CS 229 Project: Final Report

Incorporating Global Information into Local Upscaling of Fluid Flow in Porous Media

Kasama Itthisawatpan and Perth (Pu) Charernwattanagul

December 14, 2012

1 Introduction

1.1 Background

Making an accurate forecast of hydrocarbon production is one of the most important tasks in the upstream petroleum industry. To accomplish such goal, petroleum engineers use numerical simulator to model the fluid movement in the subsurface reservoir. The accurate production forecast can be achieved only if the reservoir model is accurate.

Reservoir simulation models that have finer grid sizes can generally provide more accurate results. However, performing fine-scale simulation is often not attainable due to its high computational cost, and the simulation is performed on the coarser models. The process that generates coarse-scale models from fine-scale models while trying to preserve the solution is called **upscaling**.

The project focuses on the upscaling of the single-phase flow properties (only transmissibility), which is often performed almost exclusively in the petroleum industry [2]. In particular, we tried to generate an accurate upscaled transmissibility, which measures how much fluid can flow between two simulation blocks given a pressure difference, as illustrated in Figure 1.

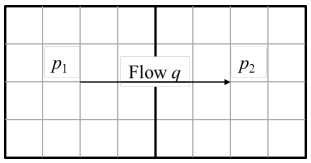

Figure 1: Schematic diagram showing the flow from coarse block 1 (left) to coarse block 2 (right). Transmissibility is defined as the ratio between flow rate and pressure drop $T_{12} = \frac{q_{12}}{p_1 - p_2}$.

1.2 Local and Global Upscaling

The upscaling methods can be classified into local upscaling and global upscaling. The difference between local and global upscaling is the amount of information required in constructing the equivalent properties. For transmissibility upscaling, local upscaling uses the areas around the interface of interest to compute the equivalent coarse-scale transmissibility. It is often very quick yet inaccurate.

On the other hand, global upscaling performs the full fine-scale single-phase simulation and use its results to solve for effective transmissibility. Global upscaling is usually more accurate than local upscaling. However, it is computationally expensive, as the full fine-scale simulation is required.

1.3 Project Description

Advantages and disadvantages of local and global upscaling form a motivation of the project. It is desirable that we use local upscaling for all geostatistical realizations. However, due to the challenging geology, such accuracy is not achieved, so some amount of global upscaling must be used. Because fine-scale models are generated by geostatistical methods, there may be some similarities among the realizations. We aimed to apply global upscaling to a small percentage of the realizations. Then, we trained machine learning models so that they can correct the local upscaling results from other realizations.

2 Data and Methods

2.1 Fine-Scale Model Generation

The data used in this study is collected from two stages. First, we generate 2D fine-scale properties by geostatistical methods. We applied training image generator and Sequential Gaussian Simulation in the software SGeMS [6] to create the permeability distribution. The models are representative of a reservoir deposited in the fluvial environment. The properties of each facies are similar to those of the top layers of the standard Stanford V reservoir [5]. An example of rock permeability distribution is shown in Figure 2.

It is generally known that this type of reservoir properties is very challenging for upscaling. Specifically, local upscaling usually results in predicting flow rate that is too low given a specified pressure drop [8].

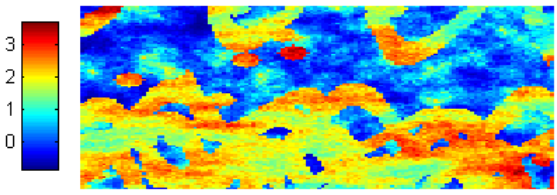

Figure 2: An example of permeability field in log-scale ($\log_{10}$ md) from one of the realizations used in this study. Hot colors represent high permeability regions, where the flow rate is high.

2.2 Upscaling Implementation

For local upscaling, we applied the standard transmissibility upscaling (see e.g. [2] for an explanation) with the fine-scale simulation result from MRST [4]. For global upscaling, we used an iterative method as outlined in [3].

Fine-scale simulation and upscaling were performed on all realizations to confirm our expectation. The results are shown as empirical CDF in Figure 3. It is seen that global upscaling can accurately reproduce fine-scale flow statistics, while local upscaling shows significant underestimation of the flow.

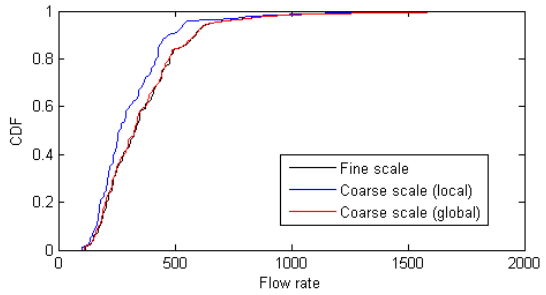

Figure 3: Empirical CDF from 100 realizations. The coarse-scale simulation from local upscaling gives consistently lower flow rate than the fine scale simulation.

