

# Two-dimensional upscaling of reservoir data using adaptive bandwidth in the kernel function

Mohammadreza Azad<sup>a,\*</sup>, Abulghasem Kamkar Ruhani<sup>a</sup>, Behzad Tokhmechi<sup>a</sup>, Mohammad Arashi<sup>b</sup>

<sup>a</sup> Faculty of Mining, Petroleum & Geophysics Engineering, Shahrood University of Technology, Shahrood, Iran

<sup>b</sup> Faculty of Mathematical Science, Shahrood University of Technology, Shahrood, Iran

## Article History:

Received: 03 December 2018,

Revised: 15 May 2019,

Accepted: 26 May 2019.

## ABSTRACT

In this paper, a new method called adaptive bandwidth in the kernel function has been used for two-dimensional upscaling of reservoir data. Bandwidth in the kernel can be considered as a variability parameter in porous media. Given that the variability of the reservoir characteristics depends on the complexity of the system, either in terms of geological structure or the specific feature distribution, variations must be considered differently for upscaling from a fine model to a coarse one. The upscaling algorithm, introduced in this paper, is based on the kernel function bandwidth, written in combination with the A\* search algorithm and the first-depth search algorithm. In this algorithm, each cell in its x and y neighborhoods as well as the optimal bandwidth, obtained in two directions will be able to be merged with its adjacent cells. The upscaling process is performed on artificial data with 30×30 grid dimensions and SPE-10 model as real data. Four modes are used to start the point of upscaling and the process is performed according to the desired pattern, and in each case, the upscaling error and the number of final upscaled blocks are obtained. Based on the number of coarsen cells as well as the upscaling error, the first pattern is selected as the optimal pattern for synthetic data and the second pattern is selected as the optimal simulator model for real data. In this model, the number of cells was 236 and 3600, and the upscaling errors for synthetic and real data were 0.4183 and 12.2, respectively. The results of the upscaling in the real data were compared with the normalization method and showed that the upscaling error of the normalization method was 15 times the upscaling error of the kernel bandwidth algorithm.

**Keywords :** *Upscaling, Bandwidth, Kernel, Cell, Upscaling error, Optimum model*

## 1. Introduction

Heterogeneous reservoirs are characterized by variation in the reservoir properties in all directions and at different length scales within the reservoirs. Generally, reservoir heterogeneity can be divided into small-scale and large-scale heterogeneities. In practice, the properties of a reservoir are described by collecting the available well data assigned to specific well locations, and then, using geostatistical methods to populate the properties throughout the reservoir. The behavior of reservoir properties is investigated by geological models. The geological models usually consist of tens or even hundreds of millions of grid blocks. Although geological models are referred to as very fine models, fine cells are still much larger than the small-scale heterogeneity, especially in the areal direction. This is due to the costs imposed by long computation times and possible rise of convergence problems, especially when it is required to run multiple fine-scale simulations in order to assess various geological and development scenarios. Therefore, building more coarse and practical models (usually referred to as simulation models) becomes important. In the simulation model, the number of fine grid cells are reduced by merging the fine cells into coarser ones. Afterwards, reservoir properties are averaged within the coarse domain. The process of coarsening the fine grid is usually referred to as up-gridding, while averaging reservoir properties within the coarse cells is referred to as upscaling. The target of upscaling is to replace very fine and detailed models with coarse models, including much less data.

These coarse models are more feasible for running simulations than fine models. However, upscaling does not aim to speed up reservoir simulations at the cost of simulation results. On the contrary, upscaling techniques aim to build coarse models that preserve the most important flow characteristics of fine models and capture the sub-grid heterogeneity [1-8].

Several upscaling methods have been introduced in the literature, some of which are analytical, and others are numerical. Analytical methods (also called averaging methods) such as arithmetic, harmonic, and geometric methods are simple and can be applied successfully to reservoir properties such as porosity and water saturation. However, applying these averaging methods to permeability requires idealized conditions that may not be present in heterogeneous reservoirs. Numerical methods are usually used for permeability determination. These methods can be divided, according to fluid phases flowing in the reservoir, into single-phase and two-phase upscaling methods [2, 4, and 6].

Normally, regardless of the type of upscaling method, the upscaling process should be such that the results of the coarse-grained model can replace those of the fine-grained model. Chen et al. (2015 & 2018) have presented the multiple boundary method for three-dimensional fractured porous rocks with the commonly used Oda upscaling method and the volume averaging method [3 and 9]. In two-dimensional upscaling, it is important to enter the effect of wells because these areas witness high changes under pressure and fluid saturation, and it is highly desirable to maintain a fine structure in these areas. The near-well upscaling technique, which can dramatically improve the accuracy of

\* Corresponding author. E-mail address: [azad66.mohammad@gmail.com](mailto:azad66.mohammad@gmail.com) (M. Azad).

