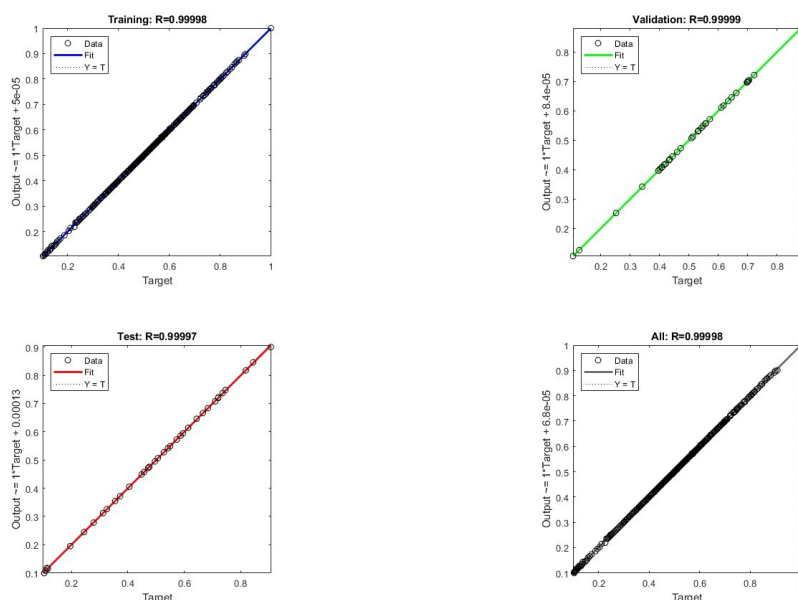


Figure 5.11 Oil and gas fractions predictions for Well 1 ($GOR_1 = GOR_2$ model) pressures, specifically $P_{Wellhead1}$ and $P_{Wellhead2}$, as input variables to predict the corresponding gas and oil fractions, represented as outputs GasFractionWell₁ and OilFractionWell₁.

The mean squared error (MSE) metric served as a quantitative measure to assess the accuracy of predictions across the training, validation, and test sets. The neural network demonstrated exceptional performance across all datasets. In the training set, encompassing 337 observations, the achieved MSE was 1.7919e-06, coupled with a correlation coefficient of 1.0000. This robust correlation coefficient suggests a strong positive linear relationship between the predicted and actual gas and oil fractions, underscoring the model's proficiency in capturing underlying relationships. Similarly, on the validation set, consisting of 42 observations, the model attained an MSE of 1.1322e-06, accompanied by an R value of

1.0000. These results imply that the model exhibits consistent performance without overfitting to the training data, showcasing its ability to generalize effectively to previously unseen data. Furthermore, the reliability and accuracy of the neural network were validated on the test set, comprising 42 observations. On this dataset, the model achieved an MSE of 1.4085×10^{-6} and an R value of 1.0000. These outcomes provide compelling evidence of the model's accuracy in predicting gas and oil fractions based on wellhead pressures, even when confronted with new data. (Table 5.10)

The consistent low MSE values and high correlation coefficients across training, validation and test datasets attest to the model's reliability and effectiveness in generalizing to new data.

	Observations	MSE	R
Training	337	1.7919×10^{-6}	1.0000
Validation	42	1.1322×10^{-6}	1.0000
Test	42	1.4085×10^{-6}	1.0000

Table 5.10 Training results for $GOR_1 \neq GOR_2$ model

The graphical plots in [Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής](#) depict the performance of the neural network model across different datasets. In the training set diagram, featuring 337 observations, data points align almost perfectly along the $Y=T$ line, highlighting the model's precision in capturing the relationship between wellhead pressures and corresponding gas and oil fractions. This alignment trend persists in the validation and test set diagrams, as well as in the overall set that combines observations from the training, test, and validation sets. This analysis underscores the model's effectiveness in predicting gas and oil fractions based on wellhead pressures across the entire dataset.

