

# Optimal Sensor Placement in Fluid Dynamics using Machine Learning and Sensitivity Analysis

Harley Hanes\*

*North Carolina State University, Raleigh, NC 27695*

Yury Lebedev†

*University of Florida, Gainesville, FL 32611*

Ralph C. Smith‡

*North Carolina State University, Raleigh, NC 27695*

Alina Zare§

*University of Florida, Gainesville, FL 32611*

**Constructing computational fluid dynamics (CFD) simulations from experimental data is a critical process in aerospace engineering design, but sparsity and errors in local sensors limits the ability to condition CFD results with experimental data. These limitations often lead to different measurements of critical quantities in experimental and computational results. Neural networks can quantify nonlinear relationships between sparse or integrated sensor data and the corresponding flow-field, potentially increasing the accuracy of other CFD methods by conditioning them on initial estimates of flow-fields produced by neural networks. However, accuracy of this method requires investigating optimal placement of sensors and sensitivity analysis of network reconstructions. In this work, we investigate the sensitivity of a shallow encoder network CFD reconstruction to the location and readings of sensors using Morris screening and network sensitivity methods. The resulting analysis facilitates the determination of flow-field regions where increased precision in sensor readings or placements most effects reconstruction accuracy.**

## I. Background and Motivation

WITH improvements in the accuracy and efficiency of computational fluid dynamics (CFD), using CFD results to compute flow-fields using experimental sensor data has become a critical step in aerospace applications of engineering design. Sensor readings from experimental data are often used to condition CFD results, which can compute the flow-field as a function of space and time, though limited to discretizations in both. However, limitations of physical sensors present numerous complications for computing velocity flow and pressure characteristics directly from sensor data. First, sensors are often limited to integrated forces and moments, supplemented by sparse velocity, pressure, or temperature sensors. Additionally, sensors can have errors in their readings, which further limits the accuracy of CFD results. As a result of these limitations, experimental and CFD results often fail to align on critical quantities such as lift and drag coefficients [1].

Neural networks present a possible intermediate reduced-order model (ROM) to efficiently expand sensor data into flow-field data with which CFD results can be better constrained. Neural network approximations of flows have already been used to accurately predict flow-fields by quantifying nonlinear relationships between the flow-field and sparse or integrated sensor data [2]. Neural network approximations may not match boundary conditions or underlying equations. However, CFD methods can effectively enforce physical limitations of the flow-field with improved accuracy from network approximation-based constraints on the entire flow-field. Therefore, we employ neural networks to efficiently approximate velocity of compressible Navier-Stokes flows directly from sensor data, which can greatly improve the accuracy of other CFD methods.

---

\*Doctoral Candidate, Department of Mathematics, North Carolina State University.

†Doctoral Candidate, Department of Electrical and Computer Engineering, University of Florida.

‡Distinguished University Professor, Department of Mathematics, North Carolina State University.

§Professor, Department of Electrical and Computer Engineering, University of Florida.

Informing high-fidelity CFD approximations with a network approximation requires determining the optimal location of sensors to maximize accuracy of reconstructions and minimize sensitivity to error in sensor readings. A common method to select optimal measurements under uncertainty is optimizing the mutual information [3]. Mutual information methods can be used to select sensor locations iteratively so that the reading of each sensor minimizes the uncertainty in predicted outputs given the previously placed sensors and uncertainty in their readings [4, 5]. However, computing the mutual information metric requires integrating over the uncertainty in each output and sensor reading along with optimizing the sensor placement [5]. This approach is computationally intractable for our application due to the high dimension of our output space, the number of discrete points in each snapshot, and the potentially long time required to retrain the networks for each sensor position when optimizing the mutual information. Therefore, we instead utilize sensitivity analysis to determine regions of a flow for which reconstructions are most sensitive to the sensor data or placement.

Sensitivity analysis quantifies how uncertainty in model outputs is apportioned to uncertainty in model parameters. Parameters with high sensitivity require more precise measurement or can be optimized to improve performance whereas parameters with low sensitivity can potentially be fixed to develop reduced-order models [6]. In this paper, we quantify the sensitivity of the network flow-field reconstructions to both the location and data of sensors placed using the leverage score method [7].

