

Available online at [www.sciencedirect.com](http://www.sciencedirect.com)

Procedia Computer Science 4 (2011) 1745–1750

---

---

**Procedia**  
Computer Science

---

---

International Conference on Computational Science, ICCS 2011

## Intelligent fracture creation for shale gas development

Craig C. Douglas<sup>a</sup>, Guan Qin<sup>a</sup>, Nathan Collier<sup>b</sup>, and Bin Gong<sup>c</sup><sup>a</sup>University of Wyoming School of Energy Resources, Laramie, WY 82071, USA<sup>b</sup>King Abdullah University of Science and Technology, Thuwal, Saudi Arabia<sup>c</sup>Peking University Department of Energy and Resource Engineering, Beijing 100871, China

---

### Abstract

Shale gas represents a major fraction of the proven reserves of natural gas in the United States and a collection of other countries. Higher gas prices and the need for cleaner fuels provides motivation for commercializing shale gas deposits even though the cost is substantially higher than traditional gas deposits. Recent advances in horizontal drilling and multistage hydraulic fracturing, which dramatically lower costs of developing shale gas fields, are key to renewed interest in shale gas deposits.

Hydraulically induced fractures are quite complex in shale gas reservoirs. Massive, multistage, multiple cluster treatments lead to fractures that interact with existing fractures (whether natural or induced earlier). A dynamic approach to the fracturing process so that the resulting network of reservoirs is known during the drilling and fracturing process is economically enticing. The process needs to be automatic and done in faster than real-time in order to be useful to the drilling crews.

Keywords: sensor-model feedback, dynamic data-driven application system, DDDAS, multiscale methods, reservoir simulation

---

### 1. Introduction

Shale gas reservoirs appear to be considerably more complicated to model and simulate than conventional oil and gas reservoirs. The added complexity is due in part to shale gas appearing in a large number of small fractures that are not naturally interconnected and are difficult to recover gas from. Instead, natural and hydraulically induced fractures are created to connect shale gas reservoirs to make recovery of shale gas economically viable [1].

A Discrete fracture model is used to represent each fracture individually and explicitly, which requires unstructured gridding of the fracture-matrix system using 3D (Delaunay) triangulation and transmissibility evaluation between each pair of adjacent cells. The near-well effects are modeled in detail by refining the unstructured 3D grid to the point where we fully resolve stimulated fractures. Very large models require an upscaling procedure, such as a multiple subregion procedure to allow fast computations.

Microseismic fracture mapping sensors are providing a level of detail never before seen in seismic imaging, which means the complexities in the fracturing is much more visible and can be leveraged to provide greater accuracy. The sensors are in the well and left there. Information can be garnered later as the fracture network is developed through multiple clusters of fracturing.

Complex geometries are common in shale gas reservoir simulation. Horizontal wells and multistage hydraulic

fracturing provide difficulties leading to only single well simulations or simple decline curve analysis. More accurate reservoir simulation is key to better field management.

Advanced simulation techniques are critical to reservoir management and sources of information to the companies that develop and operate the fields. In the past, much of the simulation development has been aimed at a working field, not at creating a working field. In this paper we discuss how a dynamic data-driven application system (DDDAS) approach can significantly enhance the creation of a fracture shale gas field.

In Sec. 2 we discuss model formulations. In Sec. 3 we discuss a DDDAS formulation. In Sec. 4, we discuss future plans.

## 2. Model formulations

Fig. 1 gives schematic representations of the patterns that are typically observed in hydraulically induced fractures in real gas shale fields [2]. Microseismic imaging techniques allow far more accurate mapping of the fractures than was available until quite recently.

Discrete fracture modeling (DFM) considers each fracture as its own geometric entity. Until recently, DFM was difficult to use since there was not enough information available to describe a fractured reservoir accurately. In addition, the computational cost was prohibitive. With the ready access of relatively inexpensive fast, parallel computers, computational cost is no longer a barrier.

DFM has been studied for finite element [3-5], finite difference [6], and finite volume [7] methods since the late 1970's extensively (see more references in [8]). Both Cartesian and unstructured grids have been studied. The unstructured grid case is the one of most interest to us since it is more realistic and accurate. A simplified finite volume model [9] applies a connection list to represent unstructured grids in two and three dimensions with multiphase flow and is used in our studies. Combined with local grid refinement in both fractures and matrix, we have enough flexibility to handle a variety of fracture distribution and connectivity type reservoirs.