coarse grid simulation for well productivity in heterogeneous media, was presented by Ding (2004) [10]. Gholinezhad et al. (2015) applied quad-tree decomposition method for areal upscaling of irregular-shaped petroleum reservoirs. The simulation results demonstrated that the resulted upscaled model could give accurate results compared to the original fine model, but it was 13.5 times faster than the original fine model [11]. Wavelet transform is also a common method for upscaling that has extensively been used. Moslehi et al. (2016) used an upscaling method based on the wavelet transformations (WTs) coarsening the computational grid based on the spatial distribution of property. The technique is applied to a porous formation with broadly distributed and correlated property values, and the governing equation for solute transport in the formation is solved numerically. The WT upscaling preserves the resolution of the initial highly-resolved computational grid in the high property zones, as well as that of the zones with sharp contrasts between the neighboring properties, whereas the low-property zones are averaged out. From a computational point of view, the wavelet transform upscaling method generates non-uniform grids with a number of grid blocks that represent, on average, only about 2.1% of the number of grid blocks in the original high-resolution models [12].

Most of the upscaling methods, developed in two dimensions, are so that the final cells in the upscaling process are obtained on a regular basis. For example, in the wavelet-based upscaling, two cells along x and y either merge or not and there is no state that due to different variability, three cells are merged and one cell remains in the fine-scale. Upscaling based on adaptive bandwidth in the kernel method will solve this problem. Due to the variability in various directions, the threshold or bandwidth can be defined in such a way that cell upscaling can be made by combining the cells until the coarse-graining condition is established. The only condition for cell integration is merely variability with no restrictions to be imposed through the method. In this method, the bandwidth of the kernel function is a function of cell variability. The bandwidth is determined by the variability of features in a non-uniform and intelligent manner. In this method, it will be possible to construct the simulator based on the upscaling error or the number of required upscaled blocks. In the following, first, the research methodology is presented and the bandwidth upscaling algorithm will be described. Then, by describing the data used in this research, the results of upscaling will be examined.

The results of upscaling based on the bandwidth of the kernel function are compared with the normalization method. The renormalization approach is based on replacing the direct upscaling process of a fine model to a coarse one by series of upscaling steps in which the initial grid is coarsened by merging cells to obtain successively coarser grids until a grid with one block only is built. The renormalization method was first used for single-phase flow upscaling and demonstrated to give accurate results [13].

## 2. Research Method

Nonparametric estimators provide an accurate estimate of the probability distribution of data without any assumptions about parameters and the density distribution of a feature. The kernel density estimator (KDE) is the most widely used method among these estimators, and has many applications in many fields [14 and 15]. A typical formula for the multivariable kernel density function is given by the following expression:

$$\hat{f}(x, H) = \frac{1}{n} \sum_{i=1}^n \frac{1}{\det(H)} K(H^{-1}(X_i - x)) \quad (1)$$

where  $x = (x_1, x_2, x_3, \dots, x_d)^T$  and  $X = (X_{i1}, X_{i2}, X, \dots, X_{id})^T$ ,  $i = 1, 2, \dots, n$  are a sequence of independent identically distributed d-variate random variables drawn from a (usually unknown) density function  $f$ .  $H$  is the bandwidth matrix or the stationary smoothing parameter, which in the univariate mode, is a scalar quantity and  $K$  is the kernel function. There are two main computational problems related to KDE: (a) the fast evaluation of the kernel density estimate  $\hat{f}$ , and (b) the fast estimation (under certain criteria) of the optimal bandwidth matrix  $H$

(or scalar  $h$  in the univariate case). Determining the optimal smoothing parameter is far more important than evaluating the type of the kernel function [16]. In practice, if bandwidth is too large, the estimate will be very smooth and there might be many mistakes. In contrast, if bandwidth is small, the estimate will be much distorted. Thus, determining the optimal bandwidth is very important. Typical methods for the determination of the optimal bandwidth or threshold are cross validation methods and plug-in methods [17]. In the kernel estimation, there are two basic approaches to determine the bandwidth: 1. fixed bandwidth and 2-variable bandwidth. In the variable bandwidth approach, the points and more precisely their variability for bandwidth determination plays a significant role while in a fixed bandwidth method, a constant fixed value for bandwidth is used everywhere in the range of observations, and the oscillation of the points has no effect on bandwidth at any point. Considering that the upscaling should be related to cell variability, the variable bandwidth approach can be considered as the bandwidth, which is a function of the variability of the reservoir property. By the use of a small bandwidth, one will have the smallest upscaling level in areas with high variations, where they will remain fine. On the contrary, in areas with low and smooth variations, when choosing a large bandwidth, most blocks will be merged and the areas will be coarse. Therefore, the bandwidth will be determined based on variability.

For upscaling in two dimensions based on the bandwidth of the kernel function, the optimal bandwidth in each direction must be determined for each row of data. This work requires an upscaling algorithm in one dimension, as displayed in Fig. 1.

