A Neural Network Model for Prediction of Liquid Holdup in Two-Phase Horizontal Flow

Mack E. Shippen, SPE, Schlumberger, and **Stuart L. Scott,** SPE, Texas A&M U.

Summary

Accurately predicting the liquid holdup associated with multiphase flow is a critical element in the design and operation of modern production systems. This prediction is made difficult by complex phase distributions and the wide range of fluid properties encountered in production operations. Consequently, the performance of existing correlations is often inadequate in terms of desired accuracy and application range. This investigation focuses on the development of a neural network model, a relatively new approach that has been applied successfully to a variety of complex engineering problems. Data from five independent studies were used to develop a neural network for predicting liquid holdup in two-phase horizontal flow. A detailed comparison with existing empirical correlations and mechanistic models reveals that for this data set, the neural network model shows an improvement in overall accuracy and performs more consistently across the range of liquid-holdup and flow patterns.

Introduction

Two-phase flow of gas and liquid in pipes is a frequent occurrence in the petroleum industry. In offshore developments, advancements in subsea processing and multiphase pumping and metering technology have extended multiphase flow across greater distances to centralized production platforms. These developments are becoming increasingly common, especially in remote and hostile locations, such as the deepwater Gulf of Mexico and offshore west Africa.

The design of multiphase production systems requires an accurate estimation of pressure loss.[1,2] Liquid holdup, defined as the in-situ liquid volume fraction, is generally the most important parameter in calculating pressure loss. Liquid holdup is also necessary to predict hydrate formation and wax deposition and to estimate the liquid volume expelled during pigging operations for sizing slug catchers.

While pressure losses in single-phase flow in pipes have long been accurately modeled with familiar expressions, such as the Bernoulli and Navier-Stokes equations, accurate predictions of pressure loss in two-phase flow have proved to be more challenging because of added complexities. The lower density and viscosity of the gas phase causes it to flow at a higher velocity relative to the liquid phase, a characteristic known as slippage. Consequently, the associated frictional pressure losses result from shear stresses encountered at the gas/liquid interface as well as along the pipe wall. Additionally, the highly compressible gas phase expands as the pressure decreases along the flow path.

Further complicating matters are the variety of physical phase distributions that are characterized by flow regimes or flow patterns (see **Fig. 1**). The prevailing flow pattern for a specific set of conditions depends on the relative magnitude of the forces acting on the fluids. Buoyancy, turbulence, inertia, and surface-tension forces are greatly affected by the fluids' relative flow rates, viscosities, and densities as well as the pipe diameter and inclination angle. The complex dynamics of the flow pattern govern slippage effects and, therefore, variations in liquid holdup and pressure gradient.

Many empirical correlations and mechanistic models have been proposed to predict liquid holdup and pressure loss.[1,2] Some are very general, while others apply only to a narrow range of conditions. Many of these approaches begin with a prediction of the flow pattern, with each flow pattern having an associated method of predicting liquid holdup. The liquid-holdup prediction is used to determine a two-phase friction factor from which a pressure gradient is calculated. One of the problems with this approach is that is is dependent on the accuracy of flow-pattern predictions and subject to discontinuities in predictions made across flow-pattern transition boundaries.[3] Comparative studies[4,5] have shown that these models perform inconsistently as flow conditions change. This limitation makes the task of selecting the most appropriate flow correlation very challenging.

Neural networks have been shown to exhibit superior performance vs. conventional approaches in a variety of problems in petroleum engineering.[6–8] In particular, data-intensive problems, characterized by a complex interaction of variables, are good candidates for this approach.

A previous study by Ternyik *et al.*[9] explored the application of neural networks in predicting the flow pattern and liquid holdup on the basis of experimental data collected by Mukherjee.[10] The liquid-holdup model used 11 input parameters consisting of inclination angles, oil and gas flow rates, oil specific gravity, inlet and outlet pressures, inlet temperature, and four binary data, each representing a flow pattern predicted by a separate neural network model. While the results were encouraging, the complexity of the model (37 hidden nodes) relative to the number of training cases (approximately 1,400) yields a low case/connection ratio (approximately 3:1), and the relatively small number of test cases (10%) may limit its ability to generalize.

Osman[11] developed neural network models to predict the flow pattern and liquid holdup using data from Minami and Brill[12] and Abdul-Majeed.[13] The liquid-holdup model uses four input parameters—gas and liquid superficial velocities, pressure, and temperature, and contains 12 hidden nodes. The network was trained using 150 training cases, with 49 test cases set aside to evaluate network performance. Comparison with a number of other correlations indicated that the neural network model was most accurate in predicting liquid holdup. As with Ternyik's model, Osman's neural network has a low training-case/connection ratio (2.5:1), which may impair generalization. Also, the use of pressure and temperature as input parameters rather than derived fluid properties limits its application to fluids with properties similar to those used in the experimental work from which the training data were taken (kerosene and water).

The objective of this study is to develop a neural network model capable of predicting liquid holdup in two-phase horizontal flow. Further details of this work can be found in Ref. 14.

Background: Neural Networks

Artificial neural networks were first conceptualized in the 1940s as research in artificial intelligence attempted to replicate the learning capabilities of biological neural systems by modeling the low-level (most basic) structure of the brain. However, difficulties in developing the necessary learning algorithms and limitations in com-

This paper (SPE 87682) was revised for publication from paper SPE 77499, first presented at the 2002 SPE Annual Technical Conference and Exhibition, San Antonio, Texas, 29 September–2 October. Original manuscript received for review 20 January 2003. Revised manuscript received 17 November 2003. Paper peer approved 29 December 2003.

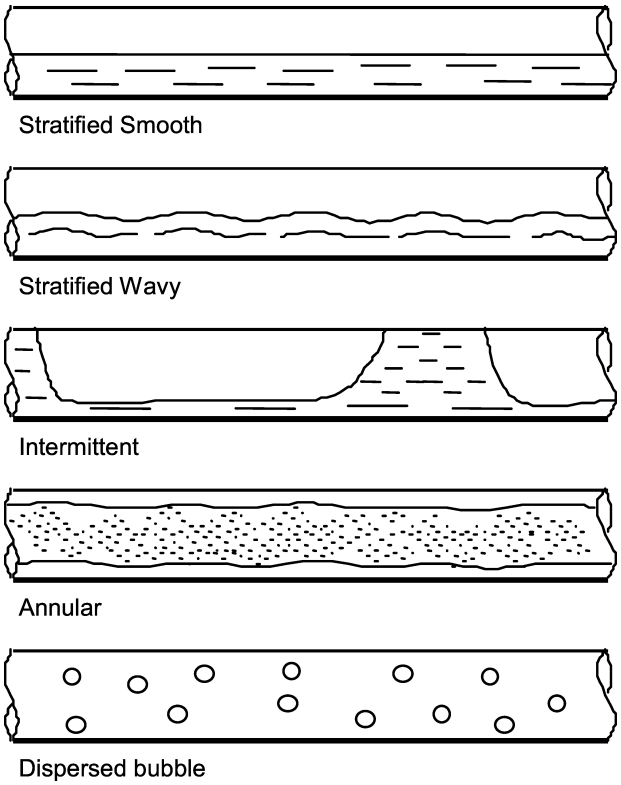

Fig. 1—Horizontal flow patterns.

puter processing ability shifted the focus toward expert systems in the 1960s. Contrary to neural networks, expert systems are based on a high-level model of reasoning processes. It was thought that programming these systems with a complex array of simple rules would generate intelligent computers capable of logic and reasoning. Although such systems appeared very useful in certain applications, it soon became evident that an extreme amount of programming was required to solve the simplest problems and that expert systems were essentially incapable of learning. Breakthrough work in the early 1980s refocused attention toward artificial neural networks.

