

Reservoir Simulation With Simultaneous Graphic Display on the Macintosh II PC

S.O. Norris,* SPE, and K.S. Capps** • Texas A&M U.

*Now with Halliburton Services.

**Now with K.W. Brown & Assocs.

Copyright 1990, Society of Petroleum Engineers

This paper (SPE 20362) was prepared for presentation at the Fifth SPE Petroleum Computer Conference held in Denver, CO, June 25–28, 1990.

Summary

This paper describes how a Macintosh II™ PC, which is usually thought of as a desk-top publishing system, can be used as an efficient tool for conducting reservoir simulation and evaluating the large amounts of data commonly generated by simulation runs. A unique feature is the machine's ability to produce simultaneous graphics with minimal programming. This ability, which allows the engineer to "monitor" the simulation, is particularly useful in history matching.

Postprocessing can be accomplished with public-domain software that yield excellent graphical presentations. Color contour maps are easily created and can be statistically evaluated. Special options—such as cross-section profiles, edge detection, and reverse imaging—allow comprehensive interpretation of simulation data.

Introduction

Comprehensive reservoir simulation on PC's is quickly becoming a reality as the speed and dynamic memory of the machines con-

tinue to increase. In the past, reservoir simulation on PC's was generally confined to implicit pressure, explicit saturation solutions for very small problems because of the limitations of the hardware and operating systems. Today, with the advent of faster, more powerful PC's and new operating systems, fully implicit solutions once thought suitable only for mainframes are now possible on PC's. The use of PC's for reservoir simulation and for other applications is attractive for the following reasons: (1) PC's can dramatically reduce expensive mainframe charges; (2) PC's eliminate the many problems and inconveniences presented by multiuser systems; and (3) PC's commonly have better graphic capabilities than their mainframe counterparts.

The industry generally considers graphics for reservoir simulation a "must" because it allows rapid interpretation of the extremely large amount of data generated by simulation runs. Points of interest and/or mistakes in the reservoir model that might go undetected in a large computer printout frequently become obvious when viewed graphically. Also, subsequent simulation runs can be compared with previous runs by juxtaposing the respective graphic outputs for a specific timestep. Many companies spend thousands of dollars a year for the very simplest graphic representations; we plan to show that the Macintosh computer, with inexpensive commercial and public-domain software, is capable of generating sophisticated graphics for a fraction of that cost.

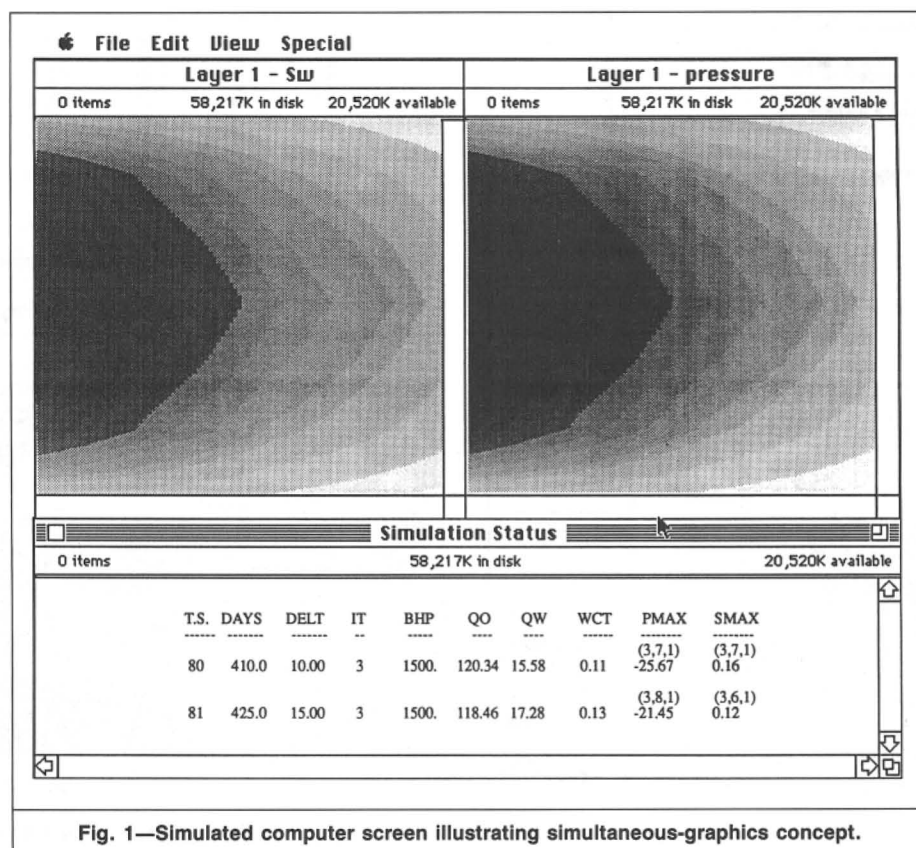


Fig. 1—Simulated computer screen illustrating simultaneous-graphics concept.