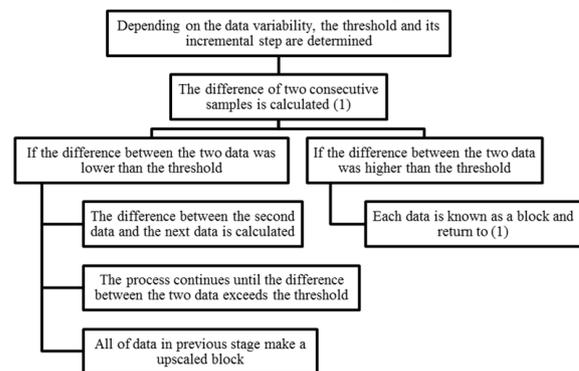


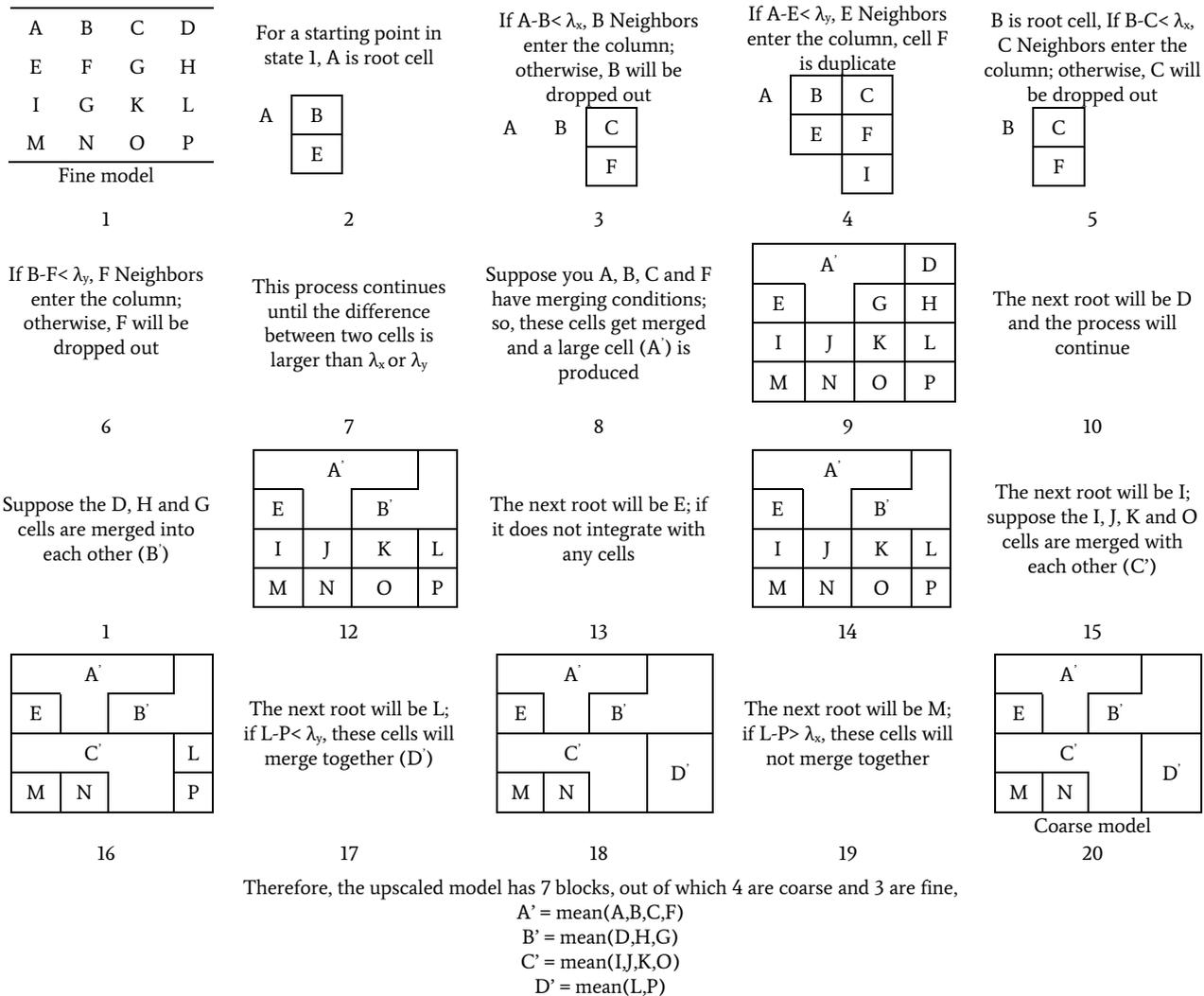
Fig. 1. Kernel bandwidth-based upscaling process in one dimensional.