Scientists estimate that the human brain consists of as many as 100 billion highly interconnected neurons.[15] The neuron itself is a specialized cell that relays electrochemical signals. It consists of branching input structures called dendrites, a cell body, and branching output structures called axons. An axon of one cell connects to a dendrite of another through a synapse. When a neuron is activated, it produces an electrochemical pulse along the axon. This signal crosses the synapses to other neurons, which may also be activated. To activate the neuron, the total signal received at the cell body from the dendrites must exceed a certain level, or threshold. It is believed that learning occurs as the strength of these synaptic connections is altered. Thus, from a complex architecture of extremely simple processing units, the brain is capable of performing extremely sophisticated tasks.

The Basic Artificial Model. There are several types of artificial neural networks,[15–17] the most common of which is the multilayer perceptron used in this study. Multilayer perceptrons consist of groups of interconnected nodes (perceptrons) arranged in layers corresponding to input, hidden, and output nodes.

As shown in **Fig. 2,** every input and output variable is represented by an individual node that is connected to all nodes in the hidden layer. Through training, these connections are assigned independent weighting factors. The input to each node is multiplied by its associated weighting factor, then summed with the product of each of the other input nodes and their respective weighting factors. An activation threshold is then subtracted from this sum, and the result is processed by a nonlinear transform function within the node. The most common transform function (and the one used in this study) is the logistic function (Eq. 2), which is an S-shaped sigmoid function. The logistic function provides nonlinearity to the model and constrains the node's output signal to a fixed range (0 to 1). It is also smooth and easily differentiable, characteristics that facilitate network training algorithms.

In most cases, the input data to the neural network must be conditioned, or preprocessed. Input variables of differing orders of magnitude are multiplied by a linear scaling factor to convert them to a range suitable for the transform function. In some instances, nonlinear scaling may be appropriate, as in the case of an exponentially distributed variable. Similarly, the output signals produced by the network are post-processed to generate meaningful results.

Mathematically, the artificial neural network can be viewed as a type of input/output model, with the weights and thresholds serving as the free parameters of the model and the number of layers and nodes determining its complexity.

Network Training. The neural network is trained using input data with corresponding known (experimentally measured) outputs in a process called supervised learning. The neural network learns to

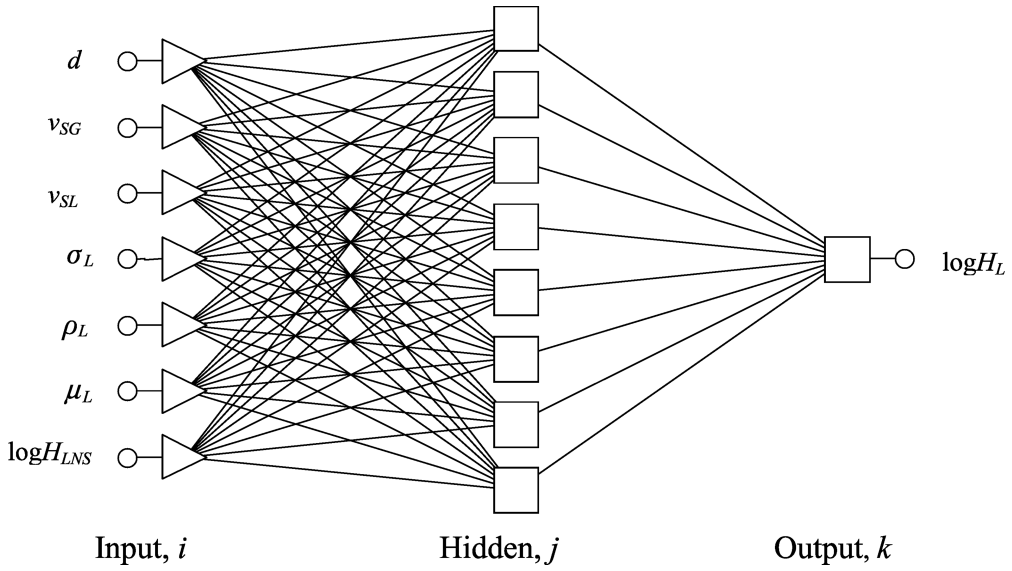

Fig. 2—Illustration of the liquid-holdup neural network.

infer the relationship between the two by iteratively adjusting the weighting factors in a two-stage propagate/adapt cycle.

In the first stage of this cycle, the input values are propagated through each layer of the network until an output is generated. The weighting factors are initially random, so the input values are transformed to output but with no meaningful pattern. These outputs are then compared to the desired one, and an error signal is calculated. This process is known as a feed-forward routine.

In the second stage, the error signals feed back through the network, and the weighting factors are corrected somewhat according to the back-propagation algorithm. This feed-forward/back-propagation cycle is iteratively executed until the weighting factors converge on values that minimize the average root mean square (ARMS) error (see Eq. 4) for all cases within the data set.

Once the initial training is complete, the weighting factors are held constant, and the predictive accuracy of the network is evaluated using only the feed-forward routine. Depending on how much known data are available, the network architecture and training procedures can be adjusted until the performance is satisfactory.

Generalization. The most important characteristic of a neural network is its ability to generalize accurately when new cases are submitted to it. The problem with the training approach discussed previously is that it is inherently empirical. Without a perfect and infinitely large set of training data, the true error of the underlying model is not necessarily minimized.

The most obvious manifestation of this problem is overlearning or overfitting, which occurs when the network begins to model the noise associated with the training data. **Fig. 3a** shows an example in which the model adopted a highly eccentric shape that is not related to the underlying function.[16]

A network with more free parameters represents a more complex function and may be more susceptible to overfitting. A network with fewer free parameters may not be powerful enough. Thus, the network's degree of complexity represents a compromise between the desired accuracy of the model and the ability to generalize well.

The solution to this issue is to apply cross-verification. That is, set some of the cases aside during training, and use these to keep an independent check on the behavior of the training algorithm. Fig 3b shows that as training progresses, the error for both the training and verification data sets declines. When the verification error quits falling or begins to rise, the network may be overfitting the data and training should stop.[16]

Still, the verification data set does actually factor into selecting a model, meaning that it is part of the training process. Therefore, it is normal practice to reserve a test data set to add further confidence in the final model's performance. The key assumption in this process is that all data sets independently represent the underlying model.

Selection of Input Variables. The variable selection process is similar to that used in dimensional analysis.[18] All factors that can conceivably influence the output are identified on the basis of theoretical significance. Like the human brain, neural networks are fault-tolerant in that the relationship between the input and output variables can be noisy, and a certain amount of data can be missing, but an underlying relationship must still exist.

Another important consideration that frequently presents difficulties is that each additional input variable adds another spatial dimension to the network complexity. As the number of variables increases, the number of cases required to properly model the added complexity increases nonlinearly, a problem known as the curse of dimensionality. One rule of thumb suggests that the number of training cases be at least 10 times the number of connections in the network; however, this depends on the complexity of the underlying function. Therefore, network performance can be improved by eliminating unnecessary input variables to placate the curse of dimensionality.

Development of the Liquid-Holdup Neural Network

Many of the empirical liquid-holdup correlations in use today were developed using data collected under a very limited range of flow conditions. Consequently, when these correlations are applied to predict liquid holdup for other conditions, the accuracy is often significantly reduced.[4,5] In light of this observation, the current study is based on five independant experimental data sets, encompassing a greater range of flow variables.