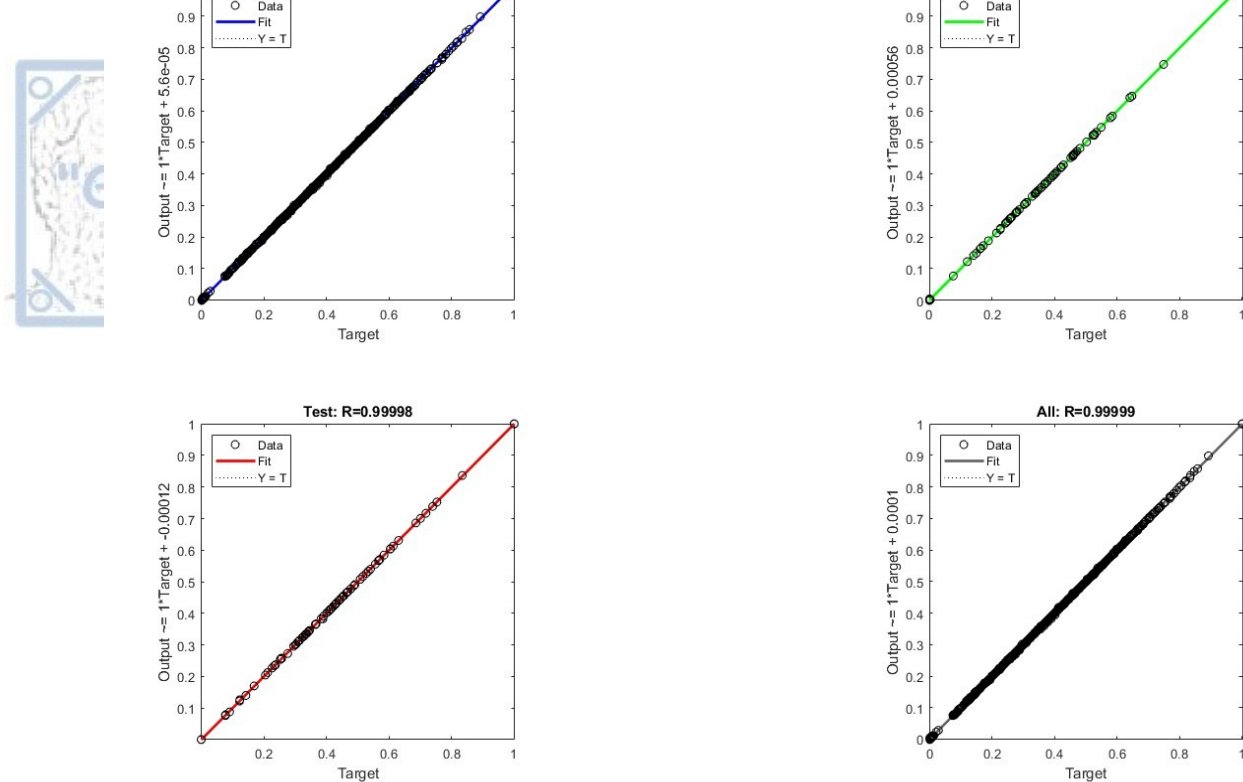


Figure 5.12 Oil and gas fractions predictions for Well 1 ($GOR_1 \neq GOR_2$ model)

5.7.3 Dataset 3 (Watercut) – Neural Net Fitting App

In the third scenario for Dataset 3, a neural network model was constructed to predict combined gas and oil fractions for Well 1. The model utilized three input variables, namely, the wellhead pressures ($P_{\text{Wellhead1}}$ and $P_{\text{Wellhead2}}$) and the produced Gas-Oil Ratio. The training process of the model employed the Levenberg-Marquardt algorithm, and its performance was evaluated using a random data division approach. The FFNN architecture was designed with a hidden layer comprising 20 neurons, enabling the model to effectively capture and learn the intricate relationships between the input variables and the target output.