We have 100 geostatistical realizations, from each of which, 257 transmissibility values are obtained. In total, we have 25700 data points for training and testing.

2.3 Feature Extraction

As discussed earlier, a major difference between local upscaling and global upscaling is the amount of information considered in the upscaling processes. While local upscaling uses only the properties adjacent to the coarse-block interface, global upscaling take the entire reservoir properties, as well as the well locations, into account. Therefore, the discrepancies between local upscaling results and global upscaling results should depend on the global permeability field that is not captured in the local domain when local upscaling is performed. Therefore, we considered the following properties as the *attributes* for the model.

1. location (x, y) of the interface,
2. permeability map of the local domain, and
3. simplified permeability map of the extended domain (12 coarse blocks around the interface).

Because the underestimation and overestimation of the transmissibility values from local upscaling depend on the distribution of *low* and *high* permeability in the domain instead of the actual permeability values, we discretized the permeability to

$$\hat{p}_i = \begin{cases} 1 & p_i \geq \bar{p}_{Di} \\ 0 & p_i < \bar{p}_{Di} \end{cases}$$

where $\bar{p}_{Di}$ is the geometric mean of permeability in the extended local domain. The attributes are illustrated in Figure 4.

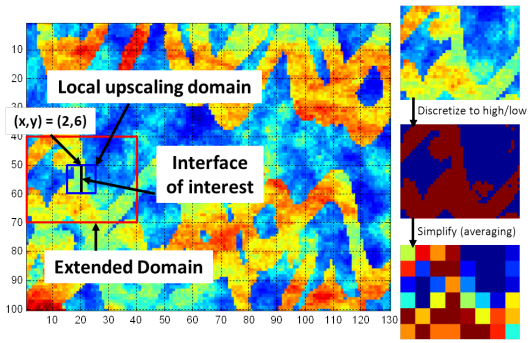

Figure 4: Processing of the attributes

All the attributes were scaled to $[0, 1]$. These attributes are supplied directly to the classification models with the training data set from 40 realizations. The rest of the realizations are used as the testing data set. For transmissibility prediction, we further simplified the attributes e.g. taking averages of some groups of attributes to reduce the complexity of the model since we have smaller training data sets.

2.4 Machine Learning Techniques

In this project, we consider the results from global upscaling as the accurate results and tried to correct the values from local upscaling. In other words, we want to predict the factors β_i of all coarse-block interfaces such that

$$T_{\text{global},i} = \beta_i T_{\text{local},i}, \text{ or}$$
$$\log T_{\text{global},i} = \log \beta_i + \log T_{\text{local},i}$$

Figure 5 shows the histogram of $\log \beta_i$.

The factors $\log \beta_i$ indicate whether the local upscaling results underestimate or overestimate the true transmissibility. We applied machine learning techniques in two stages: classification and prediction.

2.4.1 Classification

The goal of classification is to identify the interfaces with $\log \beta_i \gg 0$, $\log \beta_i \approx 0$, and $\log \beta_i \ll 0$, which correspond

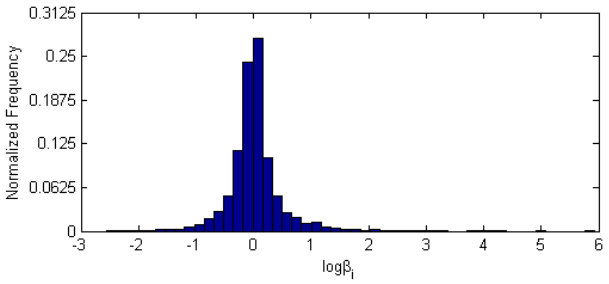

Figure 5: Histogram showing the statistics of $\log \beta_i$ in the training data set.

to the cases with transmissibility from local upscaling being a significant overestimation, a good approximate, and a significant overestimation, respectively. In particular, each interface in the training data set assigned the category from the following rule:

$$y^{(i)} = \begin{cases} -1 & \log \beta_i < -a, \text{ (Underestimation)} \\ 0 & -a \le \log \beta_i \le a, \text{ (Satisfactory Estimation)} \\ 1 & \log \beta_i > a, \text{ (Overestimation)} \end{cases}$$

where a is some threshold between 0.02 and 0.3.

