SPE-187797-MS

Hybrid Approach to Reservoir Modeling Based on Modern CPU and GPU Computational Platforms

Alexey Telishev, Kirill Bogachev, Vasilii Shelkov, Dmitry Eydinov, and Hau Tran, Rock Flow Dynamics

This paper was prepared for presentation at the SPE Russian Petroleum Technology Conference held in Moscow, Russia, 16-18 October 2017.

Abstract

For modern day reservoir simulators, it is essential to provide realistic physical description of reservoirs, fluids and hydrocarbon extraction technology and guarantee excellent performance and parallel scalability.

In the past, the advances in simulation performance were largely limited by memory throughput of CPU based computer systems. Recently, new generation of graphical processing units (GPU) became available for general purpose computing with the support of double precision floating point operations, necessary for dynamic reservoir simulations. The graphical cards currently available on the market have thousands of computational cores that can be efficiently utilized for simulations.

In this paper, for the first time we present results of running full physics reservoir simulator on CPU +GPU platform and discuss implications of this modern technology on the existing reservoir simulation workflows. We discuss challenges and developed solutions for running reservoir simulations using modern CPU+GPU hardware architecture and propose a methodology to distribute the workload between various parts efficiently. The approach is tested on several data sets on various computational platforms, such as personal computers and clusters with and without GPU's involved.

The technology proposed in this paper demonstrates multifold speed up for models with substantial number of active grid blocks. The speed up due to GPU utilization can in some cases reach as high as 3-4 times compared to the traditional GPU-based approach. Considering the recent progress in the GPU development, this factor is expected to grow in the near future, and the hybrid CPU+GPU based approach allows to utilize the exciting potential of the hardware evolution. The results, advances and potential bottlenecks combined with detailed analysis of the performance and the ‘value for money’ of the modern hardware solutions are discussed.

Introduction

In 1965 Gordon Moore, one of the Intel co-founders, predicted that the number of transistors in a dense integrated circuit doubles approximately every two years, [1]. This is known as Moore's law and the statement has proved to be true over the last 50 years. Recently the growth in computational power has been mostly achieved by growth in the number of CPU cores integrated within one shared memory chip. During the last decade, the number of cores available on the CPU's has grown from 1-2 to 20+, and keeps

growing every year [2]. Supercomputers has become available for everyone in shape of laptops or desktop workstations with large amount of memory and high-performance simulation capabilities.

There has also been a significant improvement in the cluster computing options. Only 10-15 years ago high-performance clusters were very expensive and required special infrastructure, costly cooling systems, significant efforts to support, etc. Only major companies that ran massive simulations on every day basis could afford investing in them. These days the cluster have also become quite economical and easy-to-use machines. Quite powerful machines can be installed in a simple room with air-condition and local area network connection. The power consumption was reduced to the level of regular tools, such as kettles or irons we use our everyday life.

It has been demonstrated that the reservoir simulation time can be efficiently scaled on the modern CPU-based workstations and clusters if the simulation software is implemented properly to work with the modern hardware architecture, [3,4]. Technically, the simulation time is not a principal issue anymore, and can be reduced to any time frames required by adding extra computational power. There are many options that help to optimize value for money: laptops, workstations or high-performance clusters installed in-house.

In case the workload is non-uniform and requires only occasional massive simulations, one can consider simulation resources available on the cloud with pay-per-use model, [5]. These platforms become more popular these days as they help to reduce the costs significantly, especially in the current economic environment. All it takes to run the model from the user side is to upload the input data via encrypted channels (like the secured banking online systems), chose the number of CPU's to be rented and press run. The modern remote visualization tools allow to analyze the simulation results remotely at runtime from the user terminal located anywhere. This makes the reservoir modeling solutions much more scalable in terms of time, money, human and computational resource. Reservoir engineers from large and small companies can have access to nearly unlimited simulation resources and reduce the simulation time to minimum when it is required.

There is another promising hardware technology that grows even faster than the CPU's – graphical processing units (GPU). Recently, new generation of GPU became available for general purpose computing with the support of double precision floating-point operations, necessary for dynamic reservoir simulations. The graphics cards currently available on the market have thousands of computational cores that can be efficiently utilized for high-performance simulations.

