



SPE 93906

Optimizing Cyclic Steam Oil Production with Genetic Algorithms

A. Patel, ChevronTexaco Energy Technology Co.; D. Davis, NutechSolutions Inc.; C. Guthrie, and D. Tuk, ChevronTexaco Energy Technology Co.; and Tai Nguyen and J. Williams, ChevronTexaco North America Upstream

Copyright 2005, Society of Petroleum Engineers Inc.

This paper was prepared for presentation at the 2005 SPE Western Regional Meeting held in Irvine, CA, U.S.A., 30 March – 1 April 2005.

This paper was selected for presentation by an SPE Program Committee following review of information contained in a proposal submitted by the author(s). Contents of the paper, as presented, have not been reviewed by the Society of Petroleum Engineers and are subject to correction by the author(s). The material, as presented, does not necessarily reflect any position of the Society of Petroleum Engineers, its officers, or members. Papers presented at SPE meetings are subject to publication review by Editorial Committees of the Society of Petroleum Engineers. Electronic reproduction, distribution, or storage of any part of this paper for commercial purposes without the written consent of the Society of Petroleum Engineers is prohibited. Permission to reproduce in print is restricted to a proposal of not more than 300 words; illustrations may not be copied. The proposal must contain conspicuous acknowledgment of where and by whom the paper was presented. Write Librarian, SPE, P.O. Box 833836, Richardson, TX 75083-3836, U.S.A., fax 01-972-952-9435.

Abstract

This case study describes a project applying a new technology—genetic algorithms—to the problem of scheduling oil production by cyclic steaming at an oil field in the San Joaquin Valley. The paper has seven parts.

In the first part, we discuss the nature of the problem, the importance of solving it well, and the nature of the work process as it occurred before the project began. In the second part, we discuss genetic algorithms and the way that the project adapted them to solve the problem of scheduling cyclical steam injections. In the third section, we describe a pilot project that demonstrated the potential of the approach. In the fourth, we describe the full project that created an optimization tool that has been working daily with oil field personnel to schedule production for the entire field for more than a year. In the fifth, we discuss two types of results of the project—an increase in production and changes to the work process on the part of the oil field personnel owing to their enthusiastic adoption of the optimizer. In the sixth, we discuss some project management factors that led to the success of the project. Finally, we discuss ways in which the success of this project may be relevant to a variety of other types of scheduling and resource allocation problems in the field of oil production.

The paper will center on three themes: the successful solution of a hard production problem with a new technology; the impact of that technology on the oil field personnel; and the potential of that technology to support other types of projects with similar levels of return.

1. Introduction

➤ The Cyclic Steam Process

The Antelope reservoir, located in the Cymric Field, in the San Joaquin Valley, is a siliceous shale reservoir containing heavy oil (12° to 13° API gravity). The reservoir consists primarily of diatomite, characterized by its high porosity (60%), high oil saturation (50-60%) and very low permeability (2 md). Approximately 430 wells are producing from this reservoir with an average daily production of 23,000 BOEG. The oil from the field is recovered using a Chevron-patented cyclic steam process. A fixed amount of saturated steam is injected during a three-to-four day period into the reservoir. The steam's high pressure fractures the rock and the heat from steam reduces oil viscosity. The well is shut in during the next couple of days known as the soak period. Condensed steam is absorbed by the diatomite and oil is displaced into the fractures and well bore. After the soak period the well is returned to production. The flashing of hot water into steam at the prevailing pressure provides the energy to lift the fluids to the surface. The well flows for approximately 20 to 25 days. After the well dies, the same cycle – cycle length of 26-30 days - is repeated.

Scheduling Problem for Cyclic Steam Process

Since there is no oil production during the steaming and soaking period, there is an incentive to minimize the frequency of steaming and increase the length of the cycle. On the other hand, since the well production is the highest immediately after returning to production and declines quickly thereafter, a case can be made for increasing the steaming frequency and reducing the length of the cycle. This suggests there is an optimum cycle length for every well that results in maximum productivity during the cycle. Extend this dilemma for more than 400 wells in the field, combine that with the constraints of steam availability and balancing of the steam distribution system, factor in additional facility constraints, and the result is a formidable scheduling optimization problem.