To quantify sensitivity to sensor location, we use the quasi-global method of Morris screening [8]. Morris screening approximates the local partial derivatives of output quantities with respect to each parameter and averages the approximations over an assumed distribution of parameter values. Morris screening provides less information than the total variance of Sobol analysis, a common global sensitivity analysis method, but requires significantly fewer function evaluations and is less affected by error in assumed parameter distributions [9]. To determine the sensitivity of sensor values, we use a network sensitivity method which globally averages the partial derivatives of network outputs to network inputs [10]. Network sensitivity approximation has similar sensitivity indices as Morris screening, but instead samples over the set of sensor readings from each snapshot and uses back-propagation to more efficiently and accurately compute partial derivatives.

In this paper, we detail the structure, training, and sensor placement methods for network reconstructions of fluid-flows. We then review the network sensitivity and Morris screening methods used to compute the sensitivity of reconstructions to sensor data and location respectively. Next, we present the regularized lid-driven cavity we use to test the sensor sensitivity methods. We then measure the sensitivity of reconstruction error to sensor location and data to determine optimal placement of additional sensors. We intend as future work extension of this method to determine optimal sensor placement for predicting flow-fields around an Orion lander.

## II. Methodology

### A. Network Design and Sensor Placement

To calculate the flow-field  $\mathbf{y} \in \mathbb{R}^m$ , where  $m$  is the size of the full flow-field, from a set of sensor measurements  $\mathbf{x} \in \mathbb{R}^p$ , where  $p$  is the number of sensor locations, we implement a shallow encoder network. Using a rank- $k$  approximation, we assume that a field can be approximated using

$$\mathbf{y} \approx \hat{\mathbf{y}} = \sum_{j=1}^k \boldsymbol{\phi}_j v_j = \boldsymbol{\Phi} \mathbf{v}, \quad (1)$$

where  $\boldsymbol{\phi}_j \in \mathbb{R}^m$  are the modes of the approximation and  $v_j \in \mathbb{R}$  are the coefficients [11]. Given the set of approximation modes  $\boldsymbol{\Phi} \in \mathbb{R}^{m \times k}$ , we may estimate the flow-field  $\mathbf{y}$  by learning the coefficients  $\mathbf{v}$  from the sensor observations,  $\mathbf{x}$ . A fully connected neural network with  $l$  layers is defined as [11]

$$F(x; \{\mathbf{W}_i\}_{i=1}^l) = R(\mathbf{W}_l R(\mathbf{W}_{l-1} \cdots R(\mathbf{W}_1 x))), \quad (2)$$

where  $R : \mathbb{R} \rightarrow \mathbb{R}$  is a coordinate-wise scalar nonlinear activation function and  $\{\mathbf{W}_j\}_{j=1}^l$  is a set of weight matrices, one for each layer. Given a training set  $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^n$  with  $n$  samples of flow-field  $\mathbf{y}_i$  and corresponding sensor measurements  $\mathbf{x}_i$ , the training of a shallow encoder network estimates a set of weight matrices such that  $F : x \mapsto \hat{\mathbf{y}}$  minimizes the error

in Euclidean flow space over all sensor measurements,

$$\operatorname{argmin}_F \sum_{i=1}^k \|\mathbf{y}_i - F(\mathbf{x}_i)\|_2^2. \quad (3)$$

Following Erichson [12], a shallow encoder can conceptually be mapped to a Proper orthogonal decomposition (POD) methodology as outlined in 1. This conceptual mapping involves three components. The first component is a network layer that maps from the input sensor measurements to a larger dimensional space, which conceptually corresponds to a non-linear feature extraction layer,  $z^\psi = \psi(\mathbf{x}) = R(\mathbf{W}_\psi \mathbf{x} + \mathbf{b}^\psi)$ . The second component conceptually maps so the coefficients of the POD approximation,  $z^\nu s = \nu(z^\phi) = R(\mathbf{W}_\nu z^\phi + \mathbf{b}_\nu)$ . Finally, the third component maps to the full flow-field approximation including estimation of the modes of the POD approximation,  $\hat{\mathbf{y}} = \mathbf{W}_\phi z^\nu + \mathbf{b}_\phi$ . Thus, the columns of the weight matrix  $\mathbf{W}_\phi$  conceptually correspond to the dominant modes of the approximation; i.e., similar to  $\Phi$  in 1.

Sensor positions from CFD data are randomly sampled using leverage score sampling [7, 13]. Leverage scores for the  $i^{th}$  sensor location are computed using

$$l_i = \sum_{j=1}^n \mathbf{V}_{i,j}^2, \quad (4)$$

where  $n$  is equal to the number of training samples and  $\mathbf{V} \in \mathbb{R}^{n \times p}$  is the right singular vector matrix obtained from the singular vector decomposition of the matrix of training data samples. Sensors are selected without replacement and with probability proportional to their leverage score.

