DEVELOPMENT AND APPLICATIONS OF PRODUCTION OPTIMIZATION TECHNIQUES FOR PETROLEUM FIELDS

A DISSERTATION

SUBMITTED TO THE DEPARTMENT OF PETROLEUM ENGINEERING

AND THE COMMITTEE ON GRADUATE STUDIES

OF STANFORD UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Pengju Wang

March 2003

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Dr. Khalid Aziz
(Principal Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Dr. Roland N. Horne

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Dr. Anthony Kovscek

Approved for the University Committee on Graduate Studies:

Abstract

For some petroleum fields, optimization of production operations can be a major factor on increasing production rates and reducing production cost. In this work we investigated the formulations and solution methods for the following optimization problem: determining the optimal production rates, lift gas rates, and well connections for a gathering system with tree-like structures to maximize daily operational objectives subject to multiple flow rate and pressure constraints. While some aspects of this problem have been studied by other investigators, existing methods are either inefficient or they are based on significant simplifications that lead to suboptimal solutions. Hence, it is necessary to develop approaches that can solve the problem efficiently and without unreasonable assumptions.

We first investigated efficient procedures for simulating the production system and computing sensitivity coefficients of production rates relative to system parameters. These procedures have useful applications in simulation, sensitivity analysis, and optimization of a petroleum field.

We then focused on the rate allocation problem, which refers to the determination of the optimal production and lift gas rates subject to multiple constraints. When flow interactions among different wells are not significant, the well performance can be analyzed individually. Consequently, the rate allocation problem can be formulated as a separable programming (SP) problem whose objective and constraint functions are sums of functions of one variable. The SP problem is solved by various linear optimization techniques. This method is very efficient. However, it may lead to bad solutions when the flow interactions among wells are significant. In such cases, the rate allocation problem should be appropriately formulated so that simulations capable of capturing such flow interactions can be conducted in the optimization process. Several formulations were investigated. The suggested formulation is able to handle well shut-down, avoid some numerical difficulties, and is computationally efficient. Once formulated, the optimization problem was solved by a sequential quadratic programming (SQP) algorithm.

A two-level programming approach is developed to optimize the production rates, lift gas rates, and well connections simultaneously. In this approach, the entire optimization problem is solved in two levels. The upper level masters the overall solution process and explicitly optimizes the well connections. For each new set of well connections explored in the upper level, a lower level problem is formed to determine the optimal set of production and lift gas rates. The lower level problem is a rate allocation problem that can be solved by various optimization methods developed in this study. The upper level problem is solved by a newly developed heuristic method. The method is both efficient and robust, as verified by a genetic algorithm.

Production engineers often strive to achieve multiple conflicting goals when operating a field. Several existing multiobjective optimization methods are used to address this problem. Through an example, this study demonstrated that multiobjective optimization methods can help decision makers identify the best trade-offs.

The optimization tools were coupled with models for multiphase fluid flow in reservoirs and surface pipeline networks through a commercial reservoir simulator. The coupled procedure was applied to short-term production optimization in the Prudhoe Bay field of the North-Slope of Alaska, and to long-term reservoir development studies for two fields in the Gulf of Mexico. Results demonstrated the effectiveness of the approach.

Acknowledgments

I am immensely grateful to my advisor Dr. Khalid Aziz for his sharing of my excitement as well as frustrations in this research. Without his constant support, encouragement and guidance, this work wouldn't be possible. I also would like to thank Dr. Louis Durlofsky for providing valuable advice on various occasions during this study. I sincerely thank Dr. Roland Horne and Dr. Anthony Kovscek for reading the dissertation and providing valuable comments.

Part of this work was conducted in BP. Dr. Michael Litvak provided tremendous help in integrating the optimization tools into VIP (a commercial reservoir simulator of Landmark Graphics Corporation) and testing the optimization methods on several field models. I thank him for his help and guidance. Sincere thanks also go to other BP colleagues who in one way or another contributed to my pleasant summer internship experiences at BP. I also thank BP for providing me two summer internship opportunities and the permission to put some field examples in this dissertation.

I thank the systems optimization laboratory at Stanford University for providing me SNOPT 5.3, a general purpose nonlinear constrained optimization package. I thank Mathematics and Computer Science Division at Argonne National Laboratory and the Center for Research on Parallel Computation at Rice University for providing me ADIFOR 2.0, an automatic differentiation software.

The reservoir simulation industrial affiliates program (SUPRI-B) and the Department of Petroleum Engineering have supported me financially. This support is gratefully acknowledged.

I am grateful for the friendship of and help from the faculty, staff, and fellow students in the Petroleum Engineering Department. They contributed greatly to making my stay at Stanford enjoyable. I especially thank my friend Tao Shu for supporting me in many ways and bringing so much joy to my life for the last six years.

I deeply thank my parents, Jingmei Liu and Xingchun Wang. Their love and support are a constant inspiration to me in striving to improve myself. This dissertation is dedicated to you, Mom and Dad.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The ultimate goal of virtually all effort spent on modeling a petroleum field is to devise an optimal strategy to develop, manage, and operate the field. For some petroleum fields, optimization of production operations can be a major factor in increasing production rates and reducing production costs. While for single wells or other small systems simple nodal analysis may be adequate, large complex systems demand a much more sophisticated approach to predict the response of a large complicated production system accurately and to examine alternative operational scenarios efficiently.

As optimization algorithms and reservoir simulation techniques continue to develop and computing power continues to increase, upstream oil and gas facilities previously thought not to be candidates for advanced control or optimization are being given new considerations (Clay et al., 1998). The objective of this study was to develop optimization methods for solving a class of important production operation problems for petroleum fields.

1.1 Problem Statement

In petroleum fields, hydrocarbon production is often constrained by reservoir conditions, deliverability of the pipeline network, fluid handling capacity of facilities, safety and economic considerations, or a combination of these considerations. The task of field operators is to devise optimal operating strategies to achieve certain operational goals.

These goals can vary from field to field and with time. Typically one may wish to maximize daily oil rates or minimize production costs. This research aims to develop optimization methods that ease and automate the decision making of field operators for certain operations. In this section, the major components (the objective, the control variables, and the constraints) of the petroleum field optimization problem are described in detail.

Objective. Usually, the objective is to maximize the profit from an oil field on a day-to-day basis. How to define the profit is a complicated issue that requires intensive study in its own right. To keep this research focused, we restricted our attention to simple objective functions such as maximizing weighted daily flow rates.

We emphasize that we only optimize production operations for a short-term period. The flow rate and other operational settings, once determined, are assumed to remain fixed during that period. However the optimization procedures developed in this study can be used repeatedly for long-term hydrocarbon recovery studies.

Control variables. The control variables are the production operation settings to be optimized. In this study, the control variables included the lift gas rates, the production rates, and the well connections to flow lines.

Continuous gas-lift is a common artificial lift method used in the oil industry to improve well performance. The mechanism of gas-lift is fairly simple. Gas is injected into the tubing string to lighten the liquid column and decrease the bottom hole pressure, which allows the reservoir to push more fluids into the wellbore. At the same time, increased flow rates in the tubing string and surface flow lines result in higher backpressure on the well and adjacent wells that share a common flow line. This in turn causes a reduction in well production rates. Therefore, lift gas has to be carefully allocated to achieve maximum efficiency.

The production rate of a well is usually controlled by a choke. Adjusting production rates is the most straightforward way to meet certain production targets and satisfy certain operational constraints as described later.

In some petroleum fields such as the Prudhoe Bay oil field in Alaska, USA, a production well can be connected to different flow lines that lead to different separation units. In such fields, switching a well from one flow line to another flow line can be an effective way to relieve the delivering/processing burden of one device/facility and increase the delivering/processing efficiency of the overall system. Thus, well connections were also considered as one type of decision variables in this study.

Constraints. Production operations in an oil field are usually subject to multiple capacity, safety, and economic constraints.

Capacity constraints. It is not surprising to know that oil production in some petroleum fields is constrained by the processing capacities of surface facilities, such as the gas and water processing capacities of separators, and the gas compression capacity of a central gas plant. For example, oil production in the Prudhoe Bay oil field (Barnes et al., 1990) and the Kuparuk River field in Alaska (Stoisits et al., 1992, 1994, 1999) is constrained by the gas handling capacities of surface facilities. The main reason for this is that production systems are usually designed and installed at the very early stage of reservoir development when information about the reservoir and future economics are scarce or uncertain. Thus facility capacities may not always be able to meet the production demand throughout the life of a reservoir. These constraints can be satisfied and/or fully utilized by adjusting the lift gas and/or production rates and switching well connections between flow lines.

Safety and economic constraints. For safety reasons, we may need a maximum/minimum pressure constraint at the bottom of a well or on some surface facility nodes. To operate the production system economically, we may put a maximum/minimum flow rate constraints on certain production wells or facilities. Furthermore, corrosion/erosion can lead to costly repairs. To avoid excessive corrosion/erosion, fluid velocities may have to be limited (Svedeman and Arnold, 1994, Kermani and Harrop, 1996). All these constraints can be satisfied by adjusting the production rates with chokes.

In summary, our objective was to investigate optimization approaches to guide in making operational decisions to enhance production subject to multiple capacity, safety, and economic constraints. For example the engineer may be interested in the following questions:

1. How to use chokes to control well rates?
2. How to distribute available lift-gas among specified wells?
3. How to route fluids by switching well connections to flow lines?

1.2 Approach

The problem of interest requires simultaneous allocation of production rates, lift gas rates, and well connections to flow lines. However, as discussed in Chapter 2, robust procedures for such a task are not available. Either previous investigators have addressed only a part of the problem, or ad hoc rules have been used that may lead to suboptimal operations. Hence, it was necessary to develop approaches that optimize all control variables simultaneously subject to all constraints. This dissertation investigated such approaches.

The target problem is a nonlinearly constrained optimization problem with both continuous and discrete decision variables. A two-level programming approach was developed to solve the problem. In this approach, optimization is conducted in two levels. The upper level explicitly optimizes the well connections. For each set of well connections, the upper level spawns a lower level problem to find the optimal production and lift gas rates for that set of well connections. The upper level masters the overall optimization procedure and the lower level can be viewed as a function evaluation procedure for the upper level. The advantage of this two-level programming approach is flexibility. Because the continuous and discrete variables are optimized at different levels, the approach offers us more freedom to use existing methods or develop new methods for solving the overall problem.

The lower level problem optimizes the production and lift gas rates subject to nonlinear constraints and is referred to in this study as the rate allocation problem. When there are no flow interactions among different wells or when such interaction is not significant, the performance of a well can be analyzed individually while ignoring the impact of the rest of the wells in the system. As a consequence, the rate allocation problem becomes a separable programming (SP) problem whose objective and constraint functions are sums of functions of one variable (Gill et al., 1981). The SP problem can be solved efficiently by linear optimization techniques.

The flow interaction among wells can play an important role in some rate allocation problems. In such cases, the rate allocation problem is formulated as a general nonlinear constrained optimization problem and solved by a Sequential Quadratic Programming method (Gill et al., 2002). Different formulations have been investigated. The focus is on how to handle the well chokes that control the production rates in the optimization process. The preferred formulation chooses the production rates instead of the well chokes as the decision variables and uses a set of constraints to guarantee that the optimal set of production rates is feasible for the production system.

The upper level is an integer programming problem. Two methods were investigated. The first is a heuristic method named in this study the partial enumeration method. This method is an iterative procedure that sequentially finds the best connection for one well while keeping other well connections fixed. The second method is a genetic algorithm (Goldberg, 1989). This method encodes one set of well connections into one set of binary strings, and evolves multiple sets of binary strings by means of selection, crossover, and mutation to locate the best set of well connections.

The optimization approaches for the upper level and lower level were investigated separately. The approaches can be used separately to optimize single types of operation settings such as the lift gas distribution or combined together to optimize all types of variables simultaneously.

1.3 Outline of Dissertation

General optimization techniques pertinent to this study are reviewed in Chapter 2, which introduces the concept, solution algorithms, and applications of linear programming, integer programming, and nonlinear programming. The chapter then describes the applications of optimization techniques in the development and operations of petroleum fields and sets the framework of this research.

Chapter 3 describes how the gathering system is simulated. In the oil industry, there is no generally accepted method for simulating multiphase flow in general gathering systems. The challenges lie in predicting the pressure drop across flow delivering devices, determining the phase splitting ratios at flow junctions, finding the mathematical solutions efficiently, and identifying the true physical solution from multiple mathematical solutions. This research did not address all these challenges. Rather, the study considered a simplified class of problems where the gathering system has a tree-like structure and flow directions are known. By exploiting the special properties of such systems, an efficient solution method could be investigated. It was found that care should be taken to identify the stable solution from multiple mathematical solutions. Further, an efficient method was developed to compute sensitivity coefficients for a production system. This method has useful applications in sensitivity analysis and optimization.

Chapter 4 describes how the rate allocation problem can be solved when flow interaction among wells can be ignored. This section describes how to construct the well performance information, how to formulate the rate allocation problem into a separable problem, and how to solve the separable problem by linear optimization techniques. An example is presented to demonstrate the advantages of the proposed method.

Chapter 5 describes how the rate allocation problem can be solved when flow interaction can not be ignored. This chapter discusses the appropriate formulation of the rate allocation problem under such cases. Once appropriately formulated, the problem was solved by a Sequential Quadratic Programming (SQP) method. Application examples demonstrated that the method is capable of handling rate allocation problems of varying complexities and sizes.

Chapter 6 presents the two solution methods for the upper level problem: the partial enumeration method and the genetic algorithm. Numerical experiments showed that the partial enumeration method is more efficient than the genetic algorithm. The quality of the solution by the simpler partial enumeration method was at least as good as that obtained from the genetic algorithm.

Chapter 7 discusses possible applications of multiobjective optimization techniques to production operations for a petroleum field. Often, production operations in a petroleum field are subject to multiple conflicting goals, such as maximizing the daily oil production while minimizing the water produced. Multiobjective optimization is a useful tool to help decision makers identify the best trade-offs.

Chapter 8 describes a procedure to integrate the optimization approaches with a reservoir simulator. This procedure automates the use of the approaches developed for both short-term production optimization and long-term oil recovery studies. Chapter 8 presents several such applications. One example is the use of the optimization tools in the E-Field Optimization System of Prudhoe Bay oil field, Alaska (Litvak et al., 2002).

Chapter 9 summarizes the work and recommends areas that need further study.

Chapter 2

Literature Review

Applications of optimization techniques in the upstream oil industry began in the early 1950s and have been flourishing since then. Applications have been reported for recovery processes, planning, history matching, well placement and operation, drilling, facility design and operation and so on. Optimization techniques employed in these applications cover almost all subfields in mathematical programming, such as linear programming, integer programming, and nonlinear programming. In this chapter, we first introduce concepts, solution algorithms, and applications of some major subfields in mathematical programming. Then we review their applications in areas that are pertinent to this study.

2.1 Overview of Optimization Techniques

Mathematical programming is a field born in the later 1940s (Lenstra et al., 1991). Despite its short history, mathematical programming has developed into a sophisticated field with deep specialization and great diversification. Mathematical programming encompasses subfields such as linear programming, integer programming, nonlinear programming, combinatorial optimization, stochastic programming, and so on.

Optimization problems in the most general form can be represented as

$$\min \left\{ f(\mathbf{x}) : l_i \le c_i(\mathbf{x}) \le u_i, i = 1, \ldots, m \right\} \tag{2.1}$$

where the objective function f and the constraint functions $\{c_i\}$ are functions of control variable x, and l_i and u_i are the lower and upper bounds for the ith constraint, respectively. An optimization problem can be categorized according to the type of its control variables, and objective and constraint functions.

2.1.1 Linear Programming

When the objective function f and constraint functions $\{c_i\}$ are linear functions of control variable $\mathbf{x}$, the problem described by Eq. 2.1 is a linear programming (LP) problem. Various kinds of optimization problems can be formulated as LP problems, such as transportation, production planning, resource allocation, and scheduling problems.

In 1947, Dantzig proposed the first solution method for the LP problem, the simplex algorithm. The simplex algorithm (Dantzig, 1963) finds the optimal solution by moving along the vertices of the feasible region, thus its optimal solution is always a vertex (or an extreme point) of the feasible region. In general, the simplex algorithm is very efficient. However, for some problems, the simplex method can take an very large number of iterations (Klee and Minty, 1972). Karmarkar (1984) introduced the first interior point algorithm for the LP problem. The interior point algorithm approaches the optimal solution from the interior of the feasible region. Thus the solution usually is not a vertex of the feasible region. The interior point method is very efficient both theoretically and in practice. Nowadays, LP problems with thousands or even millions of variables and constraints can be solved efficiently by both the simplex and interior point algorithm (Bertsimas and Tsitsiklis, 1997).

2.1.2 Integer Programming

When all components of the unknown $\mathbf{x}$ are discrete variables, the problem described by Eq. 2.1 becomes an integer programming (IP) problem. When some but not all components of $\mathbf{x}$ are discrete, the problem is a mixed integer programming (MIP) problem. Discrete variables are useful to model indivisibility, logical requirements, and on/off decisions. Integer programming finds its application in many real life problems, such as capital budgeting, airline crew scheduling, telecommunications, and production planning. For the production optimization problem addressed in this study, the well connections have to be modeled as discrete variables.

For linear integer programming (LIP) problems, the common solution techniques are the cutting plane method and the branch and bound method. Both methods tackle the LIP problem by solving a series of linear programming problems. The cutting plane method was proposed by Gomory (1958). The fundamental idea is to add constraints to a series of *linear programming relaxations* of the LIP problem until the optimal solution of a relaxation problem takes integer values (a linear programming relaxation is formed by allowing the integer variables in an LIP problem to take real values). In practice, the cutting plane method is not effective. The most effective technique for solving the LIP problem is the Branch and Bound method, which was first presented by Land and Doig (1960). The Branch and Bound method uses a "divide and conquer" approach to explore the set of feasible integer solutions. The method divides an optimization problem recursively into multiple subproblems. Instead of enumerating all the subproblems, the method uses bounds on the optimal objective value of a subproblem to avoid forming and solving other subproblems.

Other methods for integer programming include dynamic programming (Bellman, 1957), simulated annealing (Kirkpatrick et al., 1983), genetic algorithms (Goldberg, 1989), and many special algorithms designed for particular problems. So far, there does not exist a universal algorithm for integer programming: either the method takes a large amount of time, or it only gives an approximate solution (Bertsimas and Tsitsiklis, 1997).

2.1.3 Nonlinear Programming

Eq. 2.1 becomes a nonlinear programming problem when its objective and/or constraint functions are nonlinear. Nonlinear programming problems come in many different forms and shapes. Algorithms have been developed for individual classes of problems. It is impossible to give a thorough survey of all major optimization algorithms in a limited space. Rather, this section gives a brief description of some of the optimization methods encountered most frequently in the petroleum engineering literature. Some general references about nonlinear programming are Gill et al. (1981), Fletcher (1987), and Bertsekas (1982).

Unconstrained optimization methods. An important class of methods for unconstrained optimization problems is the so-called line search. Line search methods approach a local minimum using the following iteration scheme:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \tag{2.2}$$

where $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ are the current and next iterates, $\mathbf{p}_k$ is a *search direction* along which the function decreases, and α_k is a *step length* that ensures "sufficient" progress toward the solution.

A Newton-based line search method (usually termed as Newton's method) goes as follows. At each iteration, the method approximates the objective function $f(\mathbf{x})$ around current iterate $\mathbf{x}_k$ using a quadratic function defined as follows:

$$q_k(\mathbf{p}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{p} + \frac{1}{2}\mathbf{p}^T \nabla^2 f(\mathbf{x}_k)\mathbf{p} \tag{2.3}$$

When the Hessian matrix, $\nabla^2 f(\mathbf{x}_k)$, is positive-definite, the model function $q_k(\mathbf{p})$ has a unique minimum which can be obtained by solving the following linear system

$$\nabla^2 f(\mathbf{x}_k)\mathbf{p}_k = -\nabla f(\mathbf{x}_k) \tag{2.4}$$

The solution $\mathbf{p}_k$ of Eq. 2.4 serves as the search direction of current iteration. The step length α_k can be obtained by a linear search procedure.

Newton's method converges quadratically near a local optimum (Gill et al., 1981). The key to this exceptional convergence rate is that the method uses the curvature information of the objective function (the Hessian matrix) to compute the search direction. However, sometimes it is impossible or time-consuming to compute the Hessian matrix.

Quasi-Newton methods are based on the idea of gradually building up an approximate Hessian matrix using the gradient information collected from previous iterations. The initial Hessian approximation can be an identity matrix. In subsequent iterations, the Hessian approximation is updated by using the gradient information to mimic the behavior of the true Hessian matrix. Quasi-Newton methods are attractive for practical use because their memory and computational requirements are low while they have good convergence rate.

Constrained optimization methods. The optimum of a constrained optimization problem is characterized by a certain set of conditions. These conditions were first established by Kuhn and Tucker (1951) and they are often called the Kuhn-Tucker conditions or optimality conditions. Constrained optimization methods are developed to find a point that satisfies these conditions. The major constrained optimization methods include sequential quadratic and linear programming methods, reduced-gradient methods, and methods based on augmented Lagrangians, penalty, and barrier functions (Gill et al., 1981).

The sequential quadratic programming (SQP) methods are widely regarded (Murray, 1997) as the most effective methods for nonlinearly constrained programming (NCP) problem. The SQP methods have a structure of major and minor iterations. Each major iteration involves formulating a quadratic programming (QP) subproblem to obtain the search direction $\mathbf{p}_k$ and using a line search procedure to obtain the steplength α_k. The QP subproblem is formulated in such a way that its objective function is a *quadratic approximation* to the Lagrangian function of the original problem and its constraint functions are the linearization of the original nonlinear constraints around the current iterate. The minor iterations are used to solve a particular QP subproblem.

The sequential linearly constrained (SLC) methods (Murtagh and Saunders, 1982) are another class of methods widely used for NCP problems. Similar to the SQP methods, SLC methods also involve major and minor iterations. In contrast to SQP methods, which formulate a QP subproblem in each major iteration, the SLC methods formulate a linearly constrained subproblem whose objective function is a *general approximation* to the Lagrangian function of the original problem. In general, an SLC method tends to require fewer major iterations but more evaluations of the problem functions than an SQP method (Murray, 2000).

The barrier methods solve a nonlinearly constrained problem by solving a sequence of unconstrained optimization problems. A barrier method starts with a feasible point and creates a sequence of unconstrained problems whose successive minimum stays feasible and converges to a minimum of the constrained problem. To achieve this, the objective function of the unconstrained problem is constructed by adding a barrier function to the original objective function of the constrained problem. The barrier function is a modification of the constraint functions and becomes infinitely large when the iterate approaches the boundary of the feasible region. The barrier method was first established in the 1960s for general nonlinearly constrained optimization problems (Fiacco and McCormick, 1968). Recently the method has been developed further to address optimization problems whose objective and constraint functions are convex functions (Boyd and Vandenberghe, 2001).

Separable programming is a special class of nonlinearly constrained optimization problems whose objective and constraint functions are sums of functions of one variable. Separable programming problems are usually solved by linear programming techniques (Hillier and Lieberman, 2001).

Direct optimization methods refer to optimization methods that do not require derivatives. When selecting an optimization method, a general rule is to choose a method utilizing as much derivative information as possible (Gill et al., 1981). However, when problem functions are not smooth or the derivatives are too expensive to compute, one may choose a direct optimization method. While many direct optimization methods have been proposed, here we only review a few that are popular in petroleum engineering applications. A thorough review of the direct optimization methods is given by Powell (1998).

The polytope method was first established by Spendley et al. (1962). The method involves manipulation of $n+1$ linearly independent data points, which can be viewed as the vertices of a polytope, for a problem with dimension n. At each iteration, a new polytope is generated by replacing the worst data point with a newly generated data point. The data point is evaluated based on its objective function value. While the polytope method is the most used method for unconstrained optimization in practice, some severe cases of failure have been observed (Powell, 1998).

Genetic algorithms (GAs) are heuristic optimization algorithms introduced by Holland (1975). GAs employ the idea of natural selection and genetics in the process of searching for the global optimum of a problem. In GAs, possible solutions are encoded as chromosomes and modified by means of selection, crossover, and mutation. GAs have several advantages over other optimization algorithms. Firstly, GAs are very general optimization methods in that they handle both the discrete and continuous variables naturally and require no domain knowledge of the optimization problem. Secondly, GAs are stochastic search methods capable of avoiding local optima. Nevertheless, GAs have certain disadvantages. First of all, convergence can be slow, especially at the later stages of optimization. Secondly, GAs have been used primarily for unconstrained problems. Although nonlinear constraints can be handled through penalty functions (Powell and Sholnick, 1993; Michalewicz, 1996; Rasheed, 1998), the performance of such treatments has not been impressive.

2.2 Applications of Optimization Techniques to Petroleum Fields

Optimization techniques have been applied to virtually all aspects of the oil industry. In this section, we review applications of optimization techniques in rate allocation, production system design and operations, and reservoir development and management.

2.2.1 Lift Gas and Production Rate Allocation

Lift gas allocation. The mechanism of gas-lift is elaborate. An appropriate amount of lift gas increases the oil rate, while excessive lift gas injection will reduce the oil rate. To determine the optimal lift gas rate, the usual practice is to allocate the lift gas to a well according to a gas-lift performance curve (Nishikiori et al., 1989). A gas-lift performance curve is a plot of oil rate versus lift gas rate for a gas-lift well. When the gas supply is unlimited, the optimal lift gas rate is the one corresponding to the maximum oil rate on the performance curve. When the gas supply is limited, the lift gas is usually allocated using some optimization algorithm. The earliest lift gas allocation method is a simple heuristic method based on the concept of equal-slope, which states that at the optimal solution, the slope of the gas-lift performance curves should be equal for all wells (Hong, 1975; Kanu et al., 1981). In addition to the equal-slope method, Nishikiori et al. (1989)

also applied a formal optimization algorithm, a Quasi-Newton method, to the lift gas allocation problem. Both the equal-slope method and the Quasi-Newton method rely on derivative information to verify optimality, and therefore tend to get trapped in local optima. This limitation can be serious when some gas-lift performance curves are not concave. Buitrago et al. (1996) addressed this problem by proposing a stochastic algorithm that uses a heuristic method to calculate the descent direction. However, as suggested by their results, their method is not good at handling constraints.

Gas-lift optimization based on performance curves is simple and easy to implement. However, this approach ignores flow interactions among wells in the optimization process. Dutta-Roy and Kattapuram (1997) analyzed a gas-lift optimization problem with two wells sharing a common flow line, and pointed out that when flow interactions are significant, nonlinear optimization tools are needed to obtain satisfactory results. Dutta-Roy and Kattapuram (1997) applied a SQP method to a linearly constrained gas-lift optimization problem with 13 wells. Results showed that the SQP method does perform better than methods based on performance curves.

General rate allocation. This problem refers to allocating production rates and lift gas rate of single wells or the total production rates of reservoirs (in multireservoir cases) to achieve certain operational goals. Linear programming seems to be the most popular optimization algorithms for this kind of problems. Attra et al. (1961) applied linear programming to maximize daily income from a multireservoir case subject to well producing capacities, reservoir injection requirements, compressor capacity limits, gas-lift requirements, and sale contracts. Lo and Holden (1992) proposed a linear programming model to maximize daily oil rate by allocating well rates subject to multiple flow rate constraints. Fang and Lo (1996) applied LP to allocate both lift gas and production rates. They first approximate gas-lift performance curves by a piecewise linear curve, and then formulate the rate allocation problem as an LP problem. The LP method is very efficient for the rate allocation problem, however, suffers from the fact that the objective and constraint functions in an LP problem have to be linear. Due to this limitation, the gas-lift performance curves in Fang and Lo's method have to be concave.

Options in commercial simulators. In some commercial reservoir simulators (GeoQuest, 2000; Landmark, 2001), rate constraints on facilities are handled sequentially by ad hoc rules. For example, first, gas constraints are considered at the lowest predictive well management (PWM) level. Ad hoc rules based on the gas-oil ratio of production wells are applied to scale well rates and meet the gas constraint. Then, water, liquid, and oil rate constraints at the lowest PWM level are handled sequentially and the procedure is repeated for the next PWM level, etc. Furthermore, gas-lift optimization is usually considered separately from well rate optimization using, for example, the following approach. First, lift-gas rates are determined for specified wells based on their gas-lift performance curves or other information. Then, these lift-gas rates are scaled using some rules to match constraints on the total amount of available gas. A number of such procedures are available in commercial reservoir simulators. In some cases, these procedures may not yield the optimal solution. Also, using such approaches, it is difficult to determine how much of the gas handling capacity should be used for gas-lift and what portion should be allocated for other purposes.

Field cases. Rate allocation problems are encountered frequently in mature fields where production facilities cannot meet the field demand. For example, oil production in the Prudhoe Bay oil field in Alaska is constrained by the gas handling capacity of the separation units and the central gas plant. To utilize existing facilities fully, Barnes et al. (1990) developed the Western Production Optimization Model (WPOM) for the Prudhoe Bay field. This model allocates the oil rate and gas rate to surface facilities and wells in a top-bottom manner based on the “incremental GOR” concept that produces the next incremental barrel of oil with the lowest GOR. Litvak et al. (1997) built an integrated reservoir and gathering system model of the Prudhoe Bay field, and employed some heuristic methods to optimize well connections to manifolds. Gas-lift rates were allocated based on gas-lift tables of gas-liquid ratio, liquid well rate and water cut. The optimization algorithms in these applications are heuristic and they may lead to suboptimal solutions.

The oil production in Kuparuk River oil field in Alaska is also constrained by the gas processing capacities of surface facilities. Stoisits et al. (1992, 1994) applied a neural network model to determine the optimal allocation of lift gas to wells subject to multiple gas constraints. The neural network was trained by results generated from single well simulation. Stoisits et al. (1999) addressed the production optimization problem at Kuparuk River field using a genetic algorithm. In their study, the decision variables were the status of the production wells (close or open) and incremental GOR for drill sites. A neural network model was developed to simulate the pressure drop through the surface pipeline network. In all these applications (Stoisits et al., 1992, 1994, 1999) the neural network is the central tool for speeding up the simulation/optimization process. However, it is time-consuming to train the neural network. Also, the accuracy of the neural network, which depends on the accuracy of the training tools, the behavior of the simulated system, and many other factors, is hard to control.

In summary, the general rate allocation problem has not been fully resolved yet, especially when dealing with gigantic fields like the Prudhoe Bay and fields where flow interactions cannot be ignored.

2.2.2 Optimization of Production System Design and Operations

Traditionally, optimization of production system design and operation in a petroleum field has been performed by nodal analysis combined with trial and error (Beggs, 1991). For example, by holding all other parameters fixed, a single variable is varied to see which value of this variable gives the optimal objective function value. For multiple decision variables, such a procedure fails because of the large number of function evaluations required to cover the search space.

Carroll and Horne (1992) identified the need for formal optimization algorithms in such applications. Carroll and Horne applied a Newton-type optimization algorithm and a polytope method to optimize the tubing diameter and separator pressure of a *single-well* system and successfully demonstrated the usefulness of multivariate optimization techniques. Ravindran (1992) followed the work of Carroll and Horne and allowed the decision variables to vary with time. Fujii and Horne (1995) applied a genetic algorithm to optimize the design and operations of a *multiwell* production system. Fujii and Horne noted that multiple solutions may exist for the multiphase network flow problem. However, no procedures were developed to select the legitimate solution. Palke and

Horne (1997a) followed a similar line and applied a Newton-type algorithm, a polytope method, and a genetic algorithm to a compositional model of a single-well system. The control variables included tubing diameter, separator pressure, and volume of gas injected. A common observation of all these studies was as follows. When the tubing diameter is involved, the surface of the objective function is highly irregular and gradient-based algorithms often fail to solve the optimization problem. Palke and Horne (1997a) concluded that genetic algorithms are the most robust algorithms for such cases.

NETOPT (SIMSCI, 1999) is a commercial network optimizer that combines a general multiphase network simulator with a SQP method. NETOPT is a very general tool, and can be interfaced with Eclipse (GeoQuest, 2000), a commercial reservoir simulator, and be used to optimize various production design and operational problems. Barua et al. (1997) discussed the general usage of NETOPT in the optimization of production operations. Dutta-Roy et al. (1997) applied NETOPT to analyze compressor installation cost and operating strategy required to meet the lifetime production goals of a gas-field. Heiba et al. (1997) applied NETOPT to devise optimal strategies for a cyclic steam stimulation project. Because NETOPT employs SQP, a derivative-based optimization algorithm, it locates only local optima and accepts only continuous decision variables. The performance on individual optimization problems is not guaranteed and care has to be taken to ensure reasonable results.

Zhang and Zhu (1996) minimized the total cost of a gas distribution network by adjusting the diameters of pipelines of a fixed network layout. In their study, Zhang and Zhu treated the optimization problem as a bilevel programming problem, and then simplified the lower level and upper level problem by exploring their special properties.

2.2.3 Optimization of Reservoir Development and Planning

Reservoir development studies, such as drilling scheduling, well placement, and production rate scheduling, have been an active area for optimization. The optimization tools and simulation models employed in reservoir development studies have been evolving.

Early optimization studies of reservoir development featured simple reservoir models and linear programming techniques. Aronofsky and Lee (1958) built a linear programming model to maximize profit by scheduling production from multiple homogeneous reservoirs. Bohannon (1970) presented a linear programming model to find the optimum 15-year development plan for a multireservoir pipeline system. The main variables considered in their formulation were annual production rate from each reservoir, number of wells to be drilled each year, and timing of major capital investments such as initiating secondary recovery projects and expanding pipeline facilities. Xiao et al. (1998) proposed a multiobjective linear programming model for an oilfield injection recovery system. The main goal was to produce most oil with least investment under the limitations of the available natural resource, equipment, and manpower, etc. Aronofsky (1983) reviewed the application of linear programming in oil and gas development.

Linear programming requires the objective and constraint functions to be linear. In order to optimize the production system by linear programming, the nonlinear production system has to be approximated by a system of linear functions. This greatly limits the application of sophisticated reservoir simulators in the optimization of reservoir developments. As the computing power increases and simulation techniques advance, researchers are attempting to optimize reservoir development with more and more complex reservoir models and sophisticated optimization techniques.

Various gradient-based optimization methods have been investigated. McFarland et al. (1984) applied a generalized reduced gradient nonlinear programming method to maximize present value of profits from a reservoir by deciding how many wells to drill in each time period, the production rates, abandonment time, and platform size. They used tank-type reservoir models to describe reservoir dynamics. Lasdon et al. (1986) studied the problem of determining the optimum flows from a set of existing wells in each of several time periods subject to certain demand schedules and operating constraints, and combined numerical optimization techniques with a single phase, two-dimensional reservoir simulator. Zakirov et al. (1996) presented a method for optimizing net present value of oil production with respect to well production rates as functions of time over the whole life of the field. What distinguishes Zakirov et al.'s method from previous literature is that they used optimal control theory to obtain the gradients of the objective function in an efficient way.

Most reservoir development problems have irregular surfaces for the objective function. Therefore, stochastic optimization algorithms such as genetic algorithms have gained popularity in such applications. Palke and Horne (1997b) used a genetic algorithm to estimate the cost of uncertainty in reservoir data. In general, genetic algorithms are slow. To speed up the search process, Bittencourt and Horne (1997) hybridized a genetic algorithm with a polytope method, and applied the hybridized algorithm to a reservoir development problem in which the decision variables are well locations and the objective function is based on an economic model. Güyagüler et al. (2000) and Güyagüler and Horne (2001) also looked at well placement problems by adapting the hybridized genetic algorithm from Bittencourt and Horne (1997). In addition, they used a kriging method as a proxy for the function evaluations to reduce the time for function evaluations. In the optimization of reservoir development, a major concern is that we never possess true and complete information about the reservoir. Güyagüler (2002) used the utility theory to quantify the uncertainty in reservoir developments. Yeten et al. (2002) investigated the problem of placing a multilateral well. The decision variables include the location of the well, the number of laterals and the trajectory of the laterals. Yeten et al. (2002) employed a genetic algorithm as the optimization driver and developed a hill climber and a neural network model to speed up the optimization process. Results demonstrated that the appropriate combinations of these tools can greatly speed up the optimization process.

2.3 Concluding Remarks

Optimization techniques have been applied to many aspects of the oil industry. Due to the restrictions in computing power and simulation technologies, early applications favored simple methods such as linear programming and heuristic methods. With the increase of computing power and advances in simulation techniques, researchers and practitioners are attempting to optimize reservoir development and operations with more and more complex models and sophisticated optimization techniques. Frequently, the optimization problems in the oil industry do not have smooth objective functions. Therefore, stochastic optimization algorithms such as genetic algorithms have been the major optimization drivers in recent applications. Because reservoir simulation is a computationally intensive process and the true reservoir information is unknown, it is worthwhile to investigate ways of speeding up the simulation processes and combining uncertainty with optimization.

The type of production optimization problem addressed in this study has not been fully resolved. While extensive studies have been reported for unconstrained or linearly constrained gas-lift optimization problems (Kanu et al., 1981; Nishikiori et al., 1989; Buitrago et al., 1996; Dutta-Roy and Kattapuram, 1997), only few have been reported for optimizing both the production rate and gas-lift rate (Fang and Lo, 1996; Stoisits et al., 1999). The method proposed by Fang and Lo (1996) ignores flow interactions among wells and implicitly assumes the gas-lift performance curves are concave and the well water cut and the gas oil ratio (GOR) are constant with varying oil rates. The method proposed by Stoisits et al. (1999) determines the optimum well shut down list and a drill site IGOR (incremental gas oil ratio) list, not the production rates and lift gas rate for individual wells. Further, no previous publication has addressed the problem of optimizing production rates, lift gas rates, and well connections simultaneously subject to nonlinear constraints. Therefore, it is necessary to explore new approaches to solve this important class of production optimization problems.

This research extended the method proposed by Fang and Lo (1996) for the rate allocation problem to cases where the gas-lift performance curves can have any shape and the well water cut and GOR can vary with oil rate (Chapter 4). A SQP method was also applied to the rate allocation problem to account for flow interactions (Chapter 5). A two-level programming approach was developed to optimize all control variables simultaneously (Chapter 6). Multiobjective optimization was introduced to handle conflicting operational goals in a petroleum field (Chapter 7). Finally many of the optimization approaches developed in this study were integrated with a reservoir simulator so that the impact of the reservoir conditions could be taken into account in the optimization process (Chapter 8). Some examples are presented throughout the dissertation to demonstrate the advantages of the methods.

Chapter 3

Simulation of Gathering Systems

3.1 Introduction

A typical oil field contains a gathering system, a fluid distribution network, and an injection network. The gathering system collects the fluids from production wells and delivers them into separation units. The separated fluids are then distributed to different destinations for storage, sale, disposal, injection, or further processing. The injection network is used to inject fluids into the reservoir for enhanced oil recovery projects or for fluid disposal/storage. The multiphase flow problems in the gathering, distribution and injection network belong to network flow problems that have special properties. For the production optimization problem considered in this work, a model is required to simulate the multiphase flow in the gathering system. The distribution and injection networks are ignored.

The network problem with single-phase flow is relatively easy to solve and has been studied extensively in both the civil engineering and the petroleum engineering literature. There are two major solution approaches. Both approaches formulate the network problem according to the requirements of mass balance and Kirchoff's law. The first approach is to formulate the network problem as a set of system equations and solve the system equations by the Newton-Rapshon method or one of its variations (Donachie, 1973; Epp and Fowler, 1970; Liu, 1969). The second approach is to formulate the network problem as an optimization problem and solve it using appropriate optimization algorithms (Hall, 1976; Collins et al. 1978).

The multiphase network problem is also formulated according to the requirement of mass balance and Kirchoff's law. However, the multiphase network problem is much more difficult than the single phase network problem for the following reasons: Firstly, the system equations are more difficult to construct, because we need to determine the fluid compositions in a link before the pressure loss along the link can be evaluated. This is not easy in complex systems where the flow directions are unknown and phase splitting ratios at junctions are hard to determine. Secondly, the Newton-Raphson method may not converge or converge slowly due to the complex nature of the multiphase flow. Finally, mathematically, multiple solutions exist for the multiphase network problem and not all of them are physical. How to find the true solutions systematically is an unresolved problem. To date, only a few researchers have reported solution methods for multiphase network problems. Schiozer (1994) reported a solution method with pressure continuity requirements for the system equations for networks with tree-like structures. However, Byer (2000) showed that Schiozer's method can result in convergence problems due to incorrect Jacobian terms. Byer (2000) proposed an alternative formulation with mass balance requirements as the system equations for gathering systems with tree-like structures. Results show that even for a simple system with two wells, the solution method can take more than ten Newton iterations to converge unless a good initial guess is available. Fujii (1993) formulated the multiphase flow problem in a tree-like structure as an optimization problem and solved the problem using a Gauss-Newton method. Several commercial software packages such as Eclipse (GeoQuest, 2000), VIP (Landmark, 2001), and PIPEPHASE (SIMSCI, 2001) have options for the simulation of multiphase network problems, however, their solution algorithms are not in the public domain.

In this study, we consider gathering systems with tree-like structures. The gathering system can include such components as wells, tubing strings, chokes, and pipes. The fluids in the system can include oil, gas, and water phases. Fluid properties are represented by the black-oil model. This chapter first describes simulation models for the individual components. The study investigated efficient procedures to solve the network problem and to compute derivatives that are required in history matching, optimization, and sensitivity analysis. An example is presented to demonstrate the performance of the solution procedures that were found most suitable.

3.2 Well Model

In reservoir simulation, a well model is used to relate the flow rate of a well with its bottomhole pressure and the reservoir pressure in the vicinity of the well (such as the grid blocks in which the well is completed). The well model accounts for the geometric characteristics of the well, the reservoir properties in the vicinity of the well, and the flow behavior in the wellbore.

This study considered a vertical well completed in a single grid block and used a basic well model proposed by Peaceman (1978). This model assumes steady-state radial flow and can be expressed as

$$q_c^w = \sum_p WI\,\lambda_{c,p}\left(p_p - p^w\right) \qquad (3.1)$$

where q_c^w denotes the flow rate of component c, WI denotes the well index, $\lambda_{c,p}$ represents the well mobility for component c in phase p, p_p denotes the reservoir pressure of phase p, and p^w denotes the well pressure. Expressions for WI and $\lambda_{c,p}$ can be found in Aziz and Durlofsky (2002).

3.3 Pipe Model

In this work a homogenous model with slip was used to model multiphase pipe flow. In the homogeneous model, different phases are lumped into one single pseudophase so that pressure drop equations for single-phase flow can be utilized.

For the pseudophase, mixture density ρ_m is based on *in situ* phase fractions:

$$\rho_m = \rho_l E_l + \rho_g E_g \qquad (3.2)$$

where E_l and E_g are the liquid and gas phase *in situ* fractions, respectively, and

$$E_l + E_g = 1 \tag{3.3}$$

Computation of gas volume fraction E_g is presented in Section 3.3.2. Ouyang (1998) suggested that the mixture velocity should be defined as

$$U_m = \frac{\rho_l}{\rho_m} U_{sl} + \frac{\rho_g}{\rho_m} U_{sg} \tag{3.4}$$

where U_{sl} and U_{sg} are the superficial velocities of liquid and gas phases, respectively.

With these properties defined, we can proceed to compute the pressure loss for multiphase flow in pipes.

3.3.1 Pressure Gradient Equation

Given a flow rate and the pressure at one end of a pipe, the pressure loss (or gain) in the pipe can be computed by integrating the pressure gradient along the pipe. Consider gas/liquid two-phase flow in a pipe without perforations, the total pressure gradients along the pipe can be decomposed into three parts

$$\frac{dp}{dx} = \left(\frac{dp}{dx}\right)_f + \left(\frac{dp}{dx}\right)_g + \left(\frac{dp}{dx}\right)_a \tag{3.5}$$

where

- $\left(\frac{dp}{dx}\right)_f$ is the pressure gradient due to wall friction,
- $\left(\frac{dp}{dx}\right)_g$ is the pressure gradient due to gravity, and
- $\left(\frac{dp}{dx}\right)_a$ is the pressure gradient due to acceleration.