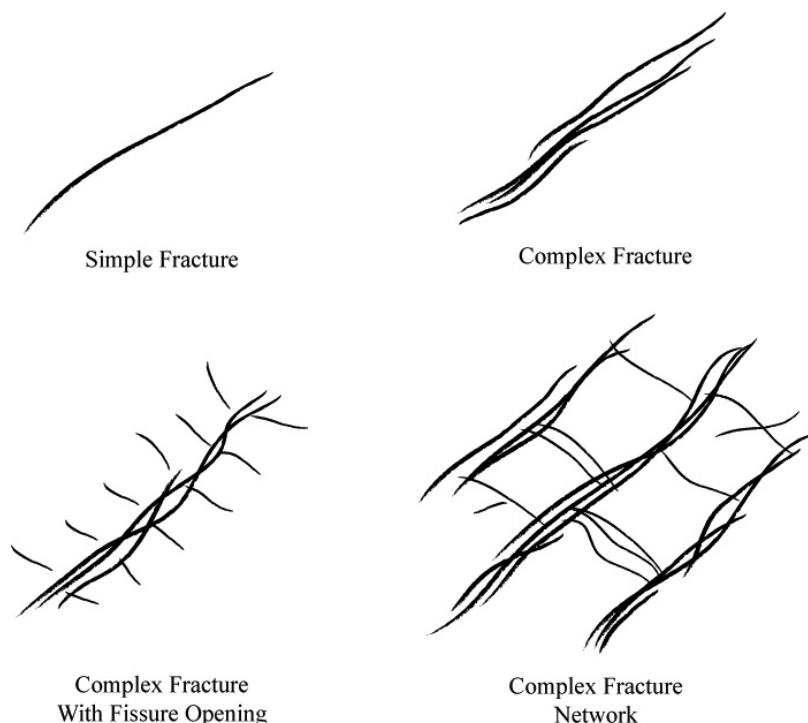


Fig. 1: Different crack formulations

The multiple subregion (MSR) method is used to upscale the problems to construct computationally attractive coarse grid problems to solve instead of fine grid ones. A direct mapping between discrete fracture characterizations and a dual porosity representation can be found in [10]. By using local subregions, the upscaled model is in a dual-porosity form. Matrix rock and fractures can then exchange fluid locally in parallel with large scale flow through the fracture network. Hence, a connection list including all internal and interblock transmissibilities can be created that is suitable to be input directly into a reservoir simulator.

There is a three step process in the upscaling procedure. First, the coarse scale equations may be different than the fine scale ones, which means that the upscaled parameters have to be computed explicitly. Second, a local or global domain must be chosen for the upscaled parameters. Third, the boundary conditions have to be determined and post processing is applied when computing the upscaled parameters.

The overall procedure can be viewed as follows. Starting with a general discrete fracture model that we wish to upscale (i.e., model via a continuum description on a coarse scale), the first step is to form the coarse grid. Ideally, this would be done in such a manner that the matrix and fractures contained within each grid block form a closed system, i.e., the fractures in (and bordering) the block drain only the matrix rock contained within the block. In this case, there would be no flow from the matrix within this block to any other block and the model would conform to the assumptions used in the homogenization procedure. Given a general fracture characterization, it may be possible to generate a grid that approximately satisfies this condition, but the grid would be unstructured with very general shaped cells, which would in turn lead to a number of numerical discretization issues. Rather than proceeding in this way, we impose a structured Cartesian grid on the system and then, in the determination of the matrix-fracture and matrix-matrix interactions, specify boundary conditions that restrict these flows to occur only within the target grid block. Large scale flow occurs from grid block to grid block and is modeled via an upscaled transmissibility that captures the effective fracture permeability.

### 3. Sensor-model interaction

Fig. 2 provides a workflow for the DDDAS. We have identified six key areas for DDDAS computing and visualization plus an initialization step. In Fig. 2(A), the drilling begins based on initial seismic imaging and simulation results. This procedure is outlined in [8] and a synthetic shale gas reservoir system is demonstrated numerically.