Although somewhat new to reservoir simulation, the concept of simultaneous graphics has been used in other areas of the industry; most notably, fracture design models frequently have simultaneous graphics that depict the creation of a hydraulic fracture and the transport of sand or artificial proppants. Simultaneous graphics is attractive for reservoir simulation because it allows the engineer to "monitor" the run as it is executing to check for points of interest or to detect errors in the model. This could be a particularly valuable tool for history matching.

Reservoir Simulation on the Macintosh Computer

The simulator used for this study is a three-phase, 3D, fully implicit, black-oil reservoir simulator. The simulator originally resided on a mainframe environment but was easily ported to the Mac. Changes to the code were minimal (the common block "include" statements had to be changed). Although the execution times were slower on the Mac than on the mainframe, problems involving several hundred gridblocks ran at a very acceptable speed. The Mac used for this project had 8M RAM and a clock speed of 16 MHz (de-

tails are given later). Apple has since introduced two newer models with significantly higher RAM memory and clock speed. These new machines should be able to execute even complex 3D problems in a reasonable amount of time.

The simulator, which was written in FORTRAN, compiled quickly with the MPW FORTRAN compiler. This compiler has many mainframe-type options, such as array boundary checking, different levels of optimization, and a run-time debugger. MPW FORTRAN supports all the Vax extensions, including structures, and does not require special applications or add-ons to use all the memory of the Macintosh (current maximum is 8M). The new operating system proposed by Apple will allow a virtual memory capacity.

Simultaneous Graphics

The objective of simultaneous graphics was to provide the engineer with the ability to monitor the simulation as it takes place. As Fig. 1 shows, this model presents three windows to the screen, two for graphics and one for text—i.e., timestep summary infor-

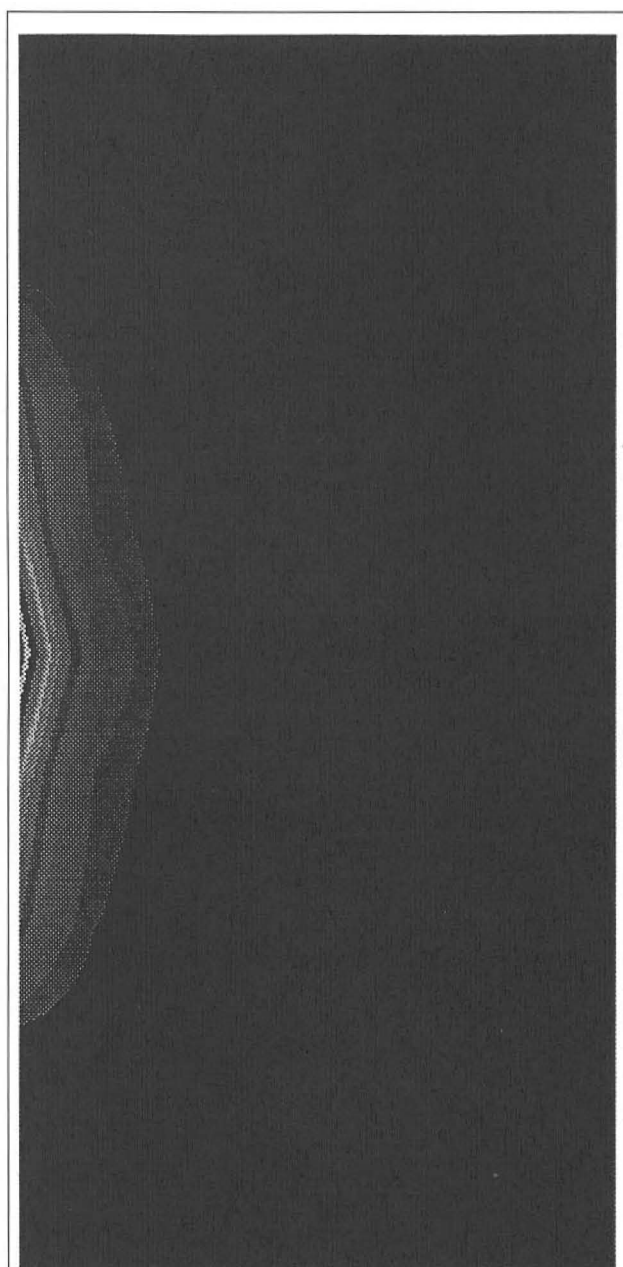


Fig. 2—Reservoir-pressure contour after 1 day of production.