Based on mass balance and momentum balance, it can be shown that (Brill and Beggs, 1998)

$$\left(\frac{dp}{dx}\right)_g = -g\rho_m \sin\theta \tag{3.6}$$

$$\left(\frac{dp}{dx}\right)_f = -\frac{\tau_w S}{A} \tag{3.7}$$

where θ is the pipe inclination angle, τ_w is the wall friction shear stress, S is the pipe perimeter, and A is the pipe area. The wall friction shear stress is computed as

$$\tau_w = \frac{1}{2} f_m \rho_m U_m^2 \tag{3.8}$$

where the Fanning friction factor f_m can be evaluated by the Colebrook equation (Colebrook, 1939) or any of its approximation forms based on the relative pipe roughness $\frac{\varepsilon}{d}$ and the mixture Reynolds number

$$R_{em} = \frac{\rho_m U_m d}{\mu_m} \tag{3.9}$$

where d is the pipe diameter and μ_m is the mixture viscosity, which can be computed from phase viscosities by (Brill and Beggs, 1998)

$$\mu_m = \mu_l E_l + \mu_g E_g \tag{3.10}$$

The acceleration pressure loss across a pipe segment depends on the difference between the velocity heads of the flowing mixture at the two ends of the segment. $\left(\frac{dp}{dx}\right)_a$ is usually small (Brill and Beggs, 1998) and it is ignored in this work.

3.3.2 Liquid Holdup Evaluation

A drift flux model (DFM) (GeoQuest, 2000; Chen, 2001) was used in this work to calculate the liquid holdup. The DFM is used because of its continuity. Unlike mechanistic models (Aziz and Petalas, 1994; Duns and Ros, 1963; Taitel and Dukler, 1976; Xiao et al., 1990) or correlations (Beggs and Brill, 1973), the DFM covers the complete range of flowing conditions without any first order discontinuities. This property makes the DFM more suitable for simulation and optimization studies than mechanistic models.

In general, in all but downward flow, the gas phase has a tendency to flow faster than the liquid phase. The DFM correlates the *in situ* gas velocity U_g with the superficial mixture velocity U_{sm} ($U_{sm} = U_{sl} + U_{sg}$) using two parameters C_0 and U_d

$$U_g = C_0 U_{sm} + U_d \tag{3.11}$$

where C_0 is a profile parameter or distribution parameter that accounts for the effect of velocity and gas concentration profiles. U_d is the drift velocity of the gas, which accounts for the local velocity differences between the two phases. Both C_0 and U_d are functions of the gas volumetric fraction E_g

$$C_0 = C_0\left(E_g\right) \tag{3.12}$$

$$U_d = U_d\left(E_g\right) \tag{3.13}$$

The specific expression of Eqs. 3.12 and 3.13 can be found in GeoQuest (2000).

The superficial gas velocity and *in situ* gas velocity are related by:

$$U_{sg} = E_g U_g \tag{3.14}$$

Substituting Eqs. 3.11, 3.12, and 3.13 into Eq. 3.14 we obtain a nonlinear equation with E_g as the unknown:

$$f\left(E_g\right) = U_{sg} - E_g\left(C_0 U_{sm} + U_d\right) = 0 \tag{3.15}$$

Solving Eq. 3.15 we can obtain E_g, thus the liquid holdup E_l.

Finally we mention that in certain cases, we found that the Newton-Raphson method (Press et al., 1992) fails to converge for Eq. 3.15. The Van Wijingaarden-Dekker-Brent method (or Brent's method) (Press et al., 1992), a method that combines root bracketing, bisection, and inverse quadratic interpolation, proved to be both efficient and robust for solving Eq. 3.15.

3.3.3 Upwards Two-Phase Flow Behavior

Under certain conditions, for upwards two-phase flow, at low flow rates, increasing the flow rate decreases the pressure drop across a pipe. Beyond a certain critical flow rate, increasing the flow rate increases the pressure drop across the pipe. This phenomenon is demonstrated in Figure 3.1. Table 3.1 presents the problem data used to generate Figure 3.1.

The phenomenon described can be explained as follows. As shown in Eq. 3.5, the pressure drop across a pipe segment is dominated by the gravity force and friction force. The lower the flow rate, the smaller the frictional force. However, at low flow rates, liquid phase tends to holdup (be held back) in the pipe causing bigger pressure drop due to larger mixture density. Hence, at low flow rates, the gravity force dominates. As the flow rate increases, the gravity force decreases due to reduced liquid holdup, the friction forces increases but not large enough to compensate fully for the gravity force reduction. Therefore, the overall pressure drop decreases as the flow rate increases. Since the increase of frictional force is proportional to the square of the mixture velocity, beyond a critical point, the friction force starts to dominant and the overall pressure drop increases as the flow rate increases. Consequently, the typical pressure drop of multiphase flow across a pipe demonstrates the behavior shown in Figure 3.1. A consequence of this flow behavior is that multiple solutions may exist for the multiphase network flow problem. This is discussed further in Section 3.5.3.

Table 3.1: Slip model problem data.

Tubing Parameter		**Fluid Property**		**Flow Conditions**	
Diameter	3.5 in.	**Oil Density at Standard Condition**	30° API	**Downstream Pressure**	1000 psia
Roughness	0.0001 in.	**Specific Gas Gravity**	0.8	**Downstream Temperature**	110 °F
Vertical Depth	8400 ft	**Water Density at Standard Condition**	62.37 lbm/cft	**Upstream Temperature**	160 °F
		Gas Oil Ratio	1000 SCF/STB		
		Water Oil Ratio	0.6		

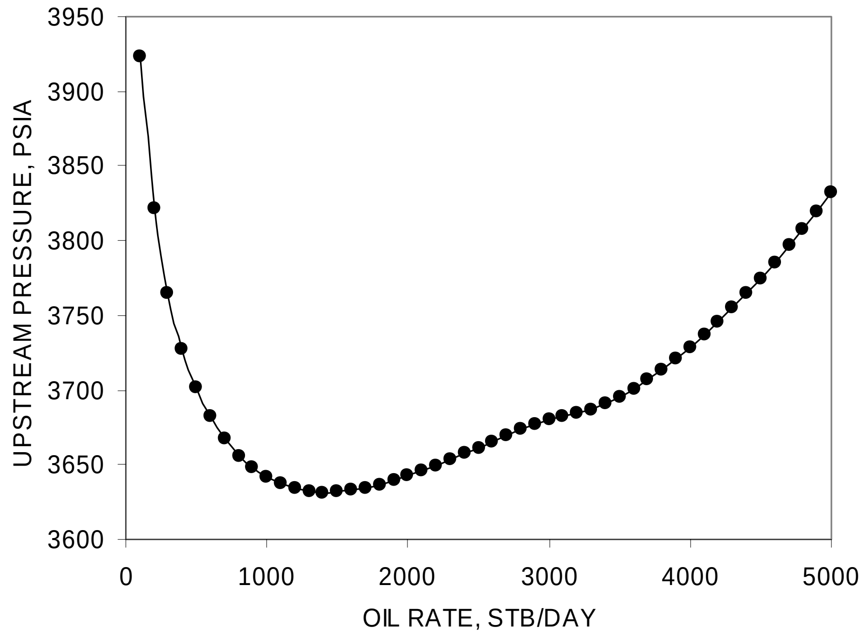


Figure 3.1: Upstream pressure in a vertical well tubing string with a fixed downstream (wellhead) pressure for different flow rates.

3.4 Choke Model

A good choke model is essential for accurately modeling multiphase flow in the gathering system. The multiphase choke model developed by Sachdeva et al. (1986) has a smooth transition between critical and subcritical flow and was used in this work. The assumptions made when building this model are:

- incompressible liquid phase,
- equal phase velocities at the choke throat,
- no mass transfer between phases,
- one-dimensional flow,
- accelerational pressure drop term dominates and the friction term is ignored,
- no distinction between free gas and solution gas, and
- liquid properties are obtained from the weighted average properties of oil and water.

The Sachdeva model can be expressed in the following context: given an upstream pressure p_1 and a downstream pressure p_2, determine the liquid flow rate across the choke.

The first step of the Sachdeva model is to determine the boundary of critical and subcritical flow. Given an upstream pressure p_1, the critical ratio of downstream to upstream pressure is determined by the following equation:

$$y_c = \left(\frac{\frac{k}{k-1} + \frac{(1-x_1)\rho_{g1}(1-y_c)}{x_1\rho_l}}{\frac{k}{k-1} + \frac{n}{2} + \frac{n(1-x_1)\rho_{g2}}{x_1\rho_l} + \frac{n}{2}\left[\frac{n(1-x_1)\rho_{g2}}{x_1\rho_l}\right]^2} \right)^{\frac{k}{k-1}} \tag{3.16}$$

$$k = C_p / C_v \tag{3.17}$$

$$n = 1 + \frac{x_1(C_p - C_v)}{x_1 C_v + (1-x_1)C_p} \tag{3.18}$$

where y_c is the critical ratio of downstream pressure to upstream pressure, k is the ratio of specific heat, C_p is the specific heat capacity at constant pressure, C_v is the specific heat capacity at constant volume, x is the mass fraction of gas, n is the polytropic exponent of gas, and the subscripts 1 and 2 represent upstream and downstream conditions, respectively. Eq. 3.16 is a nonlinear equation with y_c as the unknown. Brent's method (Press et al., 1992) is suitable for solving Eq. 3.16.

In the field units, the liquid rate across the choke is given as

$$q_l = \frac{0.525 C_d d^2}{C_{m2}} \left(p_1 \rho_{m2}^2 \left[\frac{(1-x_1)(1-y)}{\rho_{l1}} \frac{x_1 k \left(1 - y^{\frac{k-1}{k}}\right)}{\rho_{g1}(k-1)} \right] \right)^{0.5} \tag{3.19}$$

and

$$y = \max\left(\frac{p_2}{p_1}, y_c \right) \tag{3.20}$$

$$\rho_{m2} = \left[\frac{x_1}{\rho_{g1} y^{1/k}} + \frac{1-x_1}{\rho_{l1}} \right]^{-1} \tag{3.21}$$

$$C_{m2} = 8.84 \times 10^{-7} \gamma_g (GLR - f_o R_s) + 6.5 \times 10^{-5} (f_o \rho_o B_o + f_w \rho_w B_w) \tag{3.22}$$

where d is the choke inside diameter in inches, C_d is the discharge coefficient, f_o and f_w are the flowing fractions of oil and water, B is the formation volume factor, γ is the specific gravity, R_s is the solution gas-oil ratio, and GLR is the gas liquid ratio. Fluid properties used to calculate C_{m2} are evaluated at the downstream conditions.

Eq. 3.20 states that if the actual value of $y_{actual} = p_2 / p_1$ is greater than y_c, the flow is subcritical, the actual p_2 is used as the downstream pressure in Eqs. 3.21 and 3.22; if y_{actual} is less than y_c, the flow is critical, the value of $y_c p_1$ is used as the downstream pressure in Eqs. 3.21 and 3.22.

When the flow across a choke is critical, the downstream pressure can not be determined uniquely. Therefore, when simulating the network flow, we formulated the system equations in such a way that we only need to face the following problem: given a downstream pressure and a flow rate, compute the upstream pressure of the choke. This problem has a unique solution and can be obtained using appropriate root finding procedures. Computational experience showed that bisection (Press et al., 1992) and Brent's method (Press et al., 1992) work well but the Newton-Raphson often fails to converge.

3.5 Network Simulation

We considered a gathering system with a forest-like structure with multiple production trees. A production tree includes wells, links, and nodes. A well refers to a wellbore and its surrounding reservoir conditions (grid blocks in which the well is completed). Links refer to any device or facility across which pressure changes. A link can be a tubing string, a choke, or a pipeline. A node represents a flow junction or the terminal point of a link. The top of the production tree has a fixed pressure and usually represents an inlet to a separator.

In this section, we first formulate the governing equations of the network flow problem. Then we describe a solution procedure. Finally we discuss possible outcomes of the solution procedure.

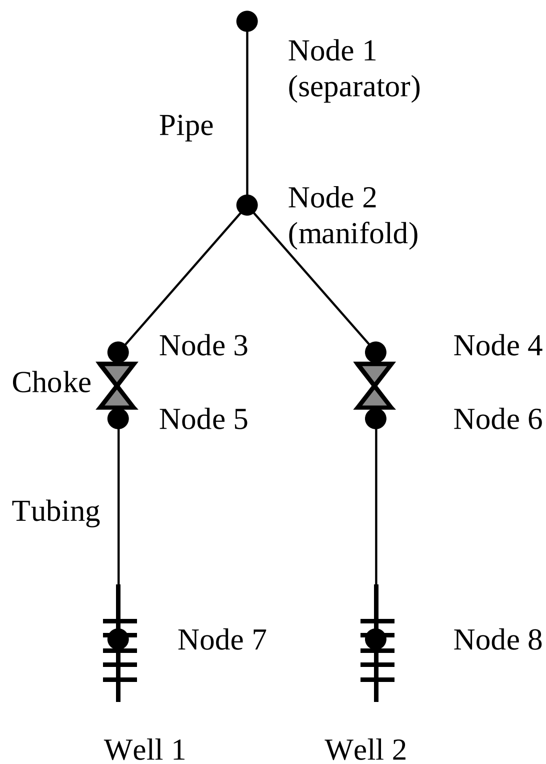


Figure 3.2: A simple gathering system.

Example problem. We will use an example problem to facilitate our discussion. In the example problem, two gas-lifted wells are connected to a common manifold, then they are connected to a separator through a common pipeline. The separator is operating at a fixed pressure. The flow rate of each well is controlled by a choke between the tubing string and the manifold. The tubing strings, chokes, and pipelines are modeled as links. The junctions of two or more links are modeled as nodes. Figure 3.2 illustrates the gathering system of this problem.

3.5.1 Governing Equations

The multiphase flow in a gathering system is described by mass conservation and Kirchoff's law. The mass conservation requires that every node should have zero net flow rate unless it is a boundary node. Kirchoff's law requires that the pressure for a node should be the same no matter from which path it is computed.

We choose the oil rates of wells as the independent variables and demonstrate how to construct the governing equations for multiphase flow in the system of this example problem.

Given the oil rate of a well, we obtain the water and gas flow rates of that well using appropriate well models. Because the gathering system has a tree-like structure and the fluid flows from the bottom up through the production tree, according to mass conservation, the flow rate in every node can be determined by summing up all the flow streams entering that node:

$$q_{p,i}^{n} = \sum_{j \in \Omega_i^w} q_{p,j}^{w}, \; p = o, g, w \tag{3.23}$$

where $q_{p,i}^{n}$ denotes the flow rate of phase p in node i, $q_{p,j}^{w}$ denotes the flow rate of phase p for well j (phase g includes both the formation gas and lift-gas, superscript n and w denote nodes and wells, respectively), and Ω_i^w denotes the set of wells whose flow streams enter node i.

Because of the tree-like structure of the gathering system, there is a unique path connecting the separator with any well. Suppose node j receives fluids only from well j. We can calculate the pressure of node j from two different pressure traverse calculations. The first calculation is from the reservoir to node j. Denote the pressure for node j obtained from this calculation as p_j^r. The second calculation is from the separator (we assume the separator is operating at a fixed pressure) to node j along the path connecting the separator and node j. Denote the pressure for node j obtained from this calculation as p_j^s. Node j is called the solution node for well j. We choose a solution node for every well and perform the described pressure traverse calculations. According to Kirchoff's law, the feasible set of oil flow rates is governed by:

$$p_j^r - p_j^s = 0, \; j \in \Omega_s^n, \; \left|\Omega_s^n\right| = n_w \tag{3.24}$$

where Ω_s^n denotes the set of solution nodes, n_w denotes the number of production wells.

In principle, for a particular well, we can choose its solution node as any node that receives fluids solely from that well. For instance, in the example problem (see Figure

3.2), we can choose either node 3, 5, or 7 as the solution node for well 1, node 4, 6, or 8 as the solution node for well 2. In practice, we always chose the wellbore bottomholes, i.e., node 7 and 8, as the solution nodes. This choice has certain computational advantages. For instance, when computing the pressure drop across a surface choke, we always compute its upstream pressure based on its downstream pressure and the flow rate across it. This guarantees a unique upstream pressure no matter the flow through the choke is critical or subcritical.

Eq. 3.24 has n_w independent variables and n_w equations, forming a closed system of equations that can be solved by the Newton-Raphson method.

3.5.2 The Newton-Raphson Method

Let the oil flow rate of each well, $q_{o,i}$, be the primary variable, and define

$$\mathbf{f}(\mathbf{q_o}) = \mathbf{p}^r(\mathbf{q_o}) - \mathbf{p}^s(\mathbf{q_o}) \tag{3.25}$$

Then Eq. 3.24 can be solved by the Newton-Raphson method with the following iterative scheme:

$$\mathbf{J}(\mathbf{q}_o^v)(\mathbf{q}_o^{v+1} - \mathbf{q}_o^v) = -\mathbf{f}(\mathbf{q}_o^v) \tag{3.26}$$

where the Jacobian matrix $\mathbf{J}(\mathbf{q}_o^v)$ is calculated as

$$\mathbf{J} = \frac{\partial \mathbf{p}^r}{\partial \mathbf{q}_o} - \frac{\partial \mathbf{p}^s}{\partial \mathbf{q}_o} = \mathbf{J}^r + \mathbf{J}^\mathbf{s} \tag{3.27}$$

where $\mathbf{J}^r$ denotes $\frac{\partial \mathbf{p}^r}{\partial \mathbf{q}_o}$ and $\mathbf{J}^s$ denotes $-\frac{\partial \mathbf{p}^s}{\partial \mathbf{q}_o}$.

The iteration terminates when the following convergence criteria are satisfied:

$$\left\| \mathbf{q}_o^v - \mathbf{q}_o^{v-1} \right\|_\infty \le \varepsilon_q \tag{3.28}$$

and

$$\left\| \mathbf{f}(\mathbf{q}_o^v) \right\|_\infty \le \varepsilon_f \tag{3.29}$$

where ε_q and ε_f are two user-specified tolerances.

The major effort required in using the Newton-Raphson method is the construction of the Jacobian matrix $\mathbf{J}$, which is the summation of matrices $\mathbf{J}^r$ and $\mathbf{J}^s$. In the current network model, matrix $\mathbf{J}^r$ is a diagonal matrix because pressure p_j^r is a function of only one primary variable $q_{o,j}$. The element of $\mathbf{J}^r$ can be evaluated analytically. Matrix $\mathbf{J}^s$ can be a dense matrix. The pattern of its nonzero elements depends on the connectivity of the gathering system. Matrix $\mathbf{J}^s$ can be constructed efficiently by the procedure described below.

First we need to define some terminology. A well or a node can have one active output node, which is the adjacent downstream node connected to that well or node. A node can have multiple input nodes or wells, which are the adjacent upstream nodes or wells connected to that node. Nodes are arranged in levels. The nodes with fixed pressures are assigned to the first level. For a node in level l, all its input nodes or wells are in level $l+1$. The pressure of all nodes and their derivatives are computed as follows:

1. For level 1, all nodes are boundary nodes. The node pressures are specified by the user and the pressure derivatives are zero.
2. For node j or well j in level $l+1$, suppose we know the pressure and pressure derivatives of its output node, say node i, in level l, then the pressure of node j can be expressed as

$$p_j = f\left(p_i, \boldsymbol{\alpha}^{ij}\right) \tag{3.30}$$

where vector $\boldsymbol{\alpha}^{ij}$ denotes the parameters that affect the pressure drop between node i and node j. For the purpose of constructing matrix $\mathbf{J}^s$, $\boldsymbol{\alpha}^{ij}$ should include the total oil, gas, and water flow rate through the link between node i and node j.

The pressure derivatives of node j can be computed as

$$\frac{\partial p_j}{\partial x_k} = \frac{\partial p_j}{\partial p_i} \bullet \frac{\partial p_i}{\partial x_k} + \sum_m \frac{\partial p_j}{\partial \alpha_m^{ij}} \bullet \frac{\partial \alpha_m^{ij}}{\partial x_k},\ k = 1,\ldots,n \tag{3.31}$$

where x_k is the kth primary variable (i.e., the oil rate of a well), α_m^{ij} is the mth parameter that affects the pressure drop between node i and node j.

Eq. 3.31 is the key for efficient evaluation of the required derivatives because of the following reasons:

- Derivative $\frac{\partial p_i}{\partial x_k}$ is known at this stage.
- Derivative $\frac{\partial \alpha_m^{ij}}{\partial x_k}$ can often be computed analytically with trivial computational time.
- Derivatives $\frac{\partial p_j}{\partial p_i}$ and $\frac{\partial p_j}{\partial \alpha_m^{ij}}$ can be approximated by the finite difference method or computed using some automatic differentiation techniques (Berz et al., 1996). Given a computer program for computing a function, automatic differentiation software reads the computer program and produces an augmented computer program capable of computing both the function and the derivatives of the function. In this study we used such an automatic differentiation software, ADIFOR (Bischof et al., 1998), to generate an augmented computer program for computing $\frac{\partial p_j}{\partial p_i}$ and $\frac{\partial p_j}{\partial \alpha_m^{ij}}$.
- Derivatives $\frac{\partial p_j}{\partial p_i}$ and $\frac{\partial p_j}{\partial \alpha_m^{ij}}$ are time consuming to compute. However, once they are evaluated, they can be used to compute the derivatives of node pressure p_j with respect to all primary variables.

3. Repeat Step 2 for all nodes or wells in level $l+1$.
4. Let $l = l+1$. Repeat Step 2 and 3 until we obtain the pressure and pressure derivatives for all nodes of the gathering system.

3.5.3 Multiple Solutions

Eq. 3.24 does not always have a unique solution. We use a single well system to illustrate possible situations and discuss the complications in finding the true solutions of a gathering system with multiple wells.

Consider a single well system with a fixed well head pressure. For a fixed flow rate, we can calculate the bottomhole pressure from two pressure traverse calculations. The first calculation is from the reservoir to the bottomhole. Denote the bottomhole pressure from this calculation as p^r. The second path is from the surface (well head) to the bottomhole. Denote the bottomhole pressure from this calculation as p^s. We define the relationship between p^r and the oil flow rate as the inflow performance curve. We define the relationship between p^s and the oil flow rate as the outflow performance curve. Several possibilities exist when we overlay the inflow and outflow performance curves:

1. The two curves have no intersections. In this case the corresponding governing equation has no feasible solutions. This is the case when the reservoir pressure is too low to support the fluid column in the tubing string.
2. The two curves have a unique intersection and the intersection represents the actual flow rate of the single well system.
3. The two curves have two intersections. As demonstrated in Section 3.1, under certain conditions, for upwards multiphase flow, at low flow rates, the pressure drop across the tubing string decreases with increasing flow rates. Beyond a critical point, the pressure drop across the tubing string increases with increasing flow rates. Consequently, p^s first decreases then increases as the flow rate increases. As a result of this the inflow and outflow performance curves may have two intersections (Figure 3.3). However, the region of decreasing p^s with increasing flow rate is unstable and can cause intermittent or no-flow condition (Lea and Tighe, 1983). Thus, the second intersection represents the stable solution while the first intersection does not.

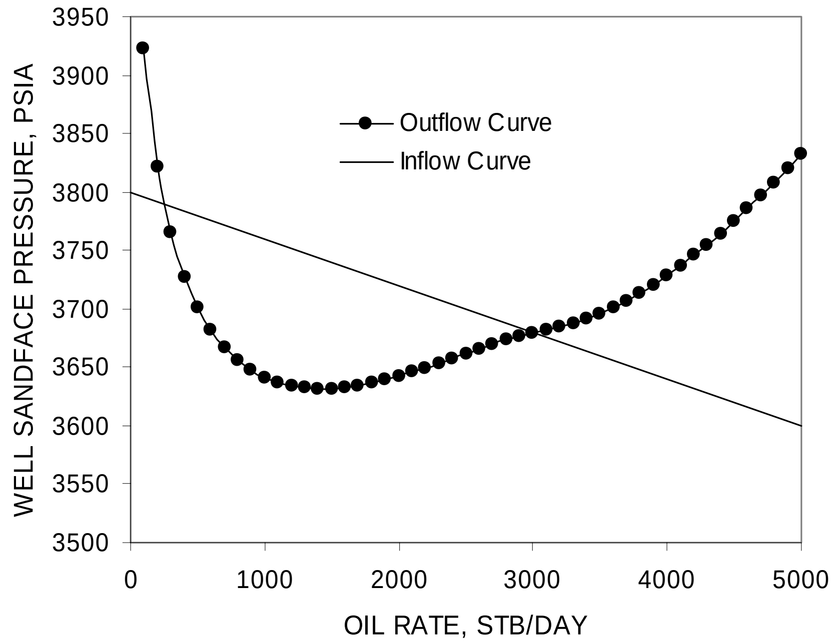


Figure 3.3: Illustration of multiple solutions for a single well system.

For a general gathering system with multiple wells, finding the true solutions to the multiphase flow problem is more complicated. Several fundamental questions regarding this problem are:

1. Is there a feasible solution? The governing equations for the network problem, Eq. 3.24, may have no feasible solutions because of the following reasons:
   - The production system may be operated with some unrealistic operational settings. For instance, in an optimization run, a well may have a large lift gas rate but a small well choke diameter that is unable to deliver the large gas rate.
   - Eq. 3.24 is formulated with the assumption that fluids always flow up a production tree, which is not necessarily true for a particular set of operational settings. The Newton-Raphson method can fail to converge no matter whether the network problem has feasible solutions or not. For these reasons, it is difficult to judge the feasibility of the network problem when the Newton-Raphson method fails to solve the problem. One possible approach to address this problem is to start the Newton-Raphson method from multiple starting points. If all runs fail to obtain a feasible solution, then there is a large chance that the network problem is

infeasible. In such cases, production settings may be adjusted and the solution process restarted.

2. How to determine if a feasible solution is stable? For naturally flowing wells, the flow is stable if the outflow performance curve has a positive slope (Lea and Tighe, 1983). Therefore, the following inequality is used to determine if a solution to Eq. 3.24 is stable:

$$\frac{\partial p_j^s}{\partial q_{o,j}} \geq 0, \quad j = 1,...,n_w \tag{3.32}$$

A solution is stable if it satisfies the inequality given by Eq. 3.32; otherwise it is not stable. For wells on gas-lift, the stability of the flow depends on many factors, and several criteria are available to determine if the flow is stable (Fitremann and Vedrines, 1985; Asheim, 1988). To keep this study focused, this work used Eq. 3.32 as the stability criteria for all wells.

3. For a fixed operational setting, is it possible for a network problem to have only feasible but no stable solutions? Is it possible for a network problem to have multiple stable solutions? If the answers to these questions are positive, how do we identify and handle these scenarios in simulation and optimization? To date, available literature does not offer a complete and systematic solution to those questions. Therefore, post-analysis is required to verify the validity of a network solution.

3.6 Sensitivity Coefficients

Sensitivity coefficients refer to the derivatives of system response with respect to some system parameters. Suppose the multiphase flow in a gathering system is governed by

$$\mathbf{f}(\mathbf{u}, \boldsymbol{\alpha}) = 0 \tag{3.33}$$

where $\mathbf{u}$ denote the system response, i.e. the flow rates from each well, and $\boldsymbol{\alpha}$ denote the system parameters, e.g., the pipe diameters. Then $\frac{\partial \mathbf{u}}{\partial \boldsymbol{\alpha}}$ are the sensitivity coefficients of system response $\mathbf{u}$ with respect to system parameters $\boldsymbol{\alpha}$.

Sensitivity coefficients are required in applications such as optimization, model tuning, history matching, and sensitivity analysis. Sensitivity analysis can assist the design of a new gathering system and potentially be an effective way to identify bottlenecks in an existing system.

3.6.1 Finite Difference Method

To obtain sensitivity coefficients $\frac{\partial \mathbf{u}}{\partial \boldsymbol{\alpha}}$, a conventional method is to use a finite difference approximation:

1. Solve $\mathbf{f}(\mathbf{u},\boldsymbol{\alpha}) = 0$ to obtain $\mathbf{u}$.
2. For all α_i, repeat the following steps:
   a. Let $\alpha_i' = (1+\varepsilon_i)\alpha_i$, where ε_i is a small perturbation.
   b. Solve $\mathbf{f}(\mathbf{u}',\boldsymbol{\alpha}') = 0$ to obtain $\mathbf{u}'$.
   c. Compute $\frac{\partial \mathbf{u}}{\partial \alpha_i} = \frac{\mathbf{u}' - \mathbf{u}}{\varepsilon_i \alpha_i}$.

The finite difference approximation is straightforward to implement, however, it is not efficient. For every perturbed parameter u_i', we need to solve an updated set of system equations. In addition, the accuracy of the sensitivity coefficients depends on the value of the perturbation ε_i and care should be taken to ensure that ε_i is appropriate (Gill et al., 1981).

3.6.2 Jacobian Method

Landa (1997) presented an efficient procedure for computing the derivatives of dynamic production data (well pressure, flow rate, and reservoir saturation) with respect to static reservoir properties (permeability and porosity in grid blocks) for a reservoir history matching problem. A brief description of this method goes as follows.

Consider a discrete reservoir system

$$\mathbf{f}\left(\mathbf{u}^{k+1}, \mathbf{u}^{k}, \boldsymbol{\alpha}, \Delta t\right) = \mathbf{0} \tag{3.34}$$

where $\mathbf{u}^k$ are the system responses (well pressure and flow rates) at time step k, $\boldsymbol{\alpha}$ are the reservoir properties (permeability and porosity), and Δt is the time step length. In order to obtain sensitivity coefficients $\frac{\partial \mathbf{u}^{k+1}}{\partial \boldsymbol{\alpha}}$, we impose a small perturbation $\boldsymbol{\delta\alpha}$ on the reservoir properties and expand Eq. 3.34 into a Taylor's series as follows:

$$\mathbf{f}^{k+1} + \frac{\partial \mathbf{f}^{k+1}}{\partial \mathbf{u}^{k+1}} \boldsymbol{\delta}\mathbf{u}^{k+1} + \frac{\partial \mathbf{f}^{k+1}}{\partial \mathbf{u}^{k}} \boldsymbol{\delta}\mathbf{u}^{k} + \frac{\partial \mathbf{f}^{k+1}}{\partial \boldsymbol{\alpha}} \boldsymbol{\delta\alpha} + O\left(\delta^2\right) = 0 \tag{3.35}$$

The first term in the left-hand size is zero. If we drop the second order term Eq. 3.35 can be expressed as

$$\frac{\partial \mathbf{f}^{k+1}}{\partial \mathbf{u}^{k+1}} \frac{\partial \mathbf{u}^{k+1}}{\partial \boldsymbol{\alpha}} = -\frac{\partial \mathbf{f}^{k+1}}{\partial \mathbf{u}^{k}} \frac{\partial \mathbf{u}^{k}}{\partial \boldsymbol{\alpha}} - \frac{\partial \mathbf{f}^{k+1}}{\partial \boldsymbol{\alpha}} \tag{3.36}$$

The implications of Eq. 3.36 are that we need to construct and invert matrix $\frac{\partial \mathbf{f}^{k+1}}{\partial \mathbf{u}^{k+1}}$ only once to obtain the derivatives of $\mathbf{u}^{k+1}$ with respect to all parameters $\boldsymbol{\alpha}$.

The same idea can be used to compute the sensitivity coefficients for a gathering system governed by Eq. 3.33. To obtain $\frac{\partial \mathbf{u}}{\partial \boldsymbol{\alpha}}$, we add a small perturbation $\delta\alpha_i$ to parameter α_i

$$\mathbf{f}\left(\mathbf{u} + \delta\mathbf{u}, \boldsymbol{\alpha} + \delta\boldsymbol{\alpha}\right) = \mathbf{0} \tag{3.37}$$

where $\delta\boldsymbol{\alpha} = \left(0,...,0,\delta\alpha_i,0,...,0\right)^T$. By expanding Eq. 3.37 around $\boldsymbol{\alpha}$ and ignoring the second order term, we obtain

$$\mathbf{f}\left(\mathbf{u}, \boldsymbol{\alpha}\right) + \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \boldsymbol{\delta}\mathbf{u} + \frac{\partial \mathbf{f}}{\partial \alpha_i} \delta\alpha_i = 0 \tag{3.38}$$

Note $\mathbf{f}\left(\mathbf{u}, \boldsymbol{\alpha}\right) = 0$. The Jacobian matrix is defined by

$$\mathbf{J} = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \tag{3.39}$$

and

$$\mathbf{J} \frac{\partial \mathbf{u}}{\partial \alpha_i} = -\frac{\partial \mathbf{f}}{\partial \alpha_i} \tag{3.40}$$

Again we need to construct and invert matrix $\mathbf{J}$ only once to obtain the derivatives of vector $\mathbf{u}$ with respect to the whole parameter vector $\boldsymbol{\alpha}$. For the network problem, constructing the Jacobian matrix requires extensive computational time. Thus the

Jacobian method is more efficient than the finite difference method, which needs to solve an updated set of system equations whenever a parameter α_i is perturbed.

3.7 Examples

In this section, we use an example to demonstrate various aspects of the procedures discussed in the previous sections for simulating the network flow and computing the sensitivity coefficients.

This is a synthetic example with ten vertical wells. The reservoir grid block pressures around these wells vary between 3400 psi to 4900 psi. All wells are connected to the same platform through tubing strings, chokes with a fixed diameter of 48/64 inches, and 3.5 inches diameter surface pipelines of lengths between 60 to 2000 ft. The platform is connected to a separator through a pipeline with a diameter of 5.5 inches and a length of 4000 ft. The separator is operated at a fixed pressure of 160 psi. The oil gravity is 31° API. The specific relative gravity of gas is 0.8. The fluid temperature in the surface pipeline is assumed to be 110 $^\circ$F. The fluid temperature at the bottomholes is assumed to be 160 $^\circ$F. The fluid temperature in the tubing strings is linearly interpolated between the bottomhole fluid temperature and the surface fluid temperature. Figure 3.4 shows the configuration of the gathering system. Appendix A.1.4 shows the input file for this example.

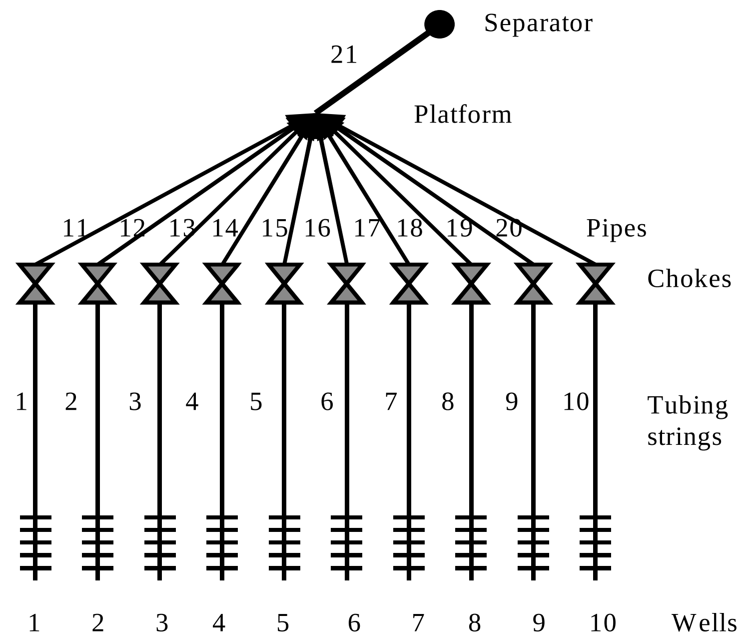


Figure 3.4: Configuration of the gathering system.

3.7.1 Simulation Results

We made five runs with different initial guesses. Runs 1, 2, 3, and 4 started the solution procedure with 20, 200, 1000, 2000 STB/d oil rate for every well, respectively. Run 5 started the solution procedure with a different flow rate for each well. We deliberately started Run 1 with a bad initial guess so that the solution procedure would be difficult to converge.

All runs were able to reach the convergence criteria. Figure 3.5 shows how the infinity norm of the residuals of the governing equations, $\left\|\mathbf{f}\left(\mathbf{q}_o^v\right)\right\|_\infty$, decreased at each iteration. For Run 3 and Run 4, $\left\|\mathbf{f}\left(\mathbf{q}_o^v\right)\right\|_\infty$ dropped below 1 psi within 3 iterations. As expected, Run 1 had the slowest convergence rate. However, even for Run 1, $\left\|\mathbf{f}\left(\mathbf{q}_o^v\right)\right\|_\infty$ dropped below 1 psi within 6 iterations. Figure 3.6 and 3.7 show the oil flow rate of each well at every iteration for Run 1 and Run 4, respectively. In both runs, the oil flow rate for each well approached its converged value within 2-4 iterations. Therefore, we concluded that the proposed solution procedure is efficient.

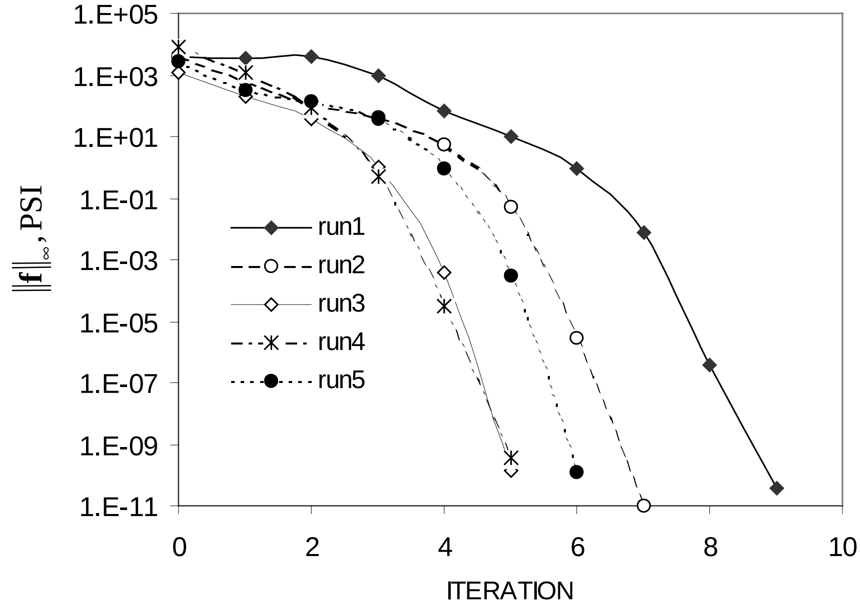


Figure 3.5: Infinity norm of the governing equation residuals versus iteration number.

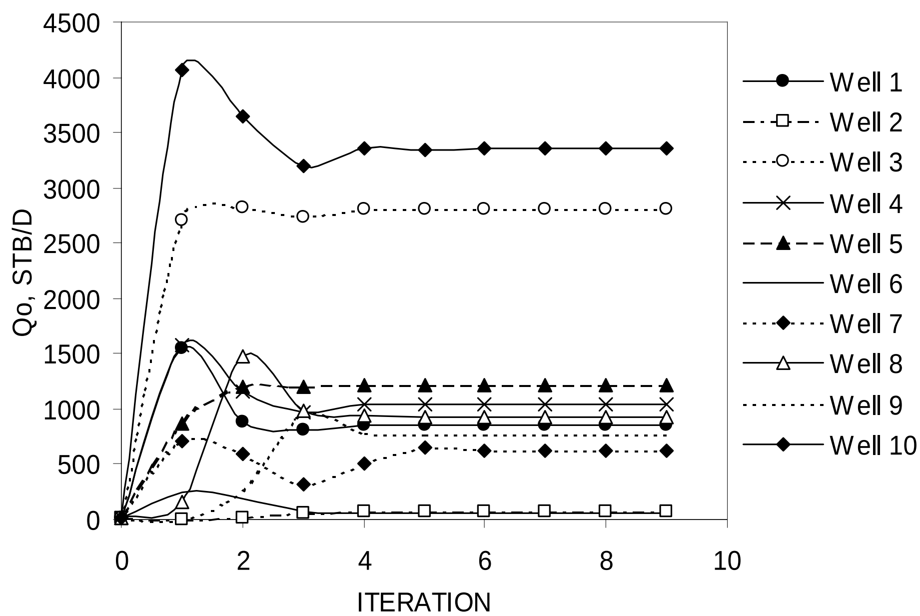


Figure 3.6: Oil rate versus iteration number for Run 1.

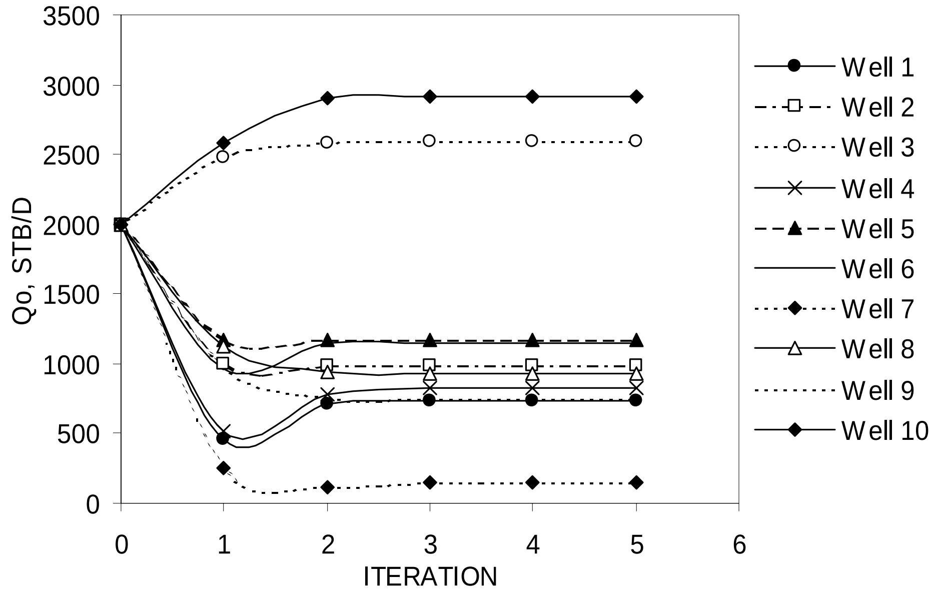


Figure 3.7: Oil rate versus iteration number for Run 4.

Multiple solutions exist for this problem. As shown in Table 3.2, the five runs converge to three sets of solutions. Evaluated by Eq. 3.32, the solutions of Run 1 and Run 2 are unstable, because for some wells $\partial p_j^s / \partial q_{o,j}$ is negative. The solutions of Runs 3, 4, and 5 are stable, because for all wells $\partial p_j^s / \partial q_{o,j}$ is positive.

Table 3.2: Network simulation results from different runs.

Well	**Run 1**		**Run 2**		**Run 3, Run 4, and Run 5**	
	Oil Rate (STB/d)	$\partial p_i^s / \partial q_{o,i}$ (psi•d/STB)	Oil Rate (STB/d)	$\partial p_i^s / \partial q_{o,i}$ (psi•d/STB)	Oil Rate (STB/d)	$\partial p_i^s / \partial q_{o,i}$ (psi•d/STB)
1	855.21	0.93	781.48	0.95	735.29	0.97
2	68.45	-4.95	1026.53	1.63	988.84	1.57
3	2809.52	0.68	2677.97	0.67	2590.56	0.66
4	1044.75	-4.95	913.44	-1.73	828.05	0.86
5	1212.72	2.86	1186.67	2.79	1168.77	2.75
6	63.76	-2.68	101.54	-1.73	1142.99	1.05
7	616.05	0.36	256.14	0.58	144.35	0.78
8	933.19	6.24	933.19	6.24	933.19	6.24
9	759.87	4.39	748.94	4.25	740.97	4.17
10	3354.15	0.37	3088.45	0.36	2910.98	0.36

3.7.2 Computation of Sensitivity Coefficients

The purpose of this section is to demonstrate the use of the Jacobian method for computing the sensitivity coefficients. We used both the finite difference and the Jacobian method to compute the sensitivity coefficients of the oil flow rate of every well with respect to the gas-lift rate of every well, the diameter of every pipe (including the tubing strings), and the diameter of every choke. Remember there are 10 gas-lifted wells, 21 pipes, and 10 chokes in the example problem.

To obtain all the sensitivity coefficients, the finite difference method needs to run 42 network simulations. In contrast, the Jacobian method requires only one network simulation and a little extra computational time. For this problem, on a Silicon Graphics Origin 200 computer with four 270 MHz CPUs and 2304MB of RAM, a typical network simulation costs about 470 ms of CPU time. The extra computational time required by the Jacobian method is about 40ms. Table 3.3 compares several sensitivity coefficients obtained from both the Jacobian method and the finite difference method. The results from the finite difference method depend on the finite difference interval. Table 3.3 suggests that a finite difference interval of between 0.01% and 0.001% of the parameter value is appropriate for this problem. The results from the Jacobian method are in good agreement with the results from the finite difference method.