In Fig. 2(B), as drilling continues, sensors provide updated fracture characterization data. The flooding process causes more fractures to open and the sensors on the drilling device are used to determine new directions to drill. However, this data is independently interpreted. The key point of this paper is that this additional local data can be used to improve the simulation results and better guide drilling while the fracture network is being created.

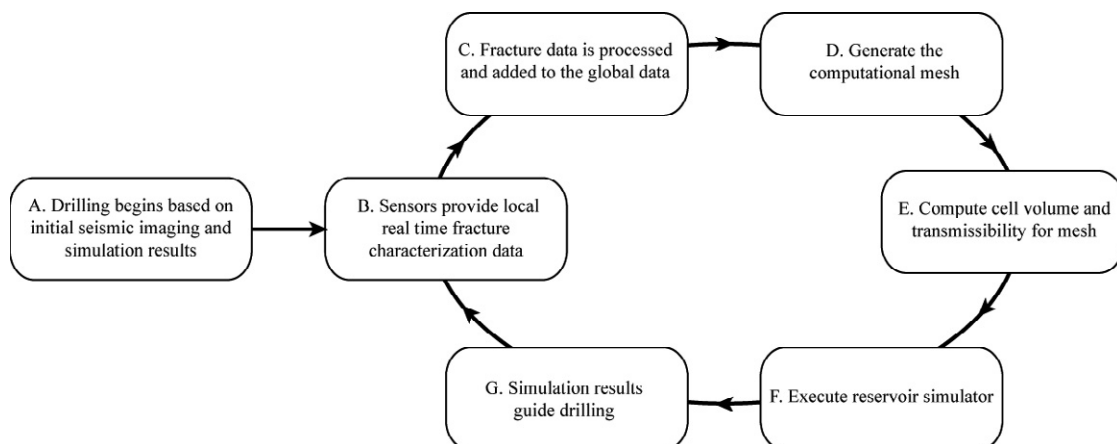


Fig. 2: Workflow

In Fig. 2(C), the additional local fracture data is added to the collection of global data. For example, GOCAD [11] provides software written that will process this data into a usable format for meshing. While GOCAD is now commercial product distributed by Paradigm, open source software can be used instead.

In Fig. 2(D), the fracture data can then be meshed. If a tetrahedral mesh is used, TetGen [12] can be used to obtain this mesh. TetGen can be downloaded and used freely for noncommercial uses. A separate commercial license is available. Other open source software can be used instead.

In Fig. 2(E), once the mesh is obtained, cell volumes and transmissibilities are computed. A software package called Grid2Trans [13] can be used to obtain this information. Once again, other open source software can be used instead.

In Fig. 2(F), the reservoir simulator is run, which can be any reservoir simulator that handles the meshes created in Fig. 2(D) and data in Fig. 2(E). The General Purpose Stanford Simulator (GPSR) was used in [8], but any simulator suffices.

Finally, in Fig. 2(G), the simulation results are then post-processed and used to more accurately determine advantageous drilling directions that uses all of the available information.

Visualization is not explicitly mentioned in any of these steps, but is normally used in areas A, C, D, F, and G. Typically, a complete cycle of steps B-G can take months using standard data collection and processing. Many parts require tedious manual intervention. A goal of DDDAS research is to identify and automate key components in an application.

Using a (near) continuous data stream that can be available while creating the fracture network, areas B-G can run as a continuous pipeline when new data is available. If manual intervention is really required for a step (e.g., C), then it can be done while the other steps are computing. If possible, no manual intervention should be done, just expert oversight to see that each step has produced reasonable results.

Consider an example shown in Fig. 3. In Fig. 3(a), the initial seismic imaging shows primarily fractures and a horizontal well that has already been drilled with places for pressure flooding or steaming to create more fractures hydraulically. There are neighboring, already existing vertical wells (one shown) that have microseismic imaging sensors in place. In Fig. 3(b), we see the result of hydraulic fracturing at the far point in the horizontal well, which is mapped from data from the neighboring microseismic imaging sensors. In Fig. 3(c), a plug was placed at the next to farthest point (to stop gas flooding of the well) and we repeat the hydraulic fracturing. Fig. 3(d) shows the results after all of the points have had hydraulic fracturing.