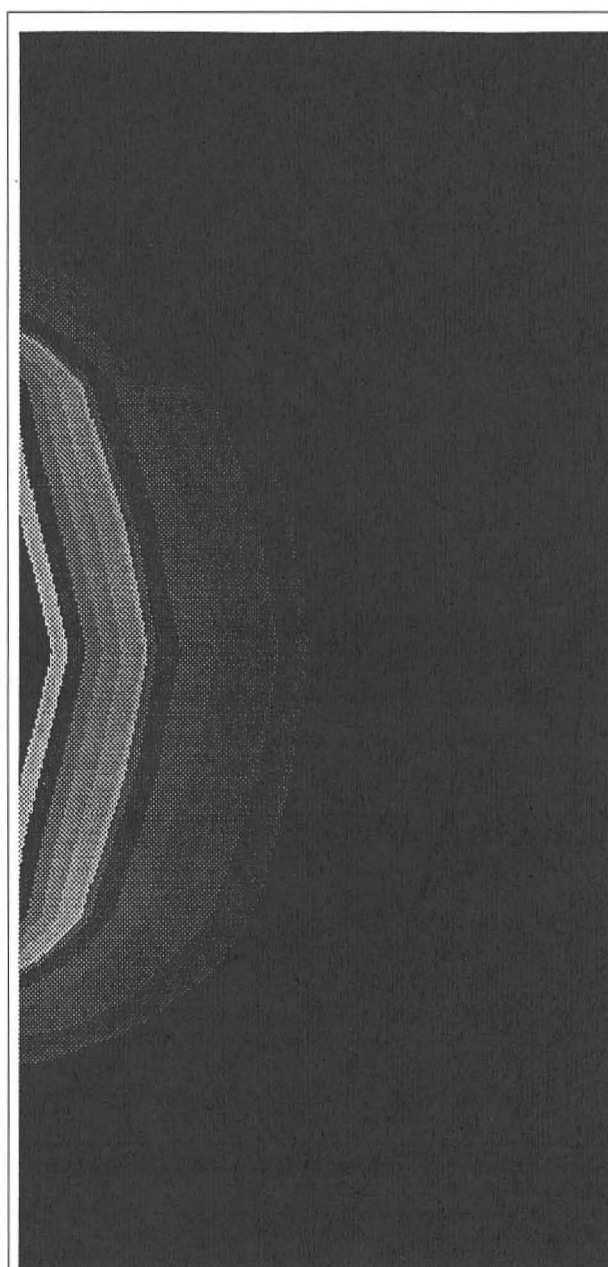


Fig. 3—Reservoir-pressure contour after 10 days of production.