Table 3.3: Comparison of sensitivity coefficients obtained from the Jacobian method and the finite difference method.

Sensitivity Coefficients	Jacobian Method	Finite Difference Approximation				
		$\varepsilon_i = 1\%$	$\varepsilon_i = 0.1\%$	$\varepsilon_i =$ 0.01%	$\varepsilon_i =$ 0.001%	$\varepsilon_i =$ 0.0001%
$\partial q_{o,1} / \partial d_{tubing,1}$ (STB/d/inch)	1.0654E+1	7.6811E+0	1.0356E+1	1.0627E+1	1.0654E+1	1.0657E+1
$\partial q_{o,1} / \partial d_{choke,1}$, (64STB/d/inch)	4.5943E+0	4.4996E+0	4.5848E+1	4.5935E+1	4.5943E+1	4.5944E-1
$\partial q_{o,1} / \partial q_{\lg,1}$, (STB/MSCF)	1.0079E-1	1.0002E-1	1.0072E-1	1.0079E-1	1.0080E-1	1.0080E-1
$\partial q_{o,6} / \partial d_{tubing,1}$, (STB/d/inch)	-4.3001E-1	-3.1003E-1	-4.1798E-1	-4.2892E-1	-4.3003E-1	-4.3020E-1
$\partial q_{o,6} / \partial d_{choke,1}$, (64STB/d/inch)	-1.8543E-1	-1.8162E-1	-1.8505E-1	-1.8540E-1	-1.8544E-1	-1.8545E-1
$\partial q_{o,6} / \partial q_{\lg,1}$, (STB/MSCF)	-1.0819E-2	-1.0790E-2	-1.0817E-2	-1.0820E-2	-1.0820E-2	-1.0820E-2

Sensitivity coefficients can give great insight about the flow behavior in a gathering system and can be used to identify bottlenecks of the system. Figure 3.8 plots the sensitivity coefficients of the well oil rates with respect to the pipe diameters (as shown in Figure 3.4, pipe 1-10 refers to the tubing strings for well 1-10, respectively; pipe 11-21 are surface pipelines). Figure 3.8 shows that the oil flow rate of all wells except well 8 are sensitive to the diameter of pipe 21. The sensitivity coefficients of the well oil rates with respect to the diameters of other surface pipelines and tubing strings are close to zero. This suggests that pipe 21 is the bottleneck of the gathering system. If we increase the diameter of pipe 21, d_{21}, the sensitivity coefficients with respect to d_{21} decrease and the sensitivity coefficients with respect to the diameters of tubing strings increase. As shown in Figure 3.9, when d_{21} increases to 7.5 inches, the diameters of the tubing strings and pipe 21 has roughly the same impact on the well oil rates. When d_{21} increases to 9.5 inches (Figure 3.10), the sensitivity coefficients with respect to d_{21} become zero. In all scenarios, the sensitivity coefficients with respect to the diameters of pipes 11-20 are approximately zero. This suggests pipes 11-12 are not properly designed and may have surplus capacity.

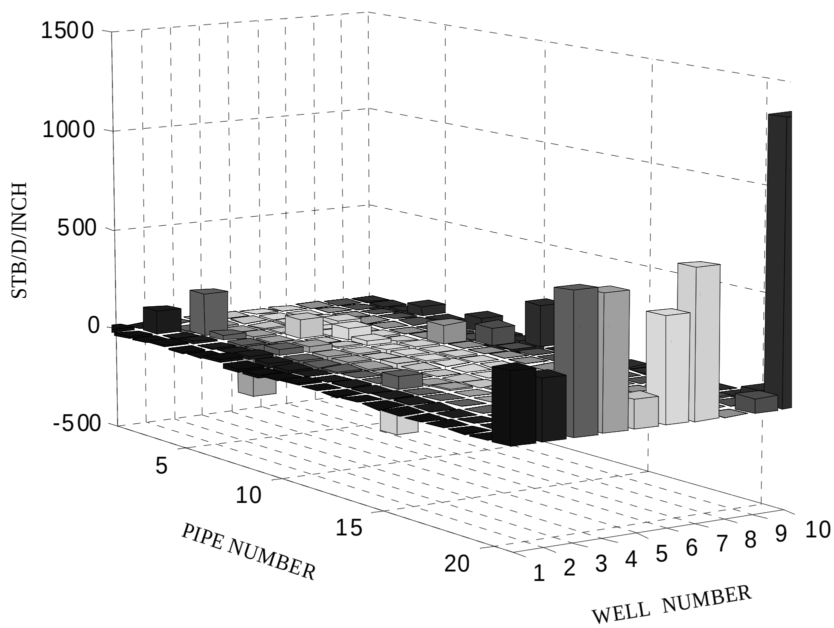


Figure 3.8: Sensitivity coefficients of well oil rates with respect to system parameters (the inner diameter of pipe 21 is 5.5 inches).

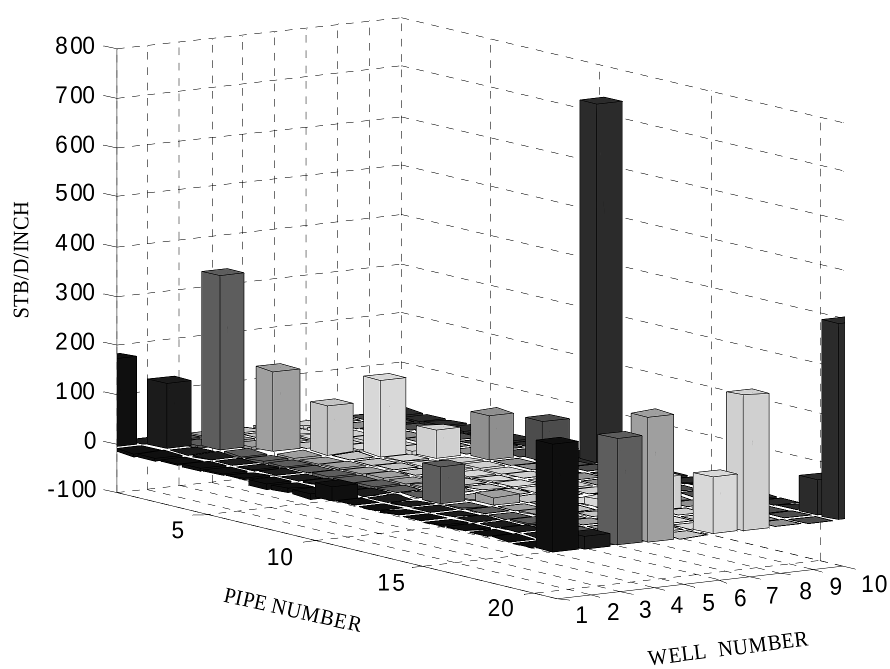


Figure 3.9: Sensitivity coefficients of well oil rates with respect to system parameters (the inner diameter of pipe 21 is 7.5 inches).

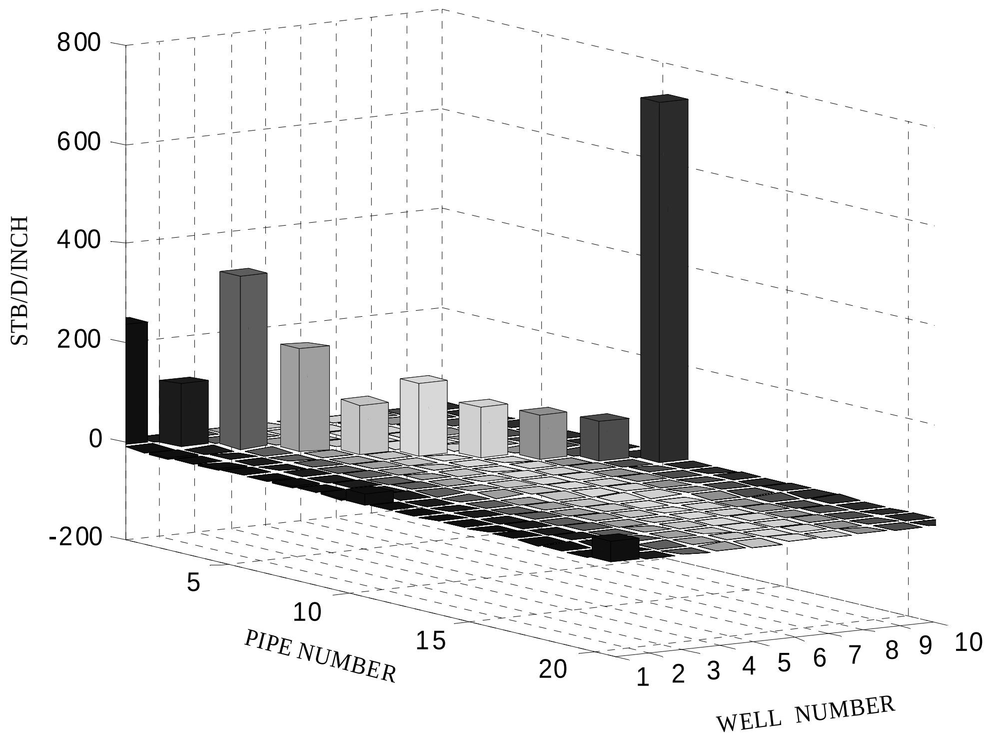


Figure 3.10: Sensitivity coefficients of well oil rates with respect to system parameters (the inner diameter of pipe 21 is 9.5 inches).

3.8 Concluding Remarks

The drift flux slip model for the multiphase pipe flow and the Sachdeva model for the multiphase choke flow were presented in this chapter. Computational experience showed that the Newton-Raphson method is not suitable for the root-finding problems encountered in these models. Brent's method (Press et al., 1992) proved to be both efficient and robust.

The multiphase network problem was formulated according to the mass balance requirement and Krichoff's law and solved by the Newton-Raphson method. The Jacobian matrix was computed efficiently by exploring the tree-like structure of the network. An efficient method, the Jacobian method, was used to compute the sensitivity coefficients. Results from an example showed that the Newton-Raphson method converges fast for our formulation, and the Jacobian method is more efficient than the finite difference method for computing sensitivity coefficients.

It is straightforward to extend the formulation and solution methods presented here to compositional models. Further research is required for the following problems:

1. How to build an efficient and robust solution procedure for multiphase flow problems of gathering systems with loops.
2. How to build a procedure capable of automatically finding the stable solutions of a multiphase flow problem.

Chapter 4

Rate Allocation through Separable Programming

4.1 Introduction

Rate allocation refers to the problem of optimally adjusting production rates and lift gas rates of production wells to achieve certain operational goals. These goals vary with the field and time. In some petroleum fields, especially mature fields, oil production can be constrained by fluid handling capacities of facilities. For such fields, rate allocation can be an effective way to increase the oil rate or reduce the production cost. For example, if the oil production in a field is constrained by the gas processing capacity of the separation units, closing or reducing production rates of wells with the highest GOR will increase the total oil rate; in addition, reducing the lift gas rate of certain wells may increase the overall oil production by utilizing the gas processing capacity more efficiently.

This study addressed the following rate allocation problem. The objective function is the total oil rate. An oil, gas, water, or liquid flow rate constraint can be put on any production well or network node. In abstract form, the optimization problem can be expressed as

$$\text{maximize} \sum_{i=1}^{n_w} \left\{ q_{o,i}^{w} \right\} \tag{4.1a}$$

$$\text{subject to } q^n_{p,j} \le Q^n_{p,j}\,,\ \ j \in \Omega^n\,,\ \ p \in \{o, g, w, l\} \tag{4.1b}$$

where $q^w_{o,i}$ is the oil rate of well i, $q^n_{p,j}$ is the flow rate of phase p in node j, $Q^n_{p,j}$ is the flow rate limit of phase p for node j, Ω^n is the set of all nodes. For gathering systems with a tree-like structure, the flow rate of network node j is the sum of the flow rates of wells connected to node j.

Physically, the control variables for the rate allocation problem are the well chokes and lift gas rates of wells. And the flow rate of each well, $q^w_{p,j}$, is a nonlinear function of the control variables. Fang and Lo (1996) studied a similar rate allocation problem. To simplify the optimization problem, Fang and Lo (implicitly or explicitly) made the following assumptions:

1. The well performance information can be evaluated individually for each well by ignoring flow interactions among wells.
2. The GOR and water cut for a well remain constant for varying oil rate.
3. The gas-lift performance curves are concave.

With the above assumptions, Fang and Lo (1996) reformulated the rate allocation problem to a linear programming problem and solved it by the simplex algorithm. The method was found to be very efficient.

In this part of the work we followed the line of Fang and Lo (1996) and developed other solution methods for the rate allocation problem. It was our aim to develop techniques that speed up the entire optimization process.

In Chapter 5, we will present a method that does not rely on assumption 1.

4.2 Well Performance Estimation

The method presented in this chapter required the performance estimation of individual wells. Specifically, for unlifted wells, the method requires (1) the maximum oil rate that is allowed or can be delivered from a well, and (2) the water and gas rates as functions of the oil rate of a well. For lifted wells, the method requires (1) the oil rate as a function of the lift gas rate (the gas-lift performance curve), and (2) the water and (formation) gas rates as functions of the oil rate of the well.

The well performance information can be described by a set of oil rate versus water, formation gas, and lift gas rate curves (Figure 4.1), which are referred to in this study as the performance curves. How to construct the performance curves is an extensively studied area (Beggs, 1991; Lea and Tighe, 1983). Two common approaches are: (1) derive the performance curves from well test information, or (2) estimate the performance curves through simulation. This study adopted the second approach.

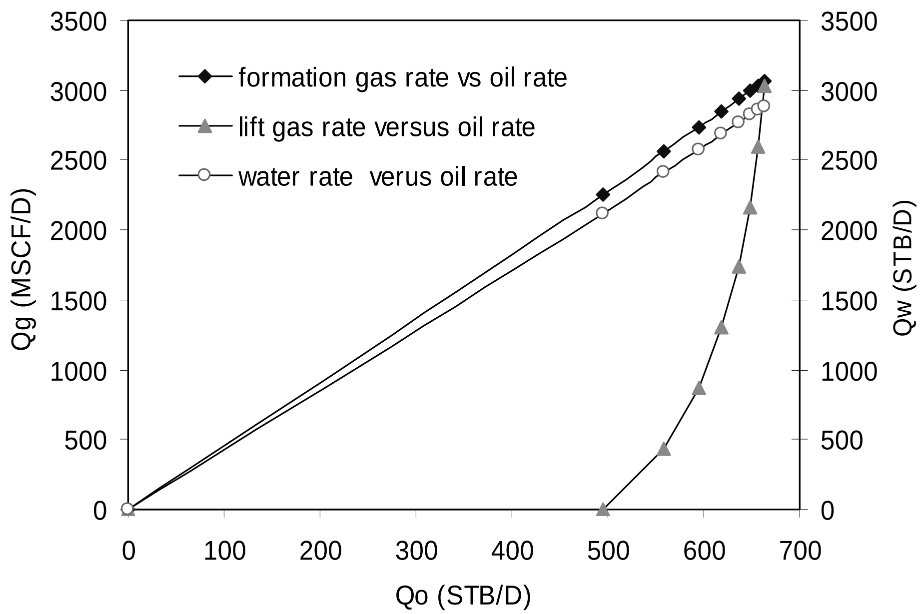


Figure 4.1: Illustration of well performance curves.

The production system is an integrated system in which the flow streams from different wells influence each other. To establish the performance curves for individual wells, a well has to be isolated from the whole system. This is done in two steps:

1. Isolate the well from the surface pipeline network by fixing the pressure of a network node before the flow stream from that well joins the flow streams from other wells.
2. Ignore well interactions in the reservoir by developing an inflow performance model for every well.

A well inflow performance model relates the oil rate of a well to its bottomhole pressure, and its water and gas rates. An appropriate way to build this model is to run a simulated well test on a numerical model of the reservoir. A simulated well test mimics well tests performed in a real field. The test calculates the bottomhole pressure, and water and gas rate of a well for various oil rates using a numerical model. A simulated well test can be time consuming. To generate performance curves efficiently, this study used the well model presented in Section 3.1 and assumed that the reservoir condition in well blocks are constant for varying oil rates.

After a well is isolated from the production system, its performance curves can be established as follows:

1. The oil rate versus water and formation gas rate curves can be solely determined by the well inflow performance model.
2. The gas-lift performance curve for a well can be constructed by determining the oil rate of the single-well system for a set of prescribed lift gas rates.

The well performance curves are represented by a set of discrete points. We assume linearity between two adjacent points, thus the performance curves are piecewise linear curves (Figure 4.1).

4.3 Rate Allocation through Separable Programming

The objective and constraint functions of the rate allocation problem (Eq. 4.1) are linear combinations of the oil, water, and formation and lift gas rates of individual wells. Further, when the well performance is approximated by piecewise linear performance curves, the water, formation gas, and lift gas rates of a well can be regarded as functions of the oil rate of that well. Therefore, if we regard the oil rate as the control variables, denoted as $\mathbf{x}$, Eq. 4.1 becomes an optimization problem whose objective and constraint functions are the sums of functions of one variable, which can be expressed as

$$\text{maximize} \sum_{j=1}^{n_w} f_j(x_j) \tag{4.2a}$$

$$\text{subject to} \sum_{j} h_{ij}(x_j) \le b_i\,,\; i = 1,\ldots,m \tag{4.2b}$$

$$x_j \ge 0\,,\; j = 1,\ldots,n_w \tag{4.2c}$$

where f_j denotes the objective functions, h_{ij} denotes the jth function involved in the ith constraint, b_i denotes the limit of the ith constraint, and m denotes the number of constraints.

Optimization problems of the form of Eq. 4.2 are separable piecewise linear problems, which can be solved by linear optimization techniques (Gill et al., 1981). This is demonstrated in the following.

First, a point $(x, f(x))$ on a piecewise linear curve defined by a set of discrete points $(x_i, f(x_i))$, $i = 0,...,r$ can be expressed as follows:

$$x = \sum_{j=0}^{r} \lambda_j x_j \tag{4.3a}$$

$$f(x) = \sum_{j=0}^{r} \lambda_j f(x_j) \tag{4.3b}$$

$$\sum_{j=0}^{r} \lambda_j = 1,\ \lambda_j \geq 0 \tag{4.3c}$$

no more than two λ_j can be positive and they must be adjacent (4.3d)

Suppose for well j, each of its well performance curves are defined by $r_j + 1$ discrete points with $x_{j0} = 0$ and $x_{jr_j} = q_{o,j}^{\max}$, where $q_{o,j}^{\max}$ is the maximum oil rate for well j. Then for all functions $f_j(x_j)$ and $g_{ij}(x_j)$ we can write

$$f_j(x_j) = \sum_{k=0}^{r_j} \lambda_{jk} f_{jk},\ f_{jk} = f_j(x_{jk}) \tag{4.4a}$$

$$g_{ij}(x_j) = \sum_{k=0}^{r_j} \lambda_{jk} g_{ijk},\ g_{ijk} = g_{ij}(x_{jk}),\ i = 1,...,m \tag{4.4b}$$

$$x_j = \sum_{k=0}^{r_j} \lambda_{jk} x_{jk} \tag{4.4c}$$

$$\sum_{k=0}^{r_j} \lambda_{jk} = 1,\ \lambda_{jk} \geq 0 \text{ for all } j,\ k \tag{4.4d}$$

for a given j, no more than two λ_{jk} can be positive and they must be adjacent (4.4e)

Substituting Eqs. 4.4a-4.4e into Eq. 4.2, we obtain Problem 4.5:

$$\text{maximize } z = \sum_{j=1}^{n_w} \sum_{k=0}^{r_j} f_{jk} \lambda_{jk} \tag{4.5a}$$

$$\text{subject to } \sum_{j=1}^{n_w} \sum_{k=0}^{r_j} g_{ijk} \lambda_{jk} \le b_i \,,\; i = 1,\ldots,m \tag{4.5b}$$

$$\sum_{k=0}^{r_j} \lambda_{jk} = 1 \,,\; j = 1,..,n_w \tag{4.5c}$$

$$\lambda_{jk} \ge 0 \text{ for all } j \,,\; k \tag{4.5d}$$

for a given j, no more than two λ_{jk} can
be positive and they must be adjacent (4.5e)

Without constraint Eq. 4.5e, we would have a general linear programming problem. It can be shown that if all $f_j(x_j)$ are concave functions and g_{ijk} are convex functions, then, the optimal solution of an LP problem defined by Eqs. 4.5a-4.5d automatically satisfies constraint Eq. 4.5e (Hartley, 1961). However, in general cases, constraint Eq. 4.5e has to be enforced explicitly.

Along this line, there are several models for the rate allocation problem depending on how constraint Eq. 4.5e are treated. These models were investigated in this study and are discussed below.

4.3.1 Model LP-I

This model was proposed by Fang and Lo (1996).

In this model, constraint Eq. 4.5e is simply ignored. The rate allocation problem is reformulated to a linear programming problem defined by 4.5a-4.5d. This model is able to solve correctly the rate allocation problem defined by Eq. 4.2 if the water and gas versus oil rate curves involved in the constraints are strictly convex. Otherwise, as demonstrated in Section 4.4.3, this model may allocate an unphysical point that is not on the performance curves.

4.3.2 Model MILP-I

This model was applied to a rate allocation problem by Güyagüler and Byer (2001). This model enforces constraint Eq. 4.5d explicitly and is suitable for rate allocation problems with performance curves of arbitrary shapes. The disadvantage of this method is that even for a rate allocation problem with moderate size, this model can contain a large number of binary variables and constraints.

To enforce the constraint that for certain j at most two consecutive coefficients λ_{jk} are nonzero, we introduce a binary variable y_{jk}, $k = 0,...,r_j - 1$, which can be equal to 1 only if $x_{j_k} \le x_j \le x_{j(k+1)}$ and 0 otherwise. Problem 4.5 is then formulated as a mixed integer linear programming (MILP) problem (Bertsimas and Tsitsiklis, 1997) to obtain Problem 4.6:

$$\text{maximize}\, z = \sum_{j=1}^{n_w} \sum_{k=0}^{r_j} f_{jk} \lambda_{jk} \tag{4.6a}$$

$$\text{subject to } \sum_{j=1}^{n_w} \sum_{k=0}^{r_j} g_{ijk} \lambda_{jk} \le b_i\,,\ i = 1,...,m \tag{4.6b}$$

$$\sum_{k=0}^{r_j} \lambda_{jk} = 1\,,\ j = 1,..,n_w \tag{4.6c}$$

$$\lambda_{j0} \le y_{j0}\,, \text{ for all } j \tag{4.6d}$$

$$\lambda_{jk} \le y_{j(k-1)} + y_{jk}\,,\ k = 1,...,r_j - 1, \text{ for all } j \tag{4.6e}$$

$$\lambda_{jr_j} \le y_{j(r_j-1)}, \text{ for all } j \tag{4.6f}$$

$$\sum_{k=0}^{r_j-1} y_{jk} = 1, \text{ for all } j \tag{4.6g}$$

$$\lambda_{jk} \ge 0 \text{ for all } j,\ k \tag{4.6h}$$

$$y_{jk} \in \{0,1\},\ k = 1,...,r_j - 1, \text{ for all } j\,. \tag{4.6i}$$

Several similar reformulations exist and their computational performances can be different (Padberg, 2000).

In principle, a MILP problem can be solved by enumeration. However, complete enumeration is computationally infeasible as soon as the number of integer variables in a

MILP problem exceeds 20 or 30 (Wolsey, 1998). So we need some strategies to cut the number of necessary enumerations. An effective method for this purpose is the *Branch and Bound* method (Wolsey, 1998).

Branch and Bound is a general search method for optimization problems over a search space that can be represented as leaves of a tree. A fundamental idea behind the Branch and Bound method is to divide and conquer. Consider a general optimization problem

$$z = \max\{\mathbf{c}^T\mathbf{x} : \mathbf{x} \in \mathbf{S}\} \tag{4.7}$$

where $\mathbf{x}$ is the control variable and $\mathbf{S}$ denotes its feasible set. Let $\mathbf{S} = \mathbf{S}_1 \cup ...\mathbf{S}_K$ be a decomposition of $\mathbf{S}$ into smaller sets, and let $z^k = \max\{\mathbf{c}^T\mathbf{x} : \mathbf{x} \in \mathbf{S}_k\}$ for $k = 1,..., K$. Then $z = \max_k z^k$ (Wolsey, 1998). Based on this idea, Branch and Bound recursively divides an optimization problem into several subproblems, and thereby forms an enumeration tree, with each node in the tree representing a subproblem. The enumeration tree is constructed implicitly, and the bound information on subproblems are used to prune the tree. A complete description of the general Branch and Bound method can be found in Chapter 7 of Wolsey (1998).

Next we present a Branch and Bound method adapted for solving Problem 4.6.

Algorithm 4.1: Branch & Bound for Problem 4.6. The algorithm maintains a lower bound $\underline{z}$, an upper bound $\bar{z}$ ($\underline{z} \le z \le \bar{z}$), and a stack of active search nodes representing subproblems to be examined.

1. *Initialization*. Let $\underline{z} = -\infty$ and $\bar{z} = \infty$. Put Problem 4.6 into the stack of active search nodes.
2. *Choosing a node*. If the stack of active nodes is empty, the entire tree has been enumerated and the search ends. If the stack of active nodes contains several nodes, pop out the node on top of the stack. This node represents a MILP subproblem generated in Step 1 or Step 6.
3. *Optimizing*. Solve the *LP relaxation* of the subproblem selected in Step 2. Denote its optimal value as $\hat{z}$. The definition of LP relaxation is discussed shortly.

4. *Bounding*. If the node selected in Step 2 is the root node, update $\bar{z} = \hat{z}$. If the solution in Step 3 is a feasible solution to Problem 4.6, update $\underline{z} = \max\{\underline{z}, \hat{z}\}$ and store the corresponding feasible solution.
5. *Pruning*. The following conditions allow us to prune the tree and thus enumerate a large number of solutions implicitly.
   (a) Pruning by optimality. If $\bar{z} - \underline{z} \le \varepsilon_z$, where ε_z is a prescribed tolerance, then the feasible solution corresponding to $\underline{z}$ can be regarded as the optimal solution of Problem 4.6. And the search ends.
   (b) Pruning by feasibility. The solution from Step 3 is a feasible solution to Problem 4.6. There is no need to divide further the subproblem represented by current node.
   (c) Pruning by bound: $\hat{z} \le \underline{z}$. The upper bound of the subproblem represented by current searching node is below the lower bound. There is no need to divide further the subproblem represented by current node.
   (d) Pruning by infeasibility. The problem examined has no feasible solution. There is no need to divide further the subproblem represented by current node.

   If condition (a) is met, stop. If condition (b), (c) or (d) is met, go to Step 2 to backtrack. Otherwise, go to Step 6 to branch.
6. *Branching*. To reach this step, the solution from Step 3 must not be a feasible solution to Problem 4.6. Some binary variable y_{jk} has a fractional value. Suppose variable y_{jk}, $1 \le j \le n_w$, $k \in \{0,\ldots,r_j - 1\}$ has a fractional value. The MILP subproblem from Step 2 (the parent subproblem) can be divided into two subproblems. The first subproblem is comprised of the parent subproblem plus a constraint of $y_{jk} = 0$. The second subproblem is comprised of the parent subproblem plus a constraint of $y_{jk} = 1$. Add the two subproblems on top of the stack of active nodes in a prescribed order (such as the first subproblem goes first, and the second subproblem goes second).
7. *Continue the search by going to Step 2.*

The purpose of Step 3 is to compute an upper bound for the MILP subproblem from Step 2. The bound information is used in Step 5 to prune the enumeration tree. The upper bound of a MILP subproblem is often obtained by solving a relaxed problem, a problem that has an optimal value no worse than that of the MILP subproblem. One such relaxation for a MILP problem is its linear programming relaxation that allows the integer variables in a MILP problem to take real values. For example, an LP relaxation of Problem 4.6 is an LP problem that replaces constraint Eq. 4.6i with the following linear constraints

$$0 \le y_{jk} \le 1,\ k = 1,\dots,r_j - 1 \text{ for all } j \qquad (4.8)$$

4.3.3 Model MILP-II

This model was developed in this study. This model is motivated by and is suitable for problems with the following properties:

1. For wells that can flow naturally, their gas-lift performance curves are concave (Curve I in Figure 4.2).
2. For wells that need a finite amount of lift gas to start flowing, their gas-lift performance curves are comprised of a horizontal line and a concave curve (Curve II in Figure 4.2).
3. The oil versus water and formation gas curves are concave.

With these properties, constraint Eq. 4.5e can be automatically satisfied if we require the coefficient λ_{j0} in Problem 4.5 to be a binary variable for wells whose gas-lift performance curves is non-concave (Curve II in Figure 4.2).

This model has the same number of variables and constraints as Model LP-I. However, some of the decision variables are binary variables. Thus this model leads to a MILP problem. The Branch and Bound method described in Section 4.3.2 can be used to solve it. An example of the application of this method is presented in Section 4.4.3.

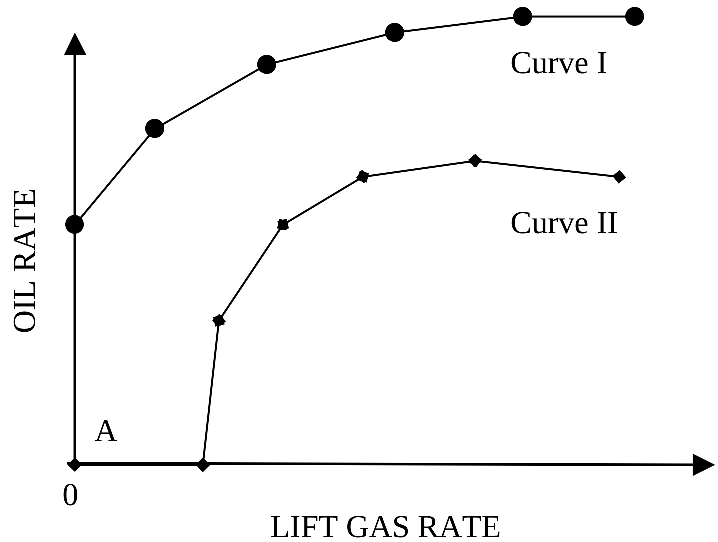


Figure 4.2: Illustration of gas-lift performance curves.

4.3.4 Model LP-II

This model was first proposed by Lo and Holden (1992). This is a simple model that does not require the performance curves and therefore there is no need to satisfy constraint Eq. 4.5e. The model assumes the well water cut and GOR are constant for varying oil rates.

This model takes a set of flow streams (either from production wells or from satellite reservoirs) as the input and scales them to meet the flow rate and velocity constraints in a way that maximizes an objective function. A flow stream is represented by the GOR, water cut, and a maximum oil rate of a well or a satellite reservoir. For example, suppose we want to maximize the total oil rate of a field subject to a total gas rate constraint. The problem can be formulated as

$$\text{maximize} \quad \sum_{i=1}^{n_w} x_i q_{o,i}^{\max} \tag{4.9a}$$

$$\text{subject to} \quad \sum_{i=1}^{n_w} x_i q_{o,i}^{\max} GOR_i \le Q_g \tag{4.9b}$$

$$0 \le x_i \le 1, \; i = 1,\dots,n_w \tag{4.9c}$$

where x_i denote the decision variable for Problem 4.9, $q_{o,i}^{\max}$ denotes the maximum oil rate for well i, Q_g denotes the limit of the total gas constraint. For gas-lift wells, the GOR includes both the formation gas and lift gas and is defined as

$$GOR = \frac{q_g^{\max} + q_{\mathrm{lg},0}}{q_o^{\max}} \tag{4.10}$$

where $q_{\mathrm{lg},0}$ denotes the lift gas rate that corresponds the maximum oil rate $q_o^{\max}$ for a gas-lift well. In the optimal solution, $x_i = 0$ indicates well i should be closed; $x_i = 1$ indicates well i should be fully open; $x_i \in (0,1)$ indicates well i should be choked back.

4.3.5 Discussions

Model LP-I and LP-II are linear programming (LP) problems. Model MILP-I and MILP-II are mixed integer linear programming (MILP) problems. These models are also called the LP-I, LP-II, MILP-I, and MILP-II methods in this study, respectively. All these methods require the objective and constraint functions be in separable form, thus they are also referred to as the separable programming (SP) techniques.

When the computational time is not a concern, the MILP-I method is the most promising method because it can be applied to rate allocation problems with arbitrary well performance curves. However, the MILP-I method can generate a MILP problem with a large number of binary variables and constraints even for a production system with a moderate number of wells. For large-scale problems, the other three methods may be more appropriate depending on the characteristics of the performance of wells and how fast we want to get the solution. For example, this study successfully applied the LP-I and LP-II methods to the Prudhoe Bay oil field with appropriate speedup techniques (see Section 4.4.2). The optimization model for Prudhoe Bay oil field has more than 1000 producers and hundreds of flowrate and velocity constraints. The MILP-II method is more appropriate for gas-lift optimization problems where many wells cannot flow without gas-lift (see Section 4.4.3).

4.4 Application to Rate Allocation Problems

When the various methods are applied to a rate allocation problem, there are some additional issues to be addressed:

1. How to handle nonlinear pressure and velocity constraints.

2. How to further speed up the solution process.

This section discusses these issues and presents a gas-lift optimization example. The application of these methods to more complicated tasks (such as in well optimization and long-term reservoir development studies) are presented in Chapter 8.

4.4.1 Handling Pressure and Velocity Constraints

For a production system, besides the flow rate constraints, a minimum pressure can be specified for a node of the gathering system:

$$p_i^n \geq p_i^{n,\min} \tag{4.11}$$

where p_i^n denotes the pressure of node i. A maximum velocity constraint can be specified for a tubing string or a surface flowline:

$$v_j \leq v_j^{\max} \tag{4.12}$$

where v_j denote the *in situ* velocity of flowline j.

The SP techniques require that the optimization problem be in separable form. However, the pressure constraints on junction nodes and velocity constraints for surface flowlines are nonlinear functions of multiple well streams. The constraints do not satisfy the requirement of the SP techniques, so special procedures are required.

In this study, the pressure constraints on network nodes are converted to flow rate constraints on individual wells using a network simulation procedure proposed by Litvak and Darlow (1995). In that procedure, whenever a pressure constraint is not honored at some node, say node j, the flow rate of related production wells are recalculated subject to the constraint:

$$p_j^n = p_j^{n,\min} \tag{4.13}$$

This procedure has been implemented in the VIP (Landmark, 2001) simulator.

The velocity constraints are incorporated into a separable problem by assuming that the velocity of a flowline is a linear function of the flow rates of that flowline:

$$v(q_o, q_g, q_w) = c_o^v q_o + c_g^v q_g + c_w^v q_w \tag{4.14}$$

where q_o denotes the oil rate of that flowline, the weighting coefficient $c_{v,o}$ is a constant defined as follows:

$$c_o^v = \frac{v_{o,0}^{\max}}{q_{o,0}} \tag{4.15}$$

where $q_{o,0}$ is a specific oil rate of that flowline (such as the oil rate of that flowline before optimization), $v_{o,0}^{\max}$ is the maximum *in situ* oil velocity along that flowline corresponding to the reference oil rate $q_{o,0}$. Variables for the gas and water phases are defined similarly.

4.4.2 Speedup Techniques

Although the SP techniques are very fast for the rate allocation problems, their efficiency is still of concern when it comes to large-scale production systems with thousands of production wells. In addition, in well connection optimization (Chapter 6) or long term reservoir development (Chapter 8) studies, a rate allocation problem has to be solved repeatedly hundreds or thousands of times. For these reasons, there is always a need to speed up the solution process.

There are several potential ways to speed up the solution process. The idea presented here is to use domain knowledge to reduce the problem size. Specifically, we want to reduce the number of decision variables and the number of constraints of an optimization problem, without sacrificing its accuracy. This can be done as follows.

Truncate the performance curves. The number of decision variables of Model LP-I, MILP-I, and MILP-II is proportional to the number of discrete points of the performance curves. To save computational time, the performance curves can be truncated at a production rate limit either specified in advance or converted from a pressure constraint. In addition, a gas-lift performance curve can be truncated at the point at which the oil rate begins to decrease as more lift gas is injected.

Eliminate redundant physical constraints. For a rate allocation problem of a large complex production system, a large number of flow rate and velocity constraints may be specified. Often some constraints are redundant (impossible to be violated) given the production limits of individual wells and other constraints specified for facilities and flowlines. Some of the constraint redundancy can be eliminated using domain knowledge (i.e., the connectivity of the production system). In this section we will describe such a *constraint elimination* procedure.

First we need to define some terminology. The *potential input* of a node is an upper bound on the oil, gas, and water flow rate that a node can receive from its input nodes. The *potential output* of a node is an upper bound on the oil, gas, and water flow rate that a node can deliver to its output node. The definition of input node and output node is given in Section 3.5.2. Nodes are arranged in levels. A node belongs to the first level if all its input nodes are wells. A node belongs to level l if the highest level of all its input nodes is level $l-1$. With these definitions, the constraint elimination procedure goes as follows.

1. Initialization. Let the potential output of all wells be their production limits. Let $l=1$.
2. Processing a node. For a node j in level l, perform the following steps.
   - 2a. Calculate the potential input of node j by summing up the potential output of all its input nodes.
   - 2b. Determine whether a user-specified constraint for node j can be violated by the potential input of node j.
     - If the constraint can *not* be violated by the potential input, it is a redundant constraint and can be eliminated. Let the potential output of node j equal to the potential input of node j.
     - If the constraint can be violated by the potential input, it is not a redundant constraint. The potential output of node j is defined by the constraint limit.
   - 2c. Repeat Step 2b for all user-specified constraints for node j.
3. Repeat Step 2 for all nodes in level l.
4. Let $l=l+1$. Repeat Step 2 and 3 until all nodes are processed.

As an example, this constraint elimination procedure has been applied to a rate allocation problem in Prudhoe Bay oil field. In that problem, the user specified 190 velocity constraints and 53 flow rate constraints for facilities (the production limits on individual wells are not counted). The speedup technique described here was able to eliminate 226 constraints, leaving only 17 constraints. When this rate allocation problem is solved by the LP-II method, a speedup factor of more than five times was achieved as a result of this constraint elimination procedure.

Avoid unnecessary variables and constraints in reformulation. The binary variables $\{y_{jk}\}$ in Model MILP-I are introduced to enforce constraint Eq. 4.5e. This is not always necessary. When the objective and constraint functions of Problem 4.5 have certain convex/concave properties, constraint Eq. 4.5e can be automatically satisfied when the LP problem defined by 4.5a-4.5d is solved by a general LP solver (Hartley, 1961). This rule can be used to reduce the number of decision variables of Model MILP-I. For example, consider a rate allocation problem whose objective is to maximize the daily oil production, there is no need to introduce the binary variables $\{y_{jk}, k = 0,..., r_j\}$ for well j unless at least one of the following conditions is met:

1. The well is involved in a water or liquid flow rate constraint while its water rate versus oil rate curve is not strictly convex.
2. The well is involved in a total gas (including formation and lift gas) flow rate constraint while its total gas rate versus oil rate curve is not strictly convex.
3. The well is involved in a lift gas volume constraint while its lift gas rate versus oil rate curve is not strictly convex.

4.4.3 A Gas-lift Optimization Example

This gas-lift optimization example has been presented in Wang et al. (2002a).

The problem of this example is from Buitrago et al. (1996). The problem is to optimize oil production from a set of 56 wells with 22,500 MSCF/d of available gas. The gas-lift performance data are shown in Table B.1 of Appendix B. In this example, wells 47-56 can not flow without gas-lift. Buitrago et al. (1996) solved the problem by both an equal-slope method and the Ex-In method, a stochastic algorithm that calculates the descent direction heuristically. In this study, this problem was solved by the MILP-II method.

Table 4.1 compares the performance of the MILP-II method with the equal-slope method and the Ex-In method. Results for the equal-slope method and the Ex-In method are taken from Buitrago et al. (1996). The MILP-II method outperforms both the equal-slope method and the Ex-In method for this example. Specifically, using the same amount of lift gas, the MILP-II method allocates 6.4% more oil than the equal-slope method. To allocate 21265 STB/d oil, the MILP-II method requires 37.0% less lift gas than the equal-slope method. To allocate 21790 STB/d oil, the MILP-II method requires 16.6% less lift gas than the Ex-In method.

Table 4.1: Gas-lift allocation results obtained from different methods.

	Equal-Slope	**Ex-In**	**MILP-II**		
			(1)	(2)	(3)
Lift Gas Rate (MSCF/d)	22,508	20,454	22,500	14,175	17,040
Oil Rate (STB/d)	21,265	21,790	22,632	21,265	21,790

(1) Allocate all available lift-gas of 22500 MSCF/d.
(2) Minimize lift-gas rate while keeping the oil rate at 21265 STB/d.
(3) Minimize lift-gas rate while keeping the oil rate at 21790 STB/d.

To investigate the advantages of the MILP-II method over the LP-I method (Fang and Lo, 1996), the LP-I method was also used to solve this problem. One shortcoming of the LP-I method is that it may allocate gas from a point not lying on the gas-lift performance cuve. This is demonstrated in Figure 4.3, which presents the gas-lift performance curve and the allocated lift gas and oil rates from both the MILP-II method and the LP-I method for well 47. For this well, the point selected by the MILP-II method is right on the gas-lift performance curve while the point selected by the LP-I method is far above the gas-lift performance curve. For the rest of wells, the data points allocated by both methods are on the gas-lift performance curves. Table B.2 of Appendix B shows the optimal lift gas and oil rate allocated by the MILP-II method for every well.

To investigate the efficiency of the MILP-II method, we applied it to other similar problems with different sizes. For example, in the second case, we have 560 wells and 225,000 MSCF/d lift gas available. All computations were performed on a Sun-Enterprise 5500 machine with 4GB of RAM and 2GB of swap space. The second row of Table 4.2 shows the computational time (elapsed time). The third row of Table 4.2 shows the number of LP problems solved in the solution process. Results demonstrated that the MILP-II method is very efficient for this problem.

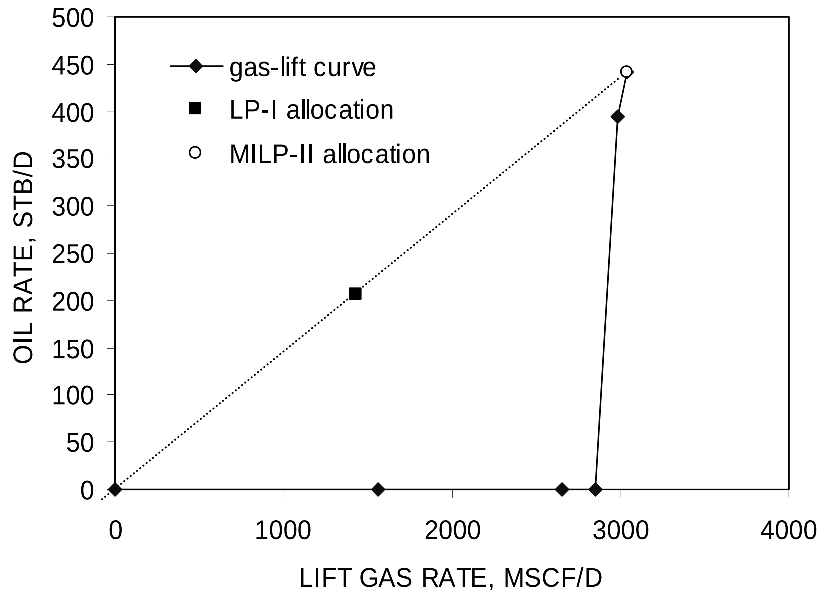


Figure 4.3: The allocated lift gas and oil rates for well 47 from the MILP-II method and the LP-I method.

Table 4.2: Computational time of the MILP-II method for various problems.