➤ Other challenges

The production stream is a mixture of heavy oil, associated gas, water and steam. It is quite difficult to accurately measure the rates of each of these components. Furthermore, the production rates for individual wells were measured only

infrequently. As a result, the data have a high degree of uncertainty. Yet another challenge was the lack of available well performance models needed to predict the production during the cycle. The final challenge was to make the new solution / process acceptable to the operational personnel who had been solving this problem quite efficiently for many years.

➤ Work process prior to this project

There used to be two independent groups of people involved in making a decision regarding which wells to select for steaming. The first group is the well managers who used to recommend a list of wells based on the observed well head pressures, the current performance of the well and its past history. These wells used to be called the PFS (Prep for Steam) wells. Note that once a well makes it to the PFS list, it has the same priority as all the others on the PFS list, thus losing the distinction between the top candidate and the last candidate on the list. The second group is the steam operators would then go through this PFS list and select the wells that can be steamed while honoring the constraints of the steam distribution system. In order to get the optimal response from the field, these two steps (selection of candidates and balancing of steam distribution system) needed to happen simultaneously, but the work process resulted in sequential implementation.

➤ Business case / opportunity

It was recognized by the engineers and operations that the factors mentioned above (work process, heuristics and uncertain measurements) led to a sub-optimal solution and that there was an opportunity to improve the performance of the field. The project team used historical data and went through an academic exercise to estimate the benefit for this project. They estimated that depending upon the quality of data and frequency of measurements it was possible to achieve a percentage increase in production in the upper single digit range. The consistency and reliability of the new decision making process were thought to be added pluses.

2. Statement of Theory and definition

➤ Genetic algorithms – background

Genetic algorithms (GA) are global optimization techniques developed by John Holland (Holland 1975). They are one of several techniques in the family of evolutionary algorithms—algorithms that search for solutions to optimization problems by “evolving” better and better solutions. In a genetic algorithm, one begins with a “population” of solutions and then chooses “parents” to reproduce. During reproduction, each parent is copied, and then parents may combine in an analogue of natural cross-breeding, or the copies may be modified, in an analogue of genetic mutation. The new solutions are evaluated and added to the population, and (typically, low-quality) solutions are deleted from the population to make room for them. As this process of parent selection, copying, cross-breeding, and mutation is repeated, the members of the population tend to get better and better.

When the algorithm is halted, the best member of the current population is taken as the algorithm’s solution to the problem posed.

One critical feature of a genetic algorithm is its procedure for selecting population members to reproduce. Selection is a random process, but a solution member’s quality biases its probability of being chosen. Because genetic algorithms promote the reproduction of high-quality solutions, they explore neighboring solutions in high-quality parts of the solution search space. Because the process is randomized, a genetic algorithm also explores parts of the search space that may be far from the best individuals currently in the population, and so some of the solutions found by the algorithm are found in an exploration of the search space.

○ **Use of GA in the industry**

In the last twenty years, genetic algorithms have been used to solve a wide range of optimization problems. Problems for which they are well-suited tend to have one or more of these characteristics: the problem cannot be well formulated as a linear programming problem; the problem can be formulated as a linear programming problem, but it takes too long to optimize in this form; or the problem is complex, and brute force and heuristic search techniques are inappropriate or find significantly sub-optimal results.

There are many examples of optimization problems in the petroleum industry for which genetic algorithms are well-suited. At ChevronTexaco, in addition to the cyclical steam scheduling problem, the following problems have been addressed with genetic algorithms: well placement, rig scheduling, portfolio optimization and facilities design. At NuTech Solutions, the following problems have been addressed with genetic algorithms: planning of rig workover projects to that overall workover time is reduced; planning production across multiple plants to reduce costs; planning distribution from multiple plants to large number of customers to reduce costs; and controlling pipeline operations to reduce costs while satisfying pipeline constraints.