## B. Morris Screening

To compute the sensitivity of network approximations to sensor placement, we use Morris screening. Morris screening is a one-at-a-time (OAT) quasi-global sensitivity analysis method, which approximates the local derivatives using finite-differences at parameter values sampled from assumed distributions [8]. Morris screening's primary sensitivity indices calculate the mean  $\mu_{ik}$ , absolute mean  $\mu_{ik}^*$ , and standard deviation  $\sigma_{ik}$  of sensitivity for the  $i^{th}$  parameter and  $k^{th}$  model output  $f_k$ . Formulae for Morris screening indices of  $f_k$  are given by

$$d_{ik}^j = \frac{f_k(\xi, \theta^j + \Delta e_i) - f_k(\xi, \theta^j)}{\Delta}, \quad \mu_{ik} = \frac{1}{N} \sum_{j=1}^N d_{ik}^j, \quad \mu_{ik}^* = \frac{1}{N} \sum_{j=1}^N |d_{ik}^j|, \quad \sigma_{ik} = \sqrt{\frac{1}{N-1} \sum_{j=1}^N (d_{ik}^j - \mu_{ik})^2}, \quad (5)$$

for a parameter  $i$  with  $N$  parameter samples, a step-size of  $\Delta$ , and unit vector  $e_i$  for parameter  $i$ . The step-size determines how finely the parameter space is searched around each parameter sample  $\theta^j$ , where very small values provide a finite-difference derivative approximation whereas large values quantify large-scale function variations [6]. For the results presented in this paper, we utilized a step-size of  $\Delta = 10^{-3}$ . To improve accuracy of Morris screening, we uniformly sample parameters with quasi-random Sobol samples [14].

The absolute mean,  $\mu_{ik}^*$  quantifies the sensitivity magnitude, where parameters with larger  $\mu_{ik}^*$  have a larger magnitude of partial derivative. The absolute mean is used to quantify sensitivity instead of the mean sensitivity since the mean sensitivity can be small or zero for highly sensitive parameters if they have a nonlinear effects on quantities due to cancellation of opposite signs in partial derivatives. It provides a similar measurement as total variance from Sobol analysis, but Sobol analysis is more computationally expensive, more impacted by error in assumed parameter distributions, and assumes parameters are independently distributed for standard implementations [6]. The indices  $\sigma_{ik}$  quantify the standard deviation of the sensitivity which identifies parameters that may have nonlinear effects on a quantity or may only be highly sensitive on a limited region of their assumed distribution [9].

## C. Network data sensitivity

We perform sensitivity analysis of the network approximation with respect to sensor data using the partial derivatives method outlined by Pizarroso, *et al* [10]. This method utilizes back-propagation to directly compute partial derivatives of network outputs to network inputs and averages these derivatives over sample data. We define the following sensitivity indices,

$$s_{ikj} = \frac{\partial y_k}{\partial x_i}(x_j), \quad S_{ik}^{avg} = \frac{1}{N} \sum_{j=1}^N s_{ikj}, \quad S_{ik}^{abs} = \frac{1}{N} \sum_{j=1}^N |s_{ikj}|, \quad S_{ik}^{sd} = \sqrt{\frac{1}{N-1} \sum_{j=1}^N (S_{ik}^{avg} - s_{ikj})^2}, \quad (6)$$

where  $N$  is the total number of data samples,  $x_j$  is the  $j$  sample of the dataset, and  $s_{ikj}$  is the sensitivity of the output of the  $k$ -th neuron in the output layer with respect to the input of the  $i$ -th neuron in the input layer calculated at  $x_j$ . We note that the network sensitivity indices  $S_{ik}^{avg}$ ,  $S_{ik}^{abs}$ , and  $S_{ik}^{sd}$  correspond to the Morris screening indices  $\mu_i$ ,  $\mu_i^*$ , and  $\sigma_i$  defined in Section II.B, although the sensitivity sampling strategies differ. Network sensitivity traditionally uses  $S_{ik}^{sq}$ , which is the square root of the mean of squares of sensitivity, but we use  $S_{ik}^{abs}$  for sensitivity magnitude to better correspondence to  $\mu^*$  in Morris screening. Similarly to  $\sigma^*$  in Morris screening, we use  $S_{ik}^{sd}$  to identify sensors that have nonlinear effects on flow-field reconstructions or have large variation in sensitivity between data samples. The package `neuralsens` provides Python and R implementation of network sensitivity with exact computation of partial derivatives. However, computation of Jacobians for each network layer can be computationally expensive for large networks [10]. Therefore, we instead use a finite-difference approximation to compute  $s_{ikj}$  using a perturbation distance of  $\Delta = 10^{-5}$ .