Number of Wells	56	560	3360
Elapsed Computational Time (Second)	0.27	3.06	140.29
Number of LP Problems Solved	5	6	40

4.5 Concluding Remarks

In this chapter we addressed the rate allocation problem for production systems where the flow in surface flowlines has little impact on well performance. The approaches used here follows the line of Fang and Lo (1996). The contributions of this study can be summarized as follows:

1. Proposed the MILP-II method, which is suitable for gas-lift optimization problems in which some wells need a finite amount of lift gas to start flowing.
2. Developed several effective speedup techniques. The speedup factor can be significant for large-scale systems.
3. The pressure and velocity constraints are handled through preprocessing and linear approximations.

The LP-I and LP-II methods have been coupled with a reservoir simulator so that the impact of the reservoir can be updated at every iteration during a time step. The coupling procedure and applications of these methods are presented in Chapter 8.

The quality of performance curves is crucial to the success of the optimization procedure presented here. Therefore further research should be conducted to address the problem of how to construct good quality performance curves efficiently.

Chapter 5

Rate Allocation through Sequential Quadratic Programming

5.1 Introduction

Chapter 4 simplifies the rate allocation problem to a separable programming problem by assuming that the flow interactions among production wells are not significant. However, this assumption does not hold for many petroleum fields, especially for offshore production systems, where the oil production is often constrained by the fluid delivering capacities of some trunk pipelines. For such cases, rigorous network simulations have to be employed in the optimization process to capture complex flow interactions among different wells. Consequently the rate allocation problem has to be formulated as a nonlinearly constrained optimization problem.

The first step in solving the nonlinear rate allocation problem is to select an appropriate optimization algorithm. The major selection criteria are the efficiency and robustness of the algorithm. Both genetic algorithms and gradient-based methods were considered. The genetic algorithms can avoid being trapped in local optima, but they are not good at handling nonlinear constraints and usually require many more function evaluations than gradient-based methods. Because a network simulation is costly, a gradient-based method is deemed appropriate for the rate allocation problem. Out of the many gradient methods for constrained problems, a sequential quadratic programming

(SQP) method was selected. These methods are regarded as the best optimization method for nonlinearly constrained problems whose number of decision variables is small or moderate (Murray, 1997).

This chapter first presents a brief description of the SQP methods. Then it discusses the appropriate formulation of the rate allocation problem and related practical issues. Finally it presents some applications of the algorithm. Results showed that the method is capable of handling rate allocation problems of varying complexities and sizes.

5.2 Sequential Quadratic Programming

SQP methods are widely considered the most effective algorithms for solving nonlinearly constrained optimization problems (Murray, 1997). This section presents a brief description of the SQP methods (Gill et al., 2002).

Consider an optimization problem, Problem 5.1, with nonlinear inequality constraints

$$\text{minimize } f(\mathbf{x}) \tag{5.1a}$$

$$\text{subject to } c_i(\mathbf{x}) \geq 0,\ i = 1,....,m \tag{5.1b}$$

where the objective function f and the constraints $\{c_i\}$ are functions of the control variable $\mathbf{x}$.

SQP methods seek a local minimum $\mathbf{x}^*$ of the constrained optimization Problem 5.1. The constraint $c_i(\mathbf{x}) \geq 0$ is said to be active at $\mathbf{x}^*$ if $c_i(\mathbf{x}^*) = 0$, and inactive if $c_i(\mathbf{x}^*) > 0$. Let the vector $\hat{\mathbf{c}}(\mathbf{x}^*)$ denote the subset of t constraint functions that are active at $\mathbf{x}^*$, and let $\hat{\mathbf{A}}(\mathbf{x}^*)$ be the matrix whose rows are the transposed gradient vectors of the active constraints. The associated Lagrangian function for Problem 5.1 can be expressed in terms of the objective function and active constraints (Gill et al., 1981)

$$L(\mathbf{x}, \boldsymbol{\lambda}) \equiv f(\mathbf{x}) - \boldsymbol{\lambda}^{\mathbf{T}} \hat{\mathbf{c}}(\mathbf{x}) \tag{5.2}$$

where $\boldsymbol{\lambda}$ denote the vector of Lagrangian multipliers. The following conditions are necessary for $\mathbf{x}^*$ to be a local minimum of Problem 5.1 (Gill et al. 1981)

1. $\mathbf{c}(\mathbf{x}) \geq 0$, with $\hat{\mathbf{c}}(\mathbf{x}^*) = 0$,

2. $\mathbf{Z}(\mathbf{x}^*)^{\mathrm{T}}\nabla\mathbf{f}(\mathbf{x}^*)=0$, or, equivalently, $\nabla\mathbf{f}(\mathbf{x}^*)=\hat{\mathbf{A}}(\mathbf{x}^*)^{\mathrm{T}}\boldsymbol{\lambda}^*$,
3. $\lambda_i^* \geq 0$, $i=1,\ldots,t$, and
4. $\mathbf{Z}(\mathbf{x}^*)^{\mathrm{T}}\nabla_{xx}\mathbf{L}(\mathbf{x}^*,\boldsymbol{\lambda}^*)\mathbf{Z}(\mathbf{x}^*)$ is positive semi-definite

where $\mathbf{Z}(\mathbf{x}^*)$ is a basis of the nullspace of $\hat{\mathbf{A}}(\mathbf{x}^*)$. These optimality conditions for a constrained optimization problem are called the second order KKT (Karush-Kuhn-Tucker) conditions.

SQP methods are based on these optimality conditions and they approach a local minimum, $\mathbf{x}^*$, by using the following iteration scheme

$$\mathbf{x}_{k+1}=\mathbf{x}_k+\alpha_k\mathbf{p}_k \tag{5.3}$$

where $\mathbf{p}_k$ is a search direction and α_k is a step length. The search direction $\mathbf{p}_k$ approximates the error $(\mathbf{x}^*-\mathbf{x}_k)$ and the optimality conditions at $\mathbf{x}^*$ should guide the definition of $\mathbf{p}_k$. The step length α_k is usually determined by a line search method.

These methods determine the search direction $\mathbf{p}_k$ by solving a linearly constrained quadratic programming (QP) subproblem

$$\text{minimize } \mathbf{g}_k^T\mathbf{p}+\frac{1}{2}\mathbf{p}^T\mathbf{B}_k\mathbf{p} \tag{5.4a}$$

$$\text{subject to } \nabla\mathbf{c}(\mathbf{x}_k)\mathbf{p}\geq-\mathbf{c}_k \tag{5.4b}$$

where $\mathbf{p}$ denote the decision variables, $\mathbf{g}_k$ denote $\nabla f(\mathbf{x}_k)$, and the matrix $\mathbf{B}_k$ represents the Hessian of the Lagrangian function.

For constrained problems the choice of step length α_k needs to ensure that the next iterate will not only decrease f but it also will come closer to satisfying the constraints. The most common approach is to choose α_k in Eq. 5.3 to produce a “sufficient decrease” in a merit function, which combines both the objective and constraint violations into one function. The two most common choices of the merit function are the l_1 merit function (Gill et al., 1981)

$$M(\mathbf{x})=f(\mathbf{x})-\sum_{i=1}^{m}\rho_i\min(c_i(\mathbf{x}),0) \tag{5.5}$$

where $\rho_i > 0$ are penalty parameters, and the augmented Lagrangian merit function (Gill et al., 2002)

$$M(\mathbf{x}, \boldsymbol{\lambda}, \mathbf{s}, \rho) \equiv f(\mathbf{x}) - \lambda^{\mathrm{T}}(\mathbf{c}(\mathbf{x}) - \mathbf{s}) + \frac{1}{2}\rho(\mathbf{c}(\mathbf{x}) - \mathbf{s})^T(\mathbf{c}(\mathbf{x}) - \mathbf{s}) \tag{5.6}$$

where $\mathbf{s}$ is a vector of slack variables to convert inequality constraint $c_i(\mathbf{x}) \geq 0$ to an equality constraint

$$c_i(\mathbf{x}) \geq 0 \text{ if and only if } c_i(\mathbf{x}) - s_i = 0,\ s_i \geq 0 \tag{5.7}$$

The objective function of the QP subproblem and the merit function of the line search step require estimates of the Lagrangian multipliers. These estimates can be obtained by solving an auxiliary problem or by using the optimal multipliers for the QP subproblem at the previous iteration (Gill et al., 1981).

5.3 Formulations of the Rate Allocation Problem

Formulation plays an important role in optimization. This section discusses the appropriate formulation for the nonlinear rate allocation problem. The focus is on how to handle the well chokes, one of the physical control variables of the rate allocation problem.

5.3.1 Optimization Problem

We considered the rate allocation problems for gathering systems with tree-like structures. A gathering system includes wells, links, and nodes. The simulation model for the gathering system is described in Chapter 3.

The objective function is the weighted sum of well rates. The physical decision variables include lift gas rates and well chokes that control the production rate of individual wells. The constraints include maximum/minimum flow rates and pressure constraints imposed on production wells and/or network nodes, and maximum amount of lift gas available for groups of gas-lift wells.

5.3.2 Formulation P1

A straightforward formulation of the well rate and lift-gas rate allocation problem is to choose the choke diameters and lift-gas rate as the decision variables and formulate the optimization problem as follows

$$\text{maximize } f(\mathbf{x}) \tag{5.8a}$$

$$\text{subject to } \mathbf{l} \le \mathbf{F}(\mathbf{x}) \le \mathbf{u} \tag{5.8b}$$

where $f(\mathbf{x})$ denotes the objective function, the weighted sum of well rates, $\mathbf{F}(\mathbf{x})$ denotes the constraint functions, the flow rates and/or pressures of wells and network nodes, and $\mathbf{l}$ and $\mathbf{u}$ denote the lower and upper bounds of the constraints. This standard formulation is named here Problem P1.

For Problem P1, given $\mathbf{x}$, a particular set of choke diameters and lift-gas rates, to evaluate $f(\mathbf{x})$ and $\mathbf{F}(\mathbf{x})$, the flow rates and pressures of some wells and facilities, the system equations of the gathering system, Eq. 3.24, have to be solved. In some cases, this can cause severe computational difficulties. For instance, if some wells are so deep that they cannot produce without an appropriate amount of lift-gas, Eq. 3.24 will have no feasible solutions unless appropriate amounts of lift-gas are allocated to those wells. In addition, Eq. 3.24 will have no feasible solutions if an excessive amount of gas is injected into a well while the choke for that well is set to a small value in the optimization process. Therefore, there are chances that the solution process for Problem P1 will fail because feasible solutions for Eq. 3.24 do not exist.

Another significant drawback of Problem P1 is that it is not good at handling well shut-down actions. When a well, say well j, is converted from open status to shut-down status, we need to solve two sets of system equations with different dimensionalities (one set without well j and one set with well j being slightly opened) to evaluate the partial derivatives of the objective function and constraint functions with respect to the choke diameter for well j. This procedure can be time-consuming and the results can be inaccurate, time-consuming because we need to solve two sets of nonlinear equations to obtain one partial derivative, inaccurate because we have to use finite differences to approximate the derivatives of a potentially nonsmooth function.

Some of these computational difficulties are inherent in the system of equations. As described in Section 3.5, Eq. 3.24 is formulated under the assumption that the fluid flows up the production trees. Replacing Eq. 3.24 with another system of equations may help resolve some of the computational difficulties, but some difficulties will persist, such as infeasible combination of lift-gas rates and choke settings, and well shut-down handling. Furthermore, as we will demonstrate by an example in Section 5.5.2, even when these computational difficulties do not appear, the computational efficiency of Formulation P1 can be disappointing.

5.3.3 Formulation P2

Here we propose a new formulation to overcome limitations of Problem P1 discussed in the previous section. The fundamental idea of the new formulation is to avoid solving the system of equations when evaluating the objective and constraint functions.

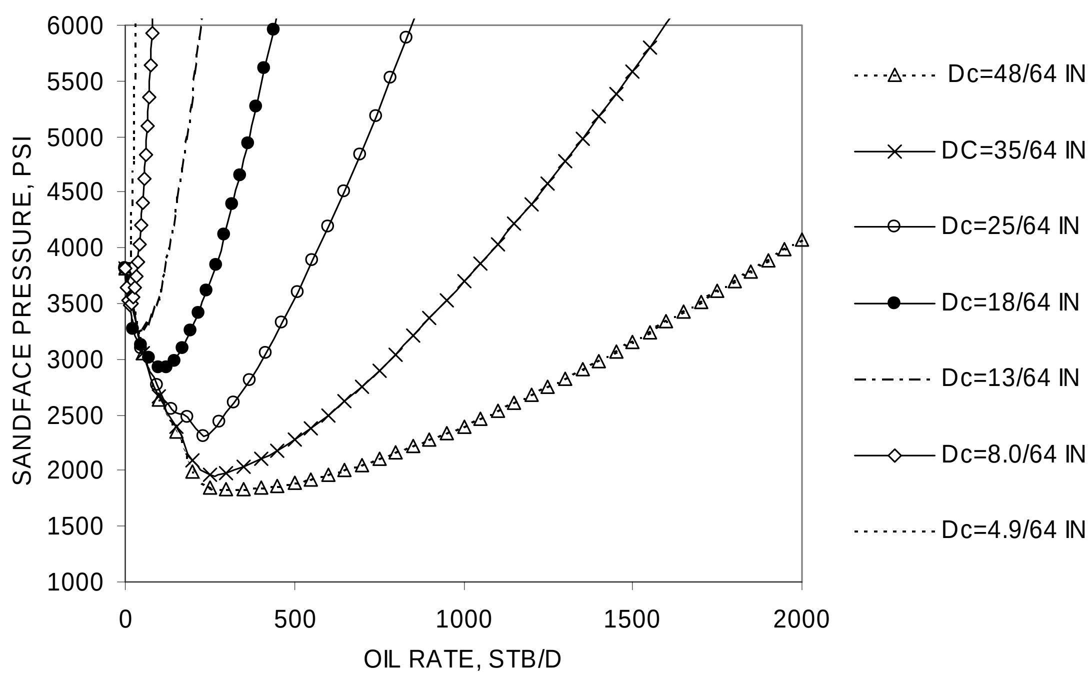


Figure 5.1: Outflow performance curves with various choke settings.

The new formulation chooses the oil rate and the lift-gas rate for each well as the decision variables and use the fact that for fixed flow rates the pressure drop across the choke increases as the choke closes. Figure 5.1 illustrates this point. For a set of oil and lift-gas rates, denoted as $\mathbf{x}$, we calculate the bottomhole pressures from two paths as described in Section 3.5.1. One path is from the reservoir side to the bottomholes. The other path is from a separator to the bottomholes. When performing the pressure transverse calculation from the separator side, we set the well chokes fully open. The set of oil and lift-gas rate $\mathbf{x}$ thus computed is within the deliverability capacity of the gathering system if we have

$$p_j^r - p_j^s \geq 0\ , \ \ j = 1,\ldots,n_w \tag{5.9}$$

where p_j^r is the bottomhole pressure for well j calculated from the reservoir side, p_j^s is the bottomhole pressure for well j calculated from the separator side. In other words, if a set of oil and lift-gas rates satisfies constraint Eq. 5.9, this set of oil and lift gas rate can be produced from the gathering system by, where appropriate, reducing choke diameters.

The well rate and gas-lift optimization problem is now formulated as Problem P2

$$\text{maximize } \ f(\mathbf{x}) \tag{5.10a}$$

$$\text{subject to } \ \mathbf{l} \leq \mathbf{F}(\mathbf{x}) \leq \mathbf{u} \tag{5.10b}$$

$$p_j^r - p_j^s \geq 0\ , \ \ j = 1,\ldots,n_w \tag{5.10c}$$

Constraint Eq. 5.10c is named as the *deliverability* constraint. This constraint should be modified to $p_j^r - p_j^s = 0$ if the choke diameter of well j is not a control variable. As in Problem P1, $f(\mathbf{x})$ and $\mathbf{F}(\mathbf{x})$ denote the flow rates and pressures of some wells or facilities. However, they are evaluated in a different way in Problem P2 than in P1. In Problem P2, given a set of oil and lift-gas rates, $\mathbf{x}$, the flow rate and bottom hole pressure of a well are computed from the reservoir side using appropriate well models when computing p_j^r; the flow rate and pressure in network nodes are computed from the separator side when computing p_j^s. Constraint 5.10c ensures that the optimal oil and lift-gas rate for Problem P2 are within the deliverability capacity of the gathering system. Thus these wells can be produced from the gathering system by appropriately adjusting the choke diameters.

In the solution process for Problem P2, we do not solve the system equations (Eq. 3.24) to evaluate the objective and constraint functions, thus most computational difficulties encountered in Formulation P1 can be avoided. After the optimization problem is solved, the required choke diameters can be determined according to the value of the control variables and the status of constraint 5.10c. This is explained further in Section 5.3.4.

Problem P2 simplifies the constraints on individual wells. For a production well, its water and gas rate, and the bottomhole pressure can be regarded as functions of its oil rate. Because Problem P2 takes the oil rate as one of the control variables, it can convert the rate and pressure constraints on individual wells to a bound constraint on the oil rate. In contrast, Formulation P1 has to treat the rate and pressure constraints on individual wells as nonlinear constraints. A nonlinear constraint is much more difficult to handle than a simple bound constraint (Gill et al., 1981).

5.3.4 Interpretation of Solutions of Problem P2

The optimal solution of Problem P2 includes the oil rate and lift gas rate of each well, and the status of deliverability constraint Eq. 5.10c (active or inactive). These results need further interpretation so that the desired choke diameters can be determined. For an arbitrary well j, its allocated oil rate, lift gas rate, and the status of corresponding deliverability constraint in the optimal solution can be interpreted as follows:

1. A zero oil rate and lift gas rate indicate that well j should be closed. Note, a zero oil rate really means that the oil rate for well j is at its lower bound, a small positive value (i.e., 1 STB/d). The lower bound of the oil rate is not allowed to be zero, because at zero oil rate, some partial derivatives required by the optimization process can not be evaluated.
2. A positive oil rate but inactive deliverability constraint ($p_j^r > p_j^s$) indicates well j should be partially closed (a positive oil rate means the oil rate for well j is greater than its lower bound). The desired choke diameter can be computed according to the desired pressure drop across the well choke for the optimal oil rate and lift gas rate of well j .

3. A positive oil rate and active deliverability constraint ($p_j^r = p_j^s$) *usually* indicates that well j should remain fully open. If the oil rate is close to its lower bound, then it may indicate well j should be closed. See next item (item 4) for explanation.
4. A zero (or close to the lower bound) oil rate, a (usually small) positive lift gas rate, and an active deliverability constraint indicate that well j should be closed. This is because, for certain wells, say well j, reducing its production rate improves the objective function. However, as its production rate reduces, the pressure drop in its tubing string increases and the corresponding deliverability constraint becomes active. To improve further the objective and keep the deliverability constraint feasible, the optimization algorithm has to reduce its production rate but allocate some lift gas to keep the deliverability constraint feasible. As shown in Figure 5.2, for a production well, increasing the oil rate and lift gas rate a little bit at low oil (or liquid) rate region can significantly reduce pressure drop in the tubing string (thus the sandface pressure calculated from the surface side decreases).
5. Infeasible problem. If certain deliverability constraints are infeasible because the corresponding wells are too weak to flow, the corresponding wells have to be manually shut down, and the optimization problem has to be re-solved.

Finally, it is necessary to check whether the optimal production rate is a stable production rate for the production system. Condition 3.32 presented in Section 3.5.3 can be used for this purpose.

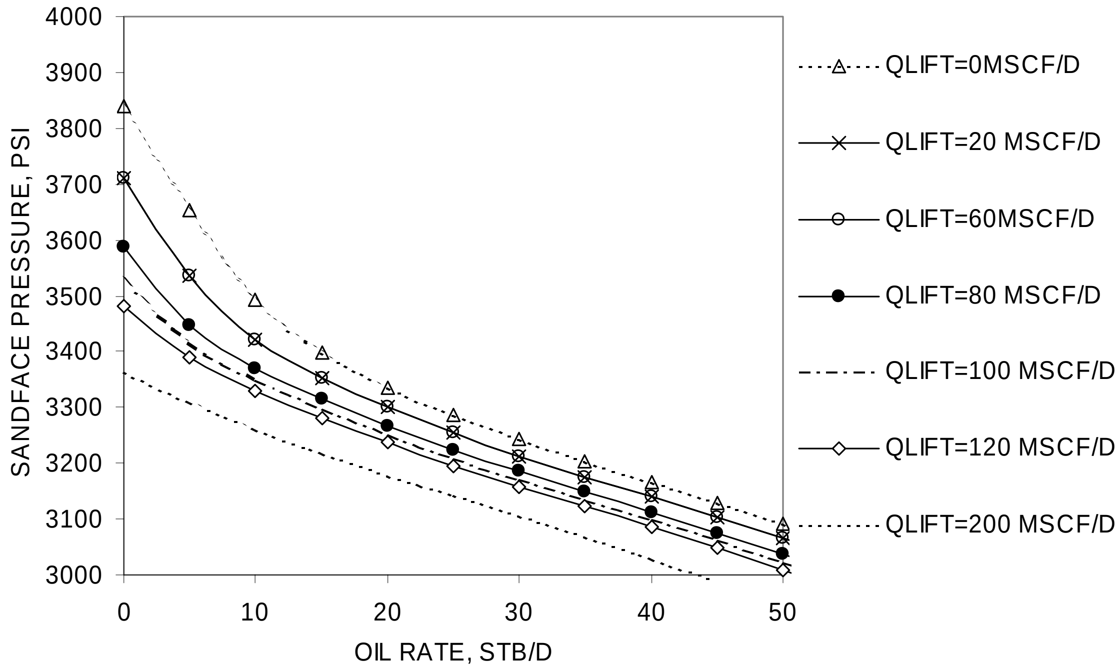


Figure 5.2 Impact of oil rate and lift gas rate on the well sandface pressure calculated from the separator side. Major characteristics of this well are as follows. The water cut is 0.26 STB/STB, GOR is 12.50 MSCF/d, tubing diameter is 3.5 inches, well depth is 8000 ft, well head pressure is 1080 psi.

5.4 Solving the Optimization Problem

Problems P1 and P2 are nonlinearly constrained optimization problems. In this work, they were solved by SNOPT (Gill et al., 1998), a general-purpose system for solving large-scale optimization problems. SNOPT implements a SQP algorithm that obtains search directions from a sequence of quadratic programming subproblems. SNOPT requires relatively few evaluations of the problem functions, hence it is especially effective if the objective or constraint functions are expensive to evaluate. The SQP algorithm used in SNOPT is described fully in Gill et al. (2002).

SQP is a derivative-based optimization algorithm. Successful application of SQP requires efficient and accurate evaluations of gradients of objective and constraint functions. In this work, we were able to compute the gradient information efficiently and accurately by exploring the tree-like structure of the gathering system and utilizing automatic differentiation techniques. Details of this procedure are presented in Section 3.5.2.

Experience with the application of SNOPT to the nonlinear rate allocation problem showed that the number of required function evaluations is sensitive to the starting point, the gathering system and the optimization problem, the convergence criteria, and the model and derivative accuracy. Therefore, the performance of SNOPT should be tuned properly whenever the condition of the production system or the optimization problem changes significantly.

5.5 Examples

We present three examples in this section. The first example compares the performance of the SQP method with the MILP-I method (see Section 4.3.2) for a well and lift-gas rate allocation problem. The second example compares the computational efficiencies of Formulations P1 and P2. The third example demonstrates the computational efficiency of Formulation P2 on problems of various sizes and properties. These examples have been presented in Wang et al. (2002b).

5.5.1 Comparison of the MILP-I Method and the SQP Method

The production system of this example is described in Section 3.7. The problem is to maximize the total oil production rates by allocating production rates and unlimited lift-gas to the ten production wells. The constraint is that the total water production rate can not exceed 5000 STB/d.

The optimization problem is solved by the MILP-I method presented in Section 4.3.2 as follows.

1. *Choose a realistic flowing pressure at the platform*. Suppose wells are producing with a set of lift-gas rate (not optimized) as shown in column 2 of Table 5.1. We simulated the multiphase flow in the gathering system and obtained the flowing pressure at the platform, which is 1545 psi. Table 5.1 also shows the oil flow rate, the water cut, and the gas oil ratio (GOR) for each well obtained in this step.

2. *Construct gas-lift performance curves*. First we fixed the platform pressure at 1545 psi. Then for any well, say well j, given its gas-lift rate, we could determine its oil flow rate by solving the multiphase flow problem for the flow path connecting well j and the platform. For each well, we computed the oil flow rate for different gas-lift rates, namely, 0, 200, 500, 1000, 2000, 3000, 4000, 5000, 6000, and 7000 MSCF/d, and obtain its gas-lift performance curve. Figure 5.3 plots the gas-lift performance curves we obtained in this step.
3. *Allocate lift-gas*. First we approximated each gas-lift performance curve using a piece-wise linear curve. Then we solved the rate allocation problem using the MILP-I method. The optimal solution predicted that the field will produce at an oil flow rate of 12449 STB/d. The allocated lift-gas rate and oil rate for each well are presented in Table 5.2.
4. *Update multiphase flow*. We allocated the optimal set of lift-gas rates from Step 3 to each well, simulate the multiphase flow in the gathering system, and obtained the oil flow rate of each well. Results indicated that the field can produce at a total oil flow rate of 12045 STB/d and a total water flow rate of 4830 STB/d, which does not violate the total water flow rate constraint. The flowing pressure at the platform is 1595 psi.

We then proceeded to solve the same optimization problem using the SQP method with Formulation P2. We selected the oil rate and lift-gas rate for each well as the decision variables. Starting with an initial guess of 400 STB/d of oil rate and 3000 MSCF/d of lift gas rate for every well, the program converged to the optimal solution using 27 function evaluations. One function evaluation refers to one complete evaluation of the objective and constraint functions and their derivatives. The allocated oil rate and lift-gas rate for each well are shown in Table 5.2. Results indicate that the field will produce at a total oil rate of 13019 STB/d and a total water rate of 5000 STB/d. The flowing pressure at the platform is 1079 psi. Note that in this example the flowing pressure at the platform is sensitive to the flow rates in the common flow line.

Table 5.1: Well rates obtained in Step 1 of the MILP-I method for Example 1.

Well	Lift-gas Rate (MSCF/d)	Oil Rate (STB/d)	Water Cut	Gas Oil Ratio (MSCF/STB)
1	3000.00	735.35	0.53	1.21
2	0	988.90	0.26	12.50
3	2000.00	2590.69	0.38	1.65
4	3000.00	828.18	0.70	1.02
5	0	1168.80	0.16	13.41
6	0	1143.09	0.07	6.84
7	2000.00	144.49	0.47	2.87
8	0	933.03	0.11	40.38
9	0	740.73	0.10	31.41
10	3000.00	2911.25	0.22	1.02
Total Flow Rate	13000.00	121842.52	6149.55*	10618.42**

* total water flow rate, STB/d
**total formation gas flow rate, MSCF/d

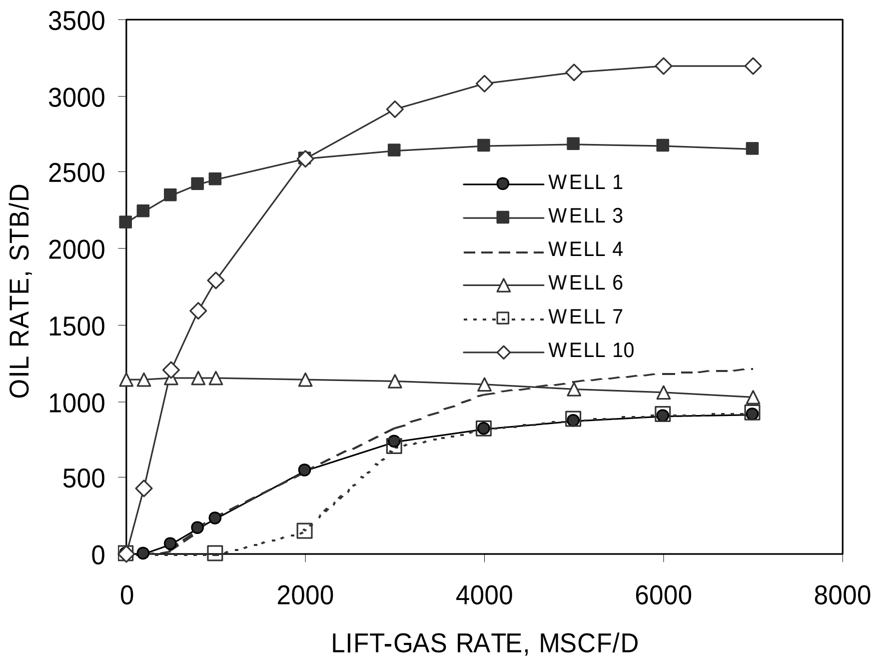


Figure 5.3: Gas-lift performance curves generated in Step 2 of the MILP-I method. For clarity, curves for high GOR wells (like well 6) are not shown.

Table 5.2: Allocated lift gas and oil rates for Example 1.

	Step 3 of the MILP-I Method		Step 4 of the MILP-I Method		The SQP Method	
Well	**Lift-gas Rate (MSCF/d)**	**Oil Rate (STB/d)**	**Lift-gas Rate (MSCF/d)**	**Oil Rate (STB/d)**	**Lift Gas Rate (MSCF/d)**	**Oil Rate (STB/d)**
1	2674.00	668.36	2674.00	649.60	-*	-*
2	0	988.64	0	947.07	0	1237.64
3	5000.00	2678.83	5000.00	2608.54	0	3233.68
4	-*	-*	-*	-*	-*	-*
5	0	1168.09	0	1149.46	0	1293.56
6	506.22	1149.29	506.00	1082.15	0	1621.08
7	7000.00	927.29	7000.00	872.57	750.15	3108.30
8	0	932.62	0	933.03	-*	-*
9	0	741.21	0	731.75	0**	270.72**
10	7000.00	3194.95	7000.00	3070.97	2769.25	4301.56
Total	22180.22	12449.24	22180.00	12045.15	3519.40	13019.66

* shut-in wells
** partially closed wells

Some comments on the MILP-I method and the SQP method are appropriate at this stage:

- In Step 2 of the MILP-I method, in order to construct the gas-lift performance curves, we needed to fix the platform pressure to a rather arbitrary value. The results of the MILP-I method will largely depend on this value. However, there is no simple way to tell which value is appropriate to choose.
- In Step 3 of the MILP-I method, the lift-gas was allocated based on existing gas-lift performance curves. This ignores the fact that increasing the flow rate from one well will decrease the flow rate from other wells sharing a common flowline. So the MILP-I method tends to over-inject lift-gas in Step 3.
- The SQP method takes the flow interaction into account through the deliverability constraint Eq. 5.10c, and is able to make more informed decisions. For example, well 8 and 9 have high GORs. For the same amount of oil produced, these two wells will cause a larger pressure drop through the common flowline than the rest of the wells, consequently the optimization algorithm will reduce the flow rates from other wells. The SQP method closed well 8 and partially closed well 9 to utilize fully the deliverability capacity of the common flowline. As a result, the SQP method allocated much less lift-gas but produces 8% more oil than the MILP-I method.
- The major computational effort of the MILP-I method is in constructing the gas-lift performance curves. Thus the computational time of the MILP-I method is

proportional to the number of wells in the system. The computational time of the SQP method depends on many factors and is hard to predict. For this particular example, the computational time required for the MILP-I method and the SQP method were of the same order.

Finally we mention that this problem cannot be solved by Formulation P1, because some wells (i.e. well 7) are too weak to flow under certain conditions and well shut-down actions (i.e. well 1, 4, and 8) must be performed in the optimization process.

5.5.2 Comparison of Formulation P1 and P2

Formulation P2 is easier to implement and is more robust than P1. However, Formulation P2 turns an unconstrained optimization problem into a constrained one. We must ensure that such a reformulation does not make an easy problem hard to solve. The purpose of this example is to show that the performance of Formulation P2 is not necessarily worse than P1 on problems that can be solved by both formulations.

In this example, the reservoir conditions and the configuration of the gathering system are exactly the same as in the example of Section 5.5.1. The difference is that the tubing strings and pipes were made shorter in this example so that all wells can produce without gas-lift.

We present two scenarios. The first scenario is an unconstrained gas-lift optimization problem. The objective is to maximize the total oil production by allocating lift-gas. Both Formulation P1 and P2 were used to solve this problem, and they yielded the same optimal solution, which is zero lift-gas rate for every well. However, Formulation P2 converged much faster than P1. Formulation P2 optimizes both the oil flow rate and lift-gas rate of every well. Starting with an initial guess of 400 STB/d oil flow rate and 5000 MSCF/d lift-gas rate for every well, Formulation P2 converged to the optimal solution with 21 function evaluations. Formulation P1 optimizes only the lift-gas rate. Starting with an initial guess of 5000 MSCF/d lift-gas rate for every well, Formulation P1 converged to the optimal solution with 67 equivalent function evaluations. The convergence histories Formulation of P1 and P2 are plotted in Figure 5.4.

The second scenario is a constrained well rates and gas-lift optimization problem. The objective is to maximize the total oil production rate by allocating lift-gas and adjusting choke diameters. The total water flow rate can not exceed 20,000 STB/d. Again, both formulations yielded the same optimal solution: zero lift-gas rate for all wells, and the well choke for well 4, the well with highest water cut, was partially closed. However, Formulation P2 was much more efficient than P1. Starting with an initial guess of 400 STB/d oil flow rate and 5000 MSCF/d lift-gas rate for every well, Formulation P2 converged to the optimal solution with 20 function evaluations. Starting with an initial guess of 5000 MSCF/d lift-gas rate and fully open choke status, Formulation P1 converged to the optimal solution with 82 equivalent function evaluations. The convergence history is plotted in Figure 5.5.

In both scenarios, Formulation P2 performed better than P1.

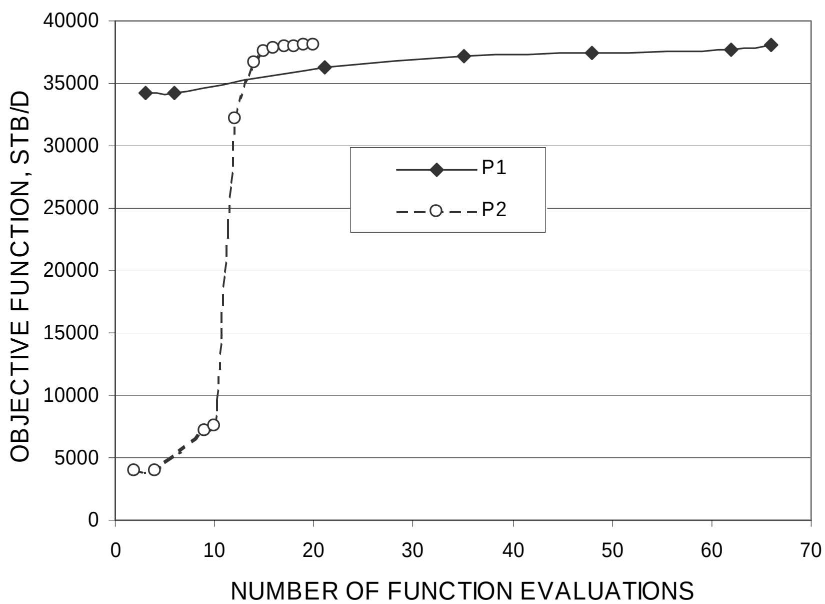


Figure 5.4: Convergence history of Formulation P1 and P2 for Scenario 1.

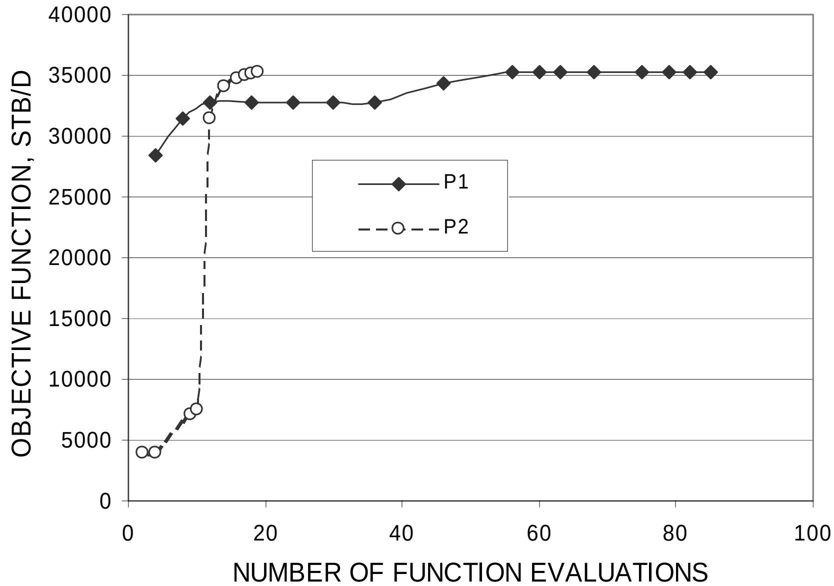


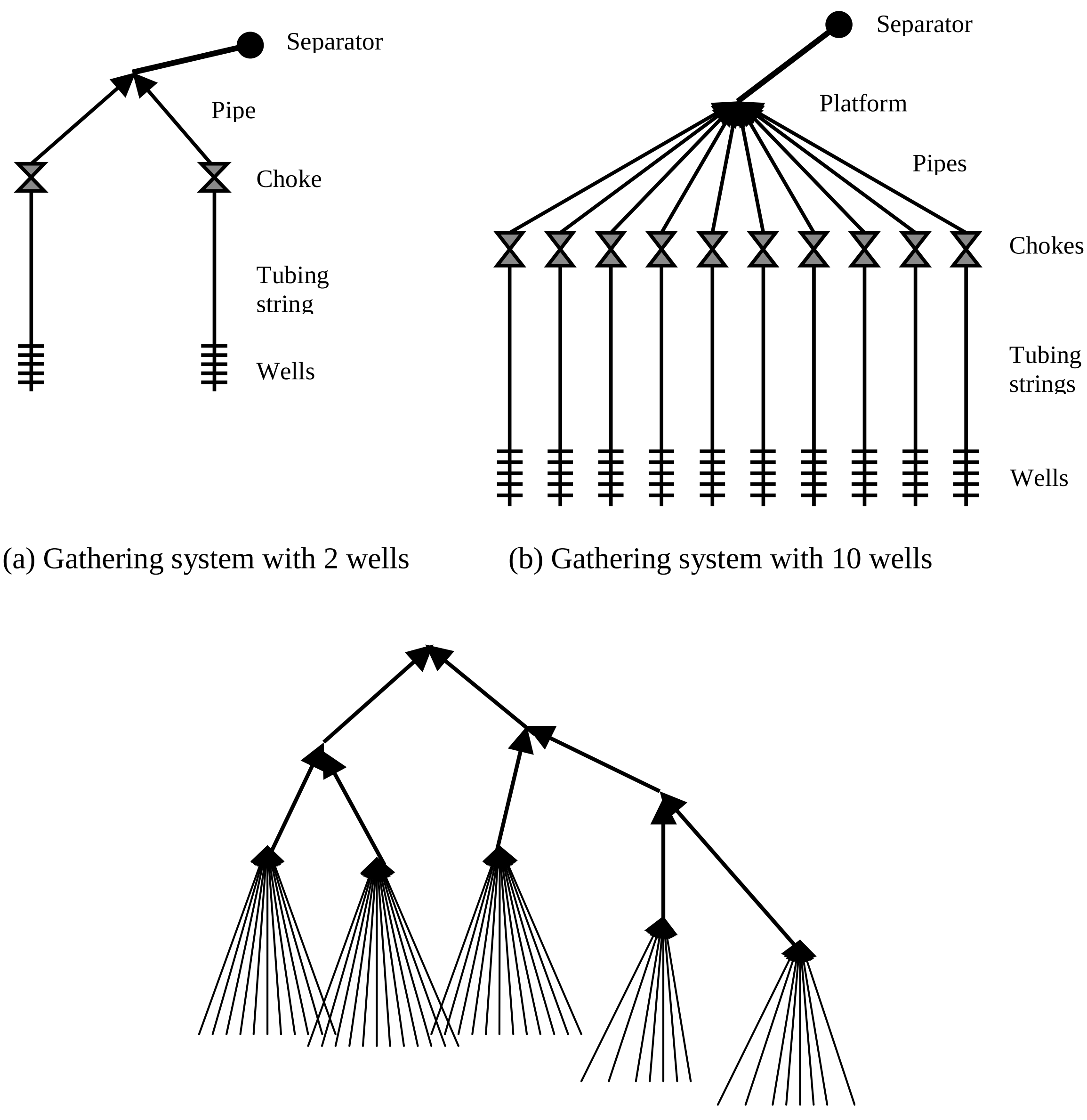
Figure 5.5: Convergence history of Formulation P1 and P2 for Scenario 2.

5.5.3 Efficiency of Formulation P2

This example demonstrates the computational efficiency of Formulation P2 on problems of various sizes and properties.

We considered three gathering systems with 2, 10, and 50 production wells, respectively. Figure 5.6 shows the configuration of these gathering systems. For each gathering system, we considered three optimization problems: unconstrained, constrained by total water flow rate, constrained by total gas and water flow rates. The objectives of all problems were to maximize the daily oil production by allocating the well and lift-gas rates. We solved the nine optimization problems using Formulation P2. The initial guess for every optimization problem was 400 STB/d oil flow rate and 3000 MSCF/d lift-gas rate for every well. The convergence criteria for the optimization process are presented in Table 5.3.

Table 5.4 shows the number of function evaluations required to solve the optimization problem and the total oil rate of the optimal solution for every problem. Constraints can greatly influence problem difficulty. For example, problems 7, 8, and 9 have the same size but require 62, 50, and 28 function evaluations, respectively. The number of function evaluations required is not sensitive to the problem size. All of the problems discussed here can be solved within about 60 function evaluations (one function evaluation is equivalent to one Newton iteration when Eq. 3.24 is solved by the Newton-Raphson method). In other words, all of the problems discussed here can be solved within approximately 10 times the number of iterations normally required to solve the network problem (defined by Eq. 3.24) by the Newton-Raphson method. Thus, we conclude that the SQP method with Formulation P2 is computationally efficient.



(a) Gathering system with 2 wells (b) Gathering system with 10 wells

(c) Gathering system with 50 wells (with a choke installed right before the manifold for every well)

Figure 5.6: Schematic illustration of gathering systems.

Table 5.3: Convergence criteria for optimization problems of Example 3.

Major Feasibility Tolerance*	5.0e-4
Major Optimality Tolerance*	1.0e-3
Minor Feasibility Tolerance*	5.0e-6
Minor Optimality Tolerance*	1.0e-6

* These are the major convergence criteria used in SNOPT (Gill et al., 1998). The major feasibility tolerance specifies how accurately the nonlinear constraints should be satisfied. The major optimality tolerance is used to judge the optimality of the optimization problem. The minor feasibility tolerance specifies how accurately the bound and linear constraints should be satisfied. The minor optimality tolerance is used to judge the optimality of each QP subproblem.

Table 5.4: Computational efficiency of Formulation P2 on various optimization problems.

Problem	Number of Wells	Constraints	Number of Function Evaluations	Total Oil Rate (STB/d)
1	2	None	21	1976.52
2	2	$q_w^t \le 1000$ STB/d	11	1480.38
3	2	$q_w^t \le 1000$ STB/d $q_g^t \le 20000$ MSCF/d	14	1267.21
4	10	None	49	14884.96
5	10	$q_w^t \le 8000$ STB/d	37	14585.92
6	10	$q_w^t \le 8000$ STB/d $q_g^t \le 50000$ MSCF/d	45	14333.10
7	50	None	62	83423.14
8	50	$q_w^t \le 35000$ STB/d	50	81698.21
9	50	$q_w^t \le 35000$ STB/d $q_g^t \le 300000$ MSCF/d	28	81067.22

5.6 Concluding Remarks

We have investigated a new formulation P2 for the optimization problem of allocating well rates and lift-gas rates simultaneously subject to multiple flow rate and pressure constraints. The optimization problem is solved by a SQP algorithm. Results demonstrated that the SQP method performs much better on certain problems than the MILP-I method. This is due to the fact that the MILP-I method optimizes the well rates and lift-gas rates based on gas-lift performance curves and ignores the deliverability constraint of the gathering system in the optimization process, while the SQP method makes no such simplifications. We determined that Formulation P2 is appropriate for the simultaneous optimization of well rates and lift-gas rates, because with this formulation, it is easier to handle well shut-down actions and avoid other computational difficulties. Furthermore, with P2 the optimization problem can be solved efficiently.

