VECTORIZATION OF A PENALTY FUNCTION
ALGORITHM FOR WELL SCHEDULING

ILYAS ABSAR

SOHIO PETROLEUM COMPANY

SAN FRANCISCO, CALIFORNIA

Vectorization of a Penalty Function Algorithm for Well Scheduling

Ilyas Absar

SOHIO Petroleum Co.

San Francisco,California

Abstract:

In petroleum engineering, the oil production profile of a reservoir can be simulated by using a finite grided model. This profile is affected by the number and choice of wells which in turn is a result of various production limits and constraints including, for example, the economic minimum well spacing, the number of drilling rigs available and the time required to drill and complete a well. After a well is available it may be shut-in because of excessive water or gas productions. In order to optimize the field performance a penalty function algorithm was developed for scheduling wells. For an example with some 343 wells and 15 different constraints, the scheduling routine vectorized for the Cyber 205 averaged 560 times faster performance than the scalar version.

Introduction:

Mathematical modelling of the fluid production from a naturally occurring reservoir involves considering the reservoir as a network of interconnected blocks. To each grid block is associated a geologic description through properties, e.g., thickness, porosity, permeability, etc. Each grid block is considered to be in material balance with its surroundings, i.e., the amount of fluid in the block at time $t + \Delta t$ is equal to the amount of fluid in that block at time t plus fluid influx in the time interval Δt minus fluid outflux in the time interval Δt.

In Figure 1A, the reservoir is shown by a curved boundary. Overlaid areally is a rectangular grid. The sizes of the blocks can be chosen to represent the geological features of the reservoir as accurately as possible. Figure 1B shows a two dimensional cross-section of a reservoir and the grid used for its simulation. Notice that the reservoir contains water, oil and gas in various regions, and only some blocks are in communication with the wells by means of perforations in the well bore. To simulate the production profile, the material balance of the grid blocks in which wells are perforated