#### D. Lid-Driven Cavity Test Case

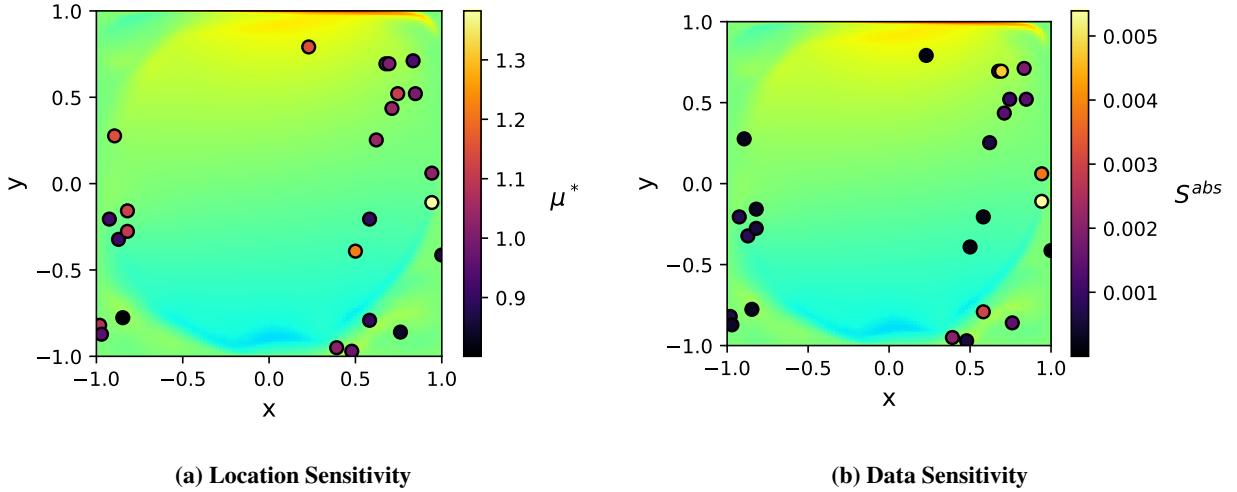
We employ an incompressible regularized lid-driven cavity model to test sensitivity analysis of network approximations of fluid flows. Lid-driven cavities are rectangular domains with homogeneous Dirichlet boundary conditions on three walls and a non-homogeneous Dirichlet boundary condition parallel to the fourth wall [15]. In the regularized case, the non-homogeneous boundary condition is a smooth non-negative definite function that converges to zero at the edges of the boundary [16].

For snapshot data, we use regularized lid-driven cavity snapshots computed via direct numerical simulation (DNS) at a discretization level of  $130 \times 130$  points and at  $Re = 15,500$  and  $Re = 30,000$  [17, 18]. We train the network with 25 x-velocity sensors placed using leverage score sampling described in Section II.A and predict the x-velocity at every cell in the snapshot. We compute the error of a network as the mean square error between reconstructed flow space and the source data. The trained networks with initial sensor positions have a relative testing error of 0.0862 for  $Re = 15,500$  and 0.0898 for  $Re = 30,000$ .

When perturbing sensor positions in Morris screening, we uniformly sample the x-position and y-position of each sensor within  $\pm 0.1$  of the initial sensor position. We then compute the Morris sensitivity indices of the network relative error to sensor perturbation in  $x$  and  $y$ . Since available sensor data is limited to the DNS discretization, we approximate the sensor values at any point in the domain with a linear-spline interpolant of the snapshot data. For sensitivity to sensor data, we compute the sensitivities  $s_{ikj}$  for each snapshot in the dataset, then compute the corresponding sensitivity indices according to Equation 6. For results present in this paper, we show the total placement sensitivity indices defined as the square root of the sum of squares of the corresponding x and y sensitivity indices. For both Morris screening and network sensitivity, we scale the sensitivity by the magnitude of the maximum perturbation distance and sensor reading respectively so that the sensitivity to sensor location and data have the same units.