In addition to the number of cores, the latest GPU's also have significantly greater memory bandwidth, which is equally important for efficient parallel simulations as it is effectively the speed of communication between the cores. In many cases, e.g. when the model grid is large or the reservoir properties are strongly heterogeneous, the time spent on the linear solver iterations become dominant. It can take more that 90% of the total time spend on the simulation, and this is mostly due to limited memory capabilities to provide the cores with data to handle.

The progress of the GPU's in this component has been extremely rapid over the last years. Fig. 1 shows the progress in the memory bandwidth component of the Nvidia GPU's, [6]. As we can see from the picture, the gap between GPU and CPU has become very significant in 2015-2016, and can be expected to grow even greater in 2017. The bandwidth of the top GPU models has grown to 700+GB/s, while the CPU remain almost one order of magnitude lower.

This achievement should not be ignored when choosing the computational platform for the reservoir simulations.

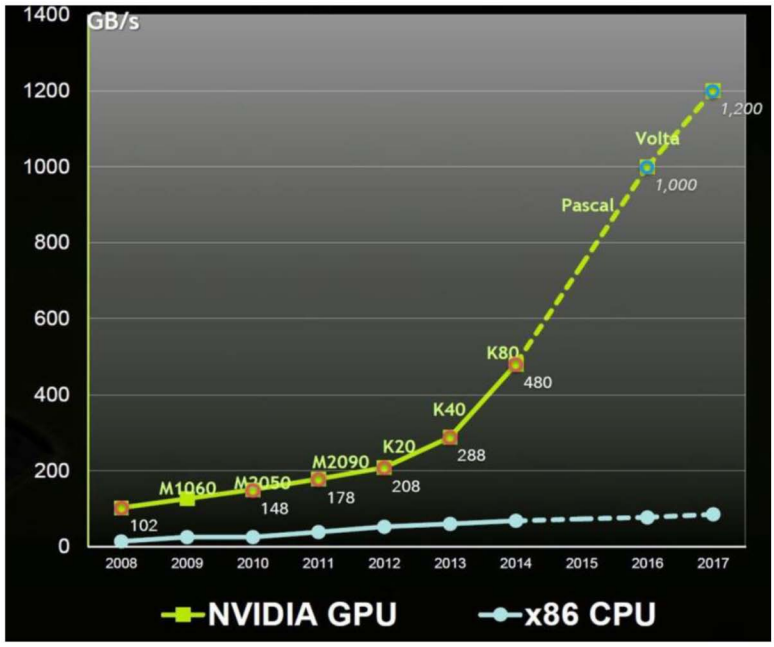

Figure 1—Memory bandwidth development

Hybrid CPU+GPU method

There have been a few recent developments applying GPU for the reservoir simulations, [7,8,9]. There various approaches to the simulation software implementations where the all calculations are delegated to the GPU cores, and some suggestions for balancing the computational efforts between the GPU and CPU.

From our point of view, the optimal solution is to share the workload between the two parts by giving the most computationally intensive parts to the GPU, and keep everything else on the CPU side, still in parallel mode between the CPU cores. One of the reasons for that is relatively small amount of memory available on the GPU side. Even the most expensive models currently limited to 24Gb, while the common ones are mostly in range of 8-12Gb. As we know from experience, typical 3 phase black-oil models require in average 3Kb of memory per active grid block of the model (2Kb per active block for 2 phase problems). Compositional cases will require as much as *(number of hydrocarbon components)* × *(active grid blocks)* Kb. This means that if we, for instance, consider a compositional simulation model with 10 components against 8Gb of memory, the highest number of grid blocks we can afford is about 800 thousand. This may not be sufficient for correct simulation results in many cases. E.g. when chemical or thermal EOR methods are applied model resolution must be sufficiently fine to provide correct description of the physical process in the reservoir.

In this work, we have tried very simple approach for the workload distribution. All parts related to the linear solver block, which is the most computationally intensive part of the solution and may take up to 90% of the overall modeling time, are given to the GPU, while the rest remains on the CPU. This approach provides an option to speed-up the calculations significantly by solving the linear system on a faster GPU card. In the meantime, it allows to run considerably larger cases on a typical workstation or laptop compared to the pure GPU software approach. The linear system does not require so much memory as the whole simulation problem. The CPU platform takes care of the direct calculations to form the matrix which is then sent to the GPU to get the results of the linear system of equations. Then the CPU generates and the next matrix and sends it to the GPU to solve.