The model was trained on a dataset comprising 348 observations. During training, the model achieved an impressively low mean squared error of $2.3822e-06$, indicating its exceptional accuracy in capturing the relationships between the input variables and the target outputs. Furthermore, the high correlation coefficient ($R = 0.9999$) between the predicted values and actual observations during training suggests a strong positive linear relationship, reinforcing the model's robustness. On the validation set, comprising 75 previously unseen observations, the model continued to perform well, yielding an MSE of $4.1854e-06$ and a high correlation coefficient ($R = 0.9999$). This outcome indicates that the model is able to maintain its accuracy on data outside the training set, further supporting its reliability. Similarly, when evaluated on the independent test set of 75 observations, the model exhibited consistent performance with an MSE of $6.4302e-06$ and a correlation coefficient ($R =$

0.9998). The combination of exceptional accuracy and consistency across all datasets underscores the model's strong predictive capability. (Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής)

	Observations	MSE	R
Training	348	2.3822e-06	0.9999
Validation	75	4.1854e-06	0.9999
Test	75	6.4302e-06	0.9998

The neural network model shows excellent performance in predicting gas and oil fractions for well 1, using wellhead pressures and GOR as input variables. The plots in

Table 5.11 Training results for Watercut model

Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής, demonstrate the model's precision and consistency, with data points closely aligning with the Y=T line across the training, validation and test sets. This high level of alignment and consistency in both training and validation sets further confirms the model's precision in its predictions.

5.7.4 Dataset 4 (Combined) – Neural Net Fitting App

In the fourth scenario for Combined Dataset, a neural network model was constructed to predict hydrocarbon fractions for Well 1. This model leveraged data from two distinct

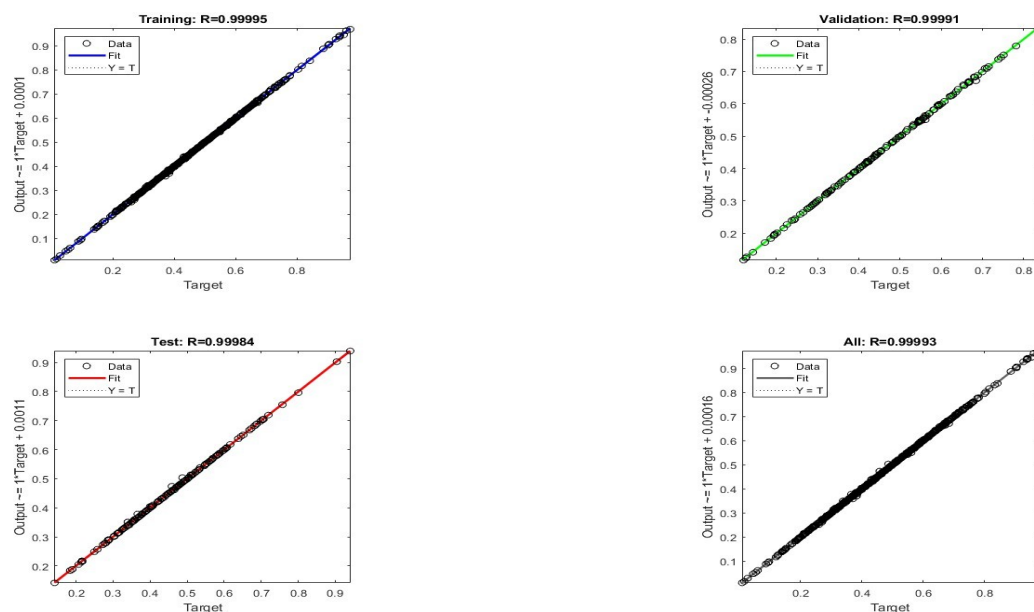


Figure 5.13 Oil and gas fractions predictions for Well 1 (Watercut model) datasets (2 and 3) and featured a two-layer FFNN. The selected input variables encompassed

wellhead pressures ($P_{\text{Wellhead1}}$ and $P_{\text{Wellhead2}}$) and the produced GOR, while the output variables specifically represented gas and oil fractions for Well 1.