### III. Results and Discussion

We use a lid-driven cavity to test the sensitivity of flow reconstruction error to sensor location using Morris screening and sensor data using network sensitivity. Figure 1 shows the means of absolute sensitivity of relative testing error with respect to sensor location ( $\mu^*$ ) and data ( $S^{abs}$ ) for  $Re = 30,000$ . We first note that the error of the network is orders of magnitude more sensitive to the location of the sensors than their sensor readings. Additionally, the error is sensitive to the location of all sensors, with the lowest sensitivity to sensor location being over 60% of the largest sensitivity, even though the error has near zero sensitivity to the data for many sensors. Therefore we conclude that the fluid-flow is approximated primarily using data from the right-side of the flow, but that alternate sensor positions for other regions can be informative. We next note that the error is highly sensitive to the location of sensors in the center-left-side of the flow but has near zero sensitivity to the data of those sensors at their initial location. Therefore, we determine this as an optimal region to place additional sensors, since the high placement sensitivity shows data in this region can influence accuracy of the network approximation, but none of the data measured by the existing sensors is weighted highly by the network.



**Fig. 1**  $Re = 30,000$  flow reconstruction total absolute sensitivity absolute mean to (a) sensor location and (b) sensor data. X-velocity contours for the first snapshots of DNS data are shown behind sensor sensitivities.

Figure 2 shows the standard deviations of sensitivity of relative testing error to sensor location ( $\sigma^*$ ) and data ( $S^{sd}$ ) for  $Re = 30,000$ . We first observe that sensors with higher magnitude of sensitivity also have higher standard deviations of sensitivity for both location and data. Additionally, the standard deviations of sensitivity for both location and data are larger than the magnitudes of sensitivity for nearly all sensors. The large standard deviations are primarily due to perturbations of both location and data increasing or decreasing error at different parameter sets, causing a mean sensitivity of approximately zero. We note that no sensors for  $Re = 30,000$  have a low magnitude of sensitivity but high standard deviation of sensitivity which may further influence the optimal placement of additional sensors.

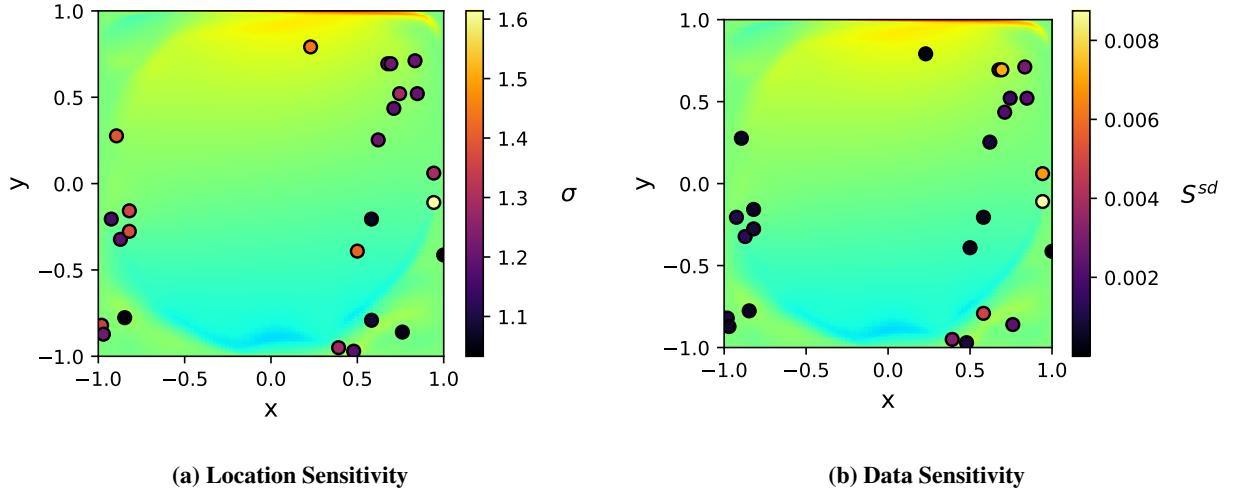
Figure 3 shows the means of absolute sensitivity of relative testing error with respect to sensor location ( $\mu^*$ ) and data ( $S^{abs}$ ) for  $Re = 15,500$ . Similarly to  $Re = 30,000$ , network error is more sensitive to sensor location than data and has significant sensitivity to the location of all sensors but the data of only a few sensors. For  $Re = 15,500$  the sensors with high data sensitivity are clustered in the bottom-right of the flow, indicating the reconstruction is approximated primarily from data in this region. However, sensors in the top-left corner of the flow have high location sensitivity but near-zero data sensitivity. Therefore, we anticipate an additional sensor in this region would be optimal to reduce reconstruction error.