This approach seems to be extremely efficient for large and mid-size problems (say, 1 million active blocks or higher) and provides many-fold acceleration compared to the conventional CPU-based simulators. In many cases we have observed 2-3 times acceleration, some have demonstrated 6x speed-up.

Upgrading a desktop to make use of the technology is just matter of purchasing a GPU for USD 500-600 and inserting it to the PCI slot. Moreover, a simple gaming laptop with a reasonable GPU for about USD 2000 in total may in many cases provide performance similar or higher than the powerful desktops commonly used for the simulations these days.

Results

In this section, we present some examples of the Hybrid CPU+GPU method performance based on synthetic and real-field models.

Case 1

As the first example, we consider model 2 from the 'Tenth SPE Comparative Solution Project' often referred to as 'SPE 10'. This model has a relatively fine grid with a simple geometry, with no top structure or faults. At the fine geological model scale, the model is described on a regular Cartesian grid. The model dimensions are 1,200×2,200×170 ft. The top 70 ft. (35 layers) and the bottom 100 ft. (50 layers). The fine-scale cell size is 20×10×2 ft. The fine-scale model has 60×220×85 cells (1 122 000 cells), [10].

The top part of the model is a Tarbert formation and is a representation of a prograding near-shore environment. The lower part (Upper Ness) is fluvial. Both formations have huge permeability contrasts in the cells: 8-12 orders of magnitude. The model used here has a kV/kH of 0.3 in the channels and a kV/kH of 10^{-3} in the background. The porosity field is correlated with the permeability.

The extreme contrasts in the permeability values make the problem very challenging for the reservoir simulators. Due to order of magnitude difference in the permeability values the matrix becomes very poor conditioned, that may take lots of linear iterations to solve. In our case, the linear solver calculations were approximately 90% of the total computation effort.

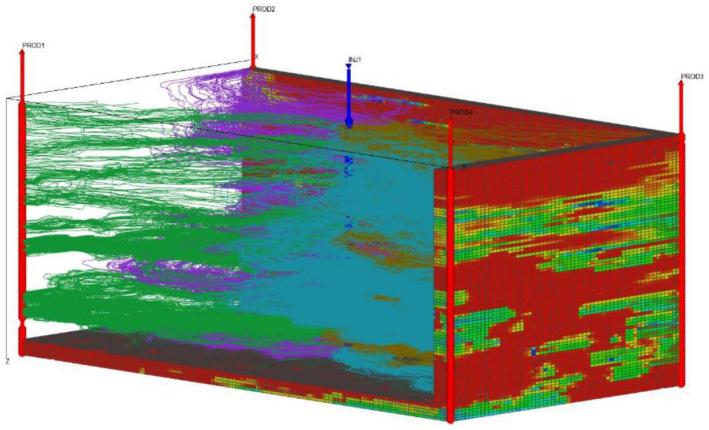

Figure 2—SPE-10 model.

In this study, we compared the simulation times with 3 various hardware configurations:

- Laptop with Quad Core i7 6700K
- The same laptop with GPU Nvidia GTX 1080
- Desktop with dual CPU 20-core Xeon E5 2698 v4 (40 cores in total)

The results of simulation performance tests are shown on Fig. 3.

As we can see from the figure, the full simulation time of the model on Quad Core laptop is 1 hour and 4 minutes. By enabling the GPU in the calculations, the simulation time is reduced to 11 minutes and 20 seconds. It also interesting to see that in this case the simulation time is shorter than the one obtained with a dual-CPU powerful workstations with 40 CPU cores in total, which delivered the results in 23 minutes.

In the meantime, if we compare the price of the hardware, the laptop with GPU will be about 6-7 times more economical.