In each row of data in two directions, the optimal bandwidth is calculated for both x and y directions. These values are required in the two-dimensional reservoir upscaling algorithm. The upscaling algorithm in two dimensions is a combination of two algorithms including;  $A^*$  search and the first-depth search pattern. In computer science,  $A^*$  is a computer algorithm that is widely used in pathfinding and graph traversal, which is the process of finding a path between multiple points, called *nodes*. It has enjoyed widespread use due to its performance and accuracy. It can be considered as an extension of Edsger Dijkstra's algorithm introduced in 1959.  $A^*$  achieves better performance by using heuristics to guide its search [18]. Depth-first search (DFS) is an algorithm for traversing or searching tree or graph data structures. The algorithm starts at the root node (selecting some arbitrary nodes as the root node in the case of a graph) and explores as far as possible along each branch before backtracking. The first depth search strategy for scrolling the graph, as its name implies, is to look deeper into the graph as long as possible. In each step, the neighboring vertices of the current head are examined and, as soon as they encounter the neighbors that have not been seen before, they execute vertically for that vertex. If all the neighbors have already been seen, the algorithm will roll back and run the algorithm for the verge from which we reach the top of the head. In other words, the algorithm goes as deep as possible and retreats in the face of a dead end. This process continues until all of the roots are accessible from the root [19].

In the two-dimensional upscaling algorithm, based on the bandwidth

of the kernel function, each cell is considered as a vertex of the graph. The starting point is arbitrary. For example, the origin of the coordinates can be the starting point of the algorithm, for which all neighborhoods are placed on a stack. Each point, depending on the position, is located in two directions x and y, and has a bandwidth obtained from the previous step. The point is compared with its neighborhoods that are on the stack. If the difference between the two data is lower than the threshold or bandwidth these two cells will be merged. Then the neighbors of this cell enter the stack, and the data of the neighbor cell will be compared with the cell data. If the difference between the two data is greater than the threshold, it means that the two data are important and the two cells should remain in its original state or fine-scale. This procedure will continue and all data will be examined. Eventually, in a computational grid, a series of cells will remain fine and a series of cells will be merged together. In upscaled cells, the average property data of the cells is calculated as the equivalent property and the cells will be considered as a coarsen cell. To control and validate the model, we can also use the sum squared error (SSE). Obviously, the lower number of upscaled cells in the final model, the more the upscaling error. Given that the thresholds or bandwidths in each row of data are optimally selected either along x or y directions, the resulting model will be optimal. The pseudo-code of the kernel bandwidth upscaling algorithm written in MATLAB is described below:

1. Based on the one-dimensional algorithm, the optimal bandwidth vector is selected as  $[\lambda_x, \lambda_y]$
2. Of the four different starting modes, the first pattern is selected.
3. For each target cell, which is indicated as the root, neighbors are identified and placed in a data column.
4. Depending on the direction of movement, the difference between the target cell and the next cell is measured with bandwidth. If x is to move, it will be calculated  $x_i - x_{i-1}$ , if y is to move, it will be calculated  $y_i - y_{i-1}$ .
5. If the difference is less than the bandwidth (see Fig. 2), then the two cells will merge, otherwise, they will remain fine. A is selected as the root cell of the graph. If  $A-B < \lambda_x$ , so A and B can get merge with each other, then B cell is targeted and its neighbors are entered into the data column. Fig. 1 shows how to perform the algorithm.
6. This process is repeated for all cells in order to reach the same extent as the ultimate number of upscaled blocks.
7. The starting pattern changes, and steps 3 through 7 are executed (According to Fig. 3).
8. The upscaling error is calculated, and each lesser is introduced as the final simulator or upscaled model.



Therefore, the upscaled model has 7 blocks, out of which 4 are coarse and 3 are fine,  
 $A' = \text{mean}(A,B,C,F)$   
 $B' = \text{mean}(D,H,G)$   
 $C' = \text{mean}(I,J,K,O)$   
 $D' = \text{mean}(L,P)$

21

Fig. 2. Pseudo-code function.

### 3. Research Data

In order to investigate the upscaling algorithm based on the bandwidth of the kernel function in two dimensions, a 30×30 grid with 900 cells is randomly generated. The porosity distribution in this grid, representing a level of the reservoir, is shown in Fig. 2. The size of fine cell is 1×1. Synthetic data are generated from a normal distribution with a mean of 0.5 and a variance of 0.15, varying from 0.2 to 0.65. After examining the efficiency of the method on synthetic data, the algorithm is tested on the actual data of the SPE-10 model. The SPE-10 model 2 (Christie and Blunt, 2001) is a heterogeneous fine model with dimensions of 1200×2200×170 ft. The model has a simple geometry, with no top structure or faults. The reason for this choice is to provide maximum flexibility in choice of upscaled grids. As a fine-scaled geological model, the model is described on a regular Cartesian grid. The model dimensions are 1200 x 2200 x 170 (ft). The top 70 ft (35 layers) represents the Tarbert formation, and the bottom 100 ft (50 layers) represents Upper Ness. The fine-scale cell size is 20 ft×10 ft ×2 ft. The SPE 10 model 2 size is 60×220×85 (1.122×10<sup>6</sup> cells). In this study, a level of the porosity model with dimensions of 60×220 was examined. The model consisted of part of a Brent sequence, originally generated to be used in the PUNQ project. The top part of the model was a Tarbert formation, a representation of a prograding near shore environment. The lower part (Upper Ness) was fluvial. Fig. 3 (a) shows the porosity for the whole model. The model was based on data from 5 wells, that all of which were vertical. All wells completed throughout the formation. The central well was used for injection and the rest of them were production wells. The position of the wells is shown in Fig. 3 (b) (Christie, 1996).