HALLIBURTON EXHIBIT 1030

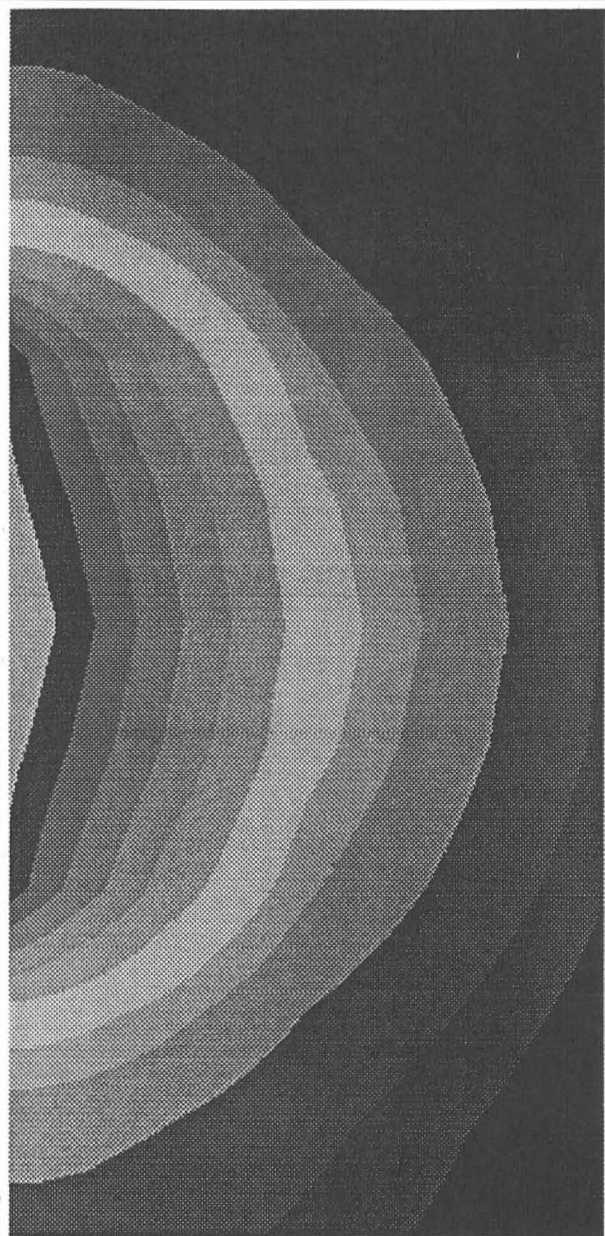


Fig. 4—Reservoir-pressure contour after 100 days of production.

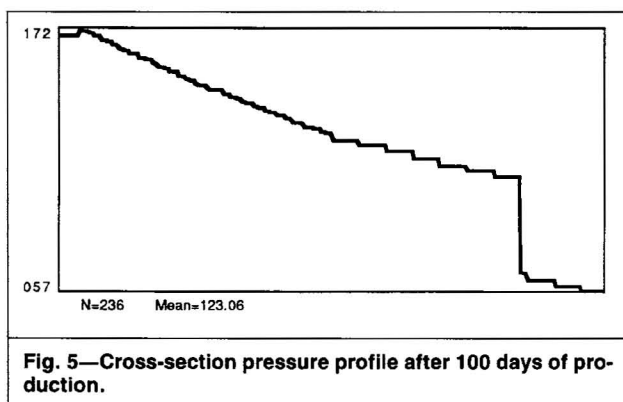


Fig. 5—Cross-section pressure profile after 100 days of production.

mation. Four variables can be selected for color contouring: pressure, oil saturation, water saturation, and gas saturation. These values must be scaled to fit into a 256 color spectrum.

Calls to graphics subroutines are done within the FORTRAN code. The subroutines are included in the Macintosh system and are fairly high level, so the amount of additional coding is minimal. As the simulator executes, arrays containing the pressure and saturation profiles are written to an off-screen bit map, along with several instruction indicators. The FORTRAN program calls the graphics subroutines at each timestep, giving the appearance of simultaneous graphics.

Postprocessing

A very elegant set of graphics packages is available from public-domain software that allows professional-quality presentation of simulated data. The first package, DATASCOPE, is used to read in the array data, modify the data if desired, display the data on a nodal basis, and interpolate between nodes to give a continuous color contour. The image generated by DATASCOPE can then be transferred to a second package, IMAGE, which is used to enhance



Fig. 6—Reversed and sharpened image of Fig. 3.

the display. IMAGE can acquire, enhance, measure, edit, animate, and print color and pseudocolor images. It contains many standard image-processing features, including histogram equalization, contrast enhancement, density profiling, smoothing, sharpening, edge detection, and noise reduction. It supports multiple windows and has eight levels of magnification. This software is very simple to use; any Mac user can master the use of the packages in just a few minutes because of the defined standard interface common to Apple software.

Figs. 2 through 7 show examples of the postprocessing capabilities of these packages. Figs. 2 through 4 represent the pressure profile data from a simulation of a horizontal well intercepting a vertical fracture. Because the reservoir is symmetrical, only half of the drainage area was simulated. The vertical fracture, which extends half the length of the drainage boundary, runs along the left edge of the figure, while the horizontal well runs perpendicular to the fracture. Fig. 2 shows the reservoir after 1 day of production and reveals that the reservoir is infinite-acting and that the linear flow period has expired. Fig. 3 represents the reservoir at 10 days and shows the familiar elliptical flow patterns around the fracture. Fig. 4 shows the reservoir at 100 days; a pseudoradial flow regime is detectable around the fracture and would continue to expand in an infinite reservoir.

A particularly useful feature is the ability to examine cross sections of the color contour maps. This is accomplished by "clicking" with the mouse on two points, which creates an x - y plot with cross-section values. This feature allows creation of Buckley-Leverett-type profiles useful in determining saturation fronts. Fig. 5 is a cross-sectional view taken at the center of Fig. 4 running parallel to the horizontal well. The numbers on the y axis indicate density values and can be scaled to represent actual pressure data.