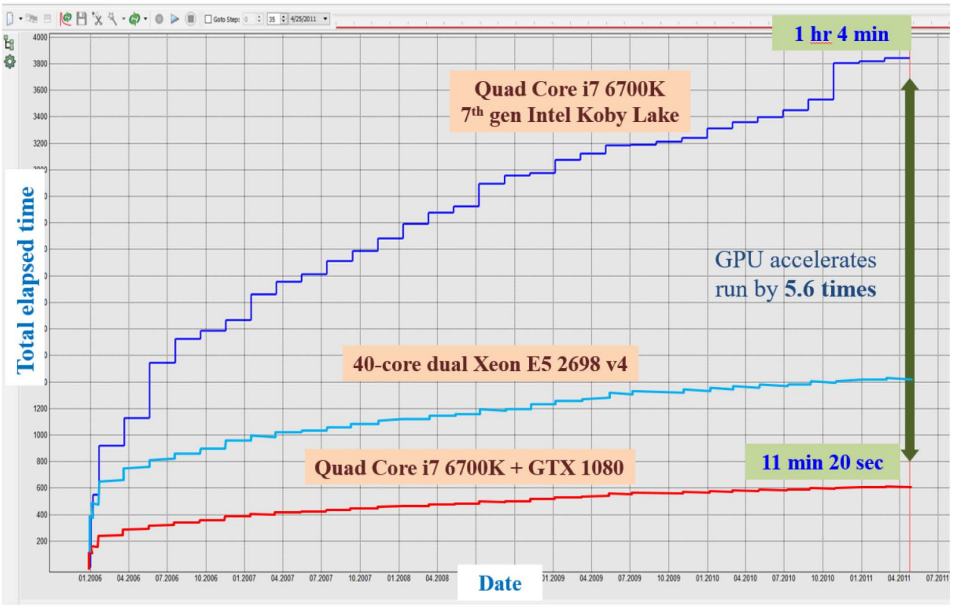

Figure 3—SPE-10 black oil model. Simulation time on several types of hardware.

We have also run a compositional version of the SPE-10 model. The model has the same grid and reservoir properties. The fluids composition is 8 hydrocarbon components plus water. We compared simulation performance of a dual-CPU desktop with 44 cores and NVidia P100 GPU. Compared to the black oil case, the fraction of time spend on solver calculation is smaller. In this case, considerable part of the total simulation time is spent on flash calculations.

The simulation time on a dual-CPU platform was equal to 10 hours 5 minutes, while the total time with NVidia P100 GPU was 4 hours 27 minutes. The speed-up factor in case equals to 2.3 times. This is smaller opposed to the black oil case. The reason is that in this work we do not do flash calculations on the GPU, and this part remains purely on the CPU side. Improvement in this directions with flash calculations on the CPU+GPU hybrid system should improve the speed-up significantly. This is going to be implemented, tested and presented in the future publications.

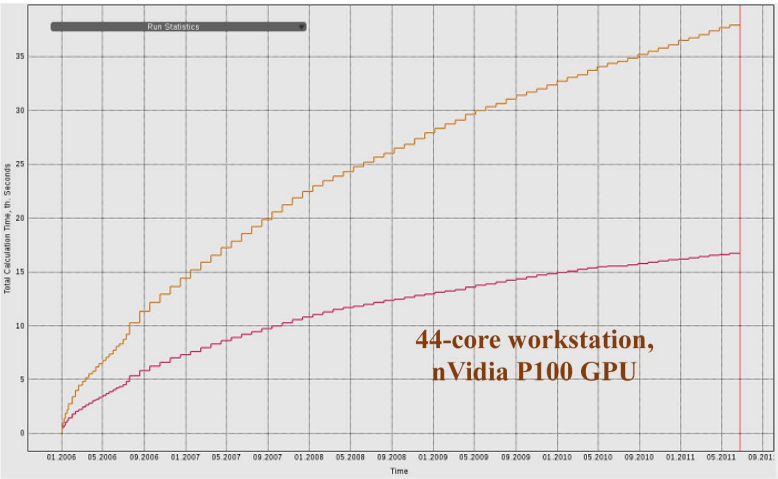

Figure 4—SPE-10 compositional model with 9 components. Simulation time on dual-CPU desktop with 44 cores (orange line) + NVidia P100 GPU (red line).

Case 2

In this case, we consider a much more complex model of a fractured reservoir. The model is a dual porosity with 200x800x150 cells. The simulation model contains 24 million active grid blocks. 6 horizontal producing wells with multistage fractures are drilled in the field. Tartan grids with local grid refinement are introduced to provide more accurate description of the fluid flow in fractured wells. The fracture modeling is based on the virtual perforations approach. The idea is that additional well completions are introduced in all grid blocks crossed by the fractures, and the connection factor for these completions are calculated based on the cross-section area between the fracture and the grid block. This approach provides accurate parameterization of the fractured well physics on the full-field model scale, [11].