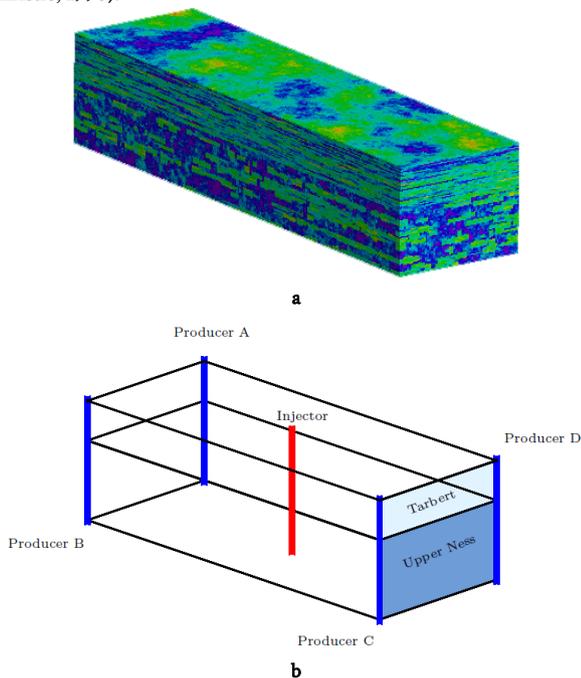


Fig. 3. a) Porosity for the whole model and b) well location (Christie, 1996)

### 4. Results and Discussion

In the two-dimensional upscaling process based on the kernel adaptive bandwidth, for each row of data along the x and y directions, depending on variability, the optimal bandwidth was determined and stored based on the number of upscaled final blocks. The optimal bandwidth vector for the dataset was computed  $[\lambda_x \lambda_y] = [0.076, 0.193]$ . Therefore, in the upscaling algorithm, depending on the position of the cell as well as the direction of the algorithm, the bandwidth was different, showing the multiscale upscaling characteristic of the method.

For upscaling the reservoir property, four modes were considered to start the upscaling process. Each of these states was the upscaling starting point from each of the grid corners. Thus, four different upscaled patterns were obtained. Fig. 4 shows the position of the scenarios and the upscaling starting points. Obviously, the results of the upscaling are different; however, the goal was to determine the optimal upscaling pattern. The selection criterion to determine the optimal pattern was the upscaling error and the number of upscaled cells in the final model. Fig. 5 shows the fine-scale model.

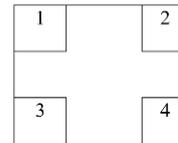


Fig. 4. Upscaling pattern in different states

The upscaling results are shown in Fig. 6. In the first mode, based on the upscaling algorithm, the coarse-grained model was the one shown in the section of Fig. 4. The upscaling error was obtained from the comparison of two coarse-scale and fine-scale models. In this case, the upscaling error was SSE = 0.4183. Moreover, the upscaled model had 236 cells. In the second mode, based on the upscaling algorithm, the coarse-grained model was the one displayed in the section b of Fig. 6. In this case, the upscaling error and the number of upscaled cells in coarse model were 0.4211 and 227, respectively. Similarly, the upscaling error in state or mode 3 was equal to 0.4055 and the number of upscaled cells was 272. In state or mode 4, the number of coarsened cells were 271 and the error resulting from the upscaling is 0.4056. The sections c and d in Fig. 4 show the upscaled models in modes 3 and 4. Parts of e to h also show zoomed parts of the upscaled models in different modes.