However there are some limitations of Formulation P2. In the optimal solution of P2, sometimes a well can have a zero oil rate and a positive lift-gas rate, which is not realistic. This positive lift-gas rate is used to lighten the oil column so that constraint Eq. 5.10c is satisfied. Fortunately, our experience shows that the value of this lift-gas rate usually is small and has little impact on the solution of the optimization problem.

More research work could be done on this subject. We can incorporate more decision variables into the optimization problem, such as pumps and subsurface chokes for multilateral and multisegment wells. We can modify Formulation P2 to allow loops in the network. It is also necessary to investigate the performance of the linear optimization methods and the nonlinear optimization method under different conditions.

Chapter 6

Optimization of Well Connections

6.1 Introduction

In some petroleum fields, a well or a flow line can have several potential output connections that join that well or flowline to different flowlines and facilities. Redirecting the well connections is an effective way to debottleneck the production system. The problem of well connection optimization is to identify the best set of well connections that maximizes an operational objective. To achieve the best results, well connections often need to be optimized simultaneously with production rates and lift gas rates (and potentially other production operations). Thus, the production optimization problem addressed in this section involves maximizing a certain operational objective (such as to maximize the daily oil rate) by optimally allocating the production and lift gas rates and well connections subject to multiple flow rate and pressure constraints. To differentiate this problem from the rate allocation problem addressed in previous chapters, the optimization problem addressed in this chapter is referred to as the *global problem*.

The global problem is a nonlinearly constrained optimization problem with both continuous (production and lift gas rates) and integer (well connections) variables. No existing formal optimization methods can be easily adapted to solve such a complicated problem. Litvak et al. (1997) presented a heuristic optimization procedure to allocate the production rates, lift gas rates, and well connections for the Prudhoe Bay oil field in Alaska. However, the performance of that procedure can be unsatisfactory because (1) the production rates, lift gas rates, and well connections are allocated sequentially rather than simultaneously; (2) the multiple flow rate constraints are checked and satisfied sequentially rather than simultaneously; and (3) the allocation procedures (either for well connections, production rates, or gas-lift rates) are too simple for a complex system like the Prudhoe Bay field.

In this chapter, a new formulation of the optimization problem is presented. Two solution methods are discussed. The performances of the solution methods were investigated and will be illustrated here by an example.

6.2 Two-Level Optimization Formulation

Although the physical decision variables for the global problem are the well connections, lift gas rates, and well chokes, it was shown in Chapter 5 that optimizing on well chokes is computationally inconvenient for the rate allocation problem. Because the rate allocation problem is a part of the global problem, production rates instead of well chokes are selected as decision variables for the global problem.

The rate allocation problem has been described extensively in Chapter 4 and 5. The global problem is formulated in such a way that existing methods for rate allocation can be utilized for the global problem. Let $\mathbf{x}$ denote a set of well connections, $\mathbf{y}$ denote lift gas rates, and $\mathbf{z}$ denote production rates. The global problem is formulated as a two-level optimization problem:

$$\text{maximize}_{\mathbf{x},\mathbf{y},\mathbf{z}} \quad f_u(\mathbf{x}) \tag{6.1a}$$

$$\text{subject to } \mathbf{x} \in \Omega_{wc} \tag{6.1b}$$

where Ω_{wc} is the domain of feasible well connections, $f_u(\mathbf{x})$ is defined as the optimal solution of a rate allocation problem

$$f_u(\mathbf{x}) = \text{maximize}_{\mathbf{y},\mathbf{z}} \quad f_l(\mathbf{x}, \mathbf{y}, \mathbf{z}) \tag{6.2a}$$

$$\text{subject to } \mathbf{l} \le \mathbf{F}(\mathbf{x}, \mathbf{y}, \mathbf{z}) \le \mathbf{u} \tag{6.2b}$$

where $f_l(\mathbf{x}, \mathbf{y}, \mathbf{z})$ is the objective function for the rate allocation problem parameterized by $\mathbf{x}$, $\mathbf{F}(\mathbf{x}, \mathbf{y}, \mathbf{z})$ are the constraint functions for the rate allocation problem parameterized by $\mathbf{x}$. $\mathbf{l}$ and $\mathbf{u}$ are the lower and upper bounds of the constraints, respectively. f_l and $\mathbf{F}$ are specified by the global problem. For example, if the global problem is to maximize the total field oil rate subject to a total field gas rate constraint, then f_l is the total field oil rate and $\mathbf{F}$ is the total field gas rate.

In this formulation, Problem 6.1 is named the *upper level* problem, and Problem 6.2 is named the *lower level* problem. The upper level problem is an unconstrained integer optimization problem. The lower level problem is a nonlinearly constrained rate allocation problem. The lower level problem can be regarded as a function evaluation procedure for the upper level problem.

This formulation separates the search on well connections and the search on production and lift gas rates into two levels. The lower level problem has been described extensively in Chapters 4 and 5. This chapter focuses on optimization methods for the upper level problem – optimization of well connections. The upper level problem is also referred to as the *well connection optimization* problem in this chapter.

6.3 Solution Methods for the Optimization of Well Connections

This section first motivates the choice of solution methods for the upper level problem, then describes the investigation of some possible solution methods.

6.3.1 Computational Issues

The upper level problem is a nonlinear integer programming (NIP) problem. In principle, a NIP problem can be solved by enumeration. However, complete enumeration is computationally impractical even for well connection optimization problems with a moderate number of connections. For example, consider a production system with 20 wells, each well has three potential output connections. A combination of these well connections represents 3^{20} or 3486784401 distinctive configurations of the production system. Suppose a computer can evaluate one configuration per second, it would take about 110 years to evaluate all the possible configurations, which is not practical.

So far, there does not exist a universal algorithm for integer programming: existing algorithms either require huge computational effort, or they only give approximate solutions (Bertsimas and Tsitsiklis, 1997). Because efficiency is an important consideration for real-time production optimization, we developed a partial enumeration method that gives only an approximate solution but is efficient for the upper level problem. A genetic algorithm was also used. The genetic algorithm has the ability to avoid local optima and it is easy to implement. The genetic algorithm can be used to crosscheck the performance of the partial enumeration method.

6.3.2 Partial Enumeration Method

The partial enumeration (PE) method is a heuristic method designed specially for the optimization of well connections. The method reconnects a well to its best output connection while keeping other well connections fixed. This process is repeated until no further significant improvement can be made. The basic steps of this method can be illustrated as follows.

Algorithm 6.1: The partial enumeration method.

1. Set iteration index $k = 0$.
2. Select a production well, well j, according to some operational rules.
3. For well j, perform the following operations:
   - 3a) Switch well j from its current output connection to another potential output connection. This step generates a new configuration of the production system.
   - 3b) Invoke optimization tools as described in Chapter 4 or Chapter 5 to reallocate the production and lift gas rate subject to imposed constraints. The optimal objective value from the rate allocation problem is regarded as the merit or value of the new output connection of well j. (This step solves the lower level problem.)
   - 3c) Repeat Step 3a)-3c) for all potential output connections of well j. Define the output connection with the maximum merit/value as the best output connection of well j.
   - 3d) Switch well j to its best output connection as determined in step 3c).

4. Repeat steps 2 and 3 for all production wells in the field.
5. Increase the iteration index k by 1.
6. Repeat Step 1 to 5 until certain user defined convergence criteria are met. One such criterion is: between iteration $k-1$ and k, no well changes its connection or no discernable increase of the objective function value is achieved.

The PE method mimics what a field operator facing the same problem would do. The method attempts to find a better system configuration at each search step, but it does not guarantee the global optimum.

The PE method is flexible. Different rate allocation algorithms can be employed in this method. Field-specific operational rules can be incorporated easily. For example, the actual implementation of Algorithm 6.1 can limit the maximum number of well reconnections so that the configuration of the production system does not change dramatically from day to day.

The performance of the PE method was evaluated and is illustrated in an example presented in Section 6.4.

6.3.3 Genetic Algorithm

The PE method is a heuristic method that examines only a portion of the complete search space. In addition, at each search step only one well is allowed to change its connection. This may cause some concerns for the quality of the solutions obtained from such a method. Genetic algorithm (GA) is a general, robust, and well developed optimization method. We used this method to crosscheck the performance of the PE method.

A genetic algorithm is a stochastic optimization algorithm based on natural selection. In GA, decision variables are encoded as a chromosome. Multiple chromosomes form a population. A GA evolves a population by means of selection, crossover, and mutation, etc. When the evolution terminates, the chromosome with the best fitness represents the optimal solution of the optimization problem. A flowchart of a simple GA is illustrated in Figure 6.1. A GA is applied to the well connection optimization problem in the following manner.

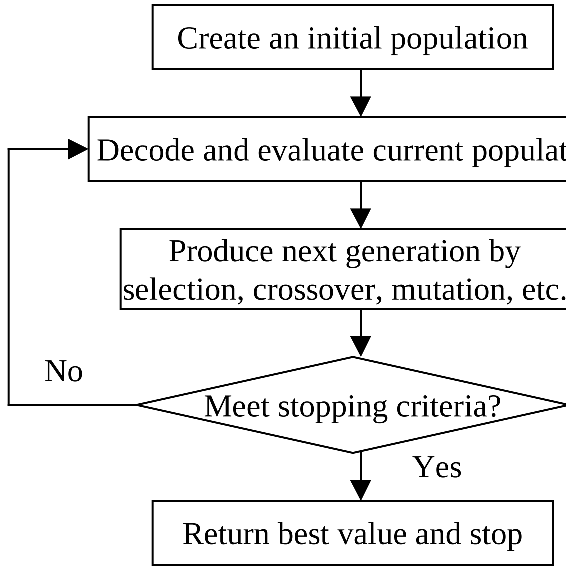


Figure 6.1: A simple GA flowchart.

Encoding/decoding. In GA, a set of decision variables is usually, though not necessarily, encoded as a binary string, with each single variable corresponding to a segment of the binary string. In GA vocabulary, the binary string is called a chromosome, and the segment of the binary string representing a variable is named a gene. With such an encoding scheme, decision variables take discrete values.

This study used the following encoding/decoding scheme for well connections. Suppose a well has n_c potential (output) connections and they are labeled by a set of consecutive numbers, $[0, n_c - 1]$. Then the encoding and decoding process for this well goes as follows:

- *Encoding.* A specific well connection is encoded into a gene by expressing its labeling number in binary format. The length of a gene is the minimum number of bits required to cover all potential connections of a well:

$$l = \min_m \{2^m \geq n_c\} \qquad (6.3)$$

  For example, suppose a well has three potential connections. Then its connections can be encoded into 00, 01, 10 respectively (Figure 6.2).

- *Decoding.* The decoding process maps a gene to a well connection by converting its binary expression to a natural number, k. If $k \in [0, n_c - 1]$, the gene is mapped to the well connection labeled as k. If $k \in [n_c, 2^l]$, the gene is mapped to the well connection labeled as $k-(n_c-1)$. Consider the above encoding example, the gene expressions 00, 01, and 10 can be mapped to well connections labeled 0, 1, and 2, respectively. A gene expression of 11 can be mapped to a well connection labeled 0. This decoding scheme is illustrated in Figure 6.2.

The genes for all decision variables comprise a chromosome.

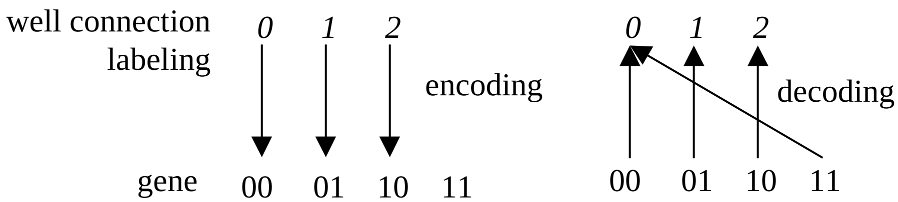


Figure 6.2: Encoding/decoding a well connection.

Initializing a population. This step generates the initial population (multiple sets of solutions or chromosomes) that GA begin to evolve with. One chromosome encodes the current set of well connections to ensure a good starting point. The rest of the chromosomes are randomly generated to ensure diversity. The number of chromosomes in the population is denoted as n.

Evolving the population. This is the major part of a GA, and is an iterative procedure with each iteration containing the following steps:

1. *Evaluate the population*. Decode each chromosome and evaluate its fitness (function value). For each chromosome, a rate allocation problem is solved and the optimal objective function value is assigned as the fitness of the corresponding chromosome.
2. *Produce next generation*. The current generation of the population is modified through selection, crossover, mutation, and other operations to produce the next generation of population.

   2a) Selection. Replicate chromosomes for mating. This study implements a tournament selection, in which a mating chromosome is selected as follows. Randomly pick two chromosomes from the current population, and the one with higher fitness value is selected as the mating chromosome. Repeat this procedure until n mating chromosomes are selected.

2b) Crossover. Randomly pick two chromosomes from the replicated population. Crossover the two chromosomes (the parent) to produce two new chromosomes (the children). There are different ways to crossover two chromosomes. One type of crossover is called single-point crossover, in which the two chromosomes swap their bits at a randomly selected position with a probability p_c. Figure 6.3 shows an example of single-point swap. Another type of crossover is called uniform crossover, in which the two chromosomes exchange their bits at every position based on a probability p_c. Figure 6.4 shows an example of uniform crossover. This study implements both crossover schemes.

1 1 1 1 1 1 1   crossover   1 1 1 1 0 0 0
0 0 0 0 0 0 0   ⟶   0 0 0 0 1 1 1

Figure 6.3: Schematic illustration of single-point crossover.

1 1 1 1 1 1 1   crossover   1 0 1 1 0 0 1
0 0 0 0 0 0 0   ⟶   0 1 0 0 1 1 0

Figure 6.4: Schematic illustration of uniform crossover.

2c) Mutation. Mutation is another way to increase the diversity of the population. In this step each bit in a chromosome flips according to some probability p_m. A mutation example is shown in Figure 6.5.

1 1 1 1 1 1 1   mutation ⟶   1 0 1 1 0 1 1

Figure 6.5: Schematic illustration of mutation.

2d) There are other techniques that can enhance the performance of the GA. This study used the following two: (1) the best parent is always copied to the next generation (elitism), and (2) if there is not enough diversity in a population (a population is converged because all chromosomes are too similar), start a new generation. The new generation copies the best chromosome from the converged population and generates the rest of the chromosomes randomly.

3. *Repeat Step 1 and 2 until stopping criteria is met*. The stopping criterion used in this study was a specified maximum number of generations.

The genetic algorithm used in this study was based on a genetic algorithm driver developed by Carroll (2000).

6.4 Comparison of the PE Method and the GA

We analyzed the performance of the PE method and the genetic algorithm on a model of the Western Operating Area of Prudhoe Bay oil field in Alaska.

Prudhoe Bay oil field is the largest oil field in north America. Currently there are 23 well pads in the Western Operating Area (WOA) and 16 drill sites in the Eastern Operating Area (EOA). Each well pad or drill site contains about 30-40 wells (Litvak et al., 1997). Oil production from Prudhoe Bay is on decline and is constrained by the gas and water handling limits of surface facilities and velocity constraints in flow lines. On the other hand, Prudhoe Bay has a flexible, highly automated and complex surface pipeline network. Production from many wells can be redirected to different flowlines and facilities (Litvak et al., 2002). For these reasons, optimization of well connections and production rates is a viable and economic way to increase the daily oil production in the Prudhoe Bay field. Therefore, the production optimization problem for the Prudhoe Bay field is to maximize the daily oil production by optimally allocating the well connections and production rates subject to multiple flow rate and velocity constraints. The field model of Western Operating Area contains about 500 hundred wells, each well has two to three potential connections. Roughly 100 wells are considered for re-connection. Other wells' connections remain fixed.

The production optimization problem was formulated as a two-level optimization problem. The lower level problem optimizes the production rates subject to all constraints. The lower level problem was solved by the LP-II method described in Section 4.3.4. The upper level optimizes the well connections. Both the PE method and the GA were used to solve the upper level problem.

The PE method is a deterministic heuristic technique. To see if the method was sensitive to its starting point, we made five test runs. Each run started the PE method from a randomly generated set of well connections. These runs found different sets of solutions after 400 function evaluations (one function evaluation refers to solving an instance of the lower level problem). However, the best objective values are close to each other. Table 6.1 shows the starting and final objective values of all runs. Figure 6.6 plots the convergence histories of the first three runs. The line labeled *PE BEST* in Figure 6.6 represents the best objective found by these runs. The objective value, which is the total oil rate, is normalized by a selected value. Figure 6.6 shows that the PE method increased the oil rate quickly at early stages, then slowed down at later stages. It is observed from Table 6.1 that the final objective values from these runs were not sensitive to the starting point. The biggest difference is 0.0013 of normalized oil rate.

Table 6.1: Performance data of the PE method.

	PE1	**PE2**	**PE3**	**PE4**	**PE5**
Starting Normalized Total Oil Rate	0.8911	0.9070	0.9352	0.8852	0.9202
Final Normalized Total Oil Rate	0.9935	0.9935	0.9922	0.9921	0.9928
Number of Lower Level Problems Solved	477	636	636	638	638

For the GA run, the following basic GA parameters were used: the number of chromosomes, n, of a population was 10, uniform crossover probability, p_c, was 0.5, and mutation probability, $p_m = 1/n = 0.1$. In this run, all initial chromosomes were generated randomly, the current set of well connections were not encoded as one of the initial chromosomes. In this way, the starting points for both the GA and the PE method were generated randomly and the comparison between the GA and the PE method is therefore more meaningful. After 1000 generations or 10,000 function evaluations (a function evaluation is defined in the same way as defined for the PE method), the best objective value obtained by the GA method was a normalized oil rate of 0.9922. The objective function was normalized in the same way as in the PE method. Figure 6.7 plots the convergence history of the GA method. The line labeled *PE BEST* represents the best objective function value obtained from the PE method. The early convergence history of the GA method is also plotted in Figure 6.6. The following observations are made from Figure 6.6 and Figure 6.7:

- The GA was not as efficient as the PE method. The GA converged to a value no larger than that from the PE method and it used 25 times more function evaluations than the PE method.
- The PE method is robust because it was not sensitive to its starting points and it achieved a better objective value than the GA.

The good performance of the PE method can be explained partially as follows. The configuration of a real production system that has been operated for a long period is already close to optimum. Such a system needs only fine adjustment in response to the changing operational conditions. The PE method, which changes at most one well connection at each step, is suitable for this purpose.

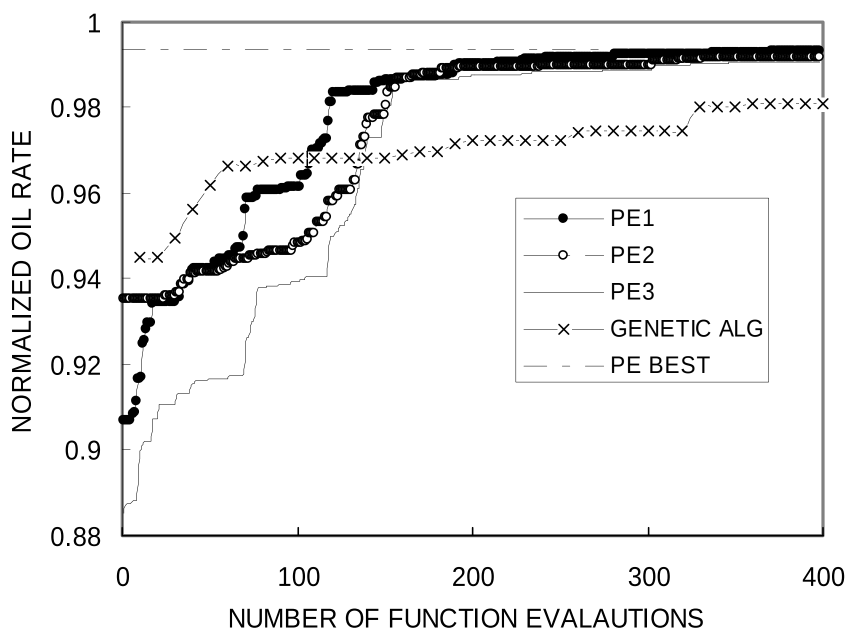


Figure 6.6: Convergence history of the PE runs.

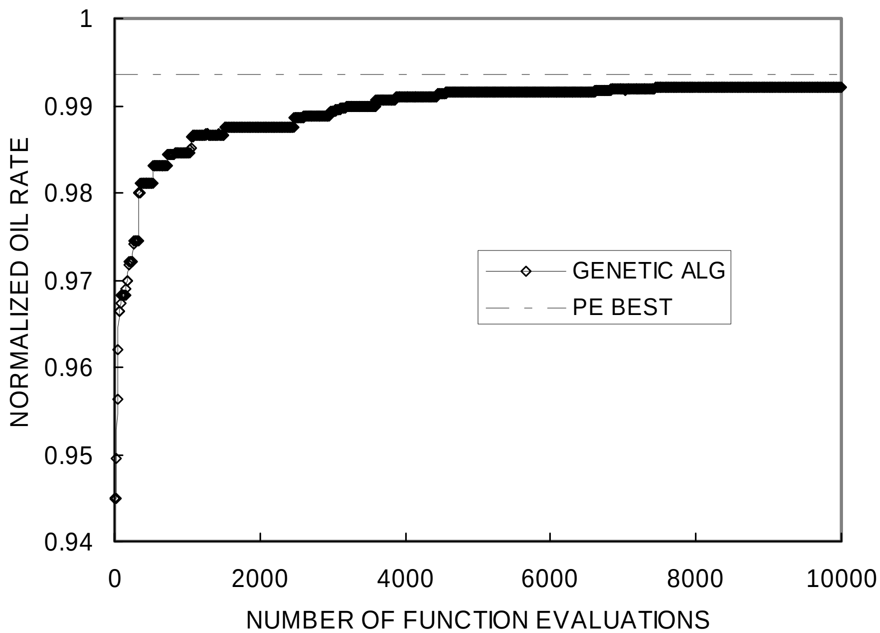


Figure 6.7: Convergence history of the GA run.

6.5 Concluding Remarks

The advantage of the two-level optimization is flexibility. The approach transforms the entire problem to an upper level problem and a lower level problem, and it allows the use of existing or new optimization methods separately for each level. This advantage is significant for the production optimization problem addressed in this chapter because no existing optimization algorithms can solve the entire problem efficiently. The solution quality of this approach depends on the properties of the global problem and which optimization methods are used for the two levels.

For the upper level problem, if the number of wells in a production system is small (say, less than 20), complete enumeration is possible to find the best set of well connections. For moderate or large-scale systems, approximate methods have to be used. The partial enumeration method proved to be both robust and efficient for several fields. However, as a specially designed heuristic method, the performance of the PE method has to be examined carefully before it is applied to a new field. Genetic algorithm provides an alternative solution method, but it is generally more expensive. The GA can, however, be used to test the robustness of the PE method.

Chapter 7

Multiobjective Optimization of Production Operations

7.1 Concept of Multiobjective Optimization

The optimization problems addressed in previous chapters assume a single objective function. In practice, petroleum production engineers often strive to achieve more than one goal when operating a field. Typical goals include maximizing the daily oil production, minimizing the production cost, and minimizing safety and environmental hazards. In most cases, it is unlikely that different objectives can be optimized by the same set of decision variables. The problem of optimizing multiple and distinct objectives is named *multiobjective optimization problem* (MOP) (Miettinen, 1999). Mathematically, a MOP can be expressed as

$$\text{minimize } \mathbf{F}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})] \tag{7.1a}$$

$$\text{subject to } \mathbf{x} \in \mathbf{S} = \{\mathbf{x} : c_i(\mathbf{x}) \le 0, i = 1, \ldots, m\} \tag{7.1b}$$

where $f_1(\mathbf{x})$, $f_2(\mathbf{x})$,..., $f_k(\mathbf{x})$ are the k objective functions or *attributes*, $\mathbf{x}$ is an n dimensional vector of decision variables, $\mathbf{S}$ denotes the feasible set of $\mathbf{x}$, and $c_i(\mathbf{x})$ denotes a constraint. The space to which the objective vector $\mathbf{F}(\mathbf{x})$ belongs is called the *objective space* or *criterion space*. The feasible set of $\mathbf{F}(\mathbf{x})$ is called the *attained set*, which is denoted by

$$\mathbf{Y} = \{\mathbf{F}(\mathbf{x}) : \mathbf{x} \in \mathbf{S}\} \tag{7.2}$$

In multiobjective optimization, no solution is optimal for all the k objectives simultaneously. Thus the optimality criterion used for single objective optimization must be replaced by a new one. One such replacement is the notion of *Pareto optimality* (Miettinen, 1999). Consider Problem 7.1 and two solution vectors $\mathbf{x}$ and $\mathbf{y}$. $\mathbf{x}$ is dominant over $\mathbf{y}$ if (Andersson, 2001):

$$\forall i \in \{1,2,\ldots,k\}: f_i(\mathbf{x}) \leq f_i(\mathbf{y}) \text{ and } \exists j \in \{1,2,\ldots,k\}: f_j(\mathbf{x}) < f_j(\mathbf{y}) \tag{7.3}$$

A point $\mathbf{x}^* \in \mathbf{S}$ is said to be *Pareto optimal* for a MOP if and only if there is no $\mathbf{x} \in \mathbf{S}$ is dominant over $\mathbf{x}^*$. All the Pareto optimal solutions form the *Pareto Optimal set*. The solutions in the Pareto optimal set are not dominant over each other. Moving from one point to another point within the set will deteriorate at least one objective or attribute. In most cases, the Pareto optimal set is on the boundary of the feasible set $\mathbf{Y}$ (see Figure 7.1).

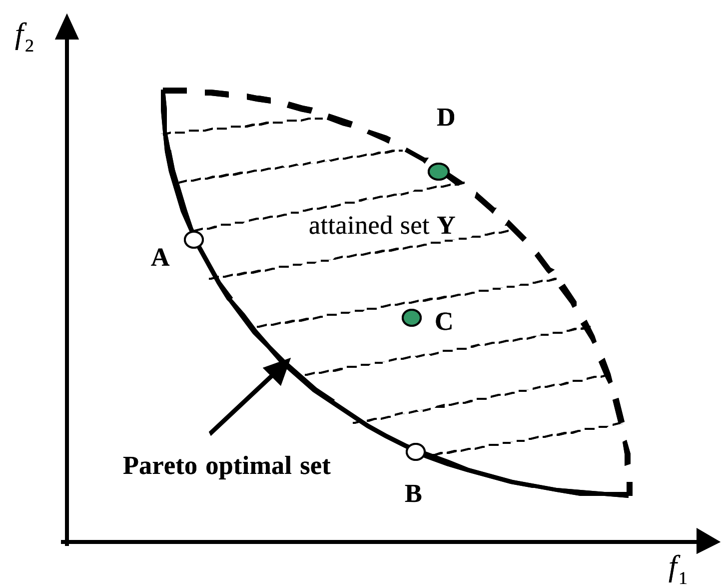


Figure 7.1: Graphical illustration of a Pareto optimal set. The problem is to minimize both f_1 and f_2. The elliptic area is the attained set of the problem. The solid line is the Pareto optimal set. Points A and B are in the Pareto optimal set, points C and D are not.

If a point $\mathbf{x}$ is not in the Pareto optimal set, there always exists a point $\mathbf{y} \in \mathbf{S}$ that is dominant over $\mathbf{x}$. If a point $\mathbf{x}$ is in the Pareto optimal set, there is no solution that is better than $\mathbf{x}$ in all attributes. Therefore, it is rational to choose the final solution for a MOP from its Pareto optimal set.

7.2 Solution Techniques for Multiobjective Optimization Problems

The solutions in a Pareto optimal set are not dominant over each other. In order to determine the final solution for a MOP, subjective judgement or preference information from the decision maker is required. The preference information is used to convert a MOP to one or a series of scalar optimization problems. According to Hwang et al. (1980), the solution methods for MOP can be classified into four categories depending on when the preference information of the decision maker is articulated in the process of multiobjective decision making: (1) no-preference methods, (2) *a priori* methods, (3) interactive methods, and (4) *a posteriori* methods. Table 7.1 lists a few major methods for each category.

Table 7.1: A classification of methods for multiobjective optimization (after Huang et al., 1980).

No-Preference Methods	Global Criterion Method (Salukvadze, 1974)
***A Priori* Methods**	Utility Functions (Keeney and Raiffa, 1976) Lexicographic Method (Fishburn, 1974) Goal Programming (Charnes and Cooper, 1961) Hierarchical Method (Azarm, 2002)
Interactive Methods	Method of Zionts-Wallenius (Wallenius, 1975) STEM method (Benayoun et al., 1971) Interactive MOLP (Steuer, 1977)
***A Posteriori* Methods**	Weighting Functions Method (Gal and Nedoma, 1972) e -constraint method (Haimes et al., 1975)

Section 7.2.1-7.2.4 gives a brief description of each category. Methods investigated in this study are described in detail where appropriate. The materials of Section 7.2.1-7.2.4 are based on Hwang et al. (1980) and Azarm (2002).

7.2.1 No-Preference Methods

No-preference methods require no subjective preference information from the decision maker. Multiple objectives are aggregated into one objective function without any opinions of the decision maker. Then a scalar optimization problem is solved and the solution is presented to the decision maker who may accept or reject the solution. The advantage of this approach is that the decision maker is not disturbed in the solution process and the method is easy to use. The disadvantage is that the decision maker has no control over the solution quality.

For example, in the *method of Global Criterion* (Salukvadze, 1974), the objective vector is aggregated into a scalar objective function in such a way that the deviation of the objective vector from an ideal objective vector is minimized:

$$f(\mathbf{x}) = \sum_{i=1}^{k} \left(\frac{f_i(\mathbf{x}) - f_i^*}{f_i^*} \right)^p \tag{7.4}$$

where f_i^* is the ideal value of the i th objective and is usually obtained by minimizing f_i individually while ignoring the rest objectives. Boychuk and Ovchinnikov (1973) suggests $p = 1$ while Salukvadze (1974) recommends $p = 2$.

7.2.2 *A Priori* Methods

A priori methods require that the preference information from the decision maker is given before the multiobjective optimization is conducted. Thus the decision maker must have some *a priori* understanding of the optimization problem.

Utility function methods. In these methods, a *utility function* is used to express the decision maker's preference information over the objectives:

$$U(\mathbf{F}(\mathbf{x})) = U(f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_3(\mathbf{x})) \tag{7.5}$$

where $U(\mathbf{F})$ is a utility function of the multiple objectives. The advantage of the utility function methods is that once the utility function is appropriately formulated, the most satisfactory solution of the MOP can be obtained. However, even for a simple problem, the utility function is very difficult to construct.

A special form of the utility function is to use weighting coefficients to indicate the importance of each objective:

$$U(\mathbf{F}(\mathbf{x})) = \sum_{i=1}^{k} w_i f_i(\mathbf{x}) \tag{7.6}$$

Methods using utility function given by Eq. 7.6 are also called *the weighted sum* methods.

Goal Programming. *Goal Programming* (GP) was first developed by Charnes and Cooper (1961). For this method, the decision maker must construct a set of goals and rank the importance of the goals. Problem 7.1 is formulated in GP as follows:

$$\text{minimize } \left[P_1 h_1(\mathbf{d}^-, \mathbf{d}^+), P_2 h_2(\mathbf{d}^-, \mathbf{d}^+), \ldots, P_l h_l(\mathbf{d}^-, \mathbf{d}^+)\right] \tag{7.7a}$$

$$\text{subject to } c_j(\mathbf{x}) + d_j^- - d_j^+ = b_j, \; j = 1,2,\ldots,m. \tag{7.7b}$$

$$f_i(\mathbf{x}) + d_i^- - d_i^+ = b_i, \; i = 1,2,\ldots,k \tag{7.7c}$$

$$d_i^-, d_i^+ \geq 0 \text{ for all } i \tag{7.7d}$$

$$d_i^- \bullet d_i^+ = 0 \text{ for all } i \tag{7.7e}$$

where b_j is the aspiration level of the jth goal, d_j^- is the negative deviation from the jth goal, d_j^+ is the positive deviation from the jth goal, $h_j(\mathbf{d}^-, \mathbf{d}^+)$ is a function of the deviation values called the jth achievement function, and P_j is the weighting coefficients for the jth achievement function. The achievement functions are absolutely ordered, which means $P_j >> P_{j+1}$ and there does not exist a number N that makes NP_{j+1} greater than P_j.

The solution procedure for GP goes as follows:

1. Minimize the first achievement function h_1, and let $\min h_1 = h_1^*$. Let $j = 1$.
2. Let $j = j + 1$. Minimize the jth achievement function h_j while requiring $h_i \leq h_i^*$ for $i = 1,\ldots, j-1$. That is, when minimizing an achievement function, achievement functions with higher rank can not be deteriorated. Denote the optimal value of the jth achievement function as h_j^*.
3. Repeat Step 2 until all achievement functions are minimized.

Hierarchical optimization method. This method (Azarm, 2002) allows the decision maker to rank and minimize the objectives in descending order of importance. Suppose the k objectives are ordered from f_1 (most important) to f_k (least important), the solution procedure can be illustrated as follows:

1. Find the optimum point $\mathbf{x}^{*,1}$ for f_1, subject to the original set of constraints

$$\text{minimize} \quad f_1(\mathbf{x}) \tag{7.8a}$$

$$\text{subject to} \quad c_i(\mathbf{x}) \le 0, \ i = 1,\ldots,m \tag{7.8b}$$

   Let $f_1^*(\mathbf{x}^{*,1})$ denote the optimal objective function value for Problem 7.8. Let $j = 1$.

2. Let $j = j + 1$. Find the optimum point $\mathbf{x}^{*,j}$ for the jth objective function f_j subject to the original and an additional set of constraints

$$\text{minimize} \quad f_j(\mathbf{x}) \tag{7.9a}$$

$$\text{subject to} \quad c_i(\mathbf{x}) \le 0, \ i = 1,\ldots,m \tag{7.9b}$$

$$f_i(\mathbf{x}) \le (1 + \boldsymbol{e}_i) f_i^*(\mathbf{x}^{*,i}), \ i = 1,\ldots, j-1 \tag{7.9c}$$

   where $\boldsymbol{e}_i \ge 0$ is the coefficient of function increment for f_i. Denote the optimal objective value of Problem 7.9 as $f_j^*(\mathbf{x}^{*,j})$.

3. Repeat Step 2 until j reaches the total number of objectives k.

7.2.3 Interactive Methods

For interactive methods, the decision maker progressively articulates preference information by interacting with the solution process. The decision maker controls the solution process by updating his preference as he learns about the problem. The advantage of these methods is that no *a priori* information is required from the decision maker. As the solution process goes, the decision maker learns about the problem and guides the solution process according to what has been learned. The solution process is a learning procedure for the decision maker and the decision maker is likely to accept the solution. The disadvantage of this method is that a great deal of effort is required from the decision maker.

Interactive methods often have a calculation phase and a decision phase. The calculation phase gathers the trade-off information, and the decision phase uses the trade-off information to guide the next search step. For example, see the methods of Zionts-Wallenius and the STEM methods reviewed in Huang et al. (1980).

7.2.4 *A Posteriori* Methods

A posteriori methods require no preference information from the decision maker before and during the solution process. The methods construct a subset of Pareto optimal from which the decision maker makes a subjective judgement and selects the most satisfactory solution. The advantage of this method is that the decision maker usually has a better understanding of the trade-offs because he/she has a discrete sampling of the Pareto optimal set. However, the computational load of this method is heavy, especially when there are more than two objective functions. Another disadvantage is that the decision maker may have too many solutions to choose from.

To construct a discrete sampling of the Pareto optimal set, the common theme of this class of methods is to vary systematically the algorithm parameters expressing the decision maker's preference information. For example, the e-constraint method (Haimes, 1975) chooses one objective for minimization and reformulates other objectives to constraints whose right-hand side is a user-controlled variable e. Solving the resulting scalar optimization problems for various e will generate a set of Pareto set of optimal solutions. The Hierarchical method (Azarm, 2002) described in Section 7.2.2 is able to generate an even-spread sampling of the Pareto optimal set by varying parameter e_i. Evolutionary multiobjective optimization algorithms (Fonseca and Fleming, 1993) try to evolve a set of solutions in such a way that the set of solutions spread to the Pareto optimal front.

The weighting functions method. The weighting functions method (Gal and Nedoma, 1972) samples the Pareto optimal set by changing the weighting coefficients of the weighted sum method:

$$\max \; f(\mathbf{x}) = \sum_{i=1}^{k} w_i f_i(\mathbf{x}) \tag{7.10a}$$

$$\text{subject to } \mathbf{x} \in \mathbf{S} \tag{7.10b}$$

The weighting functions method can be interpreted geometrically. Consider a MOP problem with two objectives f_1 and f_2. The weighting function $f = w_1 f_1 + w_2 f_2$ defines a line L with slope $-w_1 / w_2$ in the objective space (Figure 7.2). Minimization of function f can be seen as moving line L towards the origin while keeping the intersection of line L and the feasible region $\mathbf{Y}$. The optimum point is obtained when line L is tangent to the boundary of the feasible region.

If the Pareto optimal set is not convex, some points in the set can not be allocated by the weighting function method no matter what weights are used. This is illustrated in Figure 7.3. The line tangent to point A can be moved further to points B and C. Thus point A can not be allocated by the weighting function method.

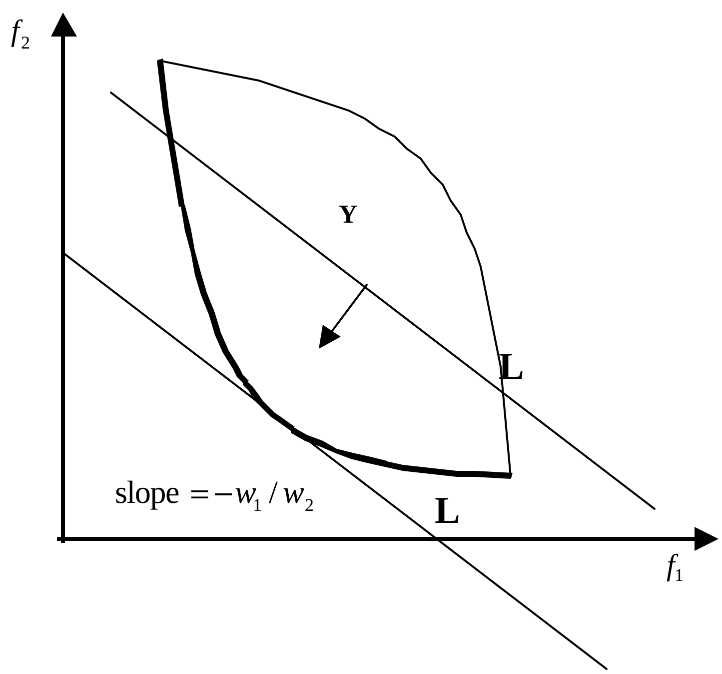


Figure 7.2: The weighting functions method with convex set (based on Huang et al. (1980) and Azarm et al. (2002)).

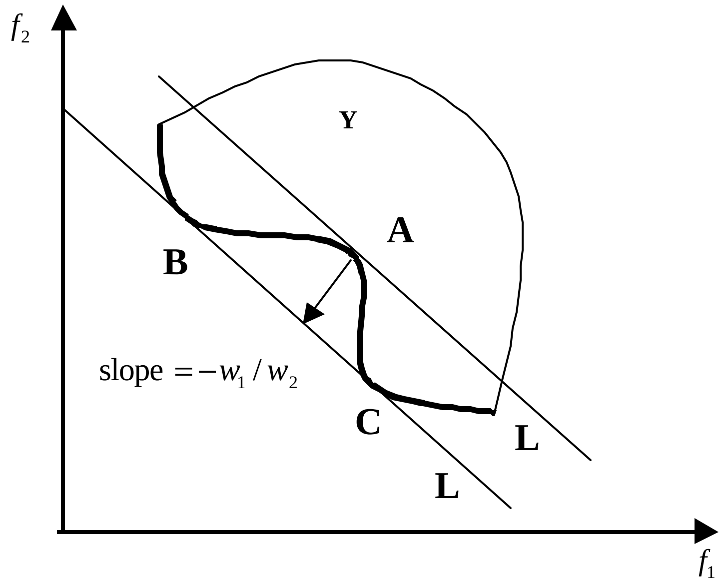


Figure 7.3: The weighting functions method with nonconvex set (based on Huang et al. (1980) and Azarm et al. (2002)).

7.3 Multiobjective Optimization of Production Operations

Production operations in a petroleum field aim to maximize profit from the field subject to restrictions of facility capacities, commercial contract, safety and environmental regulations, and other considerations. In optimization terminology, this task breaks down to the following components:

- *Objectives*. For day-to-day production operations, the objectives typically include maximizing daily oil production rate to meet contract target, and minimizing lift gas rate to minimize production cost, etc.
- *Hard constraints*. Hard constraints refer to constraints that are well defined and cannot be violated, such as the fluid processing capacities of surface facilities.
- *Soft constraints*. Soft constraints refer to constraints that are not well defined and/or can be violated, such as the maximum flow rate or pressure limit on individual wells or the target of the total oil rate from the field. Soft constraints can be viewed as objectives.

The soft constraints need to be formulated in mathematical language so that they can be handled in the optimization process.

7.3.1 Handling Soft Constraints

In this study, the soft constraints were handled in a way similar to how the goals are handled in goal programming (Charnes and Cooper, 1961). The soft constraints can be regarded as one type of goals and are converted to a set of hard constraints and an objective function by introducing some deviation variables. Consider the following set of soft constraints

$$c_i(\mathbf{x}) \le b_i\,,\; i = 1,..,m \tag{7.11}$$

where $c_i(\mathbf{x})$ is a soft constraint function and b_i is a limit value. Constraint Eq. 7.11 can be converted to the following optimization problem, which we will call Problem 7.12:

$$\text{minimize } h(\mathbf{d}) \tag{7.12a}$$

$$\text{subject to } c_i(x) - d_i \le b_i\,,\; i = 1,..,m \tag{7.12b}$$

$$d_i \ge 0\,,\; i = 1,..,m \tag{7.12c}$$

where d_i is a deviation variable and h is a function of the deviation variables. A simple example of h is the weighted sum of the deviation variables

$$h(\mathbf{d}) = \sum_i (w_i d_i) \tag{7.13}$$

where w_i is a user-supplied weighting coefficient.

Soft constraints with the following forms can be treated in a similar manner

$$c_i(\mathbf{x}) \ge b_i \text{ or } c_i(\mathbf{x}) = b_i \tag{7.14}$$

7.3.2 Solution Methods

This section presents a framework to formulate and solve the multiobjective production optimization problems. The actual implementation is limited to the nonlinear rate allocation problem described in Chapter 5.

Decision variables. It is necessary to specify which wells' lift gas rates are to be optimized. The oil flow rate of every production well and deviation variables for soft constraints are automatically selected as decision variables. The lower and upper bounds and the initial values for all decision variables are specified in advance.

Hard constraints. Multiple flow rate constraints can be specified for any well or network node.

Soft constraint. Multiple soft (oil, gas, water flow rate) constraints can be specified for any well or network node. The whole set of soft constraints can be divided into several subsets. Each subset can be converted to a set of hard constraints and an objective function defined by Eq. 7.13 (this objective function is considered as an *objective variable* below).

Objectives. The problem may include multiple objectives. Each objective is a weighted sum of a set of objective variables:

$$f_j(\mathbf{y}) = \sum_i w_i y_i \tag{7.15}$$

where f_j denotes the j th objective, y_i denotes an objective variable, and w_i denotes a weighting coefficient. An objective variable can be a single decision variable or a function of multiple decision variables. For example, an objective variable can be oil, gas, and water flow rates, or the pressure of a well or a network node, or a function of some deviation variables defined by Eq. 7.13.

Solution method. The primary solution method investigated in this study was the Hierarchical optimization method (Azarm, 2002) described in Section 7.2.2. To use this method, the user needs to rank the k objectives in descending order of importance and specify a relaxation parameter e_i for each of the first $k-1$ objectives.

The Hierarchical method was adopted because

1. The method is easy to use and its logic is easy to understand for the decision maker.
2. The method is able to locate a Pareto optimal solution that resides in the nonconvex part of the Pareto optimal set.