Like Case 1, the model is very challenging for reservoir simulations due to large number of grid blocks, heterogeneity and a relatively substantial number of well connections in the LGR blocks. In this case, the model grid is much larger, and we must consider more powerful hardware to run. We compared simulation performance of a dual-CPU desktop with 44 cores and NVidia P100 GPU and a high-performance CPU cluster with 4 nodes with 20 cores each (80 cores in total). Both machines have sufficient amount of operational memory to run the case. The choice of GPU also comes from the memory demand. P100 has 16Gb of memory, while GTX 1080 used in the previous case has only 8Gb, which will not be sufficient to handle the linear equation matrix for a model with 20 million active grid blocks.

Fig. 5 shows the simulation time comparison for the desktop and cluster. The desktop could complete the simulation in 4 hours 16 minutes, while the 4-node cluster made it in 3 hours 48 minutes. This is quite close. However, if we consider the costs of the hardware in these two cases, the price will be significantly different. The desktop will be about 4-5 times cheaper and easier to manage in the IT infrastructure.

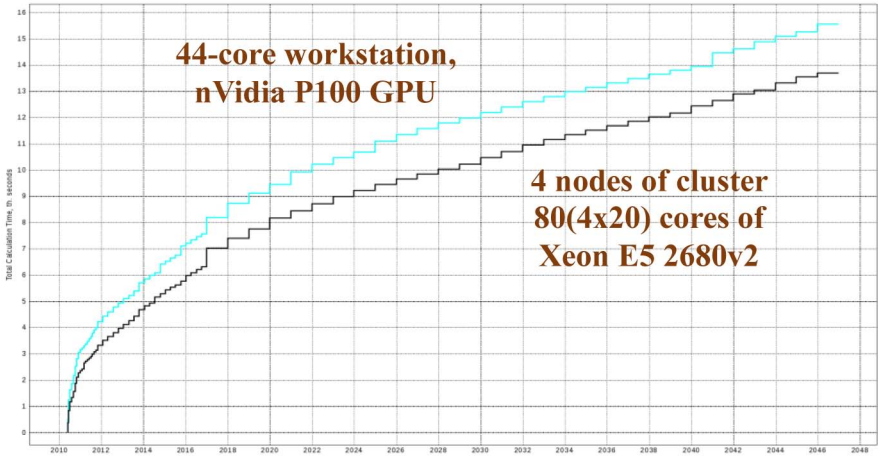

Figure 5—Case 2 simulation time.

GPU options

In this section, we would like to briefly discuss the GPU models available on the market now. Table 1 shows the variety of the most common model from the Nvidia Pascal generation that was made available recently. We have also included two CPU models to emphasize the difference between the two conceptually different approaches.

As we can see from the table, the GPU's have much greater number of cores, however, with lower speed. The second key difference is memory. CPU can have a lot more aside of it, but the speed and bandwidth is significantly lower. The first is basically related to the size of the model that can be run, while the latter is more about the simulation speed and parallel performance. As highlighted above, these parts can be efficiently used together and share the workload in the simulations. The hybrid approach proposed in this paper appears to be a reasonable trade-off for this at the moment. With further development of the CPU

and GPU chips, we will see if this approach remains the most efficient. It is quite likely that innovative ideas will be developed.

Table 1—Comparison of the key characteristics of the GPU and CPU models. Prices are rounded average as at June 2017.

Device	Cores	Clock	Memory	Size	Clock	Band-width	Price
2x Xeon E5 2630 v3	16	2400MHz	DDR4	768GB	1866MHz	56GB/s	$1340
2x Xeon E5 2680 v4	28	2600MHz	DDR4	1.54TB	2400MHz	59GB/s	$3500
2x Xeon Platinum 8170 (Skylake)	52	2100MHz	DDR4	2.31TB	2666MHz	119GB/s	N/A
Xeon Phi 7210	64	1300MHz	HBM+ DDR4	16GB + 384GB	1300MHz 2133MHz	450GB/s 102GB/s	$2500
GTX-1080	2560	1607MHz	GDDR5X	8GB	10000MHz	320GB/s	$500
GTX-1080 TI	3584	1582MHz	GDDR5X	11GB	11000MHz	484GB/s	$700
Tesla Quadro P6000	3840	1560MHz	GDDR5X	24GB	9000MHz	432GB/s	$5500
Tesla P100	3584	1480MHz	HBM2	16GB	1400MHz	720GB/s	$8900
Tesla V100	5120	1455MHz	HBM2	16GB	1750MHz	900GB/s	N/A