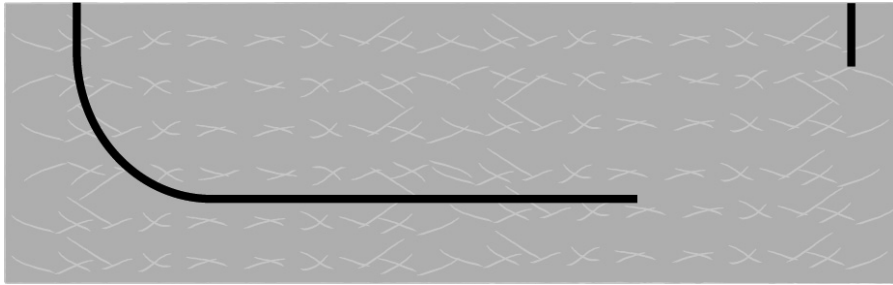
Microseismic imaging is able to show the new fractures (see Fig. 3(b)-(d)), which can be added to the fracture map (see Fig. 2(B)) as data becomes available. Now the DDDAS in Fig. 2 can use streamed data to run steps B-G in Fig. 2. This can be repeated as long as needed to create a large gas reservoir that includes as many fractures as economically feasible for a given budget.

Processing the microseismic images is much quicker than for a field since much less data is involved. However, integrating the data into the overall seismic image is nontrivial and is not automatic, and there have been few advances in automatically doing this step. It is an open research area.

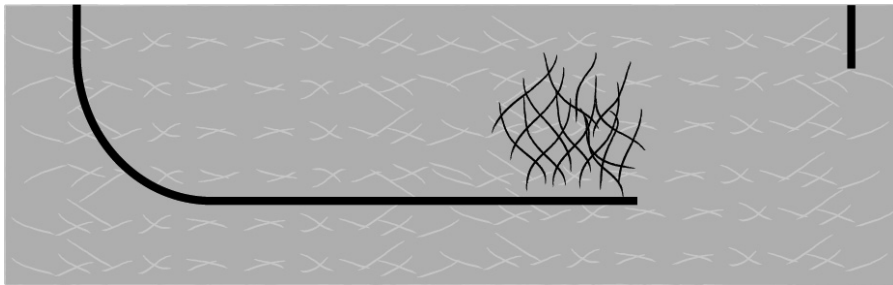
A numerical example for a synthetic fractured gas reservoir can be found in [8] that represents one time through all of the steps in Fig. 2. The example assumes that the reservoir has already been created and is a static object, not dynamic. The example shows that the model we use is very effective for simulation, but it is not a DDDAS.

#### 4. Conclusions and future work

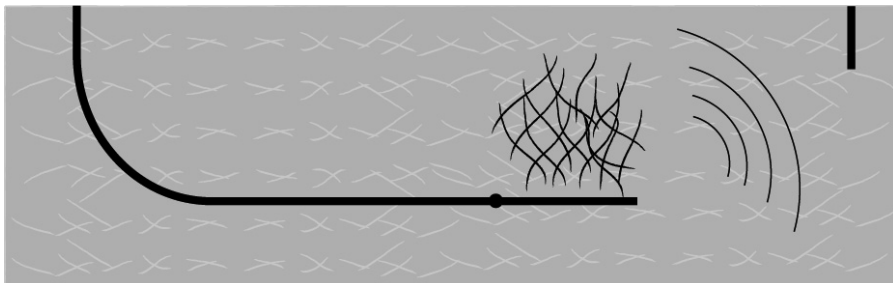
We outlined a DDDAS for network fractured shale gas reservoir creation in this paper that should work well with an established reservoir model and simulation. A systematic workflow for a DDDAS that models shale gas reservoirs with complex fractures in fine scale (DFM) and coarse scale (MSR). Ideally this methodology will be implemented to for a real shale gas development project where the natural and hydraulic fracture network is mapped through borehole imaging logs, microseismic imaging, and other characterization approaches.