The proposed framework is flexible. The Hierarchical method can be converted to other popular multiobjective methods by manipulating the algorithmic parameters. For example, the method reduces to the Lexicographic method (Fishburn, 1974) if all e_i are set to zero; the method reduces to the goal programming method (Charnes and Cooper, 1961) if all the objectives are defined as soft constraints and all e_i are set to zero; and the method reduces to the weighting functions method if all objective variables are aggregated into one objective function using Eq. 7.15. The Hierarchical method can be used as an *a priori* method, an interactive method, or an *a posteriori* method depending on how the adjustable algorithmic parameters (weighting coefficients w_i and relaxation parameters e_i) are adjusted in the optimization process (no adjustment, adjusted interactively by the decision maker, or adjusted by the analyst to generate a sampling of the Pareto optimal set, respectively).

7.4 Example

The production system is the same as the one presented in Section 3.7. The configuration of the gathering system is illustrated in Figure 3.4.

Suppose that the produced gas is not for sale and it is to be reinjected into the reservoir for storage. Let us assume that the production engineer wants to operate the field with the following goals:

1. Produce the field with a total oil production rate $q_{o,t}$ of exactly 14,000 STB/D.
2. Minimize the total water flow rate q_w^t.
3. Minimize the total gas flow rate q_g^t.

Since the field is able to produce at an oil flow rate higher than 14,000 STB/D, the first goal is to formulate the total oil production target as a hard constraint. The other two goals are formulated as two objectives.

The cost of each barrel of produced water and cubic feet of produced gas depends on many factors and may be hard to evaluate quantitatively. Therefore it is difficult to come up with utility functions to aggregate the last two objectives into one objective function. An appropriate solution is to construct a subset of the Pareto optimal set for this problem from which the production engineer can choose the most satisfactory solution.

Both the hierarchical method and the weighting functions method were used to construct the Pareto optimal set. The solution procedure of the hierarchical method goes as follows:

1. Minimize the total water flow rate of the field with the following constraint:

$$q_o^t = 14{,}000 \text{ STB/D} \tag{7.16}$$

Denote the minimum water flow rate value as $q_w^{t,*}$. Let $j = 1$.

2. Minimize the total gas flow rate of the field subject to following constraints:

$$q_o^t = 14{,}000 \text{ STB/D} \tag{7.17}$$

$$q_w^t \le (1 + j * 0.1) q_w^{t,*} \tag{7.18}$$

3. Repeat Step 3 for $j = 2,3,\ldots$ until the total gas flow rate no longer decreases as j increases.

In the weighting functions method, the multiobjective optimization problem was transformed to the following scalar optimization problem

$$\text{minimize } f = w_w q_{w,t} + w_g q_{g,t} \tag{7.19a}$$

$$\text{subject to } q_o^t = 14{,}000 \tag{7.19b}$$

Here the weighting pair (w_w, w_g) was not used to indicate cost of the produced water or gas, but was varied to generate a subset of the Pareto optimal solutions. The weighting pairs and the corresponding solutions generated for this problem are shown in Table 7.2.

Table 7.2: Pareto optimal solutions from the weighting functions method.

Weighting Pair	(1,1)	(3,1)	(3.5,1)	(5,1)	(10,1)	(15,1)	(15.5,1)	(20,1)
Total Water Rate (STB/d)	10310	10074	8709	8653	8583	7246	6904	6354
Total Gas Rate (MSCF/d)	26419	26829	31058	31295	31834	50192	55404	63923

Both methods obtained roughly the same Pareto optimal set (Figure 7.4). However, it is much easier to control the sampling point with the hierarchical method than with the weighting functions method, because for the weighting functions method, there is no clear relation between the weighting pair and the Pareto optimal solution. Furthermore, the weighting functions method is unable to locate the nonconvex parts of the Pareto optimal set, such as the curve segment roughly between 9000 STB/D and 10,000 STB/D of water flow rate (Figure 7.4). Thus, the hierarchical method is more appropriate for this problem than the weighting functions method.

Once the complete set of Pareto optimal set is sampled, the production engineers can look at the trade-offs in the sampled set and make their operational decisions according to their preference. Note that the Pareto optimal set only needs to be constructed once, no matter what preferences the production engineers have. This is an advantage of the *a posteriori* methods over the *a priori* methods.

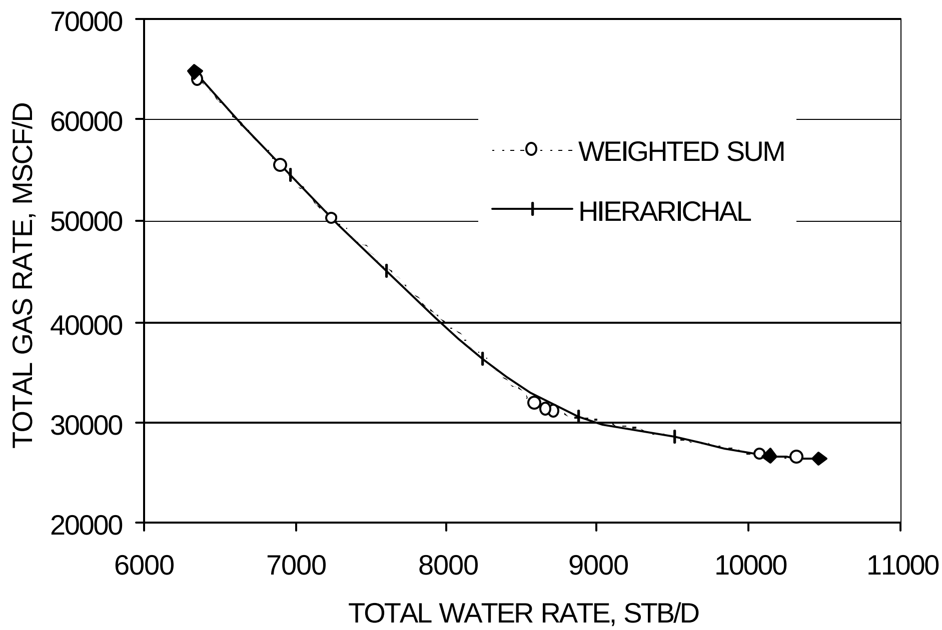


Figure 7.4: The Pareto optimal sets obtained from the Hierarchical method and the weighting functions method.

7.5 Concluding Remarks

When the multiple objectives can be appropriately aggregated into one objective function, i.e., the net present value (NPV), the utility function method is able to obtain the most satisfactory solution efficiently. However, sometimes, the utility function is difficult to construct. It appears that the weighted sum method described in Section 7.2.2 or the weighting functions method described in Section 7.2.4 is often used in the upstream oil industry to handle multiple objectives. However, as demonstrated in Section 7.4, certain multiobjective optimization problems in a petroleum field can have nonconvex Pareto optimal solutions, and the weighted sum method may fail to obtain solutions of interest.

The hierarchical method, on the other hand, does not require a utility function, and is able to sample the complete Pareto optimal set. However, as the number of objectives in an MOP increases, the computational demand of the hierarchical method may quickly becomes unmanageable, and the multidimensional Pareto optimal set get hard to visualize.

Perhaps the more useful application of multiobjective optimization to the upstream oil industry is to improve the design of a production system, because in this case there are more trade-offs involved and the impact of the trade-offs is more significant in such design problems than in production operation problems. The evolutionary multiobjective optimization methods (Fonseca and Fleming, 1993) is a promising method for such tasks because of the following reasons:

- The optimization problem of production system design has rough surfaces. Evolutionary algorithms are more appropriate than the gradient-based methods for such problems (Palke and Horne, 1997a).
- Evolutionary algorithms maintain multiple solutions and have the capability to sample various Pareto-optimal solutions in parallel (Oyama and Liou, 2002). `

Chapter 8

Coupling with a Reservoir Simulator

8.1 Introduction

The optimization methods investigated in this study require numerical tools to evaluate objective and constraint functions. For research purposes (such as evaluating the performance of the optimization algorithms without worrying about model accuracy), in Chapter 4 and 5 the objective and constraint functions were evaluated by a network simulator assuming that the required reservoir conditions around each well are given. For practical use, the optimization methods can be implemented in the following ways:

1. *Integrating* into a field simulator. In this approach, the optimization methods can be implemented in a field simulator that is capable of simulating both reservoirs and gathering systems. The function evaluation procedure of the optimization method can be implemented within the simulator. The advantage of this approach is that the optimizer is easy to use. If the engineer has built a simulation model for a field, it requires little extra effort to extend the simulation model for production optimization. The disadvantage of this approach is that its implementation requires access to the source code of the field simulator, which is not always possible.
2. *Interfacing* with a reservoir simulator. In this approach, the optimization methods are implemented in a stand-alone optimizer as in Chapter 4 and 5. The optimizer is then interfaced with a reservoir simulator, which feeds the optimizer the required reservoir

conditions. In contrast to the first approach, this approach is easy to implement, but the coupled system may be cumbersome to use.

3. *Integrating* with in-house software. For instance, the optimization methods proposed as a result of this study can replace heuristic rules of an existing optimization system developed for a specific field.

In the course of this study, some of the optimization tools developed in previous chapters were integrated into VIP-EXECUTIVE (Landmark, 2001), a commercial field simulator. This allows the optimization tools that are found to be most suitable to be applied to a large variety of petroleum fields. In particular, The VIP-EXECUTIVE with the optimization tools is an integral part of the E-Field Optimization System (EFOS) developed at Prudhoe Bay oil field in Alaska (Litvak et al., 2002). EFOS is designed to maximize the daily oil production in Prudhoe Bay field by optimally allocating the production rates, well connections to flowlines and (potentially) lift gas subject to multiple capacity constraints of facilities and velocity constraints in flowlines.

This chapter describes how the optimization tools can be integrated into/interfaced with a reservoir simulator (using VIP-EXECUTIVE as an example) and presents several examples of the integrated system.

To facilitate our discussion of the integration/interfacing procedure, we first discuss how an integrated reservoir and production network (tubing strings and surface pipeline systems) model can be solved.

8.2 Full Field Simulation

Traditionally simulation models for reservoirs and production networks have been developed independently. Various methods have been proposed to couple the reservoir model and the network model for full field solutions (Emanuel and Ranney, 1981; Litvak and Darlow, 1995; Schiozer, 1994; Hepguler et al., 1997; Byer, 2000). Full field simulation offers increased accuracy in predicting the reservoir deliverability and is critical for many reservoir development and management studies. In this section, we describe first the governing equations and then the existing solution techniques for full field simulations.

8.2.1 Simulation of Petroleum Fields

In the simulation of petroleum fields, fluid properties can be represented by a compositional model or a black-oil model. Consequently, the variables and the governing equations for the field simulation are different. The following discussion assumes a compositional model, for the black-oil model can be regarded as a special case of a compositional model.

The *Field simulator* models the multiphase flow in an integrated system of reservoirs, wellbores, well tubing strings, surface pipeline systems, and possibly the fluid separation units and other facilities. A field simulator typically is capable of obtaining the following information (Litvak and Darlow, 1995):

1. In gridblocks of a discretized reservoir model we obtain
    - pressures of each phases,
    - oil, gas, and water saturations, and
    - composition of hydrocarbon phases.
2. In production and injection wells we obtain
    - bottomhole pressures (or flow rates) and
    - molar rates of the hydrocarbon components.
3. In well tubing strings and surface pipeline systems
    - pressure distribution,
    - oil, water, and gas rates for every link and node (the notation of link and node used in this chapter is defined in Chapter 3),
    - molar rates of the hydrocarbon components for every link and node.

This information is determined from the simultaneous solution of the following governing equations:

1. A system of equations that describe multiphase flow in porous media of an underground reservoir. These equations include mass balance equations, phase equilibrium relations, capillary pressure relations, saturation constraints, and phase

constraints. In reservoir simulation, these equations are classified into primary and secondary equations and the variables are classified as primary and secondary variables

$$\mathbf{F}_p\left(\mathbf{x}_p, \mathbf{x}_s\right) = 0 \tag{8.1}$$

$$\mathbf{F}_s\left(\mathbf{x}_p, \mathbf{x}_s\right) = 0 \tag{8.2}$$

where subscript p represents primary, s represents second. The classification is not unique. As an example, in Coats model (Coats, 1980), $\mathbf{F}_p$ represents the mass balance equations for each component in each grid block. $\mathbf{x}_p$ represents the gridblock pressure, oil and water saturations, and the hydrocarbon component compositions of each gridblock. The concrete form of Eqs. 8.1 and 8.2 for various simulation models can be found in Aziz and Durlofsky (2002) and Cao (2002).

2. Wellbore flow equations that relate the well flow rate of component c, q_c^w, with the bottomhole pressure p^w

$$q_c^w = q_c^w\left(p^w\right) \tag{8.3}$$

The concrete form of Eq. 8.3 for various well models can be found in Aziz and Durlofsky (2002).

3. Multiphase flow models that relate the flow rate with the pressure drop across a tubing string:

$$p_2 = p_2\left(p_{1,} q_o, q_g, q_w\right) \tag{8.4}$$

where p_1 and p_2 denote the upstream and downstream pressure of a tubing string, and q_o, q_g, and q_w denotes the oil, gas, and water flow rate across that tubing string, respectively. Eq. 8.4 can be in the form of numerical multiphase flow models or hydraulic look-up tables.

4. The following system of equations that describe multiphase flow in a surface pipeline network:

- Mass balance equation for each component at every node

$$\sum_{j \in \Omega_i^n} q_{cij} = 0, \; i = 1, \ldots, n_n, \; c = 1, \ldots, n_c \tag{8.5}$$

where q_{cij} denotes the flow rate of component c between node i and j. Ω_i^n denotes all the adjacent nodes for node i, n_n is the number of node, and n_c is the number of components.

- Kirchoff's law, which requires that the pressure for a node should be the same no matter from which path it is computed.
- Flow equations that relate the flow rate and pressure drop through a link (i.e., a pipe, a choke, or a pump, etc.)

$$p_2 = p_2\left(p_{1,} q_o, q_g, q_w\right) \tag{8.6}$$

where p_1 and p_2 denote the upstream and downstream pressure of a link.

A *reservoir simulator* usually models multiphase flow in a reservoir up to the wellbores. The selected set of equations are solved by the Newton-Raphson method. The basic steps in a reservoir simulation can be described as follows:

1. Initialize the reservoir model.
2. March forward a time step.
   - 2a. Form a set of system equations by constructing Eq. 8.1 for every gridblock and Eq. 8.3 for every well.
   - 2b. Linearize the set of system equations.
   - 2c. Solve the linearized system equations to obtain updated estimates of the primary variables.
   - 2d. Update the secondary variables using Eq. 8.2 for every gridblock and Eq. 8.3 for every well.
   - 2e. Repeat Step 2a-2e until converge.
3. Repeat Step 2.

A *network simulator* simulates multiphase flow in well tubing strings and surface pipeline systems by treating production wells as sources with fixed pressures or flow rates. Sometimes a network model also contains wellbore models. A network simulation involves solving a set of nonlinear equations constructed from Eqs. 8.3-8.6 and Kirchoff's law. Chapter 3 presents a solution procedure for network simulation.

Full field simulators combine the reservoir model and the network model. To utilize fully existing techniques developed for reservoir simulation and network simulation, solution techniques developed for full field simulation focus on how to couple the reservoir model and the network model at the time-step level. Various coupling methods have been proposed in the literature and they are reviewed in the next section.

8.2.2 Coupling Reservoir and Production Network Solutions

The coupling methods proposed in the literature can be classified into implicit and explicit procedures.

Algorithm 8.1: Standard implicit coupling. This procedure combines the system of equations for the reservoir model and the network model and solves them simultaneously using the Newton-Raphson method. Litvak and Darlow (1995) proposed such a procedure for a compositional full field model. The steps in this procedure are described below.

1. *Solve linearized system of equations for the full field model*. Linearize and solve the flow equations in the reservoir, wellbores, and the surface pipeline system to obtain an estimate of the primary variables.
2. *Update reservoir conditions*. Update pressure, saturations, and component compositions in reservoir gridblocks using the solution from Step 1 and the secondary equations.
3. Repeat Steps 1-2 until convergence.
4. March to the next time step.

This procedure requires that the reservoir model and the network model are implemented in one simulator. Schiozer (1994) and Byer (2002) investigated a similar implicit coupling procedure for a black-oil model.

Algorithm 8.2: Iterative coupling. This procedure alternately solves a network problem and a reservoir problem until the system of equations for the full field model are satisfied. The following procedure was proposed by Litvak and Darlow (1995) for a compositional model (this method was named as an “explicit procedure” in Litvak and Darlow (1995)):

1. *Solve a network problem*. Form a network problem by taking reservoir conditions in well gridblocks from the previous iteration (or the previous time step at the first iteration) as part of the boundary conditions for the network problem. Solve the network problem and obtain well production rates and bottomhole pressures.
2. *Take a Newton iteration for the reservoir model*. Taking the bottomhole pressures or production rates from Step 1 as the well constraints, linearize and solve the reservoir flow equations. Update pressure, saturation, and component compositions in reservoir gridblocks using the solution of the linearized flow equations.
3. *Repeat Steps 1-2 until convergence*. The convergence criterion is that the production rates and bottomhole pressure obtained from Step 1 and Step 2 are within a specified tolerance for every well.
4. *March to the next time step*.

The advantage of this procedure is that it can be used to couple a reservoir simulator with an independent network simulator. However, for certain cases, this procedure may not converge or converge to a wrong solution (Litvak and Darlow, 1995). Similar procedures were also presented by Hepguler et al. (1997), Emanuel and Ranney (1981), and Breaux et al. (1985).

Algorithm 8.3: Explicit coupling. This procedure performs network simulation only once, based on reservoir conditions at the beginning of a time step. Consequently the solution may not satisfy the system equations of the full field model. However, this approach is the easiest to implement and the cheapest to run among all coupling methods.

1. *Solve a network problem*. Form a network problem by taking reservoir conditions in well gridblocks from the previous time step as part of the boundary conditions of the network problem. Solve the network problem and obtain well production rates and bottomhole pressures.
2. *Solve a reservoir problem*. Taking the bottomhole pressures or production rates from Step 1 as the well constraints, solve a reservoir simulation problem as described in Section 8.2.1.
3. *March to the next time step*.

This procedure was investigated by Schiozer (1994) and Byer (2000), and was used by Emanuel and Ranney (1981) to study an offshore reservoir.

8.3 Integration of Optimization Tools with VIP-EXECUTIVE

Some of the optimization tools we investigated were integrated with a commercial field simulator, VIP-EXECUTIVE (Landmark, 2001). This section first presents a general procedure for integrating production optimization tools into VIP-EXECUTIVE, then describes the integrated optimization tools and some related issues.

8.3.1 A General Integration Procedure

As a reservoir development and management tool, VIP-EXECUTIVE contains certain reservoir management features, such as well controls and gas-lift allocation, etc. The role of these management features in a field simulator is the same as those of our optimization tools. Therefore, the optimization tools can be integrated into the solution procedure of the field simulator in the same way as these reservoir management features are integrated.

Algorithm 8.4: Integration procedure. The optimization tools are integrated with the standard fully implicit coupling procedure of VIP-EXECUTIVE at the Newton iteration level (Figure 8.1).

1. *Perform well management routines (optional).* Based on user specifications and estimated field conditions (well water cut, GOR, total field flow rates, etc.), invoke well management routines that handle well constraints and gas-lift allocation. The results of this step may include the following:
   - Updated well status. Some wells are closed. Some wells are reopened. Some wells are worked over. Some wells are reconnected to different pipeline systems, etc.
   - Updated well constraints. For example, a well should not produce at an oil flow rate higher than 3000 STB/D to satisfy a total field gas flow rate constraint.
   - Updated lift gas rate for gas-lift wells.
   - Updated injection rate for injection wells.
2. *Convert pressure constraints (optional)*. Based on current reservoir conditions, invoke a procedure proposed by Litvak and Darlow (1995) to convert pressure and flow rate constraints on network nodes to flow rate constraints on individual wells. If

Step 3 is to be invoked, only pressure constraints are converted. The flow rate constraints on network nodes are handled in Step 3.

3. ***Perform production optimizations*** *(optional).* Perform user-specified production optimizations with current reservoir conditions. The results of this step may include
   - Updated well constraints.
   - Updated lift gas rate for gas-lift wells.
   - Updated well connections.
4. *Solve linearized system equations for the full field*. Linearize and solve the flow equations in the reservoir, wellbore, and surface pipeline systems to obtain an estimate of the primary variables. The updated operation settings from Step 1 and 2 are incorporated in this step.
5. *Update reservoir conditions*. Update pressure, saturations, and component compositions in reservoir gridblocks using the solution from Step 1.
6. *Repeat Steps 1-5 until convergence*.
7. *March to the next time step*.

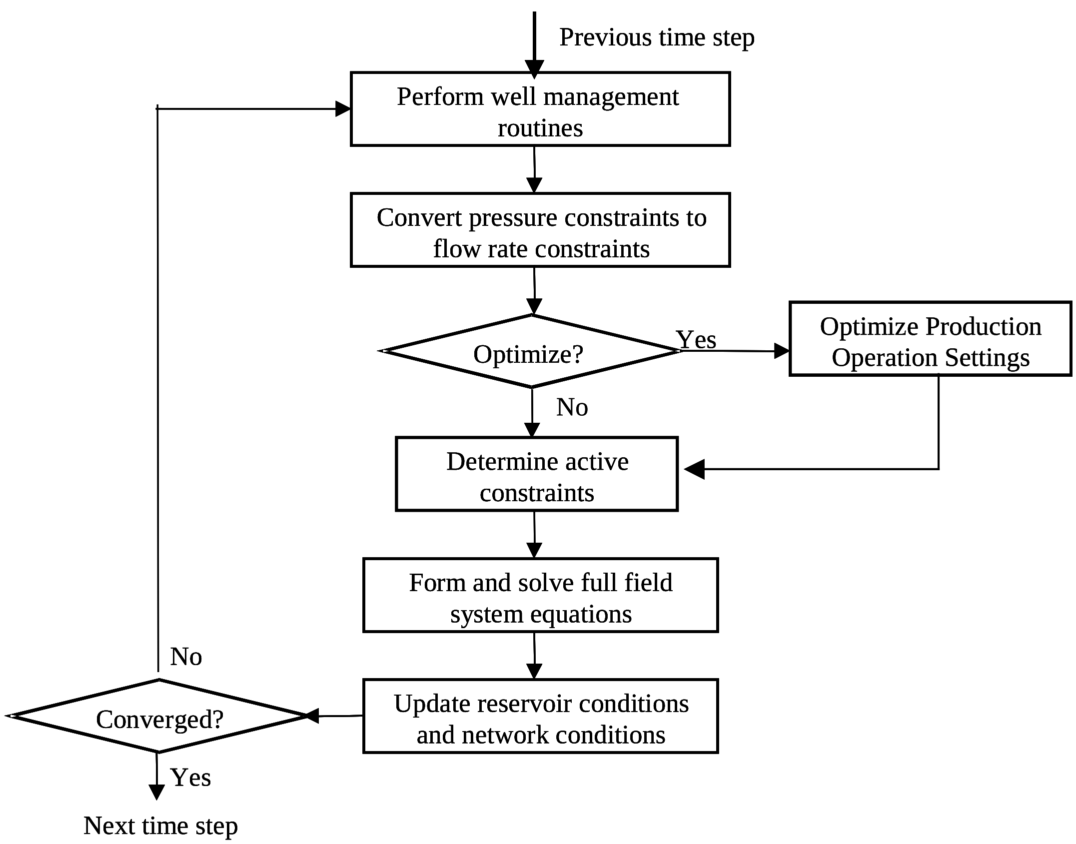


Figure 8.1: Integration of optimization tools to VIP-EXECUTIVE.

8.3.2 Integrated Optimization Tools

The following optimization tools were integrated into VIP-EXECUTIVE:

1. the LP-I method for rate allocation (Section 4.3.1),
2. the LP-II method for rate allocation (Section 4.3.4), and
3. the partial enumeration method and the genetic algorithm for well connection optimization (Chapter 6).

Specific integration issues for each optimization tool are discussed below.

The LP-II method for rate allocation. To ensure that the constraints will not be violated at the end of a time step, a rate allocation problem is solved by the LP-II method at every Newton iteration of the solution procedure for the full field model. The following procedure corresponds to Step 2 and 3 of Algorithm 8.4.

1. Convert pressure constraints on wells and network nodes to flow rate constraints on individual wells using a procedure proposed by Litvak and Darlow (1995). The weighting coefficients c_o^v, c_g^v, and c_w^v for each velocity constraint (see Section 4.4.1) are determined during this procedure.
2. Use the following information to build the LP-II model.
   - the oil rate, GOR, and water cut for each well from Step 1,
   - the weighting coefficients c_o^v, c_g^v, and c_w^v from Step 1, and
   - the user-specified constraint limits and weighting coefficients for the objective function.

   The speedup techniques described in Section 4.4.2 are used to reduce the size of the LP-II model.
3. Solve the LP problem formulated in Step 2.
4. Update the well constraints for the next Newton iteration. For every well i, if the scaling factor $x_i = 0$, well i is closed in the next iteration; if $x_i = 1$, well i remains fully open in the next iteration, and its oil rate is allowed to increase at the next iteration; if $x_i \in (0,1)$, the oil rate of well i is not allowed to exceed $x_i q_{o,i}^{\max}$ at the next iteration, where $q_{o,i}^{\max}$ is the maximum oil rate of well i used to build the LP model in Step 2.

In Step 4, once a well rate is scaled, it is not allowed to exceed its scaled value in the next iteration. Consequently in later iterations fewer constraints may be violated and the speedup techniques (the constraint elimination procedure described in Section 4.3.2) will be more effective.

At each Newton iteration, the LP-II method optimizes rate allocation based on updated reservoir conditions. Therefore, in a time step, the nonlinear interactions in the reservoir are partially taken into account by solving a rate allocation problem at every Newton iteration.

The LP-I method for rate allocation. This method is coupled into VIP-EXECUTIVE in the same manner as the LP-II method except

1. to build Model LP-I, the well performance curves have to be constructed, and
2. to reduce the computational load, the method is not invoked at all Newton iterations. The user can specify at which Newton iterations to invoke the LP-I method.

Well connection optimization. Both the partial enumeration method and the genetic algorithm described in Chapter 6 are implemented. The rate allocation problem within both algorithms are solved by the LP-II method.

Well reconnections can significantly change the flow behavior in the system. Thus performing well connections between iterations can hinder convergence of the solution procedure of the field simulator. In addition, well connection optimizations are time-consuming. For these reasons, in the current implementation, the user can specify in which Newton iterations well connection optimization will be performed.

8.4 Interfacing with an Independent Reservoir Simulator

When the optimization tools are developed as a standalone package without reservoir models, the optimization package can be interfaced with a reservoir simulator to do full field simulation and production optimization. The optimization package and the reservoir simulator can be coupled together using the iterative coupling procedure or the explicit coupling procedure described in Section 8.2.2.

8.5 Applications

The optimization system described in Section 8.3 can be used for reservoir development studies or for on-site real time production control and optimization. This section presents three examples: two for long-term reservoir prediction/development studies and one for on-site production optimization.

The first example predicted the oil production of an oil field in the Gulf Of Mexico (GOM) for five years. At each time step, the daily oil production from the field was maximized by allocating the production rates and the well connections. The second example predicted the oil production of another oil field in the GOM for the whole life of the field. At each time step, the daily oil production from the field was maximized by optimally allocating production and lift gas. The third example describes how the optimization tools can be applied to the giant Prudhoe Bay oil field in Alaska.

8.5.1 A Gulf of Mexico Oil Field Example

An integrated reservoir and surface pipeline network model had been built for one of the Gulf of Mexico oil fields. Our test example was based on this reservoir. This example has been presented in Wang, Litvak, and Aziz (2002a).

Five production wells, A1-A5, are tied to a 40-slot conventional fixed-leg platform. Ten production wells were drilled from a 10-slot subsea template and connected to the platform by two 22000 foot long pipelines. On the platform, a three-stage separator processes the produced fluids. Wells A1-A5 can be tied to any stage of the separator. Wells B1-B10 can be connected to either the low-pressure stage or the high-pressure stage of the separator. Figure 8.2 illustrates the production system.

For evaluation purposes, we imposed four hypothetical constraints on the production system. The separator has a maximum gas handling capacity of 30 MMSCF/D and a maximum of water handling capacity of 5000 STB/D. Moreover, the separator cannot receive gas from either pipeline at a rate of more than 12 MMSCF/D. Our job was to maximize the oil production by selecting appropriate well rates and well connections.

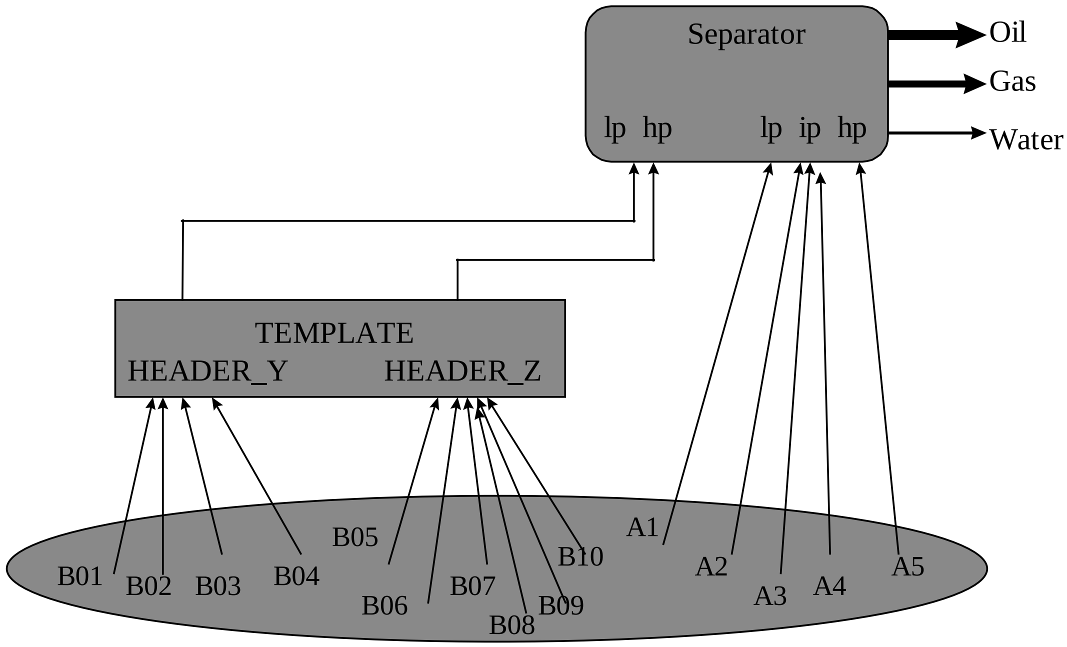


Figure 8.2: The production system of a Gulf Of Mexico oil field.

We made three predictive runs using VIP-EXECUTIVE. Each run predicted the oil production for about five years. In Run 1, at every Newton iteration of each time step, we checked flow rate constraints sequentially and scaled well rates if necessary: first we checked the two gas flow rate constraints on HEADER_Y and HEADER_Z, if either constraint was violated, the production rates of the highest GOR wells connecting to HEADER_Y (or HEADER_Z depending on which constraint was violated) were scaled; then we checked the gas flow rate constraint on the separator, if it was violated, the highest GOR wells of the whole field were scaled to meet the gas constraint; finally we checked the water constraint on the separator, if it was violated, the highest water cut wells of the whole field were scaled to meet the water constraint. This procedure is a well management option available in VIP-EXECUTIVE (Landmark, 2001). In Run 2, we applied the LP-II method described in Section 4.3.4 to select the well rates. In Run 3, we applied the partial enumeration method with the LP-II method to optimize the well rates and well connections simultaneously. The results are plotted in Figure 8.3-8.7.

The well rates in Figure 8.3-8.7 were normalized by some fixed oil, gas, and water rates. Figure 8.3 shows that Run 2 produced at a much higher rate than Run 1. Run 3 was even better. Over five years, Run 2 predicted 11.2% more oil than Run 1, and Run 3 predicted 18.9% more oil than Run 1. The excellent performance of Run 2 compared to Run 1 can be explained as follows. The rate allocation procedure in Run 1 first scaled the highest GOR wells to meet gas constraints, and then scaled the highest water cut wells to meet the water constraint. This resulted in suboptimal solutions. On the other hand, Run 2 used a rigorous optimization procedure (the LP-II method) and scaled the well rates differently. For instance, most of the time Run 2 had been producing at a lower total water rate (Figure 8.5) but a higher gas flow rate (Figure 8.4) than Run 1. And most of the time Run 2 predicted a much higher gas rate for HEADER_Z than Run 1. In other words, Run 1 did not utilize the processing capacity of HEADER_Z as efficiently as Run 2. The fact that Run 3 outperformed Run 2 can be explained as follows. At later times in Run 2, without well reconnection, most of the gas processing capacity of HEADER_Y was wasted (Figure 8.6), while HEADER_Z was constrained by its gas processing capacity (Figure 8.7). The algorithm in Run 3 identified this imbalance and switched a high GOR well from HEADER_Z to HEADER_Y.

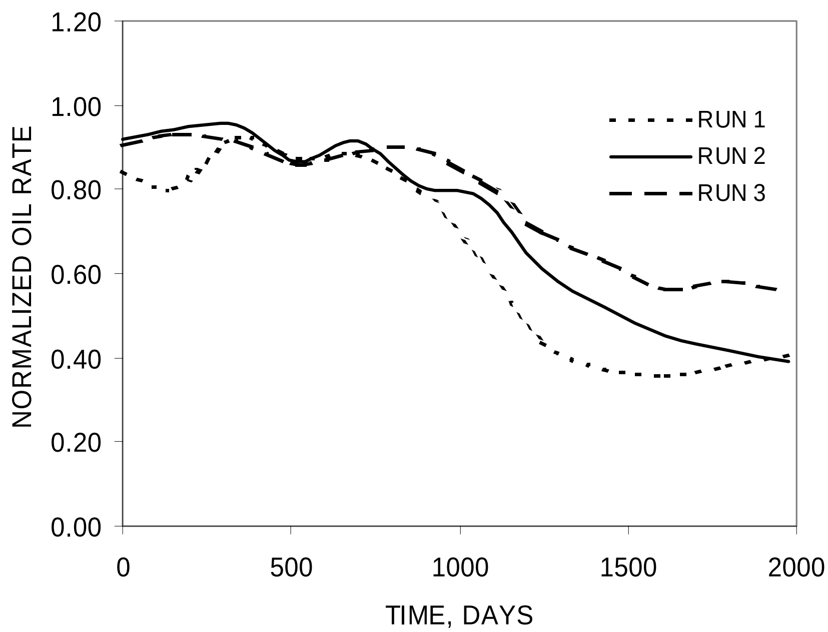


Figure 8.3: Normalized daily total oil rate.

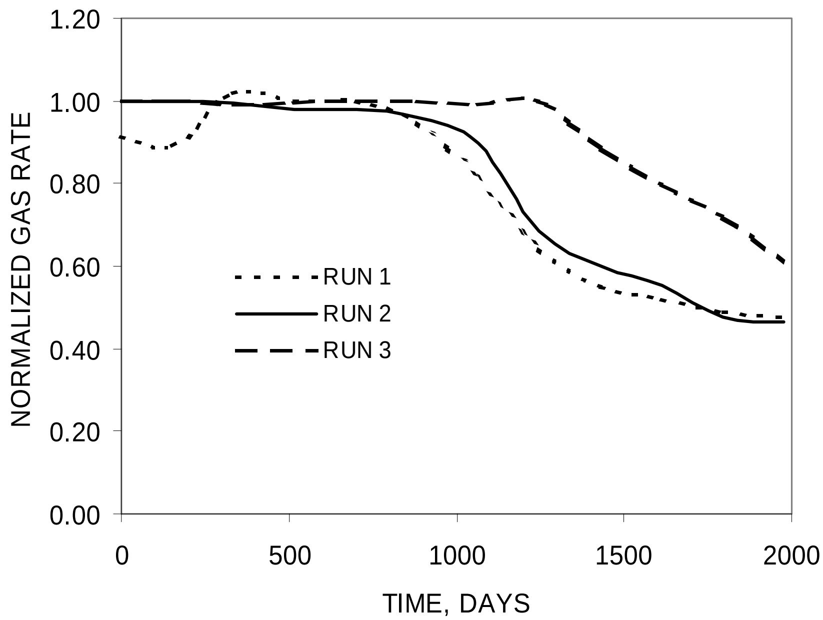


Figure 8.4: Normalized daily total gas rate.

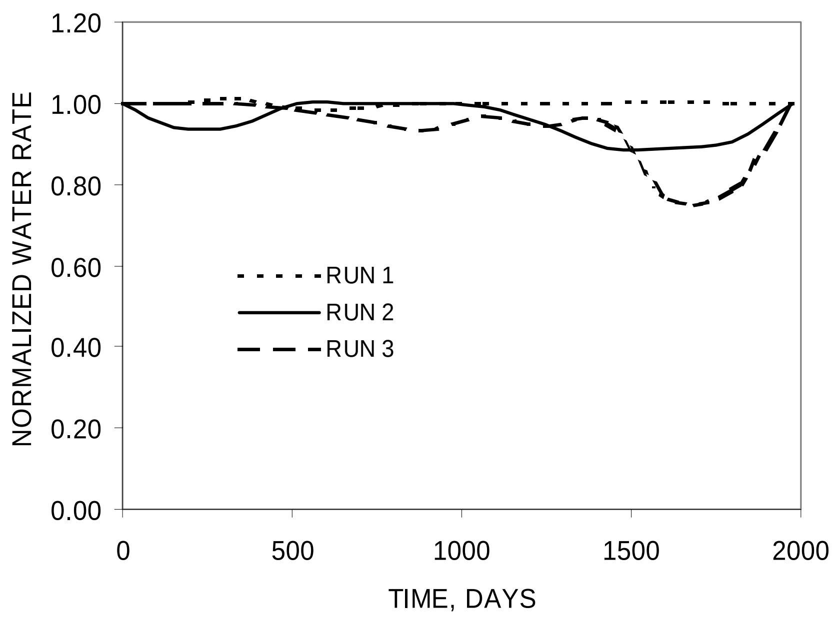


Figure 8.5: Normalized daily total water rate.

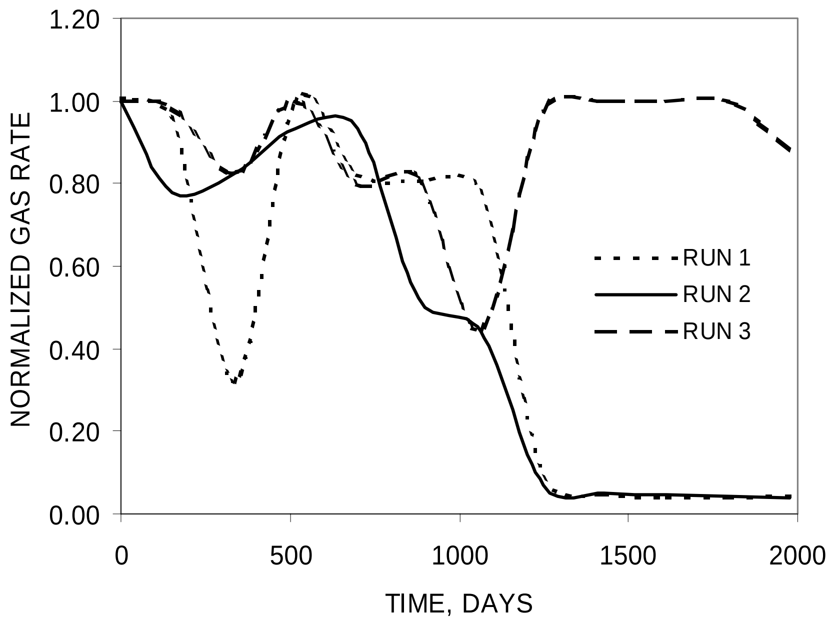


Figure 8.6: Normalized daily gas rate at HEADER_Y.

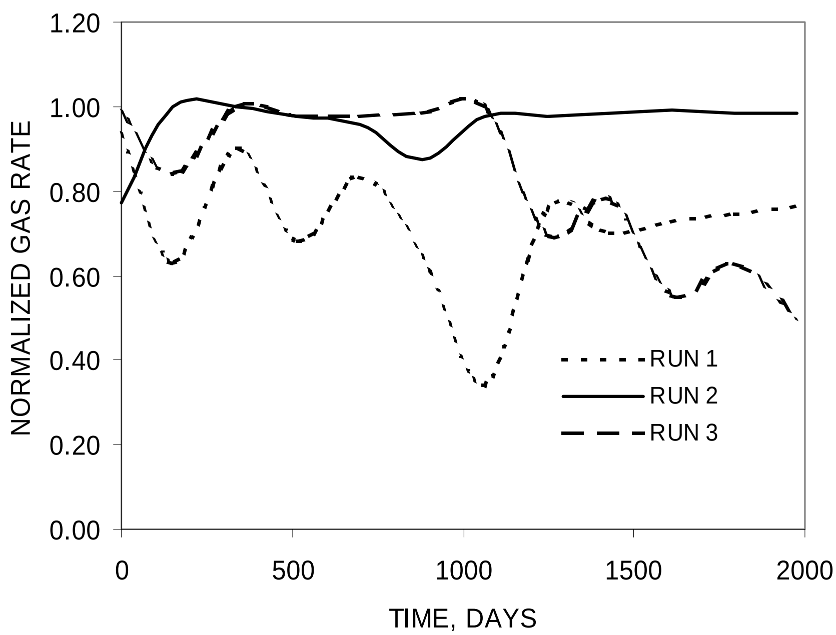


Figure 8.7: Normalized daily gas rate at HEADER_Z.

8.5.2 Another Gulf of Mexico Field Example

This is a reservoir development model for a newly discovered deep-water oil field in the Gulf of Mexico. In this model, 26 wells are scheduled to be drilled at different times. All wells are connected to a common offshore platform, on which resides a separator and possibly other surface facilities. The vertical distances between the separator and wells vary between 10,000 ft and 14,000 ft. There are no flow interactions among different wells.

The purpose of this example was to demonstrate that the application of the developed short-term optimization algorithms in long-term prediction can improve facility design. In this example, two cases were considered, each with a different gas handling capacity for the separator.

Case 1. In this case, the separator has a gas handling capacity of 98 MMSCF/D. The total field oil production rate cannot exceed a certain target rate. The lift gas rate for a gas-lift well cannot exceed 6000 MSCF/D. The prediction period is about 22 years. During the prediction period, some wells will be recompleted, placed on gas lift, converted from producers to injectors, closed for maintenance, etc. To account for the downtime between two reporting dates in simulation, an *ONTIME* factor with a value between 0 and 1 may be defined for a well. For example, if a producer is expected to produce equivalent of one month during a two month reporting period, then an ONTIME factor of 0.5 will be specified for that producer. To facilitate later discussions, we make the following definitions: *instantaneous flow rate* is the actual flow rate of a producer when it is flowing, *discounted flow rate* is the average flow rate of a well for a reporting period. For a well, its discounted flow rates for a time step are the instantaneous flow rates time its ONTIME factor for that time step.

For evaluation purposes, we made two runs: a BASE run and a LP-I run. The LP-I run uses Mode LP-I described in Section 4.3.1 to allocate the lift gas and production rates subject to the total oil rate constraint of the field and the total gas rate constraint of the separator. The BASE run allocated the lift gas and production rates using a “conventional” method available in VIP-EXECUTIVE (Landmark, 2001). This method proceeds as follows:

1. Lift-gas rates are determined for specified wells based on some gas-lift tables. These tables correlate gas-liquid ratio for a production well to its water cut and liquid production rate. If the formation gas-liquid ratio for a production well is higher than the table value, gas-lift is not implemented for the well. Otherwise, the well's lift gas rate is calculated to meet the gas-liquid ratio specified in the table (Litvak et al., 1997).
2. Update the production rates of all wells with allocated gas-lift rates.
3. Check the total gas rate constraint. If it is violated, scale both the production rates and lift gas rates of wells with the highest GOR to meet the total gas rate constraint.
4. Check the oil rate constraint. If the constraint is violated, scale the production rates of certain wells to meet the oil rate constraint.

We emphasize that when checking the flow rate constraints, the LP-I run used instantaneous production rates from each well but the BASE run used discounted production rates (these options are implemented in this way in VIP-EXECUTIVE). Thus, the same flow rate constraint appeared more strict in the LP-I run than in the BASE run.