Multiclass Support Vector Machines Using the set of feature vectors, we applied SVM to classify the interfaces. For three-class classification, *one-against-one* strategy is applied in selecting the most probable group. The optimization problem for multiclass SVM is set up as

$$\begin{aligned} \min_{\mathbf{w}_{ij}, b_{ij}, \xi_{ij}} \frac{1}{2}\mathbf{w}_{ij}^T \mathbf{w}_{ij} + C\sum_{t=1}^{m} \xi_{ij}^{(t)} \text{ subject to} \\ \mathbf{w}_{ij}^T \phi(\mathbf{x}^{(t)}) + b_{ij} \ge 1 - \xi_{ij}^{(t)}, \ \mathbf{y}^{(t)} = i \\ \mathbf{w}_{ij}^T \phi(\mathbf{x}^{(t)}) + b_{ij} \le -1 + \xi_{ij}^{(t)}, \ \mathbf{y}^{(t)} = j \\ \xi_{ij}^{(t)} \ge 0 \end{aligned}$$

From the experiment, we observe that the radial basis function (RBF) kernel $\phi(\mathbf{x}, \mathbf{z}) = \exp\left(-\gamma \|\mathbf{x} - \mathbf{z}\|^2\right)$ generally gives better results than linear or polynomial kernels, so we applied RBF kernel throughout the project. The library `libSVM` [1] is used for implementation.

We also studied the effects of the parameters C and γ. These parameters are adjusted so that we get a reasonable result ($C = 10, \ \gamma = 0.005$).

k-means. We performed k-means clustering on the training data set. The number of clusters ranges between 3 and 30 groups. After convergence, we study the characteristics of each cluster by observing the distribution of the correction factor β_i.

2.4.2 Prediction

Once we classified the interfaces into groups, we construct supervised learning models with reduced feature sets to predict the correction factor $\log \beta_i$. The model parameters are different for different groups.

Linear Regression. For each group l, we find the parameter θ_l such that

$$\theta = \arg\min_{\theta_l} \frac{1}{2} \sum_{i=1}^{m_l} \left(y^{(i)} - \theta_l^T \mathbf{x}^{(i)} \right)^2$$

Support Vector Regression (SVR). Similar to support vector machines, SVR model can be constructed from an optimization problem

$$\begin{aligned} \min_{\mathbf{w}, b, \xi, \xi^\star} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m_l} \xi_i + C\sum_{i=1}^{m_l} \xi_i^\star \text{ subject to} \\ \mathbf{w}^T \phi(\mathbf{x}^{(i)}) + b - y^{(i)} \le \epsilon + \xi_i \\ y^{(i)} - \mathbf{w}^T \phi(\mathbf{x}^{(i)}) - b \le \epsilon + \xi_i^\star \\ \xi_i, \xi_i^\star \ge 0 \end{aligned}$$

Because each model has to be constructed from the training data in each group, the number of training data points in the training set is smaller than the classification stage. To prevent against possible bias if the training set is not sufficient, we also tried a simpler linear kernel $\phi(\mathbf{x}, \mathbf{z}) = \mathbf{x}^T \mathbf{z}$ in addition to the RBF kernel for the SVR problem.

It is worthwhile to note that the approach is slightly different from the typical machine learning framework. Earlier in the project, we attempted to correct the local upscaling results directly using linear regression and support vector regression on the continuous-value $\log \beta_i$ and multiclass support vector machine on the discretized $\log \beta_i$. However, as shown in Figure 5, the large majority of $\log \beta_i$ are close to 0. However, the interfaces with $\log \beta_i$ very far from 0 are the most important contributors to the inaccuracy of local upscaling. Straightforward prediction does not capture these outliers, so it turns out to be an inaccurate correction.

It should also be noted that the number of realizations used in the training data set is limited by practical application. In reality, we do not wish to perform global upscaling on too many realizations, as it is against the initial goal of applying machine learning techniques to reduce the need for global fine-scale simulation. Around 100 to 200 realizations is a practical range for uncertainty quantification in the industry.

3 Results

3.1 Classification

Figure 6 plots an example of the classification results from **support vector machines**. Figure 7 shows how varying the parameters a (threshold to assign actual categories) affects the accuracy.