All of these problems have the common features that they can not be formulated as linear programming problems, they involve a large number of complex constraints, both hard and soft, and significant increases in profits can be obtained if the problems are solved better than they were being solved before the creation of a computerized optimization procedure.

○ **Suitability of a GA for this problem**

Genetic algorithms are applicable to the problem of scheduling cyclical steam injection of heavy oil wells because there does not seem to be a linear formulation of the steam injection problem (production falloff is exponential and constraints are nonlinear). Also, in a pilot project centered on a single gauge setting, ChevronTexaco determined that there seemed to be a significant increase in profit that could be captured if the practices currently in place at the oil field could be supplemented by the output of a global search routine that

could consider hundreds of thousands of possible schedules and “evolve” high-quality schedules that increased production while satisfying the problem’s constraints.

➤ **Problem formulation**

The scheduling problem is formulated as a GA optimization problem where the objective is to maximize cumulative production over two months’ period. The fitness function is calculated as the cumulative production minus the penalties for violating the soft constraints.

○ **Hard and soft constraints**

The problem has many constraints. The field level constraints such as the steam availability and maximum # of wells steaming; the GS level constraints such as min amount of steam used and max # of wells on production; the header level constraints such as max # of wells steaming or the individual well constraints such as max / min number of production days. Additionally, there are operational constraints such as communication – where multiple wells must be steamed together and blocked wells due to rig activities.

Although all of the field constraints could have been incorporated in the problem formulation as hard constraints – constraints that absolutely cannot be violated, it was decided to make many of the constraints soft constraints - these constraints can be violated but there is an associated penalty. An example of a hard constraint is the total steam available on a given day for the whole field whereas some of the soft constraints are maximum amount of steam used by a GS and minimum # of wells steaming in a GS or a header.

○ **Termination criteria**

The optimization is stopped when one of the following criteria is satisfied:

- a) a specified amount of generations have been created
- b) a specified amount of time is elapsed, or
- c) the fitness function has not improved over a specified number of generations

○ **Seed generation**

The genetic algorithm used multiple heuristics to enhance its performance and speed up its search for high-quality solutions. To begin with, when it created the initial population of solutions – “the seed”, it used heuristics based on those that the well operators and steam operators used at the oil field. It also used some heuristics developed for the project to find good initial schedules. An example of such a heuristic is “attempt to steam high-production wells at their optimal cycle length—the length of time between steaming at which a well’s average daily production is maximized.” The constraints of the problem made it impossible to steam all wells at their optimal cycle length, but inserting schedules based on this as a goal into the initial population gave the algorithm some high-quality solutions that could be mutated and cross-bred with other types of solutions in order to find even better solutions.

○ **Solution representation (Chromosomes)**

The technique we used for representing solutions was not the approach commonly found in textbooks on genetic algorithms. We used an indirect encoding approach—each solution was a permutation list of wells, with multiple entries allowed for the same well. We then used a decoding procedure that simulated the effects of various schedules to translate the permutation list into an actual schedule. Let us illustrate this process with an example.

The genetic algorithm encoded solutions to the cyclical steam scheduling problem as lists of wells, of the form

```
<w34, w1832, w196, w34, w9185,>
```

Such a list contains all the wells that could be steamed, and it typically has duplicate entries. (Note that the example above lists well 34 twice.) This list is not yet a schedule. We used the following procedure to convert a list of this form into a schedule: We began on day one. The schedule builder looks at the first well on the list and simulates steaming it on day 1. If this process violates no hard constraints, then the well is scheduled for steaming on day 1. The schedule builder then looks at the second well on the list—well 1832, in this case. It simulates the effects of steaming that well together with well 34 on day 1. If no hard constraints are violated, this well is added to the schedule for day 1. If hard constraints are violated, the well is not added to the schedule. The process continues, considering each well for steaming on day 1, and adding each well, in order, that can be steamed without violating a hard constraint. Then the process continues with day 2, considering each well in order that was not already steamed in day 1. And so forth. The critical point to note is that the schedule-building process will not build a schedule that violates a hard constraint. It is also worth noting that this schedule-building process uses some clever heuristics and a simulator in order to transform a list of wells into a feasible steaming schedule. Once a schedule is built, it can be evaluated, and its “score” is returned to the genetic algorithm as the evaluation of the original solution—the list of wells.