Experimental Data. Table 1 summarizes the overall range of flow conditions in the experimental data, a total of 627 liquid-holdup measurements. **Table 2** lists the flow-pattern classification based on the Xiao[19] mechanistic flow-pattern-transition criterion (a refinement of Taitel-Dukler[20]). Note that only two points fall within the dispersed-bubble region. Most models assume that liquid holdup in dispersed-bubble flow is equal to the no-slip liquid holdup. **Table 3** classifies the experimental data on the basis of the liquid-holdup range. Investigators tended to focus more on collecting data in the low-holdup range, perhaps to compensate for the increased measurement error as liquid holdup approaches zero.[5]

In all investigations, the experimental data include measurements of flow rates for each phase, temperature, and pressure at the location of sample collection and liquid holdup.

Mukherjee[10] measured liquid holdup with a capacitance meter, whereas others physically measured the liquid trapped with quick-closing ball valves. Additionally, Eaton,[21] Beggs,[22] and Mukherjee[10] measured pressure loss, and Mukherjee reported the observed flow pattern.

The data reported by Mukherjee,[10] Minami and Brill,[12] and Abdul-Majeed[13] included laboratory analysis of the testing fluids. Regression analysis was used to develop equations to estimate surface tension, liquid density, and viscosity. Eaton provided laboratory analysis of the gas composition, gas and liquid viscosities, specific gravity at 80°F, and surface tension at 80°F. In this study, general fluid-property correlations from Beggs[23] were used to es-

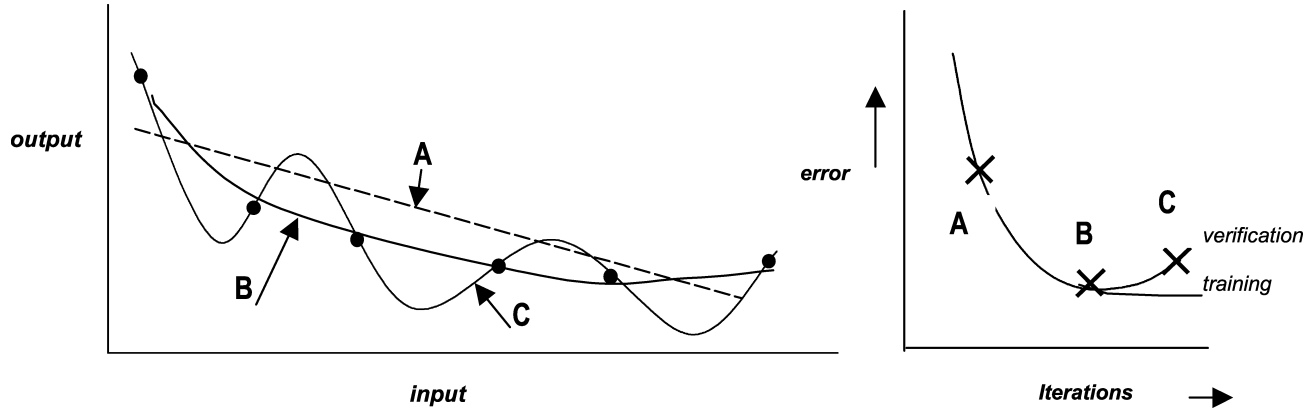

Fig. 3—(a) Illustration of neural network overlearning. (b) Stopping conditions.

TABLE 1—SUMMARY OF EXPERIMENTAL DATA

Investigator	No. Tests	Pipe Material	Pipe Diameter (in.)	Pressure (psia)	Measured Liquid Holdup
Eaton [21]	128	Steel	2.00	305 to 865	0.012 to 0.732
	110	Steel	4.00	305 to 865	0.006 to 0.728
Beggs[22]	30	Acrylic	1.00	35 to 95	0.017 to 0.856
	28	Acrylic	1.50	35 to 95	0.016 to 0.828
Mukherjee[10]	74	Steel	1.50	28 to 92	0.02 to 0.92
	57	Steel	1.50	40 to 89	0.03 to 0.99
Minami and Brill[12]	57	Steel	3.068	44 to 93	0.0095 to 0.4354
	54	Steel	3.068	46 to 80	0.0082 to 0.4515
Abdul-Majeed[13]	89	Acrylic	2.00	29 to 133	0.009 to 0.61
All Sets	627		1 to 4	28 to 865	0.006 to 0.99

Investigator	Fluids (liquid/gas)	Liquid Viscosity (cp)	Surface Tension (lbf/ft×1,000)	Liquid Density (lbm/ft^3)
Eaton[21]	Water-natural gas	0.71 to 1.33	4.22 to 4.56	62.9 to 63.6
Beggs[22]	Water-air	0.89 to 1.60	4.69 to 4.84	62.1 to 62.6
	Water-air			
Mukherjee[10]	Kerosene-air	0.92 to 2.05	1.55 to 1.79	49.1 to 51.1
	Lube oil-air	20.2 to 74.4	2.31 to 2.57	52.6 to 54.1
Minami and Brill[12]	Kerosene-air	1.34 to 1.99	1.78 to 1.94	49.4 to 50.2
	Water-air	0.58 to 0.92	4.68 to 4.94	62.5 to 63
Abdul-Majeed[13]	Kerosene-air	1.29 to 1.98	1.61 to 1.80	49.3 to 50.2
All Sets		0.58 to 74.4	1.55 to 4.94	48.3 to 63.6

timate properties at other conditions. In some cases, the estimates were normalized with these base measurements. The Beggs-Brill[22] investigation used water and air; thus, no laboratory analysis was performed, and fluid properties were estimated with the general correlations.

Optimal Network Selection. In deference to the curse of dimensionality, candidate input variables were first evaluated in terms of theoretical significance. Several variables with negligible effects on liquid holdup were ruled out on the basis of extensive experimental and theoretical work performed by Ros.[24] These variables include gas density, gas viscosity, and wall roughness. Sensitivity analysis based on preliminary network models showed the influence of these variables to be weak, further justifying their exclusion. The remaining variables include:

- Pipe inside diameter (ID) (d)
- Superficial liquid velocity (v_{SL})
- Superficial gas velocity (v_{SG})
- Liquid viscosity (μ_L)
- Liquid density (ρ_L)
- Liquid surface tension (σ_L)

The experimental data were then divided into training, verification, and test data sets. The ratio of this division represents a compromise between the desired accuracy for predictions and the ability to generalize. Much of the development process in this study used a 2:1:1 ratio of training, verification, and test cases, which is suggested as a good starting point by Bishop.[17] Initial results indicated that the network models displayed good generalization characteristics, so ratios of 3:1:1 and finally 4:1:1 (suggested by Haykin[15]) were then used.

Fortunately, the tedium of systematically testing a large number of network types and architectures was accomplished by a robust search algorithm included in the software used in this study—STATISTICA Neural Networks.[25] Initially, several network types were considered, including linear networks, generalized regression neural networks, radial base function networks, and multilayer perceptrons (both three- and four-layered). Early results indicated that the three-layer perceptron was most appropriate for this problem; therefore, the remainder of the study focused on this network type.

The number of networks to consider and, consequently, the length of time to specify for the optimal network search to operate depends on the complexity of the problem. In this study, testing the 1,000 to 3,000 neural-network candidates to determine the optimal network required 4 to 6 hours with a 500-MHz processor.