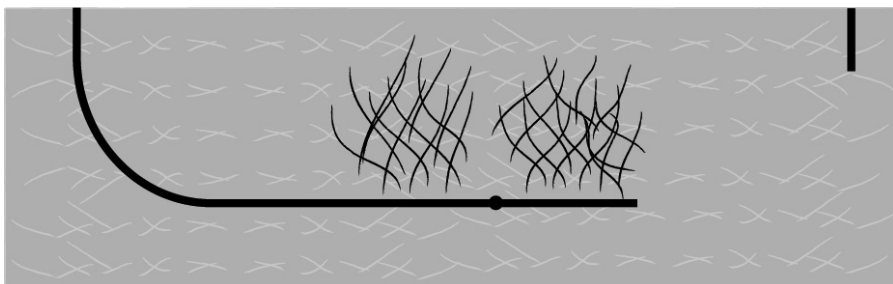
(a) Initial configuration with the horizontal well, the vertical well with microseismic sensors, and natural fractures



(b) Fractures after the first hydraulic fracturing process completed at far end of horizontal well



(c) Microseismic imaging to the vertical well with a plug in place in the horizontal well



(d) Fractures after the first hydraulic fracturing process completed at far end of horizontal well

Fig. 3: Fracture creation and microseismic imaging

## Acknowledgements

This research was supported in part by NSF grants 1018072 and 1018079, awards from the Center for Fundamentals of Subsurface Flow, School of Energy Resources, University of Wyoming, Sinopec, and Award No. KUS-C1-016-04, made by King Abdullah University of Science and Technology (KAUST).

## References

1. hydraulic-fracturing.mwv, <http://www.youtube.com/watch?v=nvnnBcxhzNA>, EnCana Corporation. Last visited 1/15/2011.
2. N.R. Warpinski., Microseismic: The new silver bullet in naturally fractured reservoirs?, AAPG Hedberg Conference: The Geologic Occurrence Hydraulic Significance of Fractures in Reservoirs, Casper, Wyoming, July 13, 2008.
3. R. Baca, R. Arnett, and D. Langford, Modeling fluid flow in fractured porous rock masses by finite element techniques, *Int. J. Numer. Meth. Fluids* 4 (1984) 337–348.
4. R. Juanes, J. Samper, and J. Molinero, A general and efficient formulation of fractures and boundary conditions in the finite element method: *Int. J. Numer. Meth. Eng.*, 54 (2002) 1751–1774.
5. S.K. Matthai, A.A. Mezentsev, and C.C. Pain, A higher-order TVD transport method for hybrid meshes on complex geological geometry: *Int. J. Numer. Meth. Fluids*, 47 (2005) 1181–1187.
6. S. Sarda, L. Jeannin, R. Basquet, and B. Bourbiaux, Hydraulic characterization of fractured reservoirs: Simulation on discrete fracture models, *SPE J. Reservoir Eval. Eng.*, 4 (2002) 154–162.
7. B. Gong, M. Karimi-Fard, and L.J. Durlofsky, Upscaling Discrete Fracture Characterizations to Dual-Porosity, Dual-Permeability Models for Efficient Simulation of Flow with Strong Gravitational Effects, *SPE J.*, 13 (2008) 58–67.
8. B. Gong, G. Qin, C.C. Douglas, and S. Yuan, Detailed Modeling of the Complex Fracture Network of Shale Gas Reservoirs, SPE-142705, 2011.
9. M. Karimi-Fard, L.J. Durlofsky, K. and Aziz, K, An Efficient Discrete-Fracture Model Applicable for General-Purpose Reservoir Simulators. *SPEJ* 9/2 (2004) 227–236, SPE-79699-PA.
10. M. Karami-Fard, B. Gong, and L.J. Durlofsky, Generation of coarse-scale continuum flow models from detailed fracture characterizations: *Water Resources Research*, 42 (2006), W10423.
11. GOCAD, <http://www.gocad.org> and <http://www.pdgm.com/products/gocad.aspx>. Last visited 1/15/2011.
12. TetGen, <http://tetgen.berlios.de>. Last visited 1/15/2011.
13. M. Karimi-Fard, Grid2Trans, available from author, see <http://www.stanford.edu/~karimi>. Last visited 1/15/2011.