○ **Special operators**

The optimization process uses heuristics to initialize the population, as well as randomly-generated solutions to fill out the initial population. It includes intelligent heuristics in the procedures used to modify new solutions. It uses “cross-breeding” procedures appropriate to combining different permutations in order to combine two parents to produce a child. It includes a good deal of domain knowledge in the schedule builder in order to produce feasible schedules. It includes a post-processor that checks to find simple changes that could be made to the best solution found in order to improve its quality.

○ **GUIs, Platforms etc**

The interface to the optimizer gives the well operators and the steam operators at the field a good deal of power and

flexibility in their interactions with the system. The operators can edit the well data that is input to the optimizer. They can select which optimization heuristics and procedures will be used in a run. They can parameterize the objective function that specifies the goals of the run. They can activate, deactivate, and parameterize the hard and soft constraints on the run. They can also edit the solutions found by the optimizer—an essential capability, particularly in cases in which there is a constraint known to the operators that is not reflected in the databases available to the optimizer.

3. Description of application of equipment and process

➤ Pilot test

A pilot test was conducted to demonstrate the feasibility as well as to determine the potential benefit of the approach. A gauge setting comprising of 21 wells was selected and the test was conducted for 60 days. During this period we operated the wells in this gauge setting only on the recommendations of scheduler without any adverse effects. The work process was changed to facilitate the scheduler needs. The operators successfully adopted the new process that relied much more on the production data that were updated daily and used the Visualizer (The next section describes in detail the functionality of Visualizer. At this juncture it is sufficient to mention that this tool /module uses the past data and predicts the well performance) to determine the well performance parameters. Daily logs were kept of the recommendations that were followed. At the end of the test period, a comparison was done against the baselines.

➤ Pilot test results

○ **Production during the test increased**

Cumulative production from this GS during the 60 day period increased 11 %. The number of active wells was same as the base period. Thus, implying the well productivity was increased.

○ **Steam injection also increased**

The scheduler recommended shorter cycles for many of the wells compared to the field practice. The cumulative steam during the same period increased 17 %.

○ **Difficult to separately quantify the benefits due to scheduler and added steam injection**

It is already known that the rate and volume of the steam injection affects the oil production. During the pilot test period, we did not control either of these, since the scheduler was asked only to increase the production. Since the total steam available for the field is constrained, it would not be possible for scheduler to increase the steam usage the way it did in the pilot test. As a result, the increase in the production due to higher steam use was backed out. The final conclusion was that the use of Scheduler would increase the production between 3 to 5%.

○ **Operators' response**

During the pilot test, well managers were asked to make a significant process change. They were asked to ignore the well head pressures and rely on the Visualizer which in turn was relying on the gauged information. The steam operators were instructed to strictly adhere to the steaming sequence suggested by the Scheduler. Not only did they quickly and successfully adopted the new work process but also found / suggested more changes that would further improve the process in the future.

○ **Other issues**

The pilot test helped identify issues that needed to be addressed before implementing the process field wide such as the need to address the steam system balance and the need for handling uncertainty in the measurements.

4. Description of the Optimization Tool

➤ System Design and Components

Figure 1 describes the overall system which is comprised of four major components: Database, Visualizer, Simulator and Optimizer.

The operator starts the main program interface and retrieves the current model values. The retrieved data can come from the local storage representing previously reviewed results or the field repositories. In either case the information is checked for completeness and consistency. The completeness and consistency checks ensure that there is enough data to run a simulation. Correctness is not ensured although some data integrity checks will be possible. Known errors are brought to the attention of the user who changes the information possibly with suggestions from the program. Results of this process are stored in a local repository.

Individual components are described below.