The actual training process involved 50 epochs (cycles) of the back-propagation training algorithm to locate the approximate position of the local minimum error. The conjugate-gradient-descent training algorithm was then applied until stopping conditions were met. Both algorithms converged to minimize the ARMS error (Eq. 4) for the training set. The training process was halted when the ARMS error for the verification set began to increase. The ARMS error is relatively insensitive to errors in the low range of liquid holdup and strongly affected by errors in the high range. Consequently, the predictive accuracy of the neural networks lessened as the liquid holdup decreased. Coupled with the fact that the amount of data collected increases as the measured liquid holdup decreases, it became obvious that the model was biased toward data in the higher ranges of liquid holdup.

TABLE 2—OCCURRENCE OF FLOW PATTERNS IN EXPERIMENTAL DATA

Flow Pattern	No. Points
Stratified smooth	39
Stratified wavy	311
Intermittent	157
Dispersed bubble	2
Annular	116

TABLE 3—LIQUID-HOLDUP RANGE IN EXPERIMENTAL DATA

Holdup Range	No. Points
$H_L > 0.1$	179
$0.1 > H_L > 0.2$	132
$0.2 > H_L > 0.4$	158
$0.4 > H_L$	158

TABLE 4—SCALING FACTORS, WEIGHTS, AND THRESHOLDS

*Weight vector, **w***

		j							
		1	2	3	4	5	6	7	8
	1	-2.646518	0.08676	0.5665431	0.9411823	1.837024	1.533598	-0.6396	3.120925
	2	-1.691418	-1.116738	0.8521293	-4.742355	-1.316329	-0.771186	2.278877	-0.9218
	3	-1.040515	-0.164	-0.1242	-0.6824	-1.836985	-0.3562	-0.8473	-0.2979
i	4	0.4478271	0.4337765	0.9611095	-1.266503	0.3729641	-2.348444	0.5887969	0.6455306
	5	-0.6928	-0.3537	1.144298	1.287419	-0.9221	1.541613	2.101707	-0.7992
	6	-0.1145	-3.099379	0.2930244	-1.802052	0.7501607	0.7043367	0.7358667	-0.08714
	7	-0.5922	-0.2966	0.2369225	0.7014839	-1.406069	-3.431447	-1.476618	0.1308323
k	1	1.71037	-1.69940	-1.37768	2.41212	-2.04494	-1.94517	1.06769	2.99608

Thresholds, κ

	j								k
	1	2	3	4	5	6	7	8	1
κ	-1.736955	0.4035331	1.194781	2.973028	-1.948319	0.9083626	0.005958	0.7877318	-0.3812

Pre-Process scaling vector

x	ξ	missing input
1	3.9656312x - .330469265	0.45827
2	.006291580x - 0.00188748	0.13682
3	0.0765495x - .00050229	0.13088
4	297.78859x - 0.4713383	0.526299
5	.06990314x - 3.4457848	0.57355
6	.013859205x - 0.00812721	0.053378
7	.295893539x + 1.0872692	0.61753

Post-Process scaling factor

	ζ
LOG (H_L)	2.18272o - 2.187087

To address this, the logarithm of the liquid holdup was used as an output to increase the data spread in the low-liquid-holdup range. The result is a model with better overall performance and only a slight performance loss in the higher liquid-holdup range, particularly in the intermittent-flow-pattern cases. Because the no-slip liquid holdup is highly correlated with the actual liquid holdup, the logarithm of no-slip liquid holdup was also used as an input.

Final Neural Network Model for Liquid Holdup. The final neural network model is illustrated in Fig. 2. It contains 7 input nodes, 8 hidden nodes, and 1 output node, giving a total of 64 connections and a case-to-connection ratio of approximately 8:1. The network training applied 50 epochs of back propagation followed by 725 epochs of conjugate gradient descent. The weights, thresholds, and scaling factors associated with this model appear in **Table 4.** Additionally, constraints imposed on the predicted liquid holdup used no-slip liquid holdup and unity as limiting values.

Table 5 shows a comparison of the errors based on training, verification, and test cases. While the correlation coefficient is slightly lower for the test case, the overall error analysis among the cases shows comparable performance, indicating that the neural network did not suffer from overlearning.

Mathematical Representation of Neural Network. The neural network model can be expressed as a nested scheme, written in the following compact form.

$$\log(H_L)' = F(\boldsymbol{x}',\boldsymbol{w}) = \left\{ \sum_j^M w_{kj}\varphi_j \left[\left(\sum_i^N w_{ji}x_i \right) - \kappa_j \right] \right\} - \kappa_k, \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

in which w_{kj} = the synaptic weights from neurons in the hidden layer, j, to the output neuron, k; w_{ji} = the synaptic weights from neurons in the input layer, i, to the neurons in the hidden layer, j; and x_i = the i-th element of the input vector $\boldsymbol{x}'$. The weight vector $\boldsymbol{w}$ denotes the entire set of synaptic weights ordered by layer, the neurons in a layer, and then synapses in a neuron. The thresholds corresponding to the hidden and output neurons are given by κ. The logistic activation function, $\varphi(*)$ is given by:

TABLE 5—COMPARISON OF NEURAL NETWORK PERFORMANCE BY DIVISION OF DATA SET

Cases (data points)	Error Parameter	Value
Training (418)	ARMS	0.0014
	APE	2.2
	AAPE	12.8
	R	0.986
Verification (104)	ARMS	0.0016
	APE	4.6
	AAPE	14.7
	R	0.984
Test (105)	ARMS	0.0018
	APE	1.4
	AAPE	12.5
	R	0.979
Total (627)	ARMS	0.0015
	APE	2.5
	AAPE	13.1
	R	0.985

$$\varphi_j = \frac{1}{[1 + e^{-(*)}]} \quad \ldots\ldots (2)$$

Note that the primes on the input vector and output represent linearly scaled data. Thus, $\boldsymbol{x}' = \boldsymbol{x} \cdot \xi$, where ξ = the preprocess scaling vector and $\boldsymbol{x}$ = the raw input vector. Similarly, $\log(H_L) = \log(H_L)' \cdot \zeta$, where ζ = the post-process scaling factor. Then, H_L is simply $10^{\log(H_L)}$.

Discussion of Results

To evaluate the accuracy of the neural network model, the results were compared with liquid-holdup correlations developed by Beggs,[22] Mukherjee,[10] Minami and Brill,[12] and Abdul-Majeed.[13] Additionally, the Taitel-Dukler[20] mechanistic model was evaluated for stratified flow cases as well as Xiao's[19] comprehensive mechanistic model for cases of stratified and intermittent flow.

Criterion for Critical Evaluation. The criterion used for this evaluation is, in part, that suggested by Mandhane *et al.*[5] Several different error measurements, each sensitive for different holdup ranges, are applied to evaluate both the accuracy and precision of the predictions. Additionally, crossplots of predicted vs. measured holdup are presented to provide an overall impression of the agreement.

Error Parameters. Error is defined as the difference between the measured and predicted values of liquid holdup for some observation, *i*.

Error:

$$e_i = (H_{L\,\text{pred}} - H_{L\,\text{meas}})_i \; i = 1,2,\ldots,n. \quad \ldots\ldots (3)$$

ARMS Error:

$$\text{ARMS} = \left(\frac{1}{n} \sum_{i=1}^{n} e_i^2 \right)^{1/2} \quad \ldots\ldots (4)$$

Average percent error (APE):

$$\text{APE} = \frac{100}{n} \sum_{i=1}^{n} \left(\frac{e}{H_{L\,\text{meas}}} \right)_i \quad \ldots\ldots (5)$$

Average absolute percent error (AAPE):

$$\text{AAPE} = \frac{100}{n} \sum_{i=1}^{n} \left| \frac{e}{H_{L\,\text{meas}}} \right|_i \quad \ldots\ldots (6)$$

The ARMS error is a measure of scatter, or lack of precision. It is relatively insensitive to errors in the low ranges of liquid holdup and strongly affected by errors in the high range. The APE measures the centering or average accuracy of the predictions but is more strongly influenced by errors in the low-liquid-holdup range. Like the ARMS error, the AAPE also measures the lack of precision; however, it is very sensitive to errors associated with small measured values of liquid holdup.