Figure 8.8 plots the daily field oil production history for both runs (the daily oil rates in both runs are normalized by the field oil target rate). It appears that the oil production histories of the two runs are the same except when the oil rate constraint in the BASE run was active. When the oil rate constraint in the BASE run was active, the LP-I run produced at a lower oil rate. This is because the LP-I method scales instantaneous production rate to meet the flow rate constraint and VIP-EXECUTIVE reports discounted flow rate. Figure 8.9 plots the normalized cumulative oil production for both runs. The trend of Figure 8.9 agrees with what we observe in Figure 8.8.

Figure 8.10 shows the daily total gas rate for both runs. The total gas constraint was not active in both runs for the whole prediction period. Thus the gas handling capacity of the separator is oversized for this reservoir development scenario. Figure 8.11 plots the normalized cumulative lift gas injection for both runs. It shows that the BASE run injected about 20% more lift gas than the LP-I run for the whole prediction period. This can be explained as follows. When allocating lift gas, the LP-I method needs to construct the gas-lift performance curve. In this procedure, a minimum gas-lift efficiency can be specified for each well, thus avoiding over-injection of lift gas. On the other hand, the lift-gas table does not have this feature and can over-inject lift gas.

Case 2. The reservoir model and the development scenario are exactly the same as in Case 1, except that the gas handling capacity of the separator is lowered to 60 MMSCF/D. Again, we made the same two runs as in case 1: a BASE run and a LP-I run.

Figure 8.12 shows the normalized daily field oil production history for both runs. Figure 8.13 plots the normalized cumulative field oil production history for both runs. Figure 8.14 plots the normalized cumulative lift gas rate for both runs. From Figure 8.12 we can see that at about 1900 days the oil production in the BASE run dropped sharply to a lower rate. This can be explained as follows. Around 1900 days, some wells were placed on gas-lift. The BASE run allocated the lift gas to these wells without the consideration of the total gas constraint. However, certain careful considerations will reveal that no lift gas should be injected (Figure 8.14 shows that the LP-I run did not inject any lift gas until about 3000 days). As a result, injecting lift gas decreased the total oil production instead of increasing it. This demonstrates the danger of allocating lift gas and production rates separately. Figure 8.12 shows that the BASE run continued to produce at a lower oil rate than the LP-I run until 4500 days, after which the BASE run produced at a higher oil rate than the LP-I run. The lower oil rate in the LP-I run at later stages of the prediction period is probably due to the fact that there was less oil in the reservoir of the LP-I run than in the reservoir of the BASE run at later stages of prediction. Figure 8.13 shows that over the whole prediction period the LP-I run produced slightly more oil than the BASE run. Figure 8.14 shows that over the whole prediction period the LP-I run allocated about 40% less lift gas than the BASE run. This is due to two facts: 1) the LP-I run optimized the production and lift gas rates simultaneously while the approach in the BASE run did not; 2) the LP-I method can control the gas-lift efficiency.

We can conclude that facility design depends on the well management tools (the “conventional” approach or the LP-I method) used to evaluate the sizes of the separator and the gas compressor for lift gas. If used appropriately, the LP-I method can save significant amount of construction and operational cost.

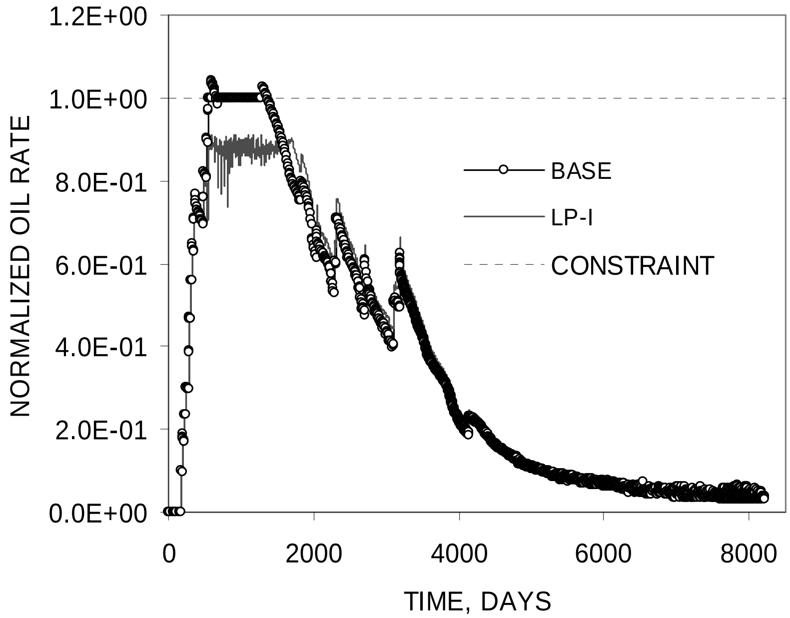


Figure 8.8: Normalized daily total oil production history for Case 1.

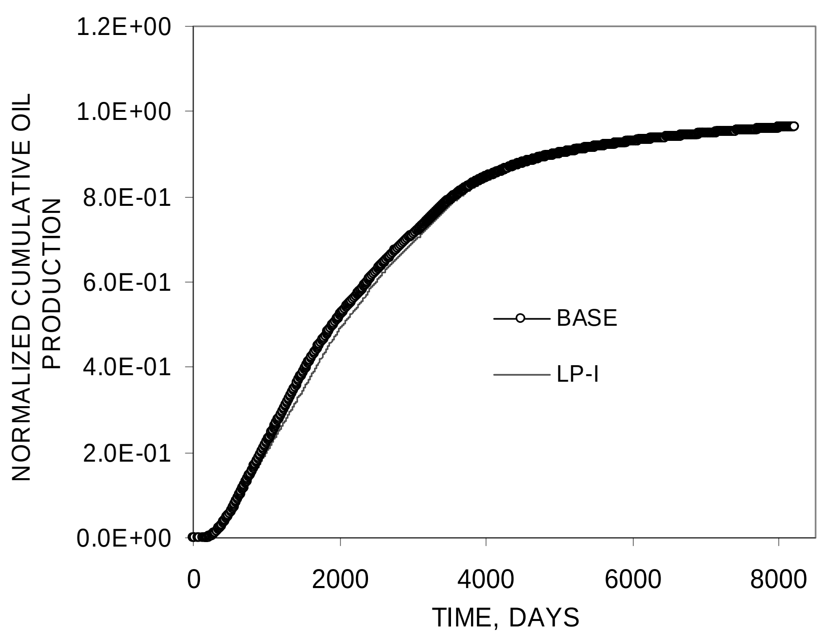


Figure 8.9: Normalized cumulative oil production history for Case 1.

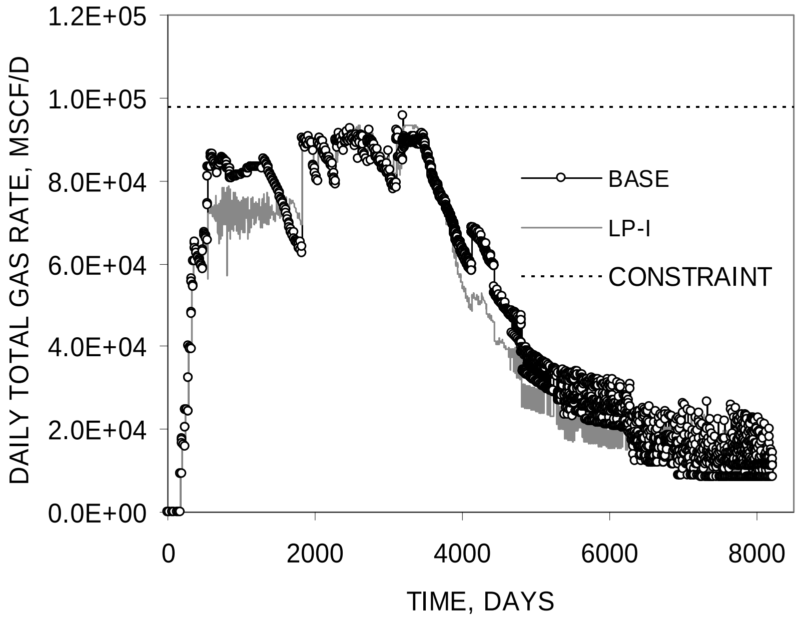


Figure 8.10: Daily total gas rate history for Case 1.

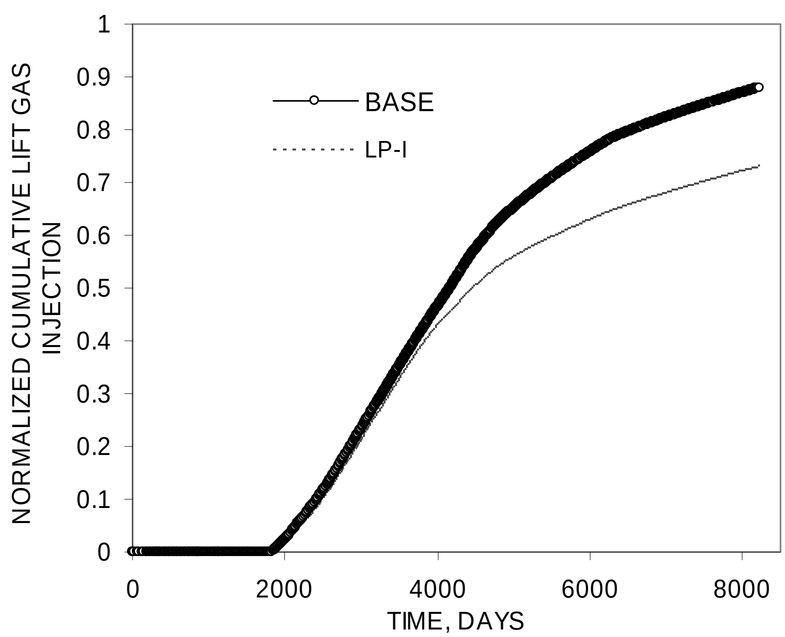


Figure 8.11: Normalized cumulative lift gas injection for Case 1.

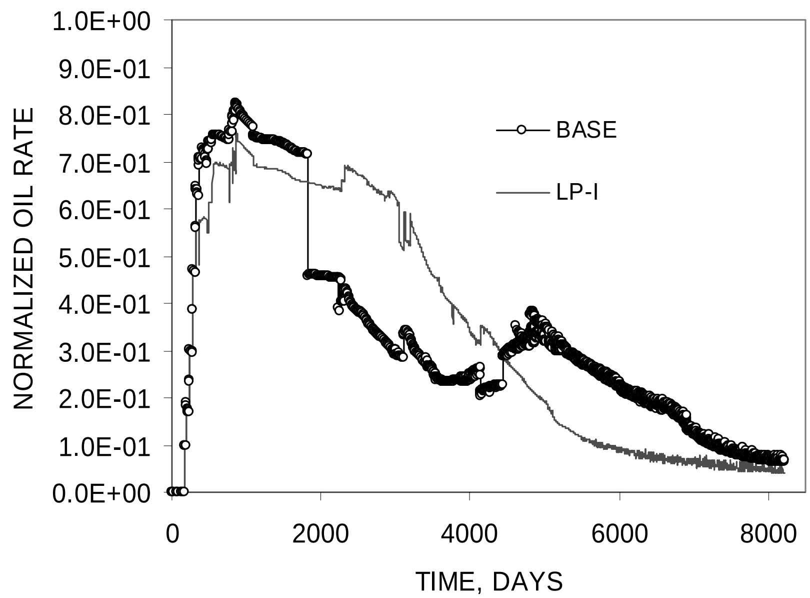


Figure 8.12: Normalized daily total oil production history for Case 2.

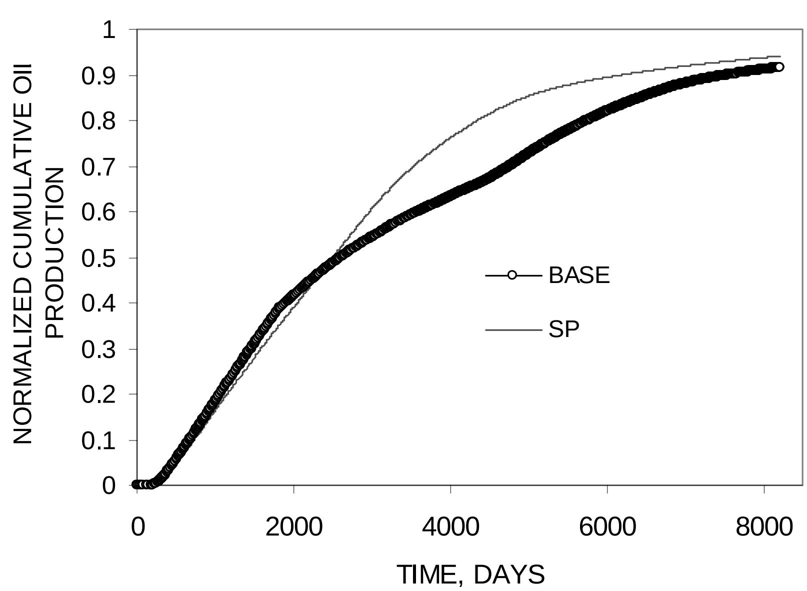


Figure 8.13: Normalized cumulative oil production history for Case 2.

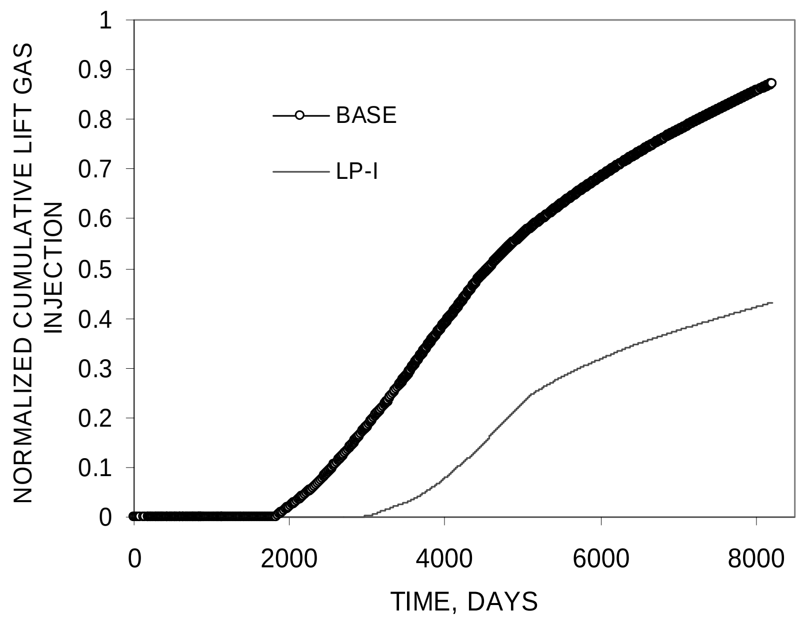


Figure 8.14: Normalized cumulative lift gas injection for Case 2.

8.5.3 Application to the Prudhoe Bay Oil Field

Prudhoe Bay oil field is located on the North Slope of Alaska and operated by BP Exploration (Alaska) Inc. Oil production from Prudhoe Bay is on decline and constrained by the gas and water handling limits of surface facilities and velocity constraints in flow lines. The E-field optimization system (EFOS) was developed at Prudhoe Bay oil field to automate and optimize the production operations of the field. EFOS contains four major modules: automatic data preparation, automatic model tuning, production optimization, and a user-friendly interface. The production optimization module uses the optimization system described in Section 8.3 to maximize the daily oil production and debottleneck the production system. EFOS shows how our optimization tools can be used for on-site production optimization in a real petroleum field. EFOS is described fully by Litvak et al. (2002).

The major purpose of this section is to demonstrate possible use of the optimization tools for Prudhoe Bay field and their high efficiency even for such a giant field. To facilitate later discussions, a brief description of the production system of Prudhoe Bay field is presented below.

Production system description. Prudhoe bay has a flexible, highly automated and complex surface pipe network and processing facilities. The following description of the production system of Prudhoe Bay is extracted from Litvak et al. (1997).

> "Production wells are tied to 22 well pads in the Western Operating Area (WOA) and 16 drill sites in the Eastern Operating Area (EOA). Most production wells can be switched between high and low-pressure headers at the well pad manifolds. Gas-lift is used in some production wells with high water cut. Gas-lift wells can be tied only to low-pressure headers. Well chokes are installed in well pad manifolds.
>
> The production stream from individual well pads and drill sites is sent without any separation to three gathering centers in the WOA and to three flow stations in EOA. Well pads and drill sites are tied to separator banks through high pressure and/or low-pressure flow lines. Each gathering center and flow station has four three-stage separator banks. Separator banks can operate in high pressure or low-pressure mode with first separation stage pressure of about 670 psig and 150 psig, respectively.
>
> Separator off-gas is routed from the gathering centers and flow stations to the central gas facility where natural gas liquids and miscible injectant are extracted. The natural gas liquids are then blended with the oil stream coming from the separators. The miscible injectant is used in the enhanced oil recovery (EOR) project. It is injected, as part of a WAG (water alternating gas) process in some 120 EOR patterns. The lean gas, after compression in the central compression plant, is reinjected into the gas cap for pressure maintenance. The produced water is either reinjected in the EOR pattern or into water disposal wells."

Possible usages of the optimization tools. For Prudhoe Bay oil field, the possible usage of the optimization tools can be summarized as follows.

1. *Optimization of production rate and well connections in response to changing operational conditions.* Oil production in Prudhoe Bay field is constrained by capacity constraints of facilities and velocity constraints of flowlines. In such a constrained environment, the maximum possible field oil rate is sensitive to the

changing operational conditions. Also the operational conditions at Prudhoe Bay can change significantly in a short period. For example, field gas compression capacity can be reduced by 1.2 BCF/D when the ambient temperature increases from 0 °F to 40 °F, which can happen during a 24-hour period in Prudhoe Bay (Litvak et al., 2002). Production capacity can also be affected by equipment maintenance and repair, and other planned or unplanned activities. Therefore, it is necessary to optimize the production rate and well connections from time to time to maximize the total field oil rate in response to the changing operational conditions.

One contribution of this study is that we developed optimization tools that can simultaneously optimize the production rate and the well connections. To demonstrate the potential advantage of the simultaneous optimization, we conducted two optimization runs for the whole Prudhoe Bay field. In the first run, the production rates were optimized using the LP-II method (see Section 4.3.4). In the second run, the production rates and well connections were optimized simultaneously by the combined use of the partial enumeration (PE) method and the LP-II method. The number of well reconnections was not allowed to exceed 50. In both runs, the choke settings for all production wells were not allowed to increase. Results showed that the second run improved oil production by 2.55% compared to the first run, which is significant for this large field. On an IBM RS6000 computer, the first run took 0.07 seconds, the second run took about 850 seconds (in the second run, the major portion of the computational time was spent on solving the network problem after each well reconnection). This demonstrates the high efficiency of the developed optimization methods.

2. *Debottlenecking studies*. Another important use of the optimization tools is to identify bottlenecks in the production system and evaluate debottlenecking opportunities. This is illustrated here by the LP-II method for rate allocation.

   The bottlenecks of the production system can be identified from the optimal solution of the LP-II model for the rate allocation problem. Suppose the rate allocation problem is formulated as a LP problem, Problem 8.7, as follows

$$\text{maximzie} \quad f(\mathbf{x}) = \mathbf{c}^T \mathbf{x} \tag{8.7a}$$

$$\text{subject to } \mathbf{a}_i^T \mathbf{x} \le \mathbf{b}_i,\ i = 1,\ldots,m \tag{8.7b}$$

$$0 \le x_i \le 1,\ i = 1,\ldots,n \tag{8.7c}$$

where $\mathbf{x}$ denotes the scaling factors for well rates, $\mathbf{b}$ denotes the maximum fluid processing capacity of facilities and velocity limits for flowlines, $f(\mathbf{x})$ is the objective function. The optimal solution of a LP problem contains a dual variable λ_i for each constraint and

$$\lambda_i = -\frac{\partial f^*}{\partial b_i} \tag{8.8}$$

where f^* denotes the optimal objective function of Problem 8.7. Eq. 8.8 states that at the optimal solution of a LP problem, the dual variable indicates how sensitive the optimal objective function is relative to the right-hand side of the constraints. $\lambda_i = 0$ indicates that increasing b_i has no impact on the optimal objective function value. The more negative the λ_i, the more sensitive the objective function to b_i, and the more likely it is that facility capacity or velocity constraint b_i is the bottleneck of the production system.

The rate allocation problem for Prudhoe Bay field contains mostly multiple gas, water constraints of facilities and velocity constraints. The dual variables for water, gas, and velocity constraints have different units. To avoid missing debottlenecking opportunities, several constraints with the most negative dual variables from each category (gas, water, or velocity constraints) can be identified as debottlenecking opportunities.

After debottlenecking opportunities are identified, a series of sensitivity runs can be performed to evaluate the selected opportunities. For example, the gas compression capacity of the most loaded compressor may be increased and the rate allocation problem re-solved to see how much oil increase can be obtained. The debottlenecking opportunity with the best economic return can be implemented.

3. *Gas-lift optimization*. In the Prudhoe Bay field model, the gas-lift performance is evaluated by the combined use of inflow performance curves and well tubing hydraulic models. However, the tubing hydraulic model is not accurate enough to

represent real physics because of the large diameters of the tubing strings (existing multiphase flow models do not perform well for large diameter tubing strings (Litvak et al., 2002)). Therefore, the LP-I method has not been applied to the real field. However, the LP-I method can be used to identify *potential benefits* of doing gas-lift optimization for the Prudhoe Bay field.

We conducted two optimization runs for the whole Prudhoe Bay field. In the first run, the production rates were optimized using the LP-II method. In the second run, the production rates and lift gas rates were optimized simultaneously by the LP-I method. Results showed that the second run improves daily oil production by 6% compared to the first run. This study demonstrated that the optimal allocation of lift gas is a potential effective way to increase the daily oil production. On an IBM RS6000 computer, the second optimization run took about 10 seconds.

8.6 Concluding Remarks

For practical use, the optimization tools can be implemented in different ways. The methods can be integrated into or interfaced with a commercial simulator. They can also be implemented in an in-house software. The use of a commercial simulator has several advantages: (1) the integrated system can be applied to a variety of fields; (2) advances in simulation technology can be easily incorporated and transferred to other fields (Litvak et al., 2002); (3) the user needs less effort to develop and maintain the simulation and optimization models, etc.

In a full field study, the network model and the reservoir model can be coupled together using either the implicit procedures or the explicit procedure. If possible, the implicit coupling procedures are recommended, because implicit coupling is able to obtain simulation results consistent with the full field model. If the optimization tools are applied for long-term prediction, their use should be managed carefully to avoid severe convergence issues of the solution procedure and to ensure no constraints violations at the end of a time step.

Application examples presented in this chapter show that optimization methods developed in this study have distinctive advantages over the heuristic methods currently employed in the oil industry. In addition, they are efficient enough to be used for both real-time optimization and long-term prediction studies for large-scale reservoirs.

The optimization tools developed in this study were designed to optimize the short-term production operation based on static reservoir conditions. The reservoir dynamics were not considered in the optimization step. Although we did apply these optimization tools to long-term predictions, the impact of decisions based on this approach on long term oil recovery needs further investigation.

Chapter 9

Conclusions

This study addressed the problem of optimizing the production rates, lift gas rates, and well connections to flowlines subject to multiple flow rate and pressure constraints to achieve certain short-term operational goals. This problem is being faced in many mature fields and is an important element to consider in planning the development of a new field. In this research we developed, compared and tested optimization methods for petroleum production problems. The major contributions of this work and areas for further study are summarized in this section.

9.1 Conclusions

1. The optimization methods investigated and developed here are effective for problems of varying complexities and sizes. The methods can be used for both short-term production optimizations and long-term reservoir development studies. For instance, they can be used to optimize well connections and well rates simultaneously, even for large production systems such as those in the Prudhoe Bay field.
2. For the rate allocation problem, when flow interactions among wells are not significant, the various separable programming methods (the LP-I, LP-II, MILP-I, MILP-II methods described in Chapter 4) proved to be better than other conventional approaches used in the oil industry (i.e., the equal-slope method for gas-lift optimization, and the rule-based methods available in commercial simulators).

Specifically, the MILP-I method can be used for problems with performance curves of arbitrary shapes. The MILP-I, LP-I, and LP-II methods are suitable for large-scale rate allocation problems. The MILP-II method is especially suitable for gas-lift optimization problems with many non-concave gas-lift performance curves.

3. For the rate allocation problem, when flow interactions have significant impact on the optimal solution, simulation models capable of capturing such flow interactions should be used in the optimization process. Formulation P2 (Chapter 5) proved to be most appropriate for this problem. The formulated nonlinearly constrained problem (NCP) can be solved efficiently by a sequential quadratic programming (SQP) algorithm.
4. For the optimization of well connections, the partial enumeration (PE) algorithm proved to be both efficient and robust compared to a genetic algorithm (GA).
5. The two-level programming approach developed for the entire production optimization problem is flexible. Different optimization algorithms can be used for the upper level problem and the lower level problem.
6. For a production optimization problem with multiple objectives, the weighted-sum method may fail to obtain solutions of interest to us.
7. Multiple solutions may exist for the multiphase flow problem of a gathering system. This may bring computational difficulties to full field simulations and production optimizations. Post-analysis should be performed to ensure the validity of a solution from a simulation or an optimization run.

9.2 Summary of Contributions

The major contributions of this study can be summarized as follows:

1. Optimization methods were developed and tested on various synthetic examples and applied successfully to a real petroleum field, the giant Prudhoe Bay oil field in Alaska. Results demonstrated that the methods developed have distinct advantages over conventional production optimization approaches.
2. A two-level programming approach was developed to optimize the production rates, lift gas rates, and well connections simultaneously subject to multiple flow rate and pressure constraints.

3. An appropriate formulation, Formulation P2 (Chapter 5), was proposed for the nonlinear rate allocation problem. The optimization problem formulated in this manner was solved by a SQP algorithm.
4. Several existing separable programming models (Model LP-I, LP-II, and MILP-I) were investigated and a new model (Model MILP-II) was proposed for the rate allocation problem. Several techniques were developed to speed up the optimization process, and they can be very effective for large-scale problems.
5. A partial enumeration (PE) method was developed to optimize well connections. The PE method is more efficient than a genetic algorithm (GA). Solutions obtained from the PE method were at least as good as those from the GA.
6. An efficient solution procedure was developed to simulate multiphase flow in a gathering system.
7. An efficient method, the Jacobian method, was proposed to compute the sensitivity coefficients of a gathering system. Sensitivity coefficients are required in applications such as optimization, model tuning, and sensitivity analysis.
8. A framework was developed to formulate and solve multiobjective production optimization problems. This framework unifies several existing multiobjective optimization algorithms. It was demonstrated through an example that the weighted-sum method may fail to obtain solutions of interest to us.
9. Some of the optimization tools were coupled with models for multiphase flow in the reservoir and surface pipeline network in a commercial reservoir simulator. This automates the application of optimization tools to short-term production optimization and long-term reservoir development studies.
10. An automatic differentiation technique was applied to compute derivatives required in network simulation and production optimization. Derivatives obtained from automatic differentiation are more accurate than those obtained from finite difference approximations. In certain cases, automatic differentiation is more efficient than finite difference approximations.

9.3 Recommendations for Further Study

1. Investigate whether well interactions within the reservoir are important for short-term production optimization problems. And if so, how to handle them efficiently.
2. Investigate the applicability of genetic algorithms on constrained production optimization problems. Genetic algorithms are slow, however, they are flexible and robust enough to handle optimization problems with complex simulation models and rough objective surfaces.
3. Investigate how the accuracy of the numerical models for multiphase pipe flow and choke flow impacts the optimal solution.
4. Investigate the impact of decisions obtained from short-term production optimizations on long-term oil recovery.
5. Develop optimization methods for gathering systems with loops.
6. Investigate multiobjective optimization methods for surface facility design problems.
7. Develop methods to find stable solutions of the multiphase network flow problem.
8. Develop methods to construct quality well performance curves efficiently.

Nomenclature

A	=	pipe area
B	=	formation volume factor
b_i	=	right-hand side of the ith constraint
C_0	=	profile parameter in the drift flux model
C_d	=	discharge coefficient in the Sachdeva choke model (Sachdeva et al., 1986)
C_p, C_v	=	specific heat capacity at constant pressure and volume, respectively
$\mathbf{c}$	=	constraint functions; cost coefficients for an objective function
c_p^v	=	velocity coefficient for flow rate of phase p (Section 4.4.1)
$\hat{\mathbf{c}}$	=	subset of constraints that are active at the optimal solution
d	=	choke inside diameter (Section 3.4); deviational variable (Chapter 7)
E_l, E_g	=	liquid and gas phase *in situ* fractions, respectively
$\mathbf{F}$	=	constraint functions; vector of objective functions (Chapter 7)
f	=	function
f_m	=	Fanning friction factor
f_o, f_w	=	flowing fractions of oil and water
g	=	gravity; gradient (Section 5.2)
h	=	achievement function (Chapter 7)
h_{ij}	=	the jth function involved in ith constraint of a separable problem
$\mathbf{J}$	=	Jacobian matrix
k	=	ratio of specific heat capacity
L	=	Lagrangian function

l	=	lower bound of a constraint
M	=	merit function
m	=	number of constraints
n	=	number of decision variables, polytropic exponent of gas (Section 3.4)
n_c	=	number of connections of a well
n_n	=	number of nodes
n_w	=	number of wells
p	=	pressure; search direction (Section 2.1.3, Section 5.2)
$Q_{p,j}^n$	=	flow rate limit of phase p for node j
q	=	flow rate
q_{cij}	=	flow rate of component c between node i and j
$q_{p,j}^n$	=	flow rate of phase p for node j
$q_{p,j}^w$	=	flow rate of phase p for well j
R_{em}	=	Reynolds number of a mixture
R_s	=	solution gas-oil ratio
r	=	number of discrete points of a piecewise linear well performance curve
$\mathbf{S}$	=	feasible set of an optimization problem
S	=	pipe perimeter
s	=	slack variable
U	=	velocity (Chapter 3); utility function (Chapter 7)
U_d	=	drift velocity of the gas in the drift flux model
U_{sg}	=	superficial velocity of the gas phase
U_{sl}	=	superficial velocity of the liquid phase
U_{sm}	=	superficial velocity of the mixture
u	=	upper bound of a constraint; system response (Section 3.6)
v	=	velocity
$v_{p,0}^{\max}$	=	the maximum *in situ* velocity along a flowline for a reference flow rate of phase p (Section 4.4.1)
w	=	weighting coefficients

x	=	decision variable, independent variable; mass fraction of gas (Section 3.4); well connections (Section 6.2)
$\mathbf{Y}$	=	feasible set of a vector of objective functions
y	=	lift gas rates (Section 6.2); ratio of downstream pressure to upstream pressure (Section 3.4); binary decision variable (Section 4.3.2)
y_c	=	critical ratio of downstream pressure to upstream pressure
z	=	value of an objective function; production rates (Section 6.2)
$\underline{z}$	=	lower bound for an objective function
$\bar{z}$	=	upper bound for an objective function

Acronyms

ADIFOR	=	an automatic differentiation software (Bischof et al., 1998)
BHP	=	bottomhole pressure
DFM	=	drift flux model for multiphase pipe flow
EFOS	=	E-field optimization system for the Prudhoe Bay oil field
EOA	=	Eastern Operating Area of the Prudhoe Bay oil field
GA	=	genetic algorithm
GOR	=	gas oil ratio
GLR	=	gas liquid ratio
GOM	=	Gulf of Mexico
GP	=	goal programming
IP	=	integer programming
LIP	=	linear integer programming
LP	=	linear programming
LP-I	=	a model for solving the rate allocation problem (Section 4.3.1)
LP-II	=	a model for solving the rate allocation problem (Section 4.3.4)
MILP	=	mixed integer linear programming
MILP-I	=	a model for solving the rate allocation problem (Section 4.3.2)
MILP-II	=	a model for solving the rate allocation problem (Section 4.3.3)
MIP	=	mixed integer programming
MOP	=	multiple objective optimization problem
NCP	=	nonlinearly constrained programming
P1	=	formulation P1, defined in Section 5.3.1
P2	=	formulation P2, defined in Section 5.3.2

PE	=	partial enumeration method for well connections optimization, defined in Section 6.3.2
SLC	=	sequential linearly constrained
SNOPT	=	a general-purpose system for solving large-scale optimization problems (Gill et al., 1998)
SP	=	separable programming
SQP	=	sequential quadratic programming
WAG	=	water alternating gas
WI	=	well index
WOA	=	Western Operating Area of the Prudhoe Bay oil field

Symbols

α	=	system parameters; steplength of a line search optimization method (Section 2.1.3, Section 5.2)
δ	=	delta (change)
ε	=	absolute pipe roughness (Section 3.3.1); tolerances (Section 3.5.2); perturbations (Section 3.6.1)
γ	=	specific gravity
λ	=	decision variable (Section 4.3); Lagrangian variable (Section 5.2); dual variable (Section 8.5.3)
$\lambda_{c,p}$	=	mobility of component c in phase p (Section 3.2)
μ	=	viscosities
θ	=	angle
ρ	=	density; penalty coefficients (Section 5.2)
τ_w	=	wall friction shear stress
Δt	=	time step length in reservoir simulation
Ω^n	=	set of all network nodes
Ω_i^n	=	set of nodes that directly connected to node i
Ω_i^w	=	set of wells whose flow streams enter node i
Ω_s^n	=	set of solution nodes
$\forall$	=	arbitrary
$\exists$	=	exist

∇	=	gradient
$\Vert \ \Vert_\infty$	=	L-infinite norm

Subscript

1	=	upstream
2	=	downstream
c	=	component
g	=	gas
l	=	liquid
lg	=	lift gas
m	=	mixture
o	=	oil
p	=	phase
w	=	water

Superscript

max	=	maximum
min	=	minimum
n	=	node
r	=	reservoir
s	=	surface
t	=	total
v	=	iteration index; velocity (Section 4.4.1)
w	=	well

Bibliography

ANDERSON, E., BAI, Z., BISCHOF, C., BLACKFORD, S., DEMMEL, J., DONGARRA, J., DU CROZ, J., GREENBAUM, A., HAMMARLING, S., MCKENNY, A., AND SORENSEN, D., 1999. *LAPACK User's Guide, Third Edition*, Society for Industrial Applied Mathematics.

ANDERSSON, J., 2001. *Multiobjective Optimization in Engineering Design – Applications to Fluid Power Systems*, dissertation No. 675, Linköping University, Linköping, Sweden.

ARONOFSKY, J.S., 1983. "Optimization Methods in Oil and Gas Development", paper SPE 12295.

ARONOFSKY, J.S., AND LEE, A.S., 1958. "The Use of Linear Programming Model for Scheduling Crude Oil Production", *Trans., AIME* 213, 51-54.

ASHEIM, H., 1988. "Criteria for Gas Lift Stability", *JPT*, November, 1452-1456.

ATTRA, H.D., WISE, W.B., AND BLACK, W.M., 1961. "Application of Optimizing Techniques for Studying Field Producing Operations", *Journal of Petroleum Technology*, January, 82-86.

AZARM, S., 2002. *Class Notes for Multiobjective Optimization*, Retrieved in October 2002 from http://www.glue.umd.edu/~azarm/optimum_notes/multi/multi.html.

AZIZ, K., AND DURLOFSKY, L., 2002. *Fundamentals of Reservoir Simulation*, Stanford University, Course material for PE223, winter 2001-2002.

AZIZ, K., AND PETALAS, N., 1994. "New PC-Based Software for Multiphase Flow Calculations", paper SPE 28249 presented at the SPE Petroleum Computer Conference, Dallas, 31 July-3 August.

BARNES, D.A., HUMPHREY, K., AND MUELLENBERG, L., 1990. "A Production Optimization System for Western Prudhoe Bay Field, Alaska", paper SPE 20653 presented at the 65th Annual Technical Conference and Exhibition of the Society of Petroleum Engineers held in New Orleans, LA, September 23-26.

BARUA, S., PROBST, A., AND DUTTA-ROY, K., 1997. "Application of a General-Purpose Network Optimizer to Oil and Gas Production", paper SPE 38838 presented at the 1997th SPE Annual Technical Conference and Exhibition held in San Antonio, Texas, 5-8 October.

BEGGS, H.D., 1991. *Production Optimization*, OGCI Publications, Oil and Gas Consultants International Inc., Tulsa.

BEGGS, H.D., AND BRILL, J.P., 1973. "A Study of Two Phase Flow in Inclined Pipes," *Trans. of Pet. Soc. of AIME*, vol. 256, 607.

BELLMAN, R., 1957. *Dynamic Programming*, Princeton University Press.

BENAYOUN, R., DE MONTGOLFIER, J., TERGNY, J., AND LARICHEV, O., 1971. "Linear Programming with Multiple Objective Functions: STEP Method (STEM)", *Mathematical Programming* **1**(3), 366-375.

BERTSEKAS, D.P., 1982. *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York.

BERTSIMAS, D., AND TSITSIKLIS, J., 1997. *Introduction to Linear Optimization*, Athena Scientific, Belmont, Massachusetts.

BERZ, M., BISCHOF, C., CORLISS, G., AND GRIEWANK, EDITORS, 1996. *Computational Differentiation: Techniques, Applications, and Tools*, SIAM, Philadelphia.

BISCHOF, C., CARLE, A., HOVLAND, P., KHADEMI, P., AND MAUER, A., 1998. *ADIFOR 2.0 User's Guide Technical Report (Revision D),* Mathematics and

Computer Science Division Technical Memorandum No. 192 and Center for Research on Parallel Computation Technical Report CRPC-95516-S.

BITTENCOURT, A.C., AND HORNE, R.N., 1997. "Reservoir Development and Design Optimization", paper SPE 38895 presented at the 1997 SPE Annual Technical Conference and Exhibition held in Antonio, Texas, 5-8 October.

BOHANNON, J.M., 1970. "A Linear Programming Model for Optimum Development of Multi-Reservoir Pipeline Systems", *Journal of Petroleum Technology*, November, 1429-1436.

BOYCHUK, L.M. AND OVCHINNIKOV, V.O., 1973. "Principal Methods of Solution Multicriterial Optimization Problems", *Soviet Automatic Control*, 6, 1-4.

BOYD, S., AND VANDENBERGHE, L., 2001. *Convex Optimization*, Course Reader for Class EE364: Convex Optimization with Engineering Applications. Stanford, CA. To be published in 2003.

BREAUX, E.J., MONROE, S.A., BLANK, L.S., YARBERRY, D.W., AND AL-UMRAN, S.A., 1985. "Application of a Reservoir Simulator Interfaced with a Surface Facility Network: A Case History", *Society of Petroleum Engineering Journal*, 397-404, June.

BRILL, J.P., AND BEGGS, H.D., 1998. *Two-Phase Flow in Pipes*, Class Notes, U of Tulsa.

BUITRAGO, S., AND RODRIGUEZ, E., AND ESPIN, D., 1996. "Global Optimization Techniques in Gas Allocation for Continuous Flow Gas Lift Systems", paper SPE 35616 presented at the Gas Technology Conference held in Calgary, Alberta, Canada 28 April - 1 May.

BYER, T.J., 2000. *Preconditioned Newton Methods for Simulation of Reservoirs with Surface Facilities*, Ph.D. dissertation, Stanford University.

CARROLL, D.L., 2000. *Fortran Genetic Algorithm Driver*, Retrieved in August 2000 from http://cuaerospace.com/carroll/ga.html.

CARROLL, J.A.III, AND HORNE, R.N., 1992. “Multivariate Optimization of Production Systems”, *Journal of Petroleum Technology*, July, 782-831.

CAO, H., 2002. *Development of Techniques for General Purpose Simulators*, Ph.D. dissertation, Stanford University, CA.

CHARNES, A., AND COOPER, W.W., 1961. *Management Models and Industrial Applications of Linear Programming*, Vol. I, Wiley, New York.

CHEN, Y.G., 2001. *Modeling Gas-Liquid Flow in Pipes: Flow Pattern Transitions and Drift-Flux Modeling*, MS thesis, Stanford University, CA.

CLAY, R.M., STOISITS, R.F., PRITCHETT, M.D., ROOD, R.C., AND COLOGGI, J.R., 1998. “An Approach to Real-Time Optimization of the Central Gas Facility at the Prudhoe Bay Field”, paper SPE 49123 presented at the SPE Annual Technical Conference and Exhibition held in New Orleans, Louisiana, 27-30 September.

COATS, K.H., 1980. “An Equation of State Compositional Model”, *SPEJ*, October, pp. 363-376.

COLEBROOK, C.F., 1939. “Turbulent Flow in Pipes with Particular Reference to the Transition Region Between the Smooth and Rough-Pipe Laws”, *J. Inst. Civil Engrs.*, **11**, 133.

COLLINS, M., COOPER, L., HELGASON, R., KENNINGTON J., AND LEBLANC, L., 1978. “Solving The Pipe Network Analysis Problem Using Optimization Techniques”, *Management Science*, Vol. 24, No. 7., March. pp. 747-760.

DANTZIG, G. B., 1963. *Linear Programming and Extensions*, Princeton University Press.

DONACHIE, R.P., 1973. “Digital Program for Water Network Analysis”, *Jo. Hydraulics Div., Proc. Amer. Soc. Civil Engineers*, Vol. 100, pp. 393-403.

DUNS, H.JR., AND ROS, N.C.J., 1963. “Vertical Flow of Gas and Liquid Mixtures in Wells”, Sixth World Petroleum Congress, Section 2, Paper No. 22, Frankfurt.

DUTTA-ROY, K., BARUA, S., AND HEIBA, A., 1997. "Computer-Aided Gas Field Planning and Optimization", paper SPE 37447 presented at the 1997 SPE Production Operations Symposium, held in Oklahoma City, Okalahoma, 9-11 March.

DUTTA-ROY, K., AND KATTAPURAM, J., 1997. "A New Approach to Gas-Lift Allocation Optimization", paper SPE 38333 presented at the 1997 SPE Western Regional Meeting held in Long Beach, California, 25-27 June.

EMANUEL, A.S., AND RANNEY, J.C., 1981. "Studies of Offshore Reservoir with an Interfaced Reservoir/Piping Network Simulator", *Journal of Petroleum Technology*, 399-406, March.

EPP, R., AND FOWLER, A.G., 1970. "Efficient Code for Steady-State Flows in Networks", *Jo. Hydraulics Div., Proc. Amer. Soc. Civil Engineers*, Vol. 96, pp. 43-56.

FANG, W.Y., AND LO, K.K., 1996. "A Generalized Well-Management Scheme for Reservoir Simulation", *SPE Reservoir Engineering*, May, 116-120.

FIACCO, A.V., AND MCCORMICK, G.P., 1968. *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley and Sons, New York and Toronto.

FISHBURN, P.C., 1974. "Lexicographic Orders, Utilities and Decision Rules: A Survey", *Mgmt Sci*. **20** (11), 1442-1471.

FITREMANN, J.M., AND VEDRINES, P., 1985. "Non Steady Gas Liquid Flow in Pipes and Gas-lifted Wells", *Proceedings of the Second International Conference on Multiphase Flow*, London, June 19-21, 245-262.

FLETCHER, R., 1987. *Practical Methods of Optimization,* 2nd ed. John Wiley & Sons, Inc., New York.

FONSECA, C.M., AND FLEMING, P.J., 1993. "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization", *Proceedings of the*

Fifth International Conference on Genetic Algorithms, Morgan Kaufmann Publishers, Inc., San Mateo, CA, pp. 416-423.

FUJII, H., 1993. *Multivafuate Production Systems Optimization in Pipeline Networks*, M.S. Thesis, Stanford University, CA.

FUJII, H., AND HORNE, R., 1995. “Multivariate Optimization of Networked Production Systems”, *SPE Production & Facilities*, August, 165-171.

GAL, T., AND NEDOMA, J., 1972. “Multiparametric Linear Programming”, *Mgmt Sci.* **18** (7), 406-421.

GEOQUEST, 2000. *Eclipse 300 Technical Description 2000A*, GeoQuest, Schlumberger.

GILL, P.E., MURRAY, W., AND SAUNDERS, M.A, 1998. *User's Guide for SNOPT 5.3: A Fortran Package for Large-scale Nonlinear Programming*, Retrieved in October 2002 from http://www.sbsi-sol-optimize.com/manuals.htm.