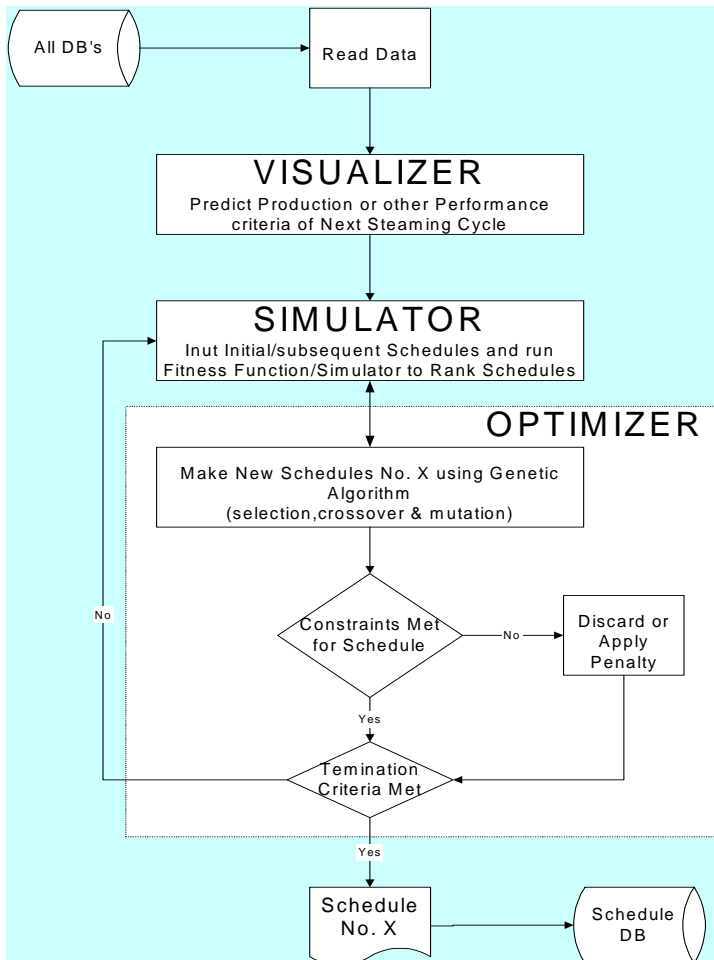
○ **Database**

A local database stores modeling information. This database serves several purposes. First, it is a collection point for field data. It provides quick access to field information, greatly reducing the sometimes extended query times from the main field database. Second, it stores model parameters derived from field data or captured by the interface. Third, this database houses field data that was changed by the operator for data quality or modeling reasons. Finally, it serves as a persistent storage of information that needs to be communicated between models, e.g. between the interface and the Simulation Engine.

- Local Database Functions
- Collects data from the field database
- Supplies raw production data to the Visualizer
- Captures decline characteristics from the Visualizer
- Supplies field data and decline characteristics to the user interface

- Maintains and captures new and adjusted model data not derived from field data. This information will be entered and edited through the interface and supplied as necessary to the Simulation Engine.
- Supplies model data to the Simulation Engine.
- Captures output from the case manager.

Figure 1.



○ Visualizer

The Visualizer provides a visual interface to the raw well production and pressure data. It can be used to explore well production for the last few steam cycles, as a place for users to correct production measurements, and as the creator of production prediction parameters. The primary function of the visualizer is to determine decline curve parameters for production wells.

- Visualizer Functions
 - Determines decline curve parameters
 - Data reconciliation
 - Allows user intervention to modify parameters

○ Simulator

The simulator consists of a Simulation Manager and a Simulation Engine. The Simulation Manager acts as the communication vehicle between the user interface, the optimizer, and the Simulation Engine. This is necessary because the system design is for three separate executables to exist along with a common simulation engine: One executable for the HEURO (optimizer) which needs to maintain its own startup entry point, one for the Visualizer, and one for our user interface. When the user instructs the interface to start an optimization run, the interface will start the simulation executable via the Simulation Manager. The Simulation Manager that exists in the optimizer process space will be given control in the HEURO framework main function prior to starting the framework.