The correlation coefficient, *R,* a more complex statistical parameter, is used to evaluate the degree of linear relationship between the measured and predicted holdup. A value of 1 indicates a perfect linear relationship, whereas a value of 0 means there is absolutely no linear dependence.

Comparison of Methods Based on Liquid-Holdup Range, Flow Pattern, and Overall Performance. Previous studies[3–5] have shown that many correlations tend to perform well over certain ranges of liquid holdup and lose accuracy when applied to other ranges. Therefore, the results of this study were analyzed on the basis of a range of liquid-holdup values.

Table 6 shows the results of this analysis, which illustrates that the neural network model performs more accurately and consistently across the liquid-holdup range. The prediction sensitivity is most evident with the Abdul-Majeed model. While the Abdul-Majeed[13] model performs very well in the low-holdup range, error increases dramatically as the liquid holdup increases. The Minami-Brill[12] results reflect the combination of two correlations—wet gas for H_L<0.15 and general for higher holdup values. Predictions made with the Minami-Brill correlations display better overall performance and greater consistency than the other empirical correlations and mechanistic models, and it is second in overall accuracy to the neural network model. The error associated with the lowest liquid-holdup range (H_L<0.10) is highest for all models. This is to be expected because the measurement error associated with collecting data increases in this range.

Liquid holdup is strongly influenced by distribution of the phases, as characterized by flow pattern.[1] Therefore, the results of this study were also evaluated on the basis of flow pattern, as determined by Xiao's[19] mechanistic criterion. This criterion is also used when applying mechanistic approaches for calculating liquid holdup. However, the Beggs-Brill[22] model uses its own flow-pattern-prediction method. All other liquid-holdup correlations, including the neural network models, were developed irrespective of flow pattern. Nevertheless, effects of flow pattern are implicit in that data collected across flow patterns were used to develop the correlations.

Table 6 presents these results. The neural network model performs favorably for all flow patterns with the exception of intermittent flow, in which the performance is comparable with Xiao's[19] mechanistic model and the Minami-Brill,[12] Mukherjee-Brill,[10] and Beggs-Brill[22] correlations. While the correlation coefficients are lower for the neural network model for intermittent flow, performance is favorable for all other error measures. This can be attributed to the use of the logarithm of liquid holdup as an output because intermittent flow occurs predominantly in the mid- to high-liquid-holdup range. Neural networks developed using only liquid holdup as an output exhibited a slight improvement in performance for intermittent flow, though at a cost to better overall performance.

The bottom of Table 6 shows overall performance. On average, the neural network will overpredict liquid holdup by 2.5%. The AAPE for the neural network is 13.1%, nearly twice as low as Beggs-Brill[22] and second in this category. The correlation coefficient for the neural network is 0.985, which again is nearly twice as good as Minami-Brill,[12] the next best.

Crossplots of Predicted vs. Experimental Liquid Holdup. Fig. 4 shows a comparison of the experimentally measured and predicted liquid holdup using the neural network model. **Figs. 5 through 10** compare experimentally measured liquid holdup with that predicted using existing empirical correlations and mechanistic models. **Fig. 11** compares the experimentally measured holdup with that assumed by the no-slip condition.

It is evident from these figures that the liquid-holdup predictions made by the neural network model are in better agreement with the experimental data than predictions made by the other methods, particularly in the low-liquid-holdup range.

Applications

While the neural network model performed favorably vs. other models for the data set used in this study, this result was expected because the neural network had the advantage of being trained with a large portion of this data.

The key differentiator of the neural network model from contemporary mechanistic models is that it represents empiricism in its purest form; that is, the neural network considers only the behavior of the experimental data used to train it, not the fundamental laws of fluid mechanics that govern multiphase flow. Therefore, while the neural network can be expected to make good predictions within the ranges of experimental data used to train it, one should be very cautious in applying neural networks for other conditions because they tend to perform poorly when forced to extrapolate. Referring to the experimental conditions summarized in Table 1, the neural network model is valid for the following data ranges.

- Pipe diameter: 1 to 4 in.
- Liquid viscosity: 0.58 to 74.4 cp.
- Surface tension: 1.55 E–03 to 4.94 E–03 lbf/ft.
- Liquid density: 49.3 to 63.6 lbm/ft^3.

TABLE 6—METHOD COMPARISON IN TERMS OF FLOW PATTERN, LIQUID-HOLDUP RANGE, AND OVERALL PERFORMANCE

Data Range (data points)	Error Measurement	Taitel-Dukler[20]	Xiao[19]	Abdul-Majeed[13]	Minami-Brill[12]	Mukherjee-Brill[1]	Beggs-Brill[22]	Neural Network
Annular	ARMS	—	—	0.001	0.002	0.001	0.001	0.000
(179)	APE	—	—	–7.4	33.8	11.6	24.2	–0.3
	AAPE	—	—	21.8	40.6	34.4	35.7	11.9
	R	—	—	0.923	0.843	0.874	0.832	0.963
Intermittent	ARMS	—	0.005	0.031	0.003	0.005	0.004	0.002
(132)	APE	—	0.6	–24.3	2.9	4.7	2.6	2.6
	AAPE	—	14.0	27.0	9.9	12.1	10.8	7.8
	R	—	0.946	0.812	0.960	0.957	0.957	0.910
Stratified smooth	ARMS	0.032	0.039	0.083	0.008	0.011	0.016	0.006
(39)	APE	11.7	15.9	47.0	2.4	–1.3	–24.4	–6.5
	AAPE	36.9	42.1	50.8	18.8	25.1	31.3	15.3
	R	0.709	0.689	0.848	0.858	0.814	0.867	0.904
Stratified wavy	ARMS	0.016	0.008	0.004	0.003	0.003	0.004	0.001
(311)	APE	58.3	–16.8	9.7	17.9	–20.8	–10.9	4.6
	AAPE	65.1	44.8	23.0	30.5	36.7	26.5	15.9
	R	0.908	0.912	0.932	0.954	0.953	0.952	0.980
Stratified	ARMS	0.018	0.011	0.013	0.003	0.004	0.005	0.002
(350)	APE	53.0	–13.1	13.9	16.1	–18.6	–12.4	3.4
	AAPE	61.9	44.5	26.1	29.2	35.4	27.0	15.8
	R	0.888	0.901	0.906	0.953	0.948	0.941	0.975
$H_L < 0.1$	ARMS	0.0027	0.0013	0.0005	0.0008	0.0007	0.0005	0.0002
Taitel-Dukler[20] (131)	APE	73.4	–38.6	17.4	41.1	–30.5	14.6	9.2
Xiao[19] (131)	AAPE	73.4	57.7	24.7	49.3	54.6	36.9	20.0
Others (180)	R	0.799	0.538	0.755	0.783	0.735	0.747	0.887
$0.1<H_L<0.2$	ARMS	0.0133	0.0050	0.0015	0.0019	0.0022	0.0022	0.005
Taitel-Dukler[20] (69)	APE	48.8	–22.9	–11.2	12.5	0.8	–5.6	–4.8
Xiao[19] (74)	AAPE	66.7	42.2	20.3	25.4	26.5	25.2	11.1
Others (132)	R	0.270	0.419	0.528	0.555	0.524	0.443	0.803
$0.2<H_L<0.4$	ARMS	0.0393	0.0186	0.0153	0.0052	0.0059	0.0063	0.0024
Taitel-Dukler[20] (97)	APE	46.5	8.2	–0.4	6.2	7.2	–9.7	3.3
Xiao[19] (150)	AAPE	58.6	35.8	27.5	18.7	21.8	22.4	12.2
Others (158)	R	0.292	0.410	0.412	0.618	0.636	0.579	0.742
$0.4<H_L$	ARMS	0.0219	0.0085	0.0469	0.0040	0.0058	0.0074	0.0030
Taitel-Dukler[20] (55)	APE	21.7	6.5	–9.4	0.2	–11.6	–7.0	0.1
Xiao[19] (155)	AAPE	23.5	12.1	27.9	8.7	10.2	11.8	7.6
Others (157)	R	0.577	0.808	0.341	0.910	0.879	0.887	0.933
Overall	ARMS	0.018	0.009	0.016	0.003	0.004	0.004	0.002
(627)	APE	53.0	–8.8	0.2	16.0	–7.2	–1.2	2.5
	AAPE	61.9	35.1	25.6	26.4	29.3	24.5	13.1
	R	0.888	0.932	0.843	0.972	0.964	0.963	0.985