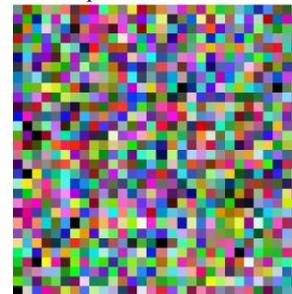
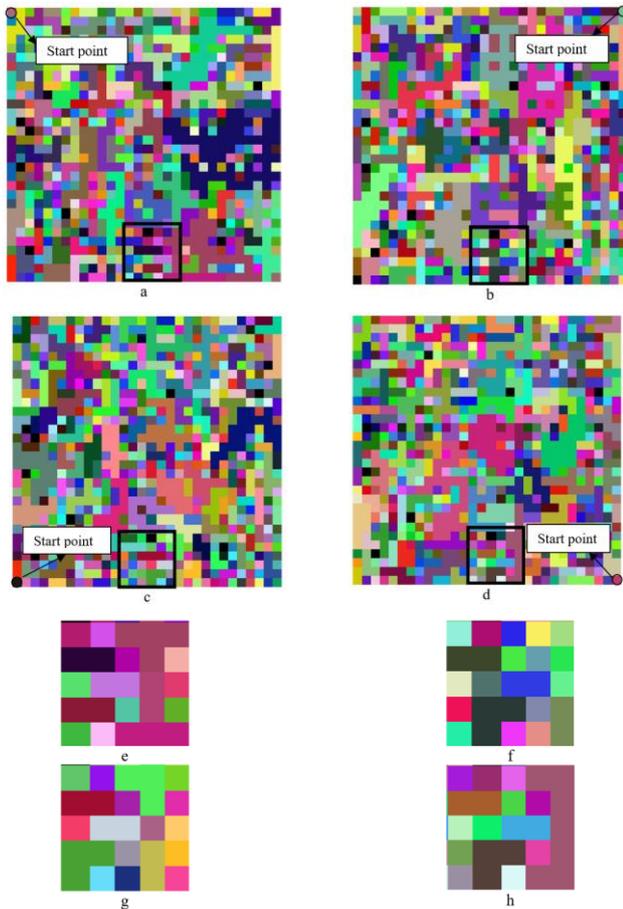


Fig 5. a) Fine-scale model

Based on this upscaling method, four different patterns were obtained. It can be seen that in places where variability was intense, and in fact, heterogeneous, the cells remained fine. In homogeneous regions, one the other hand, the cells were merged together and produced larger cells. In the two modes 3 and 4, the number of cells, as well as the error rate, were approximately equal, except that ordering the cells was different. However, the homogeneous regions in both patterns were almost in good agreement. Patterns 1 and 2 were similar in the upscaling error and cell numbers. As mentioned above, the results of upscaling were strongly dependent on the search starting point and the movement direction. Given that the aim of upscaling in porous media was to search for homogeneous regions and merging the cells in these regions, the direction of movement could be determined based on the variability of data. Suppose choosing the search point in the two directions x and y, the difference of the first 5 data was considered, and the direction for which the average of these differences was less would be selected as a pathway in the upscaling process based on the defined algorithm. For example, in pattern 1, the difference of the first 5 data, which included the starting point of the search, was calculated binary, and then, the average value of the data was obtained. The average value in the direction x was equal to 0.076, and in the direction y was 0.193, which is known as the bandwidth vector (0.076, 0.193). Therefore, because the property variability along the direction x and in the vicinity of the search point was less than that along the direction y, the former was chosen for the movement in the upscaling pattern. Similarly, the movement directions in states 2, 3 and 4, determined to be y, x and x respectively.



**Fig 6.** Upscaled model in a) First state, b) Second state, c) Third state and d) Fourth state, zoomed part of the upscaled model e) First state, f) Second state, g) Third state, and h) Fourth state.

In summary, Table 1 shows the results of upscaling at four different starting points of the search. Validation criteria in the upscaling process in this paper were including the number of upscaled blocks and the SSE. Between the two choices, in similar conditions for the number of upscaled cells, the pattern with less upscaling error would be a more appropriate pattern, and in the same condition for the upscaling error, the model with fewer upscaled cells would be a better pattern.

**Table 1.** Upscaling results in four different patterns.

Pattern	SSE	N. of upscaled blocks	Data variance	Movement direction
1	0.4183	466	0.0064	X
2	0.4211	499	0.0065	Y
3	0.4055	510	0.0065	X
4	0.4056	481	0.0064	X

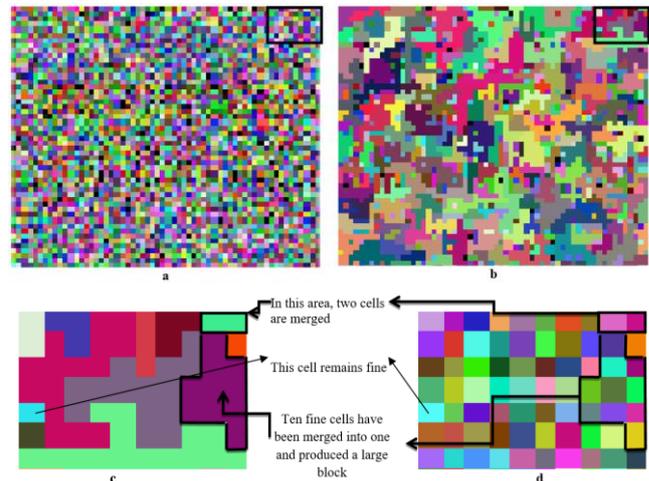
According to the upscaling parameters of four different patterns (Table 1), if the goal is to select an optimal pattern, based on the number of cells and the upscaling error, pattern 1 can be a more appropriate pattern than the other patterns. The movement direction of the upscaling algorithm is understandable in pattern 1. Of the five cells at the beginning of this pattern in the direction x, four cells were upscaled and only one remained fine, but in direction y, all five cells were kept in the fine-scale state. In the case of fluid flow simulation, selecting the appropriate pattern was possible with greater certainty, because the validation criterion could be the number of upscaled cells. Naturally, in this case, the model with the smallest number of coarsened cells is optimal. In the case of real datasets, this process was also carried out. For every starting pattern, an upscaled model was obtained. The optimal bandwidth vector was (0.13, 0.145) for this dataset. Given the upscaling error as well as the number of upscaled blocks, pattern 2 was the best pattern for the real data. A part of the initial model and the results are