The choice of the Levenberg-Marquardt algorithm as the training method for the neural network was based on its proven effectiveness in optimizing complex models. Given the greater abundance of data points and intricate relationships between variables in this dataset, a higher number of 30 neurons were selected for the feedforward neural network. This architectural decision aimed to facilitate precise forecasts by allowing the model to capture and represent the intricacies within the data effectively.

To assess the model's performance, a random data division approach was utilized, dividing the dataset into three subsets: training, validation and test sets. This division ensured that the model's ability to generalize to unseen data was evaluated while preventing overfitting.

The model demonstrated outstanding performance in predicting gas and oil fractions for Well 1, evident from the notably low Mean Squared Error values across all three subsets (Table 5.12). Specifically, the training set exhibited an impressively low MSE of $2.2951\text{e-}06$, signifying the model's effective capture of the underlying relationships within the training data. The correlation coefficient value of 1.0000 in the training set indicates a strong positive linear relationship between the predicted values and the actual values.

In the validation set, the MSE of $3.7141\text{e-}06$ and the correlation coefficient value of 1.0000 demonstrate the model's ability to generalize well to unseen data. Although it is important to note that a correlation coefficient of 1.0000 may not provide a complete assessment of the model's performance, it does indicate a strong linear relationship between the predicted values and the actual values in the validation set. For the test set, the MSE of $4.7601\text{e-}06$ and the correlation coefficient value of 0.9999 provide further evidence of the model's accuracy.

	Observations	MSE	R
Training	536	$2.2951\text{e-}06$	1.0000
Validation	179	$3.7141\text{e-}06$	1.0000
Test	179	$4.7601\text{e-}06$	0.9999

Table 5.12 Training results for Combined model

The neural network model exhibits remarkable accuracy in forecasting gas and oil fractions for well 1, using wellhead pressures and GOR as inputs. The graphical representations in [Σφάλμα: Δεν βρέθηκε η πηγή παραπομπής](#) provide clear evidence of the model's consistency and reliability across all datasets, including the training set, validation

set, test set and overall dataset. The data points closely aligning with the $Y=T$ line in the plot affirm the model's precision and further validate its performance in capturing the underlying relationships between the input and output variables.

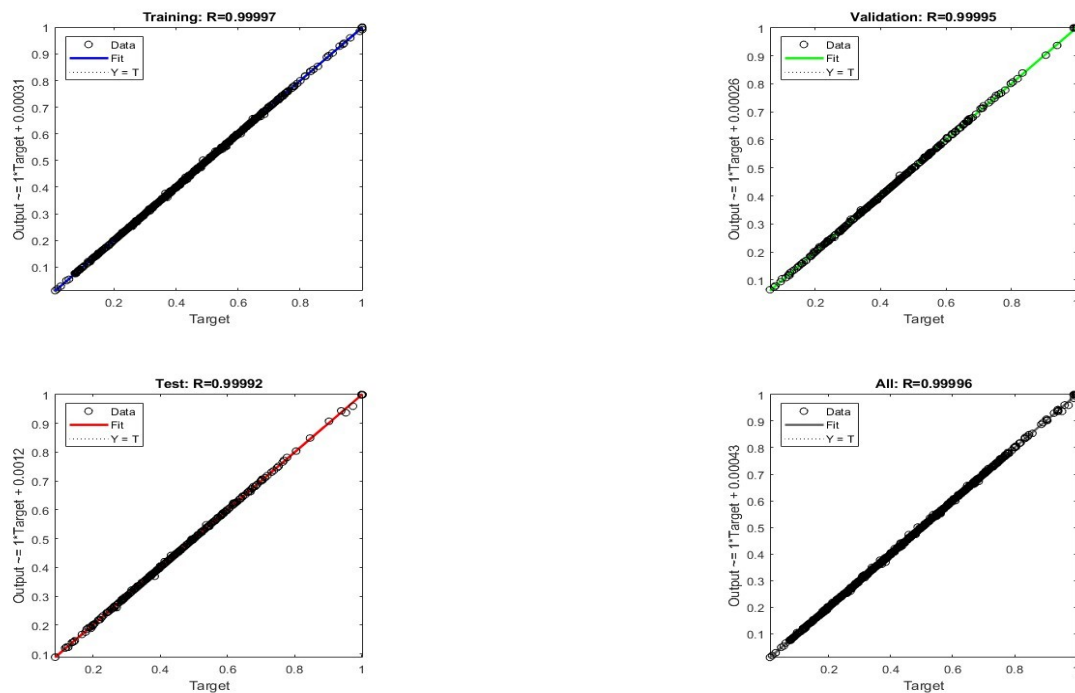


Figure 5.14 Oil and gas fractions predictions for Well 1 (Combined model)