The most limiting of these parameters is surface tension. The experimental conditions used to generate the data sets were performed at low pressures relative to typical field operating conditions. Surface tension decreases rapidly with increasing pressure and temperature and will often fall to less than the limits of applicability when the pressure rises to more than several hundred psia. This may result in underprediction of liquid holdup. Similarly, large pipe diameters and high viscosities will often yield underpredictions. The liquid densities covered in the experimental data pose the least constraining limits because actual values tend not to drop very far below the lower limit of 49.3 lbm/ft^3, except in cases of high-API-gravity fluids at high pressures and temperatures.

Considering these limitations, the neural network model is most applicable to small, low-pressure flowlines and is not recommended for general use. Though the model has not been validated with field data, it is available to use in PIPESIM,[26] a commercial multiphase flow simulator. Additionally, the FORTRAN source code for the neural network model can be obtained by contacting the authors.

Extending the range of applicability can be accomplished by training the neural network with additional cases. To account for conditions not attainable with experimental facilities, it is plausible to introduce training cases to the neural network based on liquid-holdup predictions made by mechanistic models. In this situation, the neural network would be expected to make predictions consistent with the measured data at or near experimental conditions while emulating the behavior of the mechanistic model for other operating conditions.

Contrary to the concept of training a neural network to mimic the results of mechanistic models, another plausible application

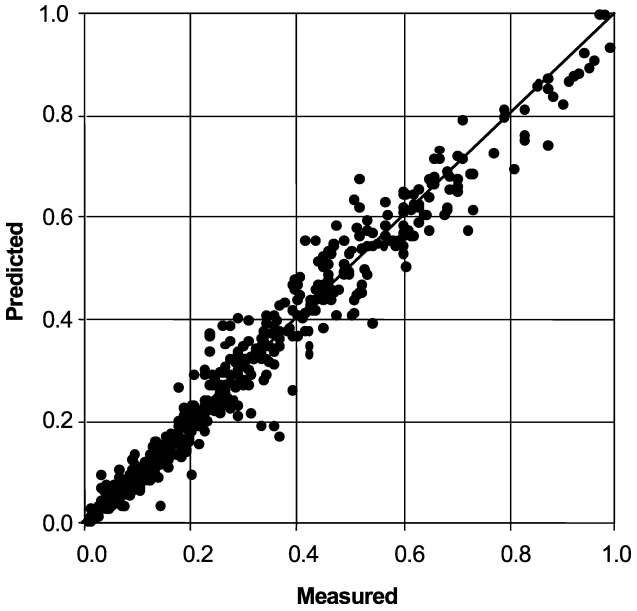

Fig. 4—Liquid-holdup crossplot—neural network.

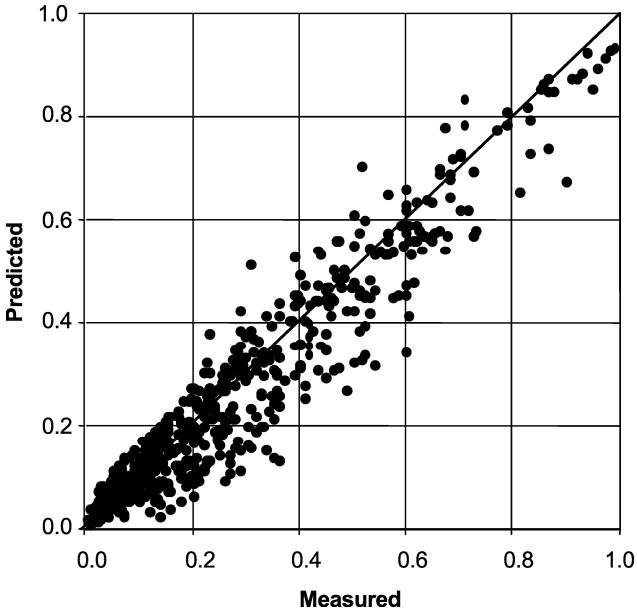

Fig. 5—Liquid-holdup crossplot—Beggs-Brill.[22]

would involve using a neural network to tune the results of a mechanistic model to match observed field data. As fiber-optic temperature measurements, pressure sensors, inline multiphase flowmeters, and other instrumentation become more commonplace, a massive amount of useful data also becomes available. Moreover, the strength of the weighting factors required to adjust the predictions of a mechanistic model to match observed data could be used to evalulate the suitability of various mechanistic models to field conditions.

Conclusions

1. A neural network model has been developed for predicting liquid holdup in two-phase horizontal flow. Performance analysis based on the data sets used in this study show the neural network model to exhibit better overall performance compared to existing empirical correlations and mechanistic models. However, because of the limitations imposed by the ranges of experimental data used to train the neural network, it is not recommended for general applications.
2. The neural network model developed in this study does not require prior estimation of flow pattern, as do many existing empirical correlations and mechanistic models. Therefore, the model is not affected by errors in flow-pattern prediction and is not subject to discontinuities in predictions made across transition boundaries.
3. Robust and systematic search algorithms have eliminated much of the arduous trial-and-error experimentation associated with finding the optimal network model. Coupled with ever-increasing processor speed, the development of a neural network solution is becoming increasingly practical for complex problems.
4. Neural networks increase in accuracy and range as more data are incorporated into the training process. Thus, with additional data, the model can adapt to make more-accurate predictions for a wider range of conditions.