Although the classification is not perfect, 60-70 percent accuracy is sufficient for proceeding to the prediction

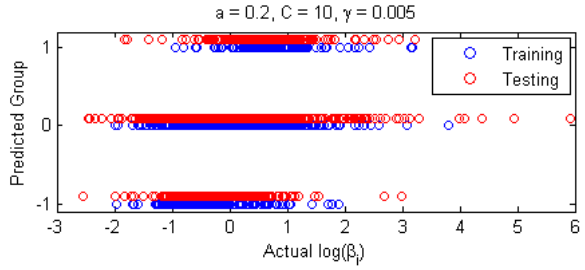

Figure 6: Classification Result, Accuracy: 62.1%

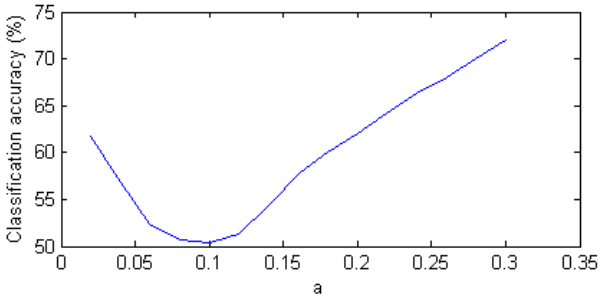

Figure 7: Accuracy of classification with different a

stage, as will be shown in the next section. The result shown in Figure 7 suggests that it is important to choose a reasonable values of the threshold a. For $a > 0.1$, larger a leads to higher accuracy of classification. This result is aligned with the observations when the feature sets are chosen (that the permeability distribution causes inaccuracy of the local upscaling results). Separation between the *very inaccurate results* and the *satisfactory results* is clearer when we define a stricter condition of deeming the result to be *very inaccurate* (when a is large).

However, too large a, while having accurate classification result, will lead to too small set of interfaces being classified as inaccurate, which will affect the performance of the next stage since the number of training data will be too small. We choose $a = 0.2$ for the rest of the study.

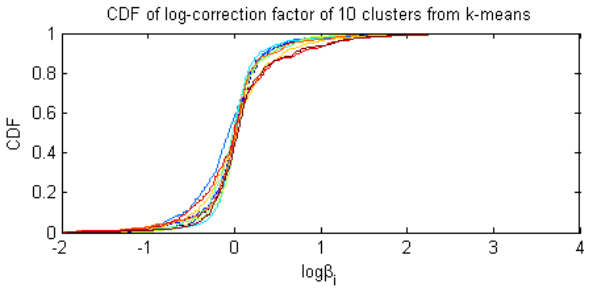

Figure 8: Distribution of correction factors of each cluster from k-means

An example of the results from **k-means** (with $n_{\text{clusters}} = 10$) is plotted in Figure 8. As the distribution of the correction factors from all clusters are very similar, it can be concluded that the correction factors cannot be differentiated by proximity of the feature vectors alone. This general result applies when we range the number of clusters from 3 to 30.

From the fact that support vector machines can perform much better with the RBF kernel than with the simple linear kernel, it is seen that the correction factor can be separated only in the high dimensional space. The results from k-means reinforces this observation, as it fails to distinguish between the clusters in the primitive feature space.

3.2 Prediction

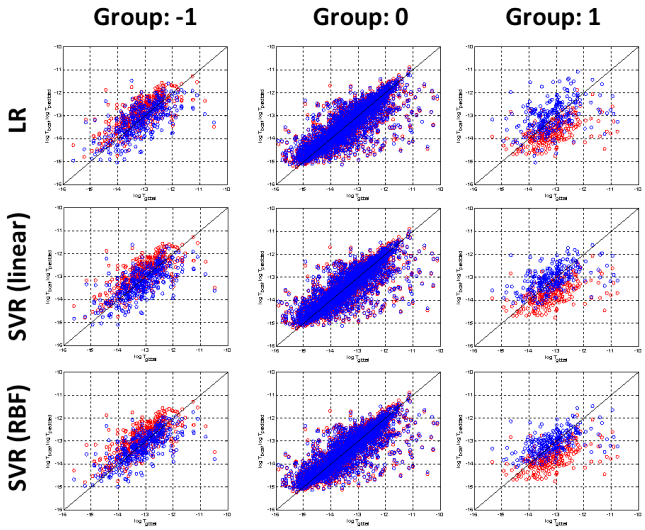

Figure 9: Comparing the prediction results from different groups and different methods. Red dots show uncorrected T_{local}. Blue dots show the corrected T. Black lines correspond to 1-1 prediction (perfect correction).

Figure 9 shows the results of the correction factor prediction on different groups by different methods. It is shown that both linear regression and support vector regression can slightly improve the transmissibility estimates from the values given by local upscaling as it is seen that the predicted values cluster slightly closer to the 1:1 line (which would correspond to perfect correction).

The performances of the predictions are measured as mean absolute error and root mean square error between the global upscaling result (treated as the correct value) and the result after correction from different methods in the testing data set. The errors are shown in the following table.