Chapter 6 Conclusions

This thesis has demonstrated the remarkable potential of a simplified approach to phase separation in subsea well networks. Through the utilization of merely two wells and the sole reliance on total production measurements from a separator and the pressure on the two wellheads, complete phase separation was achieved without the necessity for subsea metering

equipment. The encountered flowrate errors, measured at a mere 2%, underscore the effectiveness of the proposed methodology.

The virtual multiphase flow meters implemented in this thesis underwent examination across various scenarios, encompassing cases with identical fluids, differing fluid compositions, as well as those with differing pressures. Across all scenarios, all the machine learning models exhibited excellent performance. Notably, among the models tested, neural networks consistently emerged as the most effective solution. Their superior performance underscores their suitability for addressing the complexities inherent in multiphase flow metering within the oil and gas industry.

The demonstrated effectiveness of the simplified approach to phase separation in subsea well networks suggests its potential scalability to accommodate larger and more intricate production systems. In the future, as operations expand to encompass more complex systems, such as subsea networks comprising of more wells or additional drain zones per well, the methodologies presented in this study will need to be adapted accordingly. Given the experience gained from this study, there is confidence in the ability to build upon and expand these methodologies to address the challenges posed by future developments in the field.

Ideally, the insights gained from this thesis could lead to the development of software beyond the capabilities of Pipesim, specifically tailored to address the challenges of multiphase flow metering in subsea fields. Such software could offer enhanced specificity and accuracy, potentially revolutionizing the field of subsea production optimization. However, to realize this vision, it is imperative to prioritize the validation and refinement of the methodologies proposed in this thesis using real data, ensuring their effectiveness and reliability in practical scenarios.

References

- Açıkgöz, M., Franca, F., & Lahey Jr, R. T. (1992). An experimental study of three-phase flow regimes. *International Journal of Multiphase Flow*, 18(3), 327-336.
- Al-Awadi, H. (2011). Multiphase characteristics of high viscosity oil (Doctoral dissertation).
- Al-Dogail, A. S., & Gajbhiye, R. N. (2021). Effects of density, viscosity and surface tension on flow regimes and pressure drop of two-phase flow in horizontal pipes. *Journal of Petroleum Science and Engineering*, 205, 108719.