Figure 4 shows the standard deviations of sensitivity of relative testing error to sensor location ( $\sigma^*$ ) and data ( $S^{sd}$ ) for  $Re = 15,000$ . Similarly to  $Re = 30,000$ , sensors with higher magnitude of sensitivity also tend to have higher standard deviations of sensitivity and the standard deviations of sensitivity are larger than the magnitudes of sensitivity for nearly all sensors. However, the sensor at approximately  $(x, y) = (-1, .6)$  is a notable exception for location sensitivity. The sensor has a moderate magnitude of sensitivity but the largest standard deviation of sensitivity, indicating that some sensor locations in that region, likely closer to the main rotational flow, have a large effect on the reconstruction error. Since that same sensor has low data sensitivity, this further supports our identification that near the top-left corner is optimal for additional sensor placement.

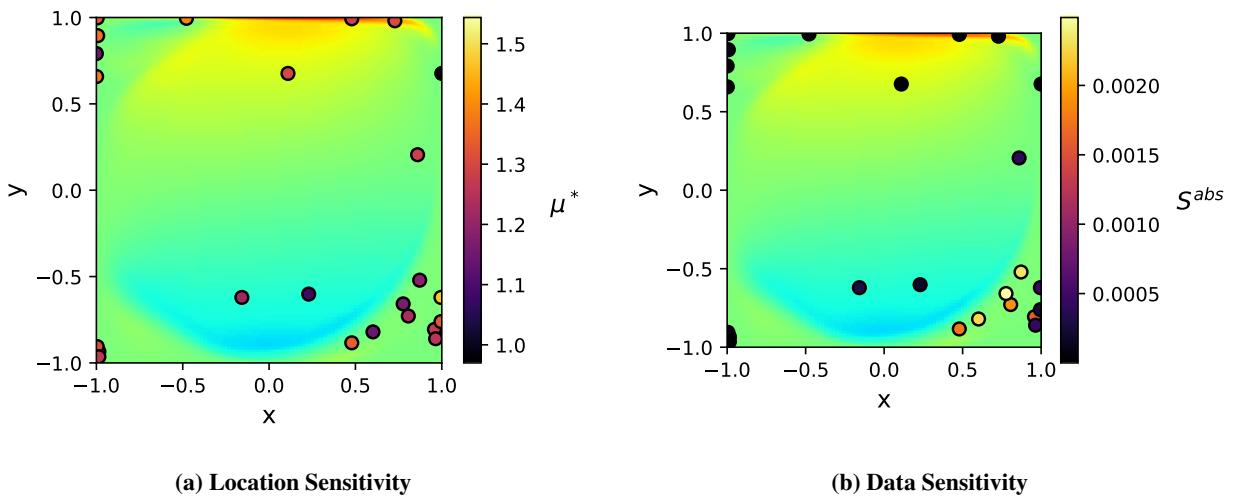
#### IV. Conclusion

Reconstructing flow-fields of physical experiments with computational fluid dynamics (CFD) has become a critical step in aerospace applications of engineering design. However, sensors often measure integrated forces and moments or sparsely measure local quantities such as velocity, limiting the accuracy of corresponding CFD results. Neural network approximations of fluid flows provide a low-cost reduced-order model directly computable from sensor data which can focus higher fidelity CFD methods and can improve the accuracy of those methods by greatly expanding the available data used for conditioning. However, understanding the variability of network approximations with respect to sensor location and error is a critical step for determining both how to improve implementation and where error in sensor readings will most impact reconstruction accuracy.