shown in Fig 7. The movement direction of the algorithm was x according to the upscaling approach. As seen, the cells were coarsened quite intelligently, non-uniformly and irregularly. Based on the bandwidth vector, the final upscaled model had 3600 upscaled blocks, in which the upscaling error scale was 12.2. Bandwidth can play a crucial role in computing efficiency, as well. If the goal of the simulator model is to depict small heterogeneities in comparison with the fine-scale model, selecting a bandwidth smaller than optimal bandwidth can meet these requirements. If the purpose of the simulation model is to lower the calculation cost and increase the efficiency of computation, choosing a bandwidth larger than the optimal bandwidth will be required as well. When the bandwidth is greater than the optimal bandwidth, the upscaling error decreases, but the number of blocks increases. For such data, if the bandwidth vector was considered (0.5, 0.6), the scale-up model with 890 blocks would have an error of about 46 units. In this case, the upscaling model was constructed with only about 7% of the data. In fact, with such a volume of data, the fine-scale model could be retrieved, but the computational error increased. Conversely, when bandwidth was considered (0.01, 0.02), the coarse-scale model would have 11000 blocks, and the upscaling error would be 0.04. Table 2 shows the upscaled parameters for the real data.

**Table 2.** Upscaled parameters.

Model	SSE	N. of blocks	Data variance	% of data
Fine model	0	13200	0.0083	100
Coarse model with bandwidth (0.13, 0.145)	12.2	3600	0.0065	27
Coarse model with bandwidth (0.01, 0.02)	0.04	11000	0.0081	83
Coarse model with bandwidth (0.5, 0.6)	46	890	0.0045	7

It can be seen that in larger bandwidth, the variance of the fine-scale model varies with the coarse-scale model. This difference signifies the loss of important information. This explains why an optimal bandwidth should be selected so that the model is also optimized. The upscaling error also supports this issue. As the bandwidth increases, the upscaling error also increases.



**Fig 7.** a) Fine-scale model, b) upscaled model, c) zoomed part of fine model and d) zoomed part of upscaled model

As observed above, the cell merging is based on the variability of the cells, and no method still exists to upscale the cells precisely. The data of the SPE model were also tested with the normalization method. In the first step, the fine-scale model would become a coarse-scale model with 3300 blocks, in which case the upscaling error of the model would be 174. Compared to the similar condition of upscaling based on the bandwidth of the kernel function, the error of the normalization method was about 15 times higher. Obviously, the kernel bandwidth-based upscaling error was far less than the normalization method, because the upscaling pattern was based on variability. Also, in the proposed

method, based on the upscaling error, the simulation model could be determined by determining the appropriate bandwidth.

The SPE-10 had a simple geometry, with no top structure or faults. The mean and variance of porosity were 0.16 and  $3.63 \times 10^{-6}$  respectively. The fine-scale model had 13200 cells. Porosity variability created a variance of  $3.63 \times 10^{-6}$  in the original grid. The goal of upscaling is to minimize the loss of the main information that is resulted from the approach used for reducing the number of cells. Data variance can be a measure of data information loss. In the coarse-scale model, with a certain bandwidth vector, the number of upscaled blocks dropped to 3600, which was 27% of the total fine cells. In fact, data were reduced to 73% by upscaling based on the kernel bandwidth method. When this volume of data is reduced, it is expected that much information will be removed from the original model. The variance of the coarse-scale model can now be calculated. The variance of the upscaled model was  $3.01 \times 10^{-6}$ . Therefore, it is clear that the variance of the original data decreased by 17%. This indicates that although 73% data were reduced, the data variance reduced only 17%.

## 5. Conclusion

In this paper, two-dimensional upscaling of the reservoir data was performed using an adaptive bandwidth algorithm in the kernel function. For this purpose, four different models were obtained based on the starting point of the upscaling algorithm. The movement direction in the algorithm was also defined based on the variability of the data. The optimum bandwidth was calculated based on the number of cells required in the upscaled model. If the number of cells in the upscaled model is lower, the upscaling error in converting the fine-scale model into the coarse-scale model will be greater. Among the four upscaling patterns examined in this paper, the third and fourth patterns were similar in terms of upscaling parameters. In terms of the number of cells and the upscaling error, the first and second patterns were almost identical. According to these results, based on the number of cells and the upscaling error for the synthetic data, the first pattern was optimal. In that case, the number of coarse blocks was 236, and the error rate due to the upscaling was 0.4183. The process was also performed on a real dataset. With the same process, the second state pattern was selected for the real data, based on the bandwidth vector (0.13, 0.145), the number of upscaled blocks was 3600, and the upscaling error was 12.2. The results of the proposed method were compared with the normalization method. On this basis, the upscaling error from the normalization procedure was 15 times of the kernel bandwidth method. Upscaling based on the bandwidth of the kernel function is an incoherent and intelligent method that performs the coarsening of the cells based on the property variability.