- Ali, S. F. (2009). Two-phase flow in a large diameter vertical riser (Doctoral dissertation, Cranfield University).
- Al-Qutami, T. A., Ibrahim, R., & Ismail, I. (2017, September). Hybrid neural network and regression tree ensemble pruned by simulated annealing for virtual flow metering application. In 2017 IEEE International Conference on Signal and Image Processing Applications (ICSIPA) (pp. 304-309). IEEE.
- Andrade, G. M., de Menezes, D. Q., Soares, R. M., Lemos, T. S., Teixeira, A. F., Ribeiro, L. D., ... & Pinto, J. C. (2022). Virtual flow metering of production flow rates of individual wells in oil and gas platforms through data reconciliation. *Journal of Petroleum Science and Engineering*, 208, 109772.
- Andrianov, N. (2018). A machine learning approach for virtual flow metering and forecasting. *IFAC-PapersOnLine*, 51(8), 191-196.
- Bannwart, A. C., Rodriguez, O. M. H., Trevisan, F. E., Vieira, F. F., & De Carvalho, C. H. M. (2009). Experimental investigation on liquid–liquid–gas flow: Flow patterns and pressure-gradient. *Journal of petroleum science and engineering*, 65(1-2), 1-13.
- Barjoui, H. S., Ghorbani, H., Mohamadian, N., Wood, D. A., Davoodi, S., Moghadasi, J., & Saberi, H. (2021). Prediction performance advantages of deep machine learning algorithms for two-phase flow rates through wellhead chokes. *Journal of Petroleum Exploration and Production*, 11, 1233-1261.
- Bello, O., Ade-Jacob, S., & Yuan, K. (2014, April). Development of hybrid intelligent system for virtual flow metering in production wells. In *SPE Intelligent Energy Conference & Exhibition*. OnePetro.
- Berry, M. W., Mohamed, A., & Yap, B. W. (Eds.). (2019). *Supervised and unsupervised learning for data science*. Springer Nature.
- Bikmukhametov, T., & Jäschke, J. (2020). First principles and machine learning virtual flow metering: a literature review. *Journal of Petroleum Science and Engineering*, 184, 106487.
- Darzi, M., & Park, C. (2017, November). Experimental visualization and numerical simulation of liquid-gas two-phase flows in a horizontal pipe. In *ASME International Mechanical Engineering Congress and Exposition* (Vol. 58424, p. V007T09A011). American Society of Mechanical Engineers.
- Dong, F., Liu, X., Deng, X., Xu, L., & Xu, L. A. (2001). Identification of two-phase flow regimes in horizontal, inclined and vertical pipes. *Measurement Science and Technology*, 12(8), 1069.

- Falcone, G., Hewitt, G. F., Alimonti, C., & Harrison, B. (2001, September). Multiphase flow metering: current trends and future developments. In SPE annual technical conference and exhibition. OnePetro.
- Gryzlov, A., Mironova, L., Safonov, S., & Arsalan, M. (2021, October). Evaluation of machine learning methods for prediction of multiphase production rates. In SPE Symposium: Artificial Intelligence-Towards a Resilient and Efficient Energy Industry. OnePetro.
- Hibiki, T., & Tsukamoto, N. (2023). Drift-flux model for upward dispersed two-phase flows in vertical medium-to-large round tubes. *Progress in Nuclear Energy*, 158, 104611.
- Ibrahim, A. F., Al-Dhaif, R., Elkatatny, S., & Al Shehri, D. (2022). Machine Learning Applications to Predict Surface Oil Rates for High Gas Oil Ratio Reservoirs. *Journal of Energy Resources Technology*, 144(1).
- Ishak, M. A. B., Ismail, I. B., & Al-Qutami, T. A. H. (2021, July). Data Driven Versus Transient Multiphase Flow Simulator for Virtual Flow Meter Application. In 2020 8th International Conference on Intelligent and Advanced Systems (ICIAS) (pp. 1-4). IEEE.
- Ishak, M. A., Al-qutami, T. A. H., & Ismail, I. Virtual Multiphase Flow Meter using combination of Ensemble Learning and first principle physics based. *International Journal on Smart Sensing and Intelligent Systems*, 15(1), 1-21.
- Ishak, M. A., Hasan, A. Q., Aziz, T., Ellingsen, H., Ruden, T., & Khaledi, H. (2020, October). Evaluation of Data Driven Versus Multiphase Transient Flow Simulator for Virtual Flow Meter Application. In Offshore Technology Conference Asia. OnePetro.
- Jafarizadeh, F., Rajabi, M., Tabasi, S., Seyedkamali, R., Davoodi, S., Ghorbani, H., ... & Csaba, M. (2022). Data driven models to predict pore pressure using drilling and petrophysical data. *Energy Reports*, 8, 6551-6562.
- Javaid, A., Mohammed, A., & Ghaithan, A. (2022). A regression-based model for prediction of flowmeters calibration cost in oil and gas industry. *Flow Measurement and Instrumentation*, 86, 102191.
- Knotek, S., Fiebach, A., & Schmelter, S. (2016, September). Numerical simulation of multiphase flows in large horizontal pipes. In Proceedings of the 17th International Flow Measurement Conference FLOMEKO.
- Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, 160(1), 3-24.