GILL, P.E., MURRAY, W., AND SAUNDERS, M.A., 2002. “SNOPT: An SQP Algorithm for Large-scale Constrained Optimization”, *SIAM J. Optimization*, Vol. 19, No. 4, pp. 979-1006.

GILL, P.E., MURRAY, W., AND WRIGHT, M.H., 1981. *Practical Optimization*, Academic Press,San Diego, CA

GOLDBERG, D.E., 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, MA.

GOMORY, R.E, 1958. “Outline of an algorithm for integer solutions to linear programs”, *Bulletin of the American Mathematical Society*, **64**, 275-278.

GREEN, A.S., JOHNSON, B.V., AND CHOI, H.J., 1993. “Flow-Related Corrosion in Large-Diameter Multiphase Flowlines”, *SPE Production & Facilities*, May, 97-100.

GÜYAGÜLER, B., 2002. *Optimization of Well Placement and Assessment of Uncertainty*, Ph.D. Disertation, Stanford Unvierstiy, CA, USA.

GÜYAGÜLER, B., AND BYER, T., 2001. *Production Optimization*. A Presentation Given at a Stanford SUPRI-O Seminar, December.

GÜYAGÜLER, B. AND HORNE, R.N., 2001. "Uncertainty Assesment of Well Placement Optimization", paper SPE 71265 presented at the 2001 SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, September 30-October 3.

GÜYAGÜLER, B., HORNE, R.N., ROGERS, L., AND ROSENZWEIG, J.J., 2000. "Optimization of Well Placement in a Gulf of Mexico Waterflooding Project", paper SPE 63221 presented at the 2000 SPE Annual Technical Conference and Exhibition held in Dallas, Texas, 1-4 October.

HAIMES, Y.Y., HALL, W.A., AND FREEDMAN, H.T., 1975. *Multiobjective Optimization in Water Resources Systems, The Surrogate Worth Trade-off Method*, Elsevier Scientific, New York.

HALL, M.A., 1976. "Hydraulic Network Analysis Using (Generalized) Geometric Programming", *Networks*, Vol. 6, pp. 105-130.

HARTLEY, H.O., 1961. "Nonlinear Programming by the Simplex Method", *Econometrica* 29, pp. 223-237.

HEIBA, A., BARUA, S., AND DUTTA-ROY, K., 1997. "Surface Facilities Management for Thermal Recovery Processes", paper SPE 37529 presented at SPE International Thermal and Heavy Oil Symposium, Bakersfield, CA, February.

HEPGULER, G., BARUA, S., AND BARD, W., 1997. "Integration of a Field Surface and Production Newtork with a Reservoir Simulator", *SPE Computer Applications*, 88-93, June.

HILLIER, F.S., AND LIEBERMAN, G.J., 2001. *Introduction to Operations Research, 7th Edition*, McCraw-Hill Companies, Inc., New York, NY,

HOLLAND, J.H., 1975. *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Harbor.

HONG, T.T., 1975. *Effect of the Variables on Optimization of Continuous Gas Lift System*, M.S. thesis, U. of Tulsa.

HWANG, C.L., PAIDY, S.R., AND YOON, K., 1980. "Mathematical Programming with Multiple Objectives: A Tutorial", *Computing and Operational Research* **7**, 5-31.

KANU, E.P., MACH, J., AND BROWN, K.B., 1981. "Economic Approach to Oil Production and Gas Allocation in Continuous Gas Lift", *Journal of Petroleum Technology*, October, 1887-1892.

KARMARKAR, N., 1984. "A New Polynomial-time Algorithm for Linear Programming", *Combinatorica*, **4**, 373-395.

KEENEY, R.L., AND RAIFFA, H., 1976. *Decision with Multiple Objectives: Preferences and Value Tradeoffs*, Wiley, New York.

KERMANI, M.B., AND HARROP, D., 1996. "The Impact of Corrosion on the Oil and Gas Industry", *SPE Production & Facilities*, August, 186-190.

KIRKPATRICK, S., GELATT, C.D.JR., AND VECCHI, M.P., 1983. "Optimization by Simulated Annealing", *Science*, 220, 4598, 671-680.

KLEE, V., AND MINTY, D.W., 1972. "How good is the simplex algorithm?", in *Inequalities - III*, O. Shisha (ed.), Academic Press, New York, NY, 159-175.

KUHN, H.W., AND TUCKER, A.W., 1951. "Nonlinear Programming", in *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability* (J. Neyman, ed.) pp. 481-492, Berkeley, University of California Press.

LAND, A.H., AND DOIG, A.G., 1960. "An Automatic Method for Solving Discrete Programming Problems", *Econometrica* 28, 497-520.

LANDA, J.L., 1997. *Reservoir Parameter Estimation Constrained to Pressure Transient, Performance History and Distributed Saturation Data*, Ph.D. dissertation, Stanford University, CA.

LANDMARK, 2001. *VIP-EXECUTIVE Technical Reference*, Landmark Graphics Corporation, September.

LASDON, L., COFFMAN, P.E., MACDONALD, R., MCFARLAND, J.W., AND SEPEHRNOORI, K., 1986. "Optimal Hydrocarbon Reservoir Production Policies", *Operations Research*, Vol. 34, No. 1, January-February, 40-54.

LEA, J.F.JR., AND TIGHE, R.E., 1983. "Gas Well Operation with Liquid Production", paper SPE 11583 prsented at the 1983 Production Operation Symposium held at Oklahoma City, Oklahoma, February 27 – March 1.

LENSTRA, J.K., RINOOY KAN, A.H.G., AND SCHRIJVER, A. (EDITORS), 1991. *History of Mathematical Programming: A Collection of Personal Reminiscences*, North Holland.

LITVAK, M.L., CLARK, A.J., FAIRCHILD, J.W., FOSSUM, M.P., MACDONALD, C.D., AND WOOD, A.R.O., 1997. "Integration of Prudhoe Bay Surface Pipeline Network and Full Field Reservoir Models", paper SPE 38885 presented at the 1997 SPE Annual Technical Conference and Exhibition held in Antonio, Texas, 5-8 October.

LITVAK, M.L., AND DARLOW, B.L., 1995. "Surface Network and Well Tubinghead Pressure Constraints in Compositional Simulation", paper SPE 29125 presented at the 13th SPE Symposium on Reservoir Simulation held in San Antonio, TX, 12-15 February.

LITVAK, M.L., HUTCHINS, L.A., SKINNER, R.C., DARLOW, B.L., WOOD, R.C., AND KUEST L.J., 2002. "Prudhoe Bay E-Field Production Optimization System Based on Integrated Reservoir and Facility Simulation", paper SPE 77643 presented at the SPE Annual Technical Conference and Exhibition held in San Antonio, Texas, 29 September – 2 October.

LIU, K.T., 1969. "The Numerical Analyses of Water Supply Networks by Digital Computer", *Thirteenth Congress of the International Association for Hydraulic Research*, Vol. 1 (Aug.-Sept.), pp. 36-43.

LO, K.K., AND HOLDEN, C.W., 1992. “Use of Well Management Schemes for Rate Forecasts”, paper SPE 24071 presented at the Western Regional Meeting held in Bakersfield, California, March 30-April 1.

LO, K.K., STARLEY, G.P., AND HOLDEN, C.W., 1995. “Application of Linear Programming to Reservoir Development Evaluations”, *SPE Reservoir Engineering*, February, 52-58.

MCFARLAND, J.W., LASDON, L., LOOSE, V., 1984. “Development Planning and Management of Petroleum Reservoirs Using Tank Models and Nonlinear Programming”, *Operations Research*, Vol. 32, No. 2, March-April, 270-289.

MICHALEWICZ, Z., 1996. *Genetic Algorithms+Data Structure = Evoluation Programs*, 3rd Edition, Springer-Verlag, Berlin.

MIETTINEN, K., 1999. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston.

MILLER, C.E., 1963. “The Simplex Method for Local Separable Programming”, in *Recent Advances in Mathematical Programming* (R. L. Graves and P. Wolfe, eds.), pp. 89-100, McGraw-Hill Book Co., New York.

MURRAY, W., 1997. “Sequential Quadratic Programming Methods for Large-Scale Problems”, *J. Comput. Optim. Appl.*, **7**, pp. 127-142.

MURRAY, W., 2000. *Class Notes for MS&E 312: Optimization Algorithms*, Stanford University, CA.

MURTAGH, B.A., 1981. *Advanced Linear Programming: Computation and Practice*. McGraw-Hill International Book Co., New York.

MURTAGH, B.A., AND SAUNDERS, M.A., 1982. “A Projected Lagrangian Algorithm and Its Implementation for Sparse Nonlinear Constraints”, *Mathematical Programming Study* 16, 84-117.

NISHIKIORI, N., REDNER, R.A., DOTY, D.R., AND SCHMIDT, Z., 1989. “An Improved Method for Gas Lift Allocation Optimization”, paper SPE 19711 presented at the 64th Annual Technical Conference and Exhibition of the Society of Petroleum Engineering held in San Antonio, TX, October 8-11.

OUYANG, L., 1998. *Single Phase and Multiphase Fluid Flow in Horizontal Wells*, Ph.D. dissertation, Stanford University, CA.

OYAMA, A., AND LIOU, M., 2002. “Multiobjective Optimization of A Multi-Stage Compressor Using Evolutionary Algorithm”, paper AIAA 2002-3535.

PADBERG, M., 2000. “Approximating Separable Nonlinear Functions Via Mixed Zero-One Programs”, *Operations Research Letters*, 27 (200), 1-5.

PALKE, M.R., AND HORNE, R.N., 1997a. “Nonlinear Optimization of Well Production Considering Gas Lift and Phase Behavior”, paper SPE 37428 presented at the 1997 SPE Production Operations Symposium held in Oklahoma City, Oklahoma, 9-11 March.

PALKE, M.R., AND HORNE, R.N., 1997b. “Determining the Value of Reservoir Data by Using Nonlinear Production Optimization Techniques”, paper SPE 28047 presented at the 1997 SPE Asia Pacific Oil & Gas Conference and Exhibition, held in Kuala Lumpur, Malaysia, 14-16 April.

PEACEMAN, D.W., 1978. “Interpretation of Well-Block Pressures in Numerical Reservoir Simulation”, *Soc. Pet. Eng. J.* (June) 183-194; *Trans. AIME*, 253.

POWELL, M.J.D., 1998. “Direct Search Algorithms for Optimization Calculations”, in *Acta Numerica*, Vol. 7, Cambridge University Press.

POWELL, D. AND SHOLNICK, M., 1993. “Using Genetic Algorithms in Engineering Design Optimization with Nonlinear Constraints”, *Proceedings of the Fifth International Conference on Genetic Algorithms*, Morgan Kaufmann. pp. 424-431.

PRESS, W.H., TEUKOLSKY, S.A., VETTERLING, W.T., AND FLANNERY, B.P., 1992. *Numerical Recipes in Fortran 77*, Cambridge University Press.

RASHEED, K., 1998. “An Adaptive Penaly Approach for Constrained Genetic Algortihm Optimization”, The Third Annual Conference on Genetic Programming (GP-98): Symposium on Genetic Algorithms (SGA-98).

RAVINDRAN, N., 1992. *Multivariate Optimization of Production Systems - the Time Dimension*, MS thesis, Stanford University, CA, USA.

SACHDEVA, R., SCHMIDT, Z., BRILL, J.P., BLAIS, R.M., 1986. "Two-Phase Flow Through Chokes", paper SPE 15657 presented at the 61st Annual Technical Conference and Exhibition of SPE. New Orleans, LA October 5-8.

SALUKVADZE, M.E., 1974. "On the Existence of Solution in Problems of Optimization under Vector Valued Criteria", *Journal of Optimization Theory and Applications*, **12**, 2, 203-217.

SCHIOZER, D.J., 1994. *Simultaneous Simulation of Reservoir and Surface Facilities*. Ph.D. dissertation, Stanford University, CA.

SIMSCI, 1999. *Input Manual for NETOPT*, Simulation Sciences, Brea, California, USA.

SIMSCI, 2001. *PIPEPHASE User's Guide*, Simulation Sciences, Brea, California, USA.

SPENDLEY, W., HEXT, G.R., AND HIMSWORTH, F.R., 1962. "Sequential Application of Simplex Designs in Optimisation and Evolutionary Operation", *Technometrics*, Vol. 4, pp. 441-461.

STEUER, R.E., 1977. "An Interactive Multiple Objective Linear Programming Procedure", in *Multiple Criteria Decision Making*, M. K. Starr and M. Zeleny (Eds), pp. 225-239, North Holland, New York.

STOISITS, R.F., BATESOLE, E.C., CHAMPION, J.H., AND PARK, D.H., 1992. "Application of Nonlinear Adaptive Modeling for Rigorous Representation of Production Facilities in Reservoir Simulation", paper SPE 24898 presented at the 67th Annual Technical Conference and Exhibition of the Society of Petroleum Engineers held in Washington, DC, October 4-7.

STOISITS, R.F., CRAWFORD, K.D., MACALLISTER, D.J., MCCORMACK, M.D., LAWAL, A.S., AND OGBE, D.O., 1999. "Production Optimization at the Kuparuk River Field Utilizing Neural networks and Genetic Algorithms", paper SPE 52177 presented at the 1999 Mid-Continent Operations Symposium held in Oklahoma City, Oklahoma, 28-31 March.

STOISITS, R.F., SCHERER, P.W., AND SCHMIDT, S.E., 1994. "Gas Optimization at the Kuparuk River Field", paper SPE 28467 presented at the 69th Annual Technical Conference and Exhibition held in New Orleans, LA, 25-28 September.

SVEDEMAN, S.J., AND ARNOLD, K.E., 1994. "Criteria for Sizing Multiphase Flowlines for Erosive/Corrosive Service", *SPE Production & Facilities*, February, 74-80.

TAITEL, Y., AND DUKLER, A.E., 1976. "A Model for Predicting Flow Regime Transitions in Horizontal and Near Horizontal Gas-Liquid Flow", *AIChE Journal*, 22, 47.

WALLENIUS, J., 1975. "Interactive Multiple Criteria Decision Methods: An Investigation and An Approach", *Acta Academiae Oecnonomicae Helsingiensis, Series A*: 14, Helsinki School of Economics, Finland.

WANG, P., LITVAK, M.L., AND AZIZ, K., 2002a. "Optimization Production from Mature Fields", paper presented at the 17th World Petroleum Congress, Rio De Janerio, Brazil, 1-5 September.

WANG, P., LITVAK, M.L., AND AZIZ, K., 2002b. "Optimization of Production Operations in Petroleum Fields", paper SPE 77658 presented at the SPE Annual Technical Conference and Exhibition held in San Antonio, Texas, 29 September – 2 October.

WOLSEY, L.A., 1998. *Integer Programming*, Wiley-Interscience, USA.

XIAO, W., LIU, Z., JIANG, M., AND SHI, Y., 1998. "Multiobjective Linear Programming Model on Injection Oilfield Recovery System", *Computers Math. Applic.*, Vol. 36, No. 5, pp. 127-135.

XIAO, J.J., SHOHAM, O., BRILL, J.P., 1990. "A Comprehensive Mechanistic Model for Two-Phase Flow in Pipelines", paper SPE 20631, 65th ATCE of SPE, New Orleans, September 23-26.

YETEN, B., DURLOFSKY, L.J., AND AZIZ, K., 2002. "Optimization of Nonconventional Well Type, Location and Trajectory", paper SPE 77565 presented at the 2002 SPE Annual Technical Conference and Exhibition, San Antonio, Texas, September 29-October 2.

ZAKIROV, I., AANONSEN, S., ZAKIROV, E., AND PALATNIK, B.M., 1996. "Optimizing Reservoir Performance by Automatic Allocation of Well Rates", paper presented at the 5th European Conference on the Mathematics of Oil Recovery, Leoben, Austria, 3-5 Sept.

ZHANG J., AND ZHU, D., 1996. "A Bilevel Programming Method for Pipe Network Optimization", *SIAM J. Optimization*, Vol. 6, No. 3, pp. 838-857, August.

Appendix A

Lists of Programs

Two computer programs were developed in the course of this study: a standalone package called NETwork Simulation and Optimization (NETSO) and a package of subroutines called Production Optimization Package (POP). Both packages are written in Fortran 77. NETSO was developed at Stanford University and implements the following simulation and optimization methods:

- The network simulation and sensitivity coefficient computation procedure described in Chapter 3.
- The MILP-I method for rate allocation described in Section 4.3.2 (The MILP-II method was implemented in another stand-alone package. The LP-I and LP-II methods were implemented in VIP-EXECUTIVE (Landmark, 2001)).
- The SQP nonlinear rate allocation method described in Chapter 5.
- The framework for multiobjective optimization of production operations described in Chapter 7.

POP was developed at BP and implements the following optimization methods:

- The LP-I and LP-II methods for rate allocation as described in Chapter 4. Pressure and velocity constraints described in Section 4.4.1 can be handled in this package. The speedup techniques described in Section 4.4.2 are implemented in this package.
- The partial enumeration method and the genetic algorithm for well connection optimization described in Chapter 6.

- The coupling procedure described in Chapter 8.

POP is a collection of subroutines that heavily depend on the data structure and subroutines of VIP-EXECUTIVE (Landmark, 2001) and is not available for use as a standalone package. Thus POP is not described further.

In this appendix, we describe NETSO in more detail.

A.1 Description of NETSO

NETSO is a standalone package written in Fortran 77 language. This section presents a brief description of program files and libraries used in NETSO, simple instructions about how to compile and run NETSO, and an example input file for NETSO.

A.1.1 Program Files

The program files can be classified roughly into the following categories:

- Simulation and sensitivity coefficient calculation.
- The SQP method for rate allocation.
- The separable programming methods for rate allocation.
- Miscellaneous files that define the global data structure, initialize the data structure, allocate memory, and perform read and write operations.

A brief description of each group of program files is given below.

Simulation and sensitivity coefficient calculation. Chapter 4 describes the simulation and sensitivity coefficient calculation procedures implemented in these program files.

netsim.f	the driver for network simulation and sensitivity calculation.
simsol.f	uses the Newton-Raphson method to solve the network problem.
calpd.f	calculate the BHP and its derivatives from the surface side, called from simsol.f.
calpu.f	calculates the BHP and its derivatives from the reservoir side.

fmseqn.f	calculates the Jacobian matrix and right-hand side of Eq. 3.28.
isconv.f	checks convergence of the Newton-Raphson method.

simsen.f	computes the sensitivity coefficients.
calpds.f	computes the BHP and its derivatives from the surface side, called from simsen.f.

fluid.f	implements functions for calculating fluid properties.
g_fluid.f	generated from fluid.f by ADIFOR.
pipe.f	implements the drift flux model for multipphase pipe flows.
g_pipe.f	generated from pipe.f by ADIFOR.
s_pipe.f	computes certain derivatives for sensitivity coefficient calculation.
choke.f	implements the Sachdeva et al. two phase choke model.
g_choke.f	generated from choke.f by ADIFOR.
s_choke.f	computes certain derivatives for sensitivity calculation.
well.f	implements a basic well model.

Separable programming method for rate allocation. Chapter 4 describes the optimization methods implemented in these program files.

ntgpc.f	generates well performance curves.
rdpc.f	reads performance curves from an input file.

genmps.f	generates model MILP-I for the rate allocation problem. The model is written to a file in MPS format (Murtagh, 1981) so that they can be solved by other stand-alone mixed integer programming solvers.

sndrv.f	implements the MILP-I method described in Section 4.3.2. The linear programming problems are solved by SNOPT. SNOPT implements a simplex method for solving an LP problem.
fmsnmilp.f	initializes the data structure for SNOPT.
snbbsol.f	implements the Branch and Bound method described in Section 4.3.2.

SQP method for Rate Allocation. Chapter 5 describes the optimization methods implemented in these program files.

netsof2.f	driver for nonlinear optimization of rate allocation problems with Formulation P2.
ntdata.f	setups the data structure for SNOPT with Formulation P2 (Section 5.3.3). It calls subroutines in files getdim.f, setopa.f, setbnd,f and setidat.f.
getdim.f	calculates the sizes of a rate allocation problem . Used for Formulation P2.
setopa.f	setups the structure of the Jacobian matrix for an optimization problem. Initialize the Jacobian matrix with appropriate values. Used for Formulation P2.
setbnd.f	setups the lower and upper bounds for decision variables and constraints. Used for Formulation P2.
setidat.f	initializes certain variables required by SNOPT. Used for Formulation P2.
netobj.f	dummy subroutines required by SNOPT. Used for Formulation P2.
netcon.f	computes the nonlinear constraint functions and their derivatives (the Jacobian matrix). Used for Formulation P2.
fmscst.f	computes the derivatives of nonlinear constraint functions. Used for Formulation P2.
setgcon.f	transposes the Jacobian matrix. Used for Formulation P2.
setnpar.f	numbers the decision variables of an optimization problem (Formulation P2) or the system parameters for sensitivity coefficient calculation.

netsof1.dat	driver for nonlinear optimization of rate allocation problems with Formulation P1.
fntdata.f	same use as ntdata.f except it is for Formulation P1 (Chapter 5).
fgetdim.f	same use as getdim.f except it is for Formulation P1.
fsetopa.f	same use as setopa.f except it is for Formulation P1.
fsetidat.f	same use as fsetidat except it is for Formulation P1.
fgcon.f	somputes the objective and constraint functions and their derivatives. Used for Formulation P1.

Miscellaneous files. These program files are used to define the global data structure, initialize the data structure, allocate memory, and perform read and write.

main.f	the main program.
inpobj.h	a header file, defines some parameters, variables, and common blocks.
net.h	a header file, defines the index variables.
ntbldt.f	data block file that initializes the index variables defined in net.h.
pnt.h	a header file, defines memory pointers for working arrays.
rd*.f90	a set of Fortran 90 files used to process the input file.
ntinit.f	initializes some variables.
chekinp.f	checks if the input variables have been properly assigned.
allocm.f	allocates memory for working arrays.
ntcnst.f	initializes the network structure.

A.1.2 Imported Libraries and Software

NETSO uses several libraries and programs from other sources. These are described below:

- **SNOPT.** SNOPT (Gill et al., 1998) is a general-purpose system for solving large-scale optimization problems. SNOPT implements a sequential quadratic programming (SQP) method that obtains search directions from a sequence of quadratic programming subproblems. SNOPT is most efficient if only some of the variables enter nonlinearly, or if the number of active constraints is nearly as large as number of variables. SNOPT requires relatively few evaluations of the problem functions, hence it is especially effective if the objective or constraint functions are expensive to evaluate. SNOPT was developed at the Stanford Systems Optimization Laboratory (SOL). More information on SNOPT can be found at http://www.sbsi-sol-optimize.com/SNOPT.htm.
- **ADIFOR.** ADIFOR (Bischof et al., 1998) is a tool for the automatic differentiation of functions written in Fortran 77. Given a Fortran 77 source code and a set of user specified dependent and independent variables, ADIFOR will generate an augmented derivative code that computes the partial derivatives of all of the specified output

variables with respect to all of the specified independent variables in addition to the original results. ADIFOR is a collaborative project between the Mathematics and Computer Science division at Argonne National Laboratory and the Center for Research on Parallel Computation at Rice University. More information on ADIFOR can be found at http://www-unix.mcs.anl.gov/autodiff/ADIFOR/.

- **LAPACK.** LAPACK (Anderson et al., 1999) is a standard linear algebra package, which can be downloaded from http://www.netlib.org/.
- **The PNNL Fortran Library.** The Pacific Northwest National Laboratory (PNNL) Fortran library is a set of general purpose Fortran 90 routines, which can be freely downloaded from http://www.pnl.gov/berc/flib/index.htm. This package was used in NETSO for processing input files.

A.1.3 Compile and Run

NETSO is written in Fortran-77 language and was developed on a Silicon Graphics Origin 200 workstation. All the source files reside in a subdirectory with a compile file named *makefile*. The libraries reside in other subdirectories that are specified in the *makefile.*

To compile NETSO, simply key in the *gmake* command to execute the instructions specified in file *makefile*. The created executable file is named *netso*.

To run NETSO, an input file named *input.dat* is required. An example input file is presented in the next section. If the run is to solve a rate allocation optimization using SNOPT, a file named *netso.spc* can be supplied, though it is not required by SNOPT. The keywords that can be used in file netso.spc are defined in Gill et al. (1998).

A.1.4 Example Input File

The following is the input file for the example problem presented in Section 3.7.

input.dat

```
C ***************************************************
C Fluid and rock property
```

```
C ***************************************************
OIL
API 31.0
GAS
sgpg 0.8
WATER
SGW  1.0
KROWTAB
       SW        KRW       KROW     PCOW
    0.12000  0.00000  1.00000  0.00000E+00
    0.13000  0.00000  1.00000  0.00000E+00
    0.92000  1.00000  0.00000  0.00000E+00
KROGTAB
         SG        KRG       KROG      PCOG
       0.00000  0.00000  1.00000  0.00000E+00
       0.00100  0.00000  1.00000  0.00000E+00
       0.02000  0.00000  0.99700  0.00000E+00
       0.05000  0.00500  0.98000  0.00000E+00
       0.12000  0.02500  0.70000  0.00000E+00
       0.20000  0.07500  0.35000  0.00000E+00
       0.25000  0.12500  0.20000  0.00000E+00
       0.30000  0.19000  0.09000  0.00000E+00
       0.40000  0.41000  0.02100  0.00000E+00
       0.45000  0.60000  0.01000  0.00000E+00
       0.50000  0.72000  0.00100  0.00000E+00
       0.60000  0.87000  0.00010  0.00000E+00
       0.70000  0.94000  0.00000  0.00000E+00
       0.85000  0.98000  0.00000  0.00000E+00
       1.00000  1.00000  0.00000  0.00000E+00
C ***************************************************
C Gathering system components
C ***************************************************
 PIPES
 NN NAME   DIAMET LENGTH DEPTH  TEMPUP TEMPDW  ROUGHN
C           inch   foot    foot   F       F       in
 1   TUB1    3.0    8000.0 8000.0  160  110    0.0003
 2   TUB2    3.5    8000.0 8000.0  160  110    0.0003
 3   TUB2    3.5    7500.0 7500.0  160  110    0.0003
 4   TUB2    3.5    7800.0 7800.0  160  110    0.0003
 5   TUB2    3.5    8000.0 8000.0  160  110    0.0003
 6   TUB2    3.5    7700.0 7700.0  160  110    0.0003
 7   TUB2    3.5    8500.0 8000.0  160  110    0.0003
 8   TUB2    3.5    7000.0 7000.0  160  110    0.0003
 9   TUB2    3.5    8000.0 8000.0  160  110    0.0003
 10  TUB10   3.5    8000.0 8000.0  160  110    0.0003
 11  P1      3.0     500.0    0.0  110  110    0.0003
 12  P2      3.5      60.0    0.0  110  110    0.0003
 13  P3      3.5    1200.0    0.0  110  110    0.0003
 14  P4      3.5     300.0    0.0  110  110    0.0003
 15  P5      3.5     400.0    0.0  110  110    0.0003
 16  P6      3.5     700.0    0.0  110  110    0.0003
 17  P7      3.5    1800.0    0.0  110  110    0.0003
 18  P8      3.5    2000.0    0.0  110  110    0.0003
 19  P9      3.5    1000.0    0.0  110  110    0.0003
 20  P10     3.5     600.0    0.0  110  110    0.0003
 21  P11     5.5    4000.0    0.0  110  110    0.0003
C
VALVE
 NN NAME DIAMET DISC  TEMPUP TEMPDW CTRTYP
C      1/64th inch     deg F deg F  DIAM/DP/FIXED
 1   C1   48   0.85    110    110   DIAM
```

```
 2   C2   48   0.85    110    110   DIAM
 3   C3   48   0.85    110    110   DIAM
 4   C4   48   0.85    110    110   DIAM
 5   C5   48   0.85    110    110   DIAM
 6   C6   48   0.85    110    110   DIAM
 7   C7   48   0.85    110    110   DIAM
 8   C8   48   0.85    110    110   DIAM
 9   C9   48   0.85    110    110   DIAM
 10  C10  48   0.85    110    110   DIAM
C
C Reservoir well grid blocks
BLOCK
NN NAME DX DY DZ    KX  KY   SW   SG   PRES   TEMP
1  B1   100 100 30  50  50   0.6  0.1  3800.  160.0
2  B2   100 100 30  100 100  0.3  0.4  3400.  160.0
3  B3   200 100 50  70  70   0.5  0.15 4500   160.0
4  B4   100 150 40  500 500  0.7  0.0  3900   160.0
5  B1   100 100 30  50  50   0.25 0.4  4800.  160.0
6  B2   100 100 30  100 100  0.2  0.3  3400.  160.0
7  B3   200 100 50  70  70   0.5  0.25 3500   160.0
8  B4   100 150 40  500 500  0.2  0.5  4900   160.0
9  B3   200 100 50  70  70   0.2  0.45 4200   160.0
10 B4   100 150 40  500 500  0.4  0.   3900   160.0
C
C Well flow rate or pressure constraints can be imposed here
C LFTTYP has following choices
C    AUTO:  used in option OPTF1, automatic allocation of lift gas.
C           QLFT is the upper limit
C    FIXED: QLFT is the fixed amount
WELL
 NN NAME IBLOC DIAM SKIN PMIN  QOM  QGM   QWM    QLM     LFTTYP QLFT
 1   W1    1  0.400   1  1000 1E23  1.0E15 1.0E15 1.0E16 AUTO 3000
 2   W2    2  0.475   0  1000 1E23  1.0E15 1.0E15 1.0E16 AUTO   0
 3   W1    3  0.400   1  1000 1E23  1.0E15 1.0E15 1.0E16 AUTO 2000
 4   W2    4  0.475   0  1000 1E23  1.0E15 1.0E15 1.0E16 AUTO  3000
 5   W1    5  0.400   1  1000 1E23  1.0E15 1.0E15 1.0E16 AUTO  0
 6   W2    6  0.475   0  1000 1E23  1.0E15 1.0E15 1.0E16 AUTO  0
 7   W1    7  0.400   1  1000 1E23  1.0E15 1.0E15 1.0E16 AUTO 2000
 8   W2    8  0.475   0  1000 1E23  1.0E15 1.0E15 1.0E16 AUTO  0
 9   W1    9  0.400   1  1000 1E23  1.0E15 1.0E15 1.0E16 AUTO  0
 10  W2    10  0.475   0  1000 1E23  1.0E15 1.0E15 1.0E16 AUTO 3000
C
C Nodes number, node name, and constraints
C nodes flowrate and pressure constraints can be imposed here
NODES
NN NAME PMIN  QOM    QGM       QWM      QLM
 1  H1  50  1.0E15  1.0E15   1.0E15   1.0E15
 2  H2  50  1.0E15  1.0E15   1.0E15   1.0E15
 3  H3  50  1.0E15  1.0E15   1.0E15   1.0E15
 4  H4  50  1.0E15  1.0E15   1.0E15   1.0E15
 5  H5  50  1.0E15  1.0E15   1.0E15   1.0E15
 6  H6  50  1.0E15  1.0E15   1.0E15   1.0E15
 7  H7  50  1.0E15  1.0E15   1.0E15   1.0E15
 8  H8  50  1.0E15  1.0E15   1.0E15   1.0E15
 9  H9  50  1.0E15  1.0E15   1.0E15   1.0E15
 10 H10 50  1.0E15  1.0E15   1.0E15   1.0E15
 11 I1  50  1.0E15  1.0E15   1.0E15   1.0E15
 12 I2  50  1.0E15  1.0E15   1.0E15   1.0E15
 13 I3  50  1.0E15  1.0E15   1.0E15   1.0E15
 14 I4  50  1.0E15  1.0E15   1.0E15   1.0E15
 15 I5  50  1.0E15  1.0E15   1.0E15   1.0E15
```

```
 16 I6  50  1.0E15  1.0E15   1.0E15   1.0E15
 17 I7  50  1.0E15  1.0E15   1.0E15   1.0E15
 18 I8  50  1.0E15  1.0E15   1.0E15   1.0E15
 19 I9  50  1.0E15  1.0E15   1.0E15   1.0E15
 20 I10 50  1.0E15  1.0E15   1.0E15   1.0E15
 21 M1  50  1.0E15  1.0E15   1.0E15   1.0E15
 22 SEP 160 1.0E15  7.0E24    8.0E23   1.0E15
C ****************************************************
C Gathering system structure
C ****************************************************
C For wells there are three types of STYP
C SEQ:  fixed open well choke
C SIEQ: adjustable well choke
C SHUT: well is shut-down
C for nodes, there are two types of STYP
C BND: boundary nodes, pressure are specified
C      this must be the root of a production tree
C CON: connection/junction node, pressure are continuous
C      at this node
C ====================================================
WELCON
NN OUTNOD OUTCON OUTCNT  STYP
 1    1     1    PIPE    SIEQ
 2    2     2    PIPE    SIEQ
 3    3     3    PIPE    SIEQ
 4    4     4    PIPE    SIEQ
 5    5     5    PIPE    SIEQ
 6    6     6    PIPE    SIEQ
 7    7     7    PIPE    SIEQ
 8    8     8    PIPE    SIEQ
 9    9     9    PIPE    SIEQ
 10   10    10   PIPE    SIEQ
NODCON
NN OUTNOD OUTCON OUTCNT  STYP
 1   11     1    CHOKE   CON
 2   12     2    CHOKE   CON
 3   13     3    CHOKE   CON
 4   14     4    CHOKE   CON
 5   15     5    CHOKE   CON
 6   16     6    CHOKE   CON
 7   17     7    CHOKE   CON
 8   18     8    CHOKE   CON
 9   19     9    CHOKE   CON
 10  20     10   CHOKE   CON
 11  21     11    PIPE   CON
 12  21     12    PIPE   CON
 13  21     13    PIPE   CON
 14  21     14    PIPE   CON
 15  21     15    PIPE   CON
 16  21     16    PIPE   CON
 17  21     17    PIPE   CON
 18  21     18    PIPE   CON
 19  21     19    PIPE   CON
 20  21     20    PIPE   CON
 21  22     21    PIPE   HEAD
C
C Define boundary nodes
 NODCON
 NN STYP
 22 BND
C
```

```
C*****************************************************
C Total gas-lift volume, only used in option OPTF1
C*****************************************************
LFTVOL  3e23
C ***************************************************
C Optimization according to operation policies
C ***************************************************
INTVAR
NVAR LBND UBND XINIT
  1   0   4000   300
  2   0   4000   400
  3   0   4000   1200
  4   0   4000   400
  5   0   4000   500
  6   0   4000   1000
  7   0   4000   20
  8   0   2000   400
  9   0   2000   400
  10  0   5000   1500
DVAR
NVAR  DEVTYP IDEV  PAR  LBND  UBND  XINIT
 1     WELL   1    GLFT   0  6000   3000
 2     WELL   2    GLFT  0   6000   3000
 3     WELL   3    GLFT   0  6000   3000
 4     WELL   4    GLFT   0  6000   3000
 5     WELL   5    GLFT   0  6000   3000
 6     WELL   6    GLFT   0  6000   3000
 7     WELL   7    GLFT   0  6000   3000
 8     WELL   8    GLFT   0  6000   3000
 9     WELL   9    GLFT   0  6000   3000
 10    WELL   10   GLFT   0  6000   3000
C
HCON
NCON  DEVTYP IDEV PAR   LBND    UBND
c 1     NODE   22  OIL   14000      14000
 1     NODE   22  WAT    0          6400
 2     NODE   22  GAS    0         900000
C
GOAL 1
 NOBJ  DEVTYP  IDEV PAR   OBJ  TARG COEF
  1    FIELD    0   OIL   MAX  2000   1
GOAL 2
 NOBJ  DEVTYP  IDEV PAR  OBJ  TARG COEF
  1    FIELD    0    GAS  MAX  2000  1
GOAL 3
 NOBJ  DEVTYP  IDEV PAR OBJ  TARG  COEF
  1    FIELD    0   WAT  MIN  2000   1
C
C define the jobs
JOBS
RANK  IGOAL MAX   RELF
 1      1   YES   0.8
c 2      3    NO    0.1
C
SOLVER SEN
```

Appendix B

Problem Data for the Gas-lift Example in Section 4.4.3

Table B.1: Gas injection and oil production rates for a set of 56 wells (from Buitrago et al. (1996)).

Well	Lift Gas Rate, MSCF/d Oil Rate, STB/d									
1	0.	225.	314.	672.	768.	1485.				
	0.	290.	324.	386.	391.	433.				
2	0.	68.	144.	266.	450.	693.	861.	1035.		
	487.	530.	560.	591.	626.	652.	665.	676.		
3	0.	60.	129.	210.	348.	521.	748.	792.	1027.	
	441.	481.	516.	541.	575.	605.	629.	632.	651.	
4	0.	42.	86.	542.	945.					
	280.	284.	287.	307.	315.					
5	0.	35.	71.	464.	1164.					
	281.	285.	289.	306.	315.					
6	0.	74.	157.	294.	505.	890.	926.	1114.		
	287.	315.	333.	352.	366.	376.	376.	379.		
7	0.	114.	235.	386.	577.	1279.	1507.			
	790.	815.	836.	857.	876.	908.	914.			
8	0.	58.	97.	268.	651.	1725.				
	0.	209.	233.	276.	297.	311.				
9	0.	551.	716.	1295.	1706.					
	0.	1038.	1199.	1568.	1424.					
10	0.	66.	135.	215.	308.	413.	553.	756.	975.	1053.
	233.	240.	246.	252.	258.	263.	268.	274.	279.	281.
11	0.	91.	201.	397.	675.	1054.	1378.	1740.		
	548.	656.	727.	814.	896.	958.	987.	1021.		
12	0.	617.	800.	1345.	1513.	2880.	6765.			
	0.	459.	510.	590.	614.	679.	687.			

Table B.1: Gas injection and oil production rates for a set of 56 wells (cont).

Well	Lift Gas Rate, MSCF/d Oil Rate, STB/d									
13	0.	16.	33.	181.	1218.						
	108.	109.	111.	118.	137.						
14	0.	128.	186.	413.	1105.	1111.					
	0.	277.	302.	336.	361.	360.					
15	0.	65.	153.	252.	381.	598.	884.	1112.	1392.		
	351.	420.	493.	549.	594.	648.	688.	709.	735.		
16	0.	40.	69.	253.	460.	926.	982.	907.			
	0.	170.	198.	310.	361.	415.	417.	455.			
17	0.	128.	262.	370.	494.	647.	823.	1081.	2174.		
	892.	909.	931.	946.	960.	975.	986.	1001.	1029.		
18	0.	138.	282.	468.	705.	1027.	2297.				
	1151.	1185.	1213.	1240.	1266.	1290.	1317.				
19	0.	86.	175.	295.	469.	686.	943.	1362.			
	310.	317.	323.	329.	335.	342.	347.	353.			
20	0.	136.	215.	465.	541.	975.	1948.	4225.	12161.		
	0.	162.	213.	303.	316.	391.	458.	496.	476.		
21	0.	189.	289.	772.	894.	1479.					
	0.	251.	307.	455.	466.	523.					
22	0.	270.	370.	727.	816.	1929.					
	0.	195.	214.	234.	234.	237.					
23	0.	110.	221.	1669.							
	944.	948.	951.	957.							
24	0.	49.	105.	182.	283.	399.	556.	725.	767.	1030.	3621.
	956.	1062.	1149.	1240.	1338.	1420.	1505.	1578.	1594.	1680.	1955.
25	0.	51.	103.	160.	225.	551.	839.				
	487.	495.	502.	509.	516.	536.	548.				
26	0.	54.	120.	200.	319.	579.	734.	1023.	1972.	3317.	
	82.	94.	105.	113.	119.	127.	129.	133.	141.	139.	
27	0.	54.	109.	590.	1100.						
	353.	354.	355.	358.	357.						
28	0.	105.	211.	1618.							
	1044.	1051.	1056.	1079.							
29	0.	69.	141.	224.	322.	439.	579.	754.	760.		
	184.	188.	191.	195.	198.	201.	204.	206.	206.		
30	0.	56.	112.	545.	561.						
	308.	309.	310.	309.	308.						
31	0.	37.	75.								
	354.	354.	354.								
32	0.	64.	131.	217.	1093.						
	618.	638.	654.	664.	728.						
33	0.	184.	280.	682.	897.	1646.	4485.				
	0.	185.	211.	257.	265.	291.	308.				
34	0.	31.	63.	366.	1299.						
	209.	210.	210.	215.	221.						
35	0.	40.	86.	144.	195.	283.	389.	515.	717.	801.	1273.
	162.	177.	190.	207.	216.	229.	242.	251.	263.	267.	281.
36	0.	28.	59.	108.	166.	235.	330.	404.	1145.		
	179.	189.	195.	204.	213.	220.	227.	232.	253.		

Table B.1: Gas injection and oil production rates for a set of 56 wells (cont).

Well	Lift Gas Rate, MSCF/d Oil Rate, STB/d								
37	0.	34.	71.	126.	199.	292.	409.	564.	772.	1346.
	64.	67.	69.	72.	74.	77.	79.	82.	84.	87.
38	0.	112.	157.	345.	690.	1891.				
	0.	270.	282.	310.	326.	339.				
39	0.	105.	168.	301.	800.	860.				
	0.	131.	174.	207.	246.	251.				
40	0.	98.	116.	186.	330.	757.	1149.			
	0.	27.	28.	29.	31.	35.	38.			
41	0.	63.	126.	636.	801.					
	372.	373.	373.	374.	374.					
42	0.	28.	57.	361.	1701.					
	200.	201.	202.	207.	201.					
43	0.	538.	797.	976.	1300.	2920.	3650.			
	0.	256.	337.	362.	407.	467.	198.			
44	0.	61.	123.	666.	1303.					
	397.	403.	406.	422.	427.					
45	0.	11.	21.	137.	712.					
	83.	83.	83.	87.	93.					
46	0.	7.	14.	22.	31.	40.	50.	61.	91.	700.
	47.	48.	50.	51.	52.	53.	55.	56.	59.	80.
47	0.	1558.	2653.	2849.	2982.	3042.				
	0.	0.	0.	0.	394.	441.				
48	0.	1450.	1633.	2466.	5096.					
	0.	0.	197.	483.	490.					
49	0.	1228.	1268.	1418.	1690.					
	0.	0.	0.	232.	248.					
50	0.	1135.	1441.	2538.	3980.					
	0.	0.	146.	188.	188.					
51	0.	162.	369.	2009.	2224.	3488.	4899.			
	0.	0.	0.	0.	223.	232.	237.			
52	0.	1512.	2536.	2671.	2750.	2830.				
	0.	0.	0.	194.	260.	317.				
53	0.	1174.	1304.	1484.						
	0.	0.	186.	267.						
54	0.	1559.	2340.	2594.	2650.	3342.				
	0.	0.	0.	278.	300.	328.				
55	0.	465.	1333.	2169.	2317.	2445.	2596.	3100.		
	0.	0.	0.	0.	152.	160.	167.	184.		
56	0.	761.	1247.	1590.	1655.	1730.	1770.	2300.	5360.	
	0.	0.	0.	350.	403.	444.	452.	505.	658.	

Table B.2: Gas injection and oil production rates for a set of 56 wells obtained from the MILP-II method.

Well	Oil Rate STB/d	Gas Rate MSCF/d	Well	Oil Rate STB/d	Gas Rate MSCF/d	Well	Oil Rate STB/d	Gas Rate MSCF/d
1	386	672	20	391	975	39	207	301
2	626	450	21	455	772	40	27	98
3	605	521	22	214	370	41	372	0
4	280	0	23	944	0	42	200	0
5	281	0	24	1680	1030	43	337	797
6	333	157	25	487	0	44	397	0
7	836	235	26	105	120	45	83	0
8	276	268	27	353	0	46	50	14
9	1568	1295	28	1044	0	47	441	3042
10	233	0	29	184	0	48	483	2466
11	957	1048	30	308	0	49	232	1418
12	510	800	31	354	0	50	0	0
13	108	0	32	654	131	51	0	0
14	302	186	33	211	280	52	0	0
15	648	598	34	209	0	53	267	1484
16	361	460	35	216	195	54	0	0
17	892	0	36	204	108	55	0	0
18	1213	282	37	64	0	56	452	1770
19	310	0	38	282	157			