Nomenclature

d = pipe ID, ft
e = error

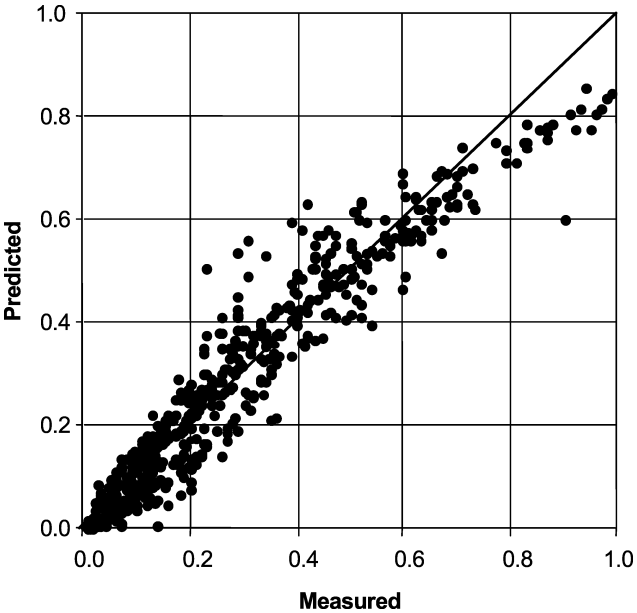

Fig. 6—Liquid-holdup crossplot—Mukherjee-Brill.[10]

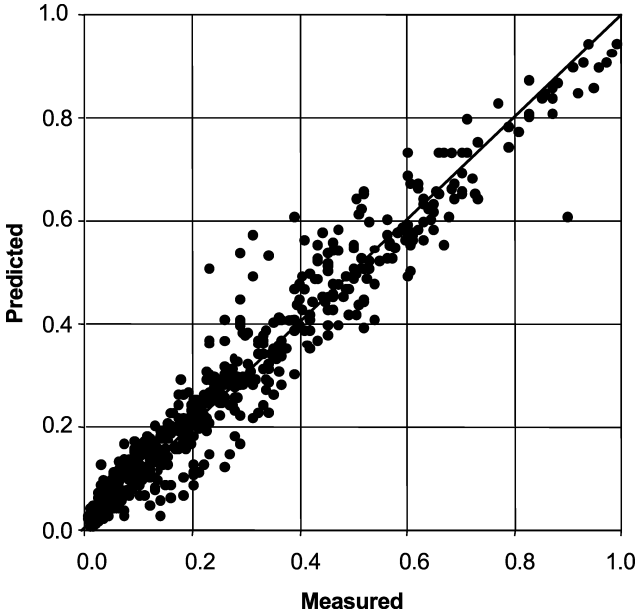

Fig. 7—Liquid-holdup crossplot—Minami-Brill.[12]

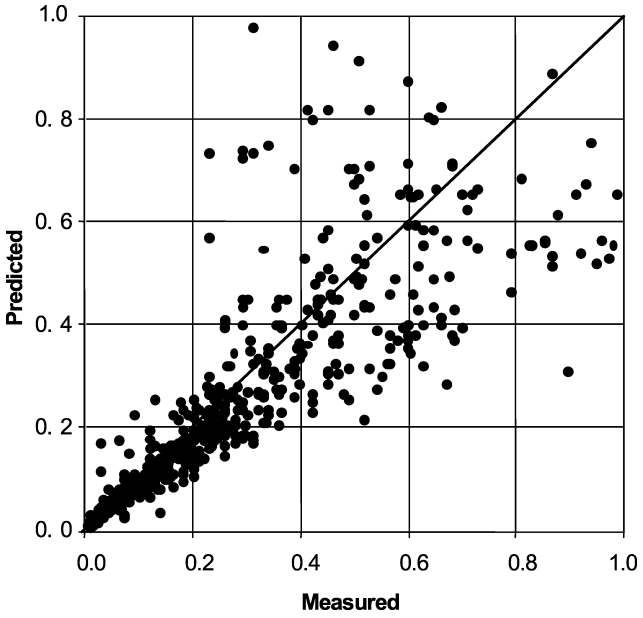

Fig. 8—Liquid-holdup crossplot—Abdul-Majeed.[13]

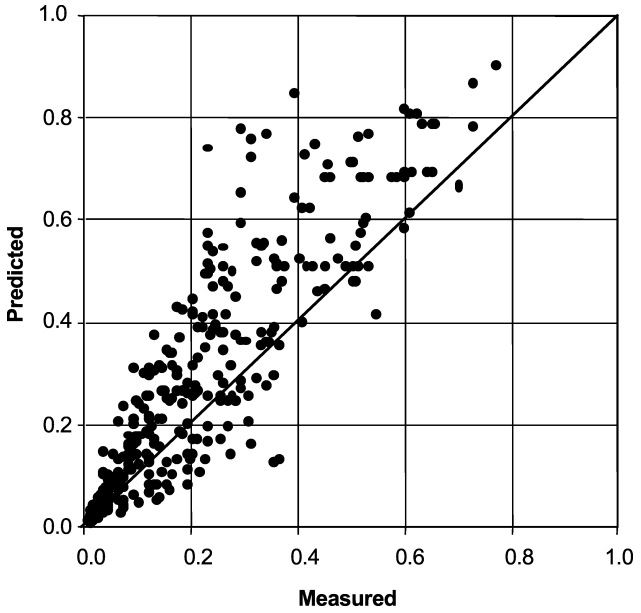

Fig. 9—Liquid-holdup crossplot—Taitel-Dukler[20] mechanistic model (stratified case).

F = function of
H_L = liquid holdup
H_{LNS} = no-slip liquid holdup
i = input layer, also counter for error points
j = hidden layer
k = output layer
M = number of hidden nodes
n = counter for data points
N = number of input nodes
o = raw output
R = correlation coefficient
v = velocity, ft/s
w = individual weight
$\boldsymbol{w}$ = weight vector
x = individual input
$\boldsymbol{x}$ = input vector
$\boldsymbol{x}'$ = linearly scaled input vector
φ = logistic transform function
κ = neuron threshold
μ = viscosity, cp
ρ = density, lbm/ft^3
σ = liquid surface tension, lbf/ft
ξ = preprocessing scaling vector
ζ = post-processing scaling factor

Subscripts

G = gas
i = input layer
j = hidden layer
k = output layer
L = liquid
meas = measured

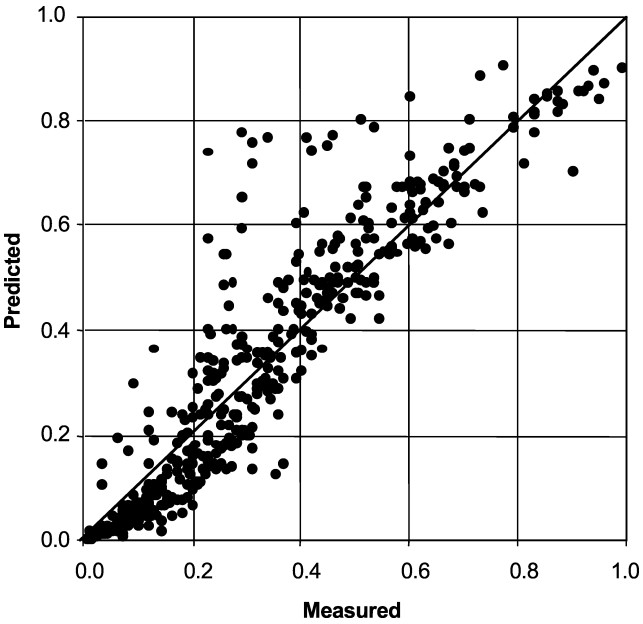

Fig. 10—Liquid-holdup crossplot—Xiao[19] mechanistic model (stratified and intermittent cases).