- Lee, J. Y. , Ishii, M., and Kim, N. S. , 2008, "Instantaneous and Objective Flow Regime Identification Method for the Vertical Upward and Downward Co-Current Two-Phase Flow," *Int. J. Heat Mass Transfer*, 51, pp. 3442–3459.
- Liu, M., Kim, G., Bauckhage, K., & Geimer, M. (2022). Data-Driven Virtual Flow Rate Sensor Development for Leakage Monitoring at the Cradle Bearing in an Axial Piston Pump. *Energies*, 15(17), 6115.
- Lv, X., Zhang, D., Song, G., Su, G., Tian, W., & Qiu, S. (2023). Experimental study on flow patterns in narrow rectangular channels. *Progress in Nuclear Energy*, 156, 104562.
- Ma, Y., Li, C., Pan, Y., Hao, Y., Huang, S., Cui, Y., & Han, W. (2021). A flow rate measurement method for horizontal oil-gas-water three-phase flows based on Venturi meter, blind tee, and gamma-ray attenuation. *Flow Measurement and Instrumentation*, 80, 101965.
- Manami, M., Seddighi, S., & Örlü, R. (2023). Deep learning models for improved accuracy of a multiphase flowmeter. *Measurement*, 206, 112254.
- Mekisso, H. M. (2013). Comparison of frictional pressure drop correlations for isothermal two-phase horizontal flow. Oklahoma State University.
- Mercante, R., & Netto, T. A. (2022). Virtual flow predictor using deep neural networks. *Journal of Petroleum Science and Engineering*, 213, 110338.
- Mohammed, S. K., Hasan, A. H., Ibrahim, A., Dimitrakakis, G., & Azzopardi, B. J. (2019). Dynamics of flow transitions from bubbly to churn flow in high viscosity oils and large diameter columns. *International Journal of Multiphase Flow*, 120, 103095.
- Mohammed, S. K., Hasan, A., Dimitrakakis, G., & Azzopardi, B. J. (2018). Churn flow in high viscosity oils and large diameter columns. *International Journal of Multiphase Flow*, 100, 16-29.
- Montoya, G., Lucas, D., Baglietto, E., & Liao, Y. (2016). A review on mechanisms and models for the churn-turbulent flow regime. *Chemical Engineering Science*, 141, 86-103.
- Mu, J., McArdle, S., Ouyang, J., & Wu, H. (2023). Single well virtual metering research and application based on hybrid modeling of machine learning and mechanism model. *Journal of Pipeline Science and Engineering*, 100111.
- Mukherjee, H., & Brill, J. P. (1999). Multiphase flow in wells. Memorial Fund of AIME, Society of Petroleum Engineers Inc.: Richardson, TX, USA.
- Mukherjee, T., Das, G., & Ray, S. (2014). Sensor-based flow pattern detection—gas–liquid–liquid upflow through a vertical pipe. *AIChE Journal*, 60(9), 3362-3375.