Method	Mean Abs. Err.	RMS Err.
No Correction	0.3034	0.5118
Linear Regression	0.2925	0.5018
SVR (Linear kernel)	0.2802	0.4890
SVR (RBF kernel)	0.2741	0.4839

From the table, both linear regression and support vector regression can slightly correct the transmissibility values from local upscaling. Comparison between the methods shows that support vector machines with RBF

kernel gives the best correction, but the correction is still small. From all the methods, the reduction of the mean absolute error is very similar in magnitude to the reduction of the root mean square error. This suggests that the extremes (data with large $|\log \beta_i|$) are not corrected.

4 Discussion and Conclusion

In this project, we presented the potential uses and limitations of an attempt to correct the effective transmissibility from local upscaling using the permeability distribution information. We have seen that in general, the permeability distribution information can be used to predict whether the transmissbility from local upscaling will be an underestimation or an overestimation (classification stage). For the prediction stage, linear regression and support vector regression can slightly predict the correct values of the effective transmissibility. However, the correction is small and may not have a large effect on the flow behavior. From the results, we make the following observations:

- The simplified attributes applied in this project cannot capture the complex relationship between transmissibility from local and global upscaling. This is seen during the project that if the actual permeability values are used as inputs instead of the processed values, SVM gives perfect prediction on the training data set but inaccurate prediction on the testing data set. This observation suggests that the model has high variance, and the result may improve with a more complex model.
- The extended domain considered in this project is not large enough to capture the general pattern of the permeability distribution from different geostatistical realizations. The physical problem that governs the single-phase upscaling is an elliptic PDE, so the property at a single point is affected by all points in the global domain. This project assumes that the extended local domain is large enough to represent the entire reservoir. This assumption may be invalid. Consideration of a larger domain may enhance accuracy.
- The extreme values of transmissibility (the ones whose results from local upscaling are highly incorrect) are still problematic. One possible remedy is to increase the number of categories in the classification stages, so the extremes can be grouped together.

All the discussions above would implies the necessity of a more complex model, as its degrees of freedom increases, which would in turn demand for a larger training data set. As discussed earlier, this may not be practical as we aimed to limit the number of global upscaling runs to reduce computational burden. However, it could be more useful in a much larger reservoir model (with a much larger number of coarse blocks), where many more coarse block interfaces can provide more training data. The success of this method on a larger model needs investigation.

Despite the imperfection of the prediction stage, this project provides an insight on the possibility of applying the machine learning techniques to aid the upscaling problems. The classification stage suggests that even with the simplified attributes, the permeability distribution maps can be used to predict the fluid flow properties at least qualitatively. This finding can be useful in other upscaling processes, especially those that only focus on local domain such as relative permeability—a multiphase parameter—upscaling, which is not considered in this project. In fact, several attempts have been successful in using k-means to aid local relative permeability upscaling [7]. It is seen in this project that SVM can work better than k-means under the non-clustering permeability distribution. It will be worth investigating the use of SVM in relative permeability upscaling.

5 Acknowledgment

We would like to thank Professor Louis Durlofsky (Energy Resources Engineering) for the motivation and guidance for the project. We also thank Matthieu Rousset (Ph.D. student, Energy Resources Engineering) for providing the global upscaling code.

References

[1] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[2] Y. Chen. *Upscaling and Subgrid Modeling of Flow and Transport in Heterogeneous Reservoirs*. PhD thesis, Stanford University, 2005.

[3] Y. Chen, B. T. Mallison, and L. J. Durlofsky. Nonlinear two-point flux approximation for modeling full-tensor effects in subsurface flow simulations. *Computational Geosciences*, 2008.

[4] K. A. Lie, S. Krogstad, I. S. Ligaarden, J. R. Natvig, H. M. Nilsen, , and B. Skaflestad. Open source matlab implementation of consistent discretisations on complex grids. *Comp. Geo.*, 16(2):297–322, 2012.

[5] S. Mao and A. G. Journel. Generation of a reference petrophysical and seismic 3d data set. Technical report, SCRF Report, Stanford Univ., 1999.

[6] N. Remy, J. Wu, and A. Boucher. *Applied Geostatistics with SGeMS: A User's Guide*. Cambridge University Press, 2009.

[7] S. Suzuki. Pattern-based approach to multiphase flow upscaling using distance-based clustering. In *Proceeding to SPE Annual Technical Conference and Exhibition*, Denver, CO, USA, 2011.

[8] X. H. Wen, L. J. Durlofsky, and M. G. Edwards. Use of border regions for improved permeability upscaling. *Math. Geol.*, 35:521–547, 2003.