For gray-scale images, the "reverse-image," "edge-detection," and "sharpening" options can be used to enhance the presentation. Fig. 6 is a reversed image of Fig. 3 to which edge detection has been applied. This feature is very helpful in that information not clearly visible in the contour mode is now revealed. In Fig. 6, it is possible to detect the fact that the pressure transient has extended much farther into the reservoir than Fig. 3 would indicate. Fig. 7 shows the application of the edge detection option applied to the contour of Fig. 4.

Hardware and Software Requirements

This project was conducted with the following hardware and software.

Hardware: Macintosh II PC
68030 16.5 MHz Motorola
68882 coprocessor
80M hard disk
13" Sony color monitor
256 color video card
8M RAM

Software: MPW FORTRAN 1.2.1
MPW 3.0
Macintosh Finder 6.1
Macintosh System 6.0.2

Manuals: Inside Macintosh Vo. I-IV

Conclusions

Fully implicit reservoir simulation is now possible and desirable on PC's because of the reduction of mainframe cost, the attrac-

tiveness of a single-user environment, and the enhanced graphics capability. As the memory and speed of PC's continue to increase, many reservoir simulation problems once thought too large to run on PC's will become ideal candidates for PC simulation.

The Macintosh PC is capable of executing medium-size, fully implicit simulation runs and presents an excellent tool for simulation analysis.

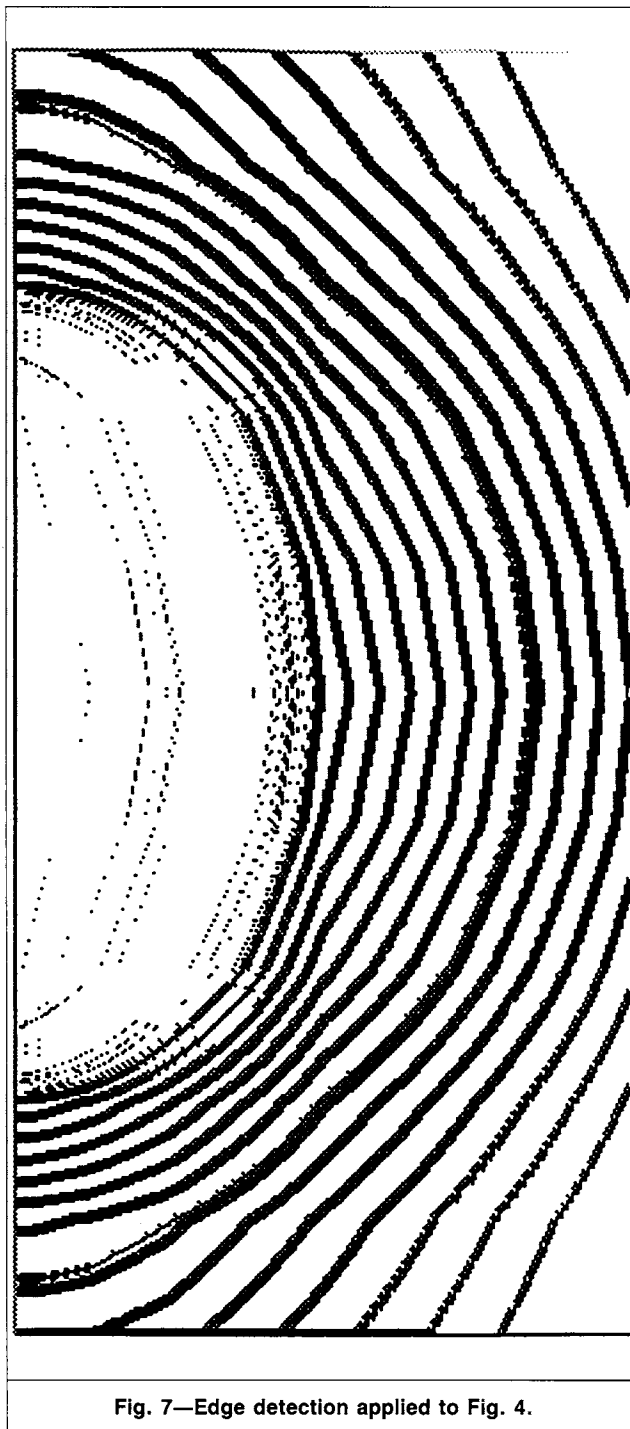


Fig. 7—Edge detection applied to Fig. 4.