- Naidek, B. P., Kashiwakura, L. Y., Alves, R. F., Bassani, C. L., Stel, H., & Morales, R. E. (2017). Experimental analysis of horizontal liquid-gas slug flow pressure drop in d-type corrugated pipes. *Experimental Thermal and Fluid Science*, 81, 234-243.
- Najmi, K., McLaury, B. S., Shirazi, S. A., & Cremaschi, S. (2015). Experimental study of low concentration sand transport in wet gas flow regime in horizontal pipes. *Journal of Natural Gas Science and Engineering*, 24, 80-88.
- Oyelade, A. O., & Oyediran, A. A. (2021). Nonlinear dynamics of horizontal pipes conveying two phase flow. *European Journal of Mechanics-A/Solids*, 90, 104367.
- Pagan, E., Williams, W. C., Kam, S., & Waltrich, P. J. (2017). A simplified model for churn and annular flow regimes in small-and large-diameter pipes. *Chemical Engineering Science*, 162, 309-321.
- Perez, V. H. (2007). Gas-liquid two-phase flow in inclined pipes. University of Nottingham: Nottingham, UK.
- Quintino, A. M., da Rocha, D. L. L. N., Fonseca Júnior, R., & Rodriguez, O. M. H. (2021). Flow pattern transition in pipes using data-driven and physics-informed machine learning. *Journal of Fluids Engineering*, 143(3).
- Roxas II, R., Evangelista, M. A., Sombillo, J. A., Nnabuife, S. G., & Pilario, K. E. (2022). Machine Learning Based Flow Regime Identification using Ultrasonic Doppler Data and Feature Relevance Determination. *Digital Chemical Engineering*, 3, 100024.
- Schmid, D., Verlaat, B., Petagna, P., Revellin, R., & Schiffmann, J. (2022). Flow pattern observations and flow pattern map for adiabatic two-phase flow of carbon dioxide in vertical upward and downward direction. *Experimental Thermal and Fluid Science*, 131, 110526.
- Shaban, H., & Tavoularis, S. (2014). Measurement of gas and liquid flow rates in two-phase pipe flows by the application of machine learning techniques to differential pressure signals. *International journal of multiphase flow*, 67, 106-117.
- Song, S., Wu, M., Qi, J., Wu, H., Kang, Q., Shi, B., ... & Gong, J. (2022). An intelligent data-driven model for virtual flow meters in oil and gas development. *Chemical Engineering Research and Design*, 186, 398-406.
- Spedding, P. L., Donnelly, G. F., & Cole, J. S. (2005). Three phase oil-water-gas horizontal co-current flow: I. Experimental and regime map. *Chemical Engineering Research and Design*, 83(4), 401-411.

- Thome, J. R., & Cioncolini, A. (2016). Two-phase flow pattern maps for microchannels. In Encyclopedia of Two-Phase Heat Transfer and Flow I: Fundamentals and Methods (pp. 47-84).
- Thome, J. R., & Cioncolini, A. (2017). Flow boiling in microchannels. In Advances in Heat Transfer (Vol. 49, pp. 157-224). Elsevier.
- Van der Meulen, G. P. (2012). Churn-annular gas-liquid flows in large diameter vertical pipes (Doctoral dissertation, University of Nottingham).
- Vanvik, T., Henriksson, J., Yang, Z., & Weisz, G. (2022, October). Virtual flow metering for continuous real-time production monitoring of unconventional wells. In Unconventional Resources Technology Conference, 20–22 June 2022 (pp. 739-746). Unconventional Resources Technology Conference (URTeC).
- Wambsganss, M. W., Jendrzeczyk, J. A., & France, D. M. (1991). Two-phase flow patterns and transitions in a small, horizontal, rectangular channel. International journal of multiphase flow, 17(3), 327-342.
- Woods, G. S., Spedding, P. L., Watterson, J. K., & Raghunathan, R. S. (1998). Three-phase oil/water/air vertical flow. Chemical Engineering Research and Design, 76(5), 571-584.
- Xu, Q., Wang, X., Luo, X., Tang, X., Yu, H., Li, W., & Guo, L. (2022). Machine learning identification of multiphase flow regimes in a long pipeline-riser system. Flow Measurement and Instrumentation, 88, 102233.
- Xu, X. X. (2007). Study on oil–water two-phase flow in horizontal pipelines. Journal of Petroleum Science and Engineering, 59(1-2), 43-58.
- Yang, K., Liu, J., Wang, M., Wang, H., & Xiao, Q. (2023). Identifying flow patterns in a narrow channel via feature extraction of conductivity measurements with a support vector machine. Sensors, 23(4), 1907.
- Yang, Y., Guo, J., Ren, B., Zhang, S., Xiong, R., Zhang, D., ... & Fu, S. (2019). Oil-Water flow patterns, holdups and frictional pressure gradients in a vertical pipe under high temperature/pressure conditions. Experimental Thermal and Fluid Science, 100, 271-291.
- Zimmer, M. D., & Bolotnov, I. A. (2019). Slug-to-churn vertical two-phase flow regime transition study using an interface tracking approach. International Journal of Multiphase Flow, 115, 196-206.