## REFERENCES

- [1] Christie, M. A., (1996). Upscaling for Reservoir Simulation. *Journal of Petroleum Technology*, 48(11), 1-4. <https://doi.org/10.2118/37324-JPT>.
- [2] Christie, M. A., & Blunt, M. J. (2001). Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques. *Journal of Petroleum Technology*, 4(4), 1-10. <https://doi.org/10.2118/72469-PA>.
- [3] Chen, T., Clauser, C., Marquart, G., Willbrand, K. and Mottaghy, D. (2015). A new upscaling method for fractured porous media. *Journal of Advances in Water Resources*, 80, 60-68. <https://doi.org/10.1016/j.advwatres.2015.03.009>.
- [4] Farmer CL., (2000). Upscaling: a review. *International Journal Numerical Methods Fluids*, 40, 63 – 78. <https://doi.org/10.1002/flid.267>.
- [5] Dadvar, M., & Sahimi, M., (2007). The effective diffusivities in porous media with and without nonlinear reactions. *Chemical engineering science*, 62, 1466-1476. <https://doi.org/10.1016/j.ces.2006.12.002>
- [6] Hochstetler, D. L., & Kitanidis, P. K., (2013). The behavior of effective rate constants for bimolecular reactions in an asymptotic transport regime. *Journal of contaminant hydrology*, 144, 88-98. <https://doi.org/10.1016/j.jconhyd.2012.10.002>.
- [7] Pereira, J. M. C., Navalho, J. E. P., Amador, A. C. G., & Pereira, J. C. F., (2014). Multi-scale modeling of diffusion and reaction-diffusion phenomena in catalytic porous layers: comparison with the 1D approach. *Chemical Engineering Science*, 117, 364-375. <https://doi.org/10.1016/j.ces.2014.06.028>.
- [8] Ratnakar, R. R., Bhattacharya, M., & Balakotaiah, V., (2012). Reduced order models for describing dispersion and reaction in monoliths. *Chemical Engineering Science*, 83, 77-92. <https://doi.org/10.1016/j.ces.2011.09.056>.
- [9] Chen, T., Clauser, C., Marquart, G., Willbrand, K. and Hiller, T., (2018). Upscaling permeability for three-dimensional fractured porous rocks with the multiple boundary method, *Hydrogeology Journal*, 1-14. <https://doi.org/10.1007/s10040-018-1744-z>.
- [10] Ding, D. Y., (2004). Near-Well Upscaling for Reservoir Simulations. *Oil & Gas Science and Technology*, 59(2), 157-165.
- [11] Gholinezhad, S., Jamshidi, S. and Hajizadeh, A., (2015). Quad-Tree decomposition method for areal upscaling of heterogeneous reservoirs: Application to arbitrary shaped reservoirs. *Fuel*, 139, 659-670. <https://doi.org/10.1016/j.fuel.2014.09.039>.
- [12] Moslehi, M., Felipe P.J., de Barros, Ebrahimi, F. and Sahimi, M., (2016). Upscaling of solute transport in disordered porous media by wavelet transformations, *Advances in Water Resources*, 96, 180-189. <https://doi.org/10.1016/j.advwatres.2016.07.013>.
- [13] King, P.R. (1989). The Use of Renormalization for Calculating Effective Permeability, *Transport in Porous Media*, 4, 37-58.
- [14] Silverman, B.W. (1986). Density Estimation for Statistics and Data Analysis. Published in Monographs on Statistics and Applied Probability, London: *Chapman and Hall*, 176 P.
- [15] Scott, D. W., 1992. Multivariate Density Estimation: Theory, Practice, and Visualization. New York: John Wiley & Sons Inc., pp. 384.
- [16] Raykar, V.C., Duraiswami, R., Zhao, L.H., (2010). Fast computation of kernel estimators. *Journal of Computational and Graphical Statistics*, 19 (1), 205–220.
- [17] Hardle, W. K. K., Muller, M. and Sperlich., (2004). Nonparametric and semiparametric models, *Springer Series in Statistics*, 277 P.
- [18] Delling, D., Sanders, P., Schultes, D. and Wagner, D., (2009). Algorithmic of Large and Complex Networks: Design, Analysis, and Simulation. *Springer-Verlag Berlin Heidelberg* 401 P.
- [19] Even, S., (2011), Graph Algorithms (2nd Ed.). *Cambridge University Press*, 48 P.