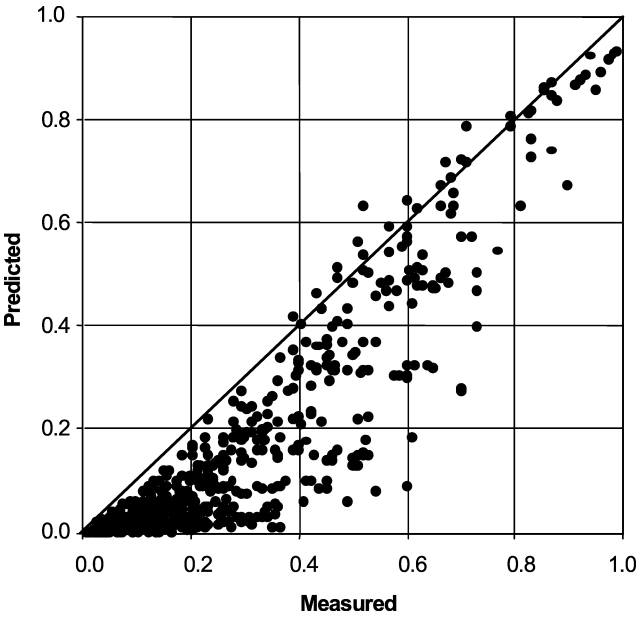

Fig. 11—Liquid-holdup crossplot—no-slip assumption.

pred = predicted
SG = superficial gas
SL = superficial liquid

References

1. Brill, J.P. and Mukherjee, H.: *Multiphase Flow in Wells,* Monograph Series, SPE, Richardson, Texas (1999) 17.
2. Brown, K.E.: *The Technology of Artificial Lift Methods,* 1 and 4, Pennwell Publishing Co., Tulsa (1977).
3. Arya, A. and Gould, T.L.: "Comparison of Two-Phase Liquid Holdup and Pressure Drop Correlations Across Flow Regime Boundaries for Horizontal and Inclined Pipes," paper SPE 10169 presented at the 1981 SPE Annual Technical Conference and Exhibition, San Antonio, Texas, 5–7 October.
4. Vohra, I.R. *et al.:* "Comparisons of Liquid-Holdup and Friction-Factor Correlations for Gas-Liquid Flow," *JPT* (May 1975) 564.
5. Mandhane, J.M., Gregory, G.A., and Aziz, K.: "Critical Evaluation of Holdup Prediction Methods for Gas-Liquid Flow in Horizontal Pipes," *JPT* (August 1975) 1017; *Trans.*, AIME, **259.**
6. Ali, J.K: "Neural Networks: A New Tool for the Petroleum Industry?" paper SPE 27561 presented at the 1994 SPE European Petroleum Computer Conference, Aberdeen, 15–17 March.
7. Kumoluyi, A.O.: "Higher-Order Neural Networks in Petroleum Engineering," paper SPE 27905 presented at the 1994 SPE Western Regional Meeting, Long Beach, California., 23–25 March.
8. Mohaghegh, S.: "Virtual Intelligence Applications in Petroleum Engineering: Part I—Artificial Neural Networks," *JPT* (September 2000) 64.
9. Ternyik, J. IV, Bilgesu, H.I., and Mohaghegh, S.: "Virtual Measurement in Pipes: Part 2—Liquid Holdup and Flow Pattern Correlations," paper SPE 30976 presented at the 1995 SPE Eastern Regional Meeting, Morgantown, West Virginia, 17–21 September.
10. Mukherjee, H.: "An Experimental Study of Inclined Two-Phase Flow," PhD dissertation, U. of Tulsa, Tulsa (1979).
11. Osman, E.-S.A.: "Artificial Neural Network Models for Identifying Flow Regimes and Predicting Liquid Holdup in Horizontal Multiphase Flow," *SPEPF* (February 2004) 33.
12. Minami, K. and Brill, J.P.: "Liquid Holdup in Wet-Gas Pipelines," *SPEPE* (February 1987) 36.
13. Abdul-Majeed, G.H.: "Liquid Holdup in Horizontal Two-Phase Gas-Liquid Flow," *J. of Petroleum Science and Engineering* (1996) **15,** 271.
14. Shippen, M.: "Development of a Neural Network Model for Prediction of Liquid Holdup in Two-Phase Horizontal Flow," MS thesis, Texas A&M U., College Station, Texas (2001).
15. Haykin, S.: *Neural Networks: A Comprehensive Foundation,* second edition, Prentice Hall, Upper Saddle River, New Jersey (1999) 215.
16. *User's Manual Statistica Neural Networks Release 4.0 F,* Statsoft Inc., Tulsa (2000).
17. Bishop, C.: *Neural Networks for Pattern Recognition,* Oxford U. Press, Oxford, England (1995).
18. Taylor, E.S.: *Dimensional Analysis For Engineers,* Clarendon Press, Oxford, England (1974).
19. Xiao, J.J., Shoham, O., and Brill, J.P.: "A Comprehensive Mechanistic Model For Two-Phase Flow in Pipelines," paper SPE 20631 presented at the 1990 SPE Annual Technical Conference and Exhibition, New Orleans, 23–26 September.
20. Taitel, Y. and Dukler, A.E.: "A Model for Predicting Flow Regime Transitions in Horizontal and Near Horizontal Gas-Liquid Flow," *AIChE J.* (January 1976) **22,** 47.
21. Eaton, B.A.: "The Prediction of Flow Patterns, Liquid Holdup and Pressure Losses Occurring During Continuous Two-Phase Flow in Horizontal Pipelines," PhD dissertation, U. of Texas at Austin, Austin, Texas (1966).
22. Beggs, H.D: "An Experimental Study of Two-Phase Flow in Inclined Pipes," PhD dissertation, U. of Tulsa, Tulsa (1973).
23. Beggs, H.D.: *Production Optimization Using Nodal Analysis,* OGCI Publications, Tulsa (1991) 75–82.
24. Ros, N.C.: "Simultaneous Flow of Gas and Liquid as Encountered in Well Tubing," *JPT* (October 1961) 1037; *Trans.*, AIME, **222.**
25. STATISTICA Neural Networks, Release 4.0 F, Statsoft Inc., Tulsa (2000).
26. PIPESIM 2003, Schlumberger Information Solutions, Abingdon, England (2003).

SI Metric Conversion Factors

cp	× 1.0*	E–03	= Pa·s
ft	× 3.048*	E–01	= m
ft^3	× 2.831 685	E–02	= m^3
°F	(°F–32)/1.8		= °C
in.	× 2.54*	E+00	= cm
lbf	× 2.248 089	E–01	= N
lbm	× 4.535 924	E–01	= kg
psi	× 6.894 757	E+00	= kPa

*Conversion factor is exact.

Mack E. Shippen is a senior petroleum engineer at Schlumberger in Houston, where he provides user support, training, and consulting services for PIPESIM, a multiphase-flow simulator. e-mail: mshippen@slb.com. Shippen holds BS and MS degrees in petroleum engineering from Texas A&M U. He is currently serving as an Editor of the SPE Reprint on *Offshore Multiphase Production Operations* (2004) and serves on the Tulsa U. Fluid Flow Projects Advisory Board, the SPE.org Advisory Board, and the OTC Program Committee. **Stuart L. Scott** is an associate professor of petroleum engineering at Texas A&M U., where he conducts research on a variety of multiphase-flow aspects, including compact separation, multiphase leak detection, and multiphase pumping. e-mail: scott@spindletop.tamu.edu. Since 2001, he has chaired the Production Committee for the ASME Petroleum Division, and in 2003, he was awarded the ASME Henry R. Worthington Medal for his work in the area of multiphase pumping. Before joining Texas A&M U., he was an assistant professor at Louisiana State U. and also worked 9 years for Phillips Petroleum. Scott holds a BS degree in petroleum engineering, an MS degree in computer science, and a PhD degree in petroleum engineering, all from the U. of Tulsa. He currently serves as the Chairman of the Production Systems Technical Interest Group (TIG) and has served on the SPE Program Coordinating, Reprint Series, Mid-Continent Gas Symposium, Annual Meeting Program, and Computer Applications Review Committees.