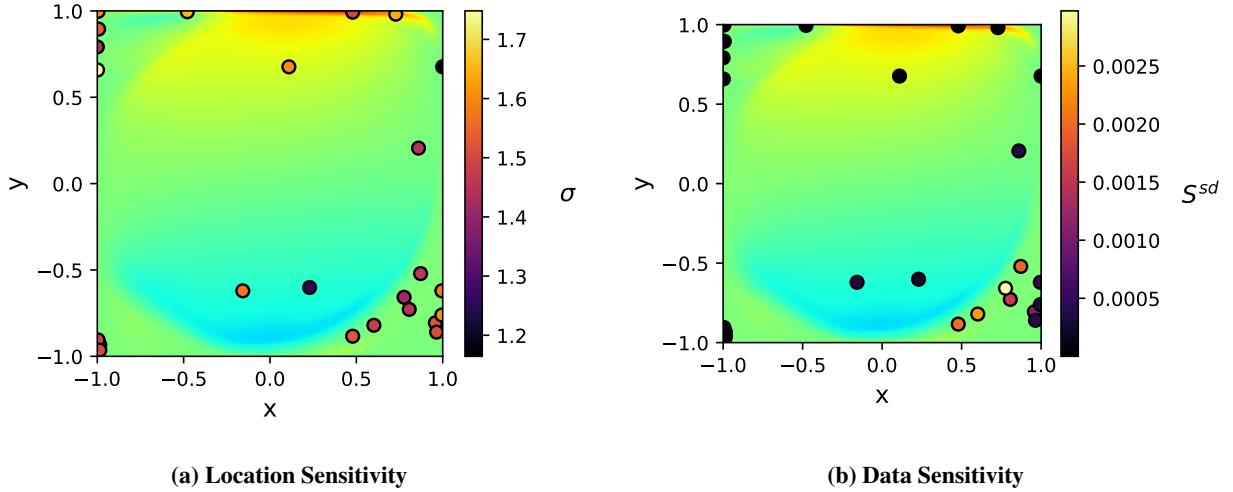
In this paper, we employ Morris screening and network sensitivity analysis methods to determine where in a fluid



**Fig. 2**  $Re = 30,000$  flow reconstruction total absolute sensitivity standard deviation to (a) sensor location and (b) sensor data. X-velocity contours for the first snapshots of DNS data are shown behind sensor sensitivities.



**Fig. 3**  $Re = 15,500$  flow reconstruction total sensitivity absolute mean to (a) sensor location and (b) sensor data. X-velocity contours for the first snapshots of DNS data are shown behind sensor sensitivities.



**Fig. 4**  $Re = 15,500$  flow reconstruction total sensitivity standard deviation to (a) sensor location and (b) sensor data. X-velocity contours for the first snapshots of DNS data are shown behind sensor sensitivities.

flow a network reconstruction is most sensitive to the location and data of sensors. Areas where sensors have high sensitivity to location but low sensitivity to data may require greater fidelity by adding more sensors while areas with high sensitivity to data may require improving the accuracy of sensors. We use this method as an alternative approach to commonly used mutual information methods which determine optimal sensor placement under uncertainty but are computationally intractable for our approach.

We computed the sensitivity of reconstruction error to sensor location and data on a lid-driven cavity test problem at  $Re = 15,500$  and  $Re = 30,000$ . We found significant sensitivity to location for all sensors, thus indicating that there are not regions in the test problem where perturbation of sensors does not affect network accuracy. However, at both Reynolds numbers, nearly all sensors with high data sensitivity were clustered in a single region of the domain. This shows that the network approximation is reconstructing the entire fluid-flow primarily from variations in sensor data of a single subset of the flow. We determined that the optimal locations for additional sensors would be in the center-right of the cavity for  $Re = 30,000$  and top-left of the cavity for  $Re = 15,500$  since nearby sensors have high sensitivity to location but near-zero sensitivity to data, indicating that nearby locations better inform flow-field reconstruction.

Sensitivity analysis provides an effective method to determine locations for additional sensors where other methods such as mutual information metrics are computationally intractable. For future work, we intend to extend this method to more complex flows around an Orion Lander. Network approximations can be trained directly on the integrated forces and moments and sparse velocity, pressure, and temperature sensors that are used in wind-tunnel experiments, allowing estimation of fluid-flows direct from experimental data to better condition high-fidelity CFD approximations. Sensitivity analysis of those network approximations could determine where to put additional sensors in wind-tunnel experiments to produce more accurate high-fidelity CFD approximations.

### Acknowledgments

This work was authored by employees of North Carolina State University under Contract No. 80LARC21CA003 with the National Aeronautics and Space Administration and by employees of University of Florida under Contract No. 80LARC21CA005 with the National Aeronautics and Space Administration. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, worldwide license to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, or allow others to do so, for United States Government purposes. All other rights are reserved by the copyright owner.