As for the specific GPU options, available on the market currently, we can make some observations. There is significant difference in price between the Tesla and GTX series. This is partly related to the amount of memory. Another reason is the positioning – the GTX cards are designed for gaming and supposed to be used at home, while the Teslas are designed for enterprise applications. There are quite a few things that are different between those two related to server compatibility, maintenance, etc. But practically, if GTX has sufficient to run the model, this is the most practical choice for PC's from the price/performance point of view. The may still see difference in the simulation performance with the Tesla compared to the GTX, but it will incomparable to the price gap. If the model is large and cannot fit in 8-11Gb available with GTX, Tesla is the right option. As illustrated by Case 2 in this paper, it is a very powerful tool that is still available for a very reasonable price.

Conclusions and way forward

It has been demonstrated that the hybrid CPU+GPU approach with proper balancing of the computational workload between these parts can provide fast and scalable solution for the reservoir simulations. The high memory bandwidth available on the GPU's can be efficiently utilized for the most computationally intensive parts of the simulations. In particular, the linear solver.

Based on the cases presented in this paper, we can see that the simulation time can be reduced by many-fold compared to the conventional CPU-based machines utilized for calculations. Case 2 illustrates that the desktops equipped with powerful GPU's may become equally efficient to more powerful cluster machines.

For the real projects, application of the GPU can significantly reduce the costs of hardware and software required for the project. The choice of hardware options should be made carefully with account for specific model features. It remains model dependent and should consider the grid size, amount of memory, overall computational power, and of course, the time frames and hardware price. The variety of the options, especially with reasonable price, has become wider with GPU introduction.

Next generations of the GPU's such as Volta expected in the end of 2017 should have even higher memory bandwidth compared to the current Pascal generation. That will most likely allow to improve the simulation time even further.

Another direction for improvement is clusters with CPU+GPU nodes. This option remains beyond this paper and requires further investigation. The early work that we have done in this direction demonstrates that on one hand the CPU-GPU cluster should be able to solve the memory limitation problems for large models, which will have the linear system matrix larger than a single GPU can host. On the other hand, the communication with between various parts in such a multi-layered computational system becomes an issue and should be implemented properly to provide the best performance.

References

1. Moore, Gordon E. (1965). "Cramming more components onto integrated circuits" (PDF). *Electronics Magazine*. p. 4. Retrieved 2006-11-11.
2. https://ark.intel.com/#@PanelLabel595
3. Dzyuba, V. I., Bogachev, K. Y., Bogaty, A. S., Lyapin, A. R., Mirgasimov, A. R., & Semenko, A. E. (2012). Advances in Modeling of Giant Reservoirs. *Society of Petroleum Engineers*. doi: 10.2118/163090-MS
4. Tolstolytkin, D. V., Borovkov, E. V., Rzaev, I. A., & Bogachev, K. Y. (2014, October 14). Dynamic Modeling of Samotlor Field Using High Resolution Model Grids. *Society of Petroleum Engineers*. doi:10.2118/171225-MS
5. https://aws.amazon.com/
6. https://www.enterprisetech.com/2014/11/17/nvidia-doubles-tesla-gpu-accelerators/
7. Yu, S., Liu, H., Chen, Z. J., Hsieh, B., & Shao, L. (2012, January 1). GPU-based Parallel Reservoir Simulation for Large-scale Simulation Problems. *Society of Petroleum Engineers*. doi:10.2118/152271-MS
8. Khait, M., & Voskov, D. (2017, February 20). GPU-Offloaded General Purpose Simulator for Multiphase Flow in Porous Media. *Society of Petroleum Engineers*. doi:10.2118/182663-MS
9. Manea, A. M., & Tchelepi, H. A. (2017, February 20). A Massively Parallel Semicoarsening Multigrid Linear Solver on Multi-Core and Multi-GPU Architectures. *Society of Petroleum Engineers*. doi:10.2118/182718-MS
10. Christie, M. A., & Blunt, M. J. (2001, August 1). Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques. *Society of Petroleum Engineers*. doi:10.2118/72469-PA
11. Bogachev, K., Shelkov, V., Zhabitskiy, Y., Eydinov, D., & Robinson, T. (2011, January 1). A New Approach to Numerical Simulation of Fluid Flow in Fractured Shale Gas Reservoirs. *Society of Petroleum Engineers*. doi:10.2118/147021-MS