- Simulator Functions
 - Accept calls from the user interface to write the simulation data to disk.
 - Launch the simulation executable and obtain evaluation.
 - The Simulation Manager performs initialization, establishes communication with the interface, and proceeds into the HEURO framework.
 - Manage the control of the program.

○ Optimizer

The Optimizer, HEURO is an engine supplied by NuTech Solutions. It consists of library functions that allow one to carry out all of the GA operations, namely populating, reproduction, crossover, mutation and selection. It also has functions that can be added to carry out any special operations as desired by the user. Finally, it has wrapper functions for easy integration with the rest of the system.

- Optimizer functions
 - Communicate with Simulation Manager.
 - Perform all necessary GA operations.
 - The Optimizer will supply a solution by calling the Simulation Manager with the solution in the chromosome format.

5. Results

As a risk mitigation measure, the plan for developing the Scheduler with all of its functionality was broken down into three Releases. Each subsequent release would be approved to move forward after the prior release was considered to be the success. The Release 1 of the Scheduler for the whole field was deployed in the first week of Jan 2003. It was developed jointly by ChevronTexaco and NuTech Solutions, who were primarily responsible for the optimization module. The system's performance was evaluated as a part of the look back study that is required of all major projects in SJVBU. The evaluation data presented below refers to the first 60 days of operation.

➤ Increase in Production

- *Production during the evaluation period has increased*

- The production from the field during the period Jan 8, 2003 to March 8 2003 increased 6.4% compared to the baseline. Some of this increase, however, is not due to the Scheduler. Since the field situation was dynamic and new wells were drilled, the production from these wells was a part of the increased production.
- In order to estimate the economic benefit of the scheduler, we needed to deduct the production gain attributable to the factors other than Scheduler. We also wanted to ensure that we erred on the side of underestimating the gain from the Scheduler rather than otherwise. Of the total increase of 6.4%, our best quantitative estimate showed that there was approximately 1.4% gain that cannot be attributable to absolutely any other factor. Although we believe that the use of Scheduler resulted in higher gain than 1.4%, for our economic analysis we assigned only 1.4% production increase to the Scheduler.
 - A preliminary look back of the project indicates that it has DPI of 4.19 and Pay-out of 2.5 months.

➤ Experience in the field

Scheduler development and deployment have consistently enjoyed the support from the operations personnel. After the installation at the site, the tool has been checked for functionality as well as accuracy by the operators and well managers. They have indicated that the Scheduler correctly provides the answers based on whatever is being asked from it.

- The Scheduler is in daily use since 8/03.
- Operators love it for its consistency and fast response.
- They have added other constraints and continue to ask for more functionality.

➤ Enthusiastic adoption

They have adopted the new work process which has the following features

- No PFS wells
- Spend time looking at the production curves, adjust visualizer parameters
- Focus on the right amount of steam to be used
- Duplicate this process elsewhere in the business unit

➤ Intangible benefits

Although the project was justified on the basis of accelerated production from the field, we have noticed additional benefits that cannot be quantified:

- The operators are claiming that the process is now consistent

- Operators are able to focus more on the field problems as the schedule is generated ahead of the time. A typical response is - "...there is no urgency to get back to the computer because the steam operators are waiting..."
- The schedule implementation is done system wide so steam system is always balanced
- The operators are able to analyze the results of the scheduler and are able to justify the results – making the tool more credible

6. Conclusions

Based on our experience on this project, we can reach the following conclusions:

➤ GAs can be effectively used for production optimization
Production optimization problems can be successfully solved using Genetic Algorithms. Although this specific application was related to scheduling, genetic algorithms can be applied to any kind of resource allocation problems as well as other constrained combinatorial optimization problems.

➤ Technology implementation is as much about right stakeholders as it is about right technology

The success of Antelope 1Y has been due to the involvement of the stakeholders throughout the development and deployment. The technology, no matter how great, will always need the assistance from right people. From the beginning to the end, the role of each of the stakeholder is critical. Having the right stakeholders in the loop ensures that throughout the project issues are being worked on and a belief in the ultimate success of the project is maintained. Many of the issues and decisions are non-technical in nature and they can only be resolved with the help of right stakeholders.

➤ Measurement accuracy / frequency very important for optimization

One of the issues that has been identified early on and which is still on our radar is the issue related to the quality of measurements. Since scheduler is a data driven tool, it is very important to have good quality data (measurements) as an input. The business unit operations have also recognized this issue and are in the process of implementing the necessary steps to improve the quality of both steam and oil measurements. We expect that the scheduler benefit will be even more after the data quality is improved.

➤ Other factors that led to the success of the project

- Excellent collaboration amongst all partners, and
- Frequent communication / weekly calls

The team was comprised of subgroups working from multiple locations – Boston, Richmond & Bakersfield. Constant communications helped resolve the issues quickly.

7. Relevancy and extensibility

The success of this project has led us to look for other areas where Genetic Algorithms and other Artificial Intelligence techniques may be relevant to a variety of other types of scheduling and resource allocation problems in the field of oil production. Indeed, we have already started several other projects in the area of well workovers and rig scheduling (well work scheduling).

Other areas where such applications may be successfully deployed are chemical injection optimization, choke optimization scheduler, and surface steam distribution optimization.

8. Acknowledgements

The authors would like to thank SJVBU –for allowing the publication of this paper, Jim Lederhos and Jim Ouimette for championing this project as well as providing valuable guidance throughout the implementation of this project, and Bob Fragione for providing assistance related to the field operations.

Figure 2.

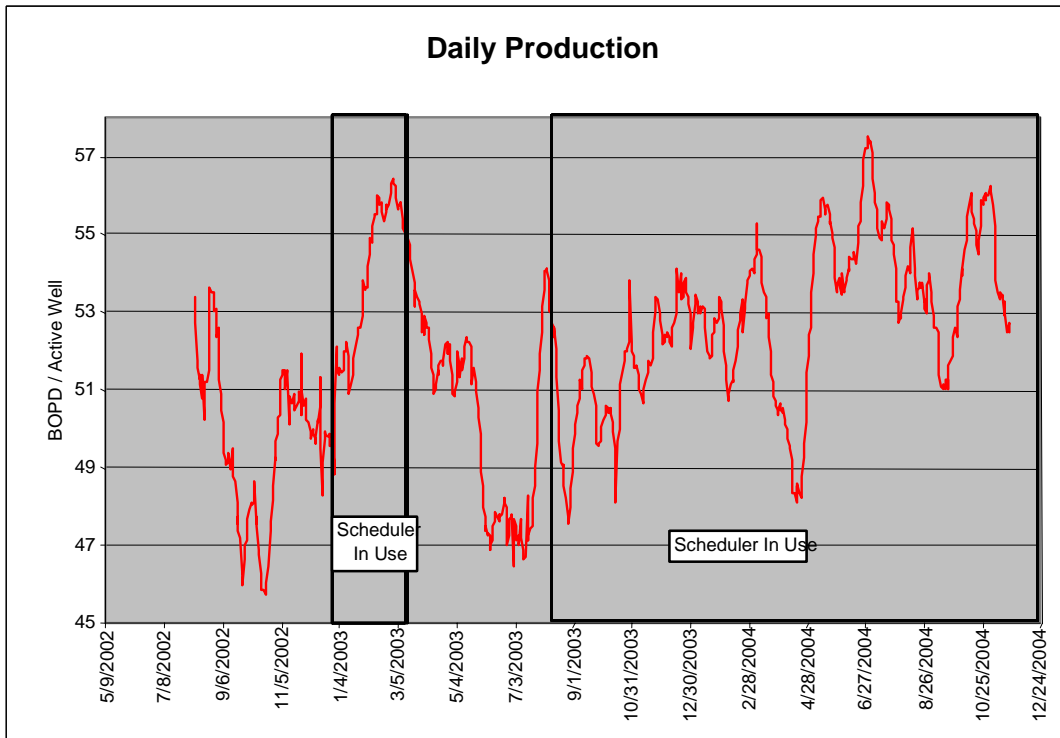


Figure 2 indicates the field performance since the scheduler has been deployed field wide.