## References

- [1] Edwards, J. M., Danao, L. A., and Howell, R. J., “PIV measurements and CFD simulation of the performance and flow physics and of a small-scale vertical axis wind turbine,” *Wind Energy*, Vol. 18, No. 2, 2015, pp. 201–217. <https://doi.org/https://doi.org/10.1002/we.1690>.
- [2] O. San, R. M., and Ahmed, M., “An artificial neural network framework for reduced order modeling of transient flows,” *Commun. Nonlinear Sci. Numer. Simul.*, Vol. 77, 2019, pp. 271–287.
- [3] Qu, G., Zhang, D., and Yan, P., “Information measure for performance of image fusion,” *Electronics Letters*, Vol. 38, No. 7, 2002, pp. 1–2. <https://doi.org/10.1049/el:20020212>.
- [4] Schmidt, K., Smith, R. C., Hite, J., Mattingly, J., Azmy, Y., Rajan, D., and Goldhahn, R., “Sequential optimal positioning of mobile sensors using mutual information,” *Statistical Analysis and Data Mining*, Vol. 12, No. 6, 2019. <https://doi.org/10.1002/sam.11431>.
- [5] Gordon, N., Gilkey, L., Smith, R. C., Michaud, I., Williams, B., Mousseau, V., Hooper, R., and Jones, C., “A Mutual Information-Based Experimental Design Framework to Use High-Fidelity Nuclear Reactor Codes to Calibrate Low-Fidelity Codes,” *Nuclear Technology*, Vol. 205, No. 12, 2019, pp. 1685–1696. <https://doi.org/10.1080/00295450.2019.1590073>.
- [6] Smith, R. C., *Uncertainty Quantification: Theory, Implementation, and Applications*, SIAM Computational Science and Engineering, 2014.
- [7] Alaoui, A. E., and Mahoney., M. W., “Fast randomized kernel methods with statistical guarantees,” *Neural Information Processing Systems*, 2015.
- [8] Morris, M. D., “Factorial Sampling Plans for Preliminary Computational Experiments,” *Technometrics*, Vol. 33, No. 2, 1991, pp. 161–174. <https://doi.org/10.1080/00401706.1991.10484804>.
- [9] Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M., *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*, John Wiley & Sons Ltd, West Sussex, England, 2004.
- [10] Pizarroso, J., Portela, J., and Muñoz, A., “NeuralSens: Sensitivity Analysis of Neural Networks,” *CoRR*, Vol. abs/2002.11423, 2020. URL <https://arxiv.org/abs/2002.11423>.
- [11] Adler J, O. O., “Solving ill-posed inverse problems using iterative deep neural networks,” *Inverse Probl*, Vol. 33, 2017.
- [12] Erichson, N. B., Mathelin, L., Yao, Z., Brunton, S. L., Mahoney, M. W., and Kutz, J. N., “Shallow neural networks for fluid flow reconstruction with limited sensors,” *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, Vol. 476, No. 2238, 2020, p. 20200097.
- [13] Drineas, P., Magdon-Ismail, M., Mahoney, M. W., and Woodruff, D. P., “Fast approximation of matrix coherence and statistical leverage,” *The Journal of Machine Learning Research*, 2012.
- [14] Campolongo, F., Cariboni, J., and Saltelli, A., “An effective screening design for sensitivity analysis of large models,” *Environmental Modelling & Software*, Vol. 22, No. 10, 2007, pp. 1509–1518. <https://doi.org/https://doi.org/10.1016/j.envsoft.2006.10.004>.
- [15] Lee, M. W., Dowell, E. H., and Balajewicz, M. J., “A study of the regularized lid-driven cavity’s progression to chaos,” *Communications in Nonlinear Science and Numerical Simulation*, Vol. 71, 2019, pp. 50–72. <https://doi.org/https://doi.org/10.1016/j.cnsns.2018.11.010>.
- [16] Shen, J., “Hopf bifurcation of the unsteady regularized driven cavity flow,” *Journal of Computational Physics*, Vol. 95, No. 1, 1991, pp. 228–245.
- [17] Lee, M. W., “Steady State 2D Regularized Lid-Driven Cavity Low-resolution Spatiotemporal Snapshots - Part 1,” *Mendeley Data*, Vol. 1, 2021. <https://doi.org/10.17632/cptf8c5h74.1>.
- [18] Lee, M. W., “Steady State 2D Regularized Lid-Driven Cavity Low-resolution Spatiotemporal Snapshots - Part 3 of 3,” *Mendeley Data*, Vol. 1, 2021. <https://doi.org/10.17632/74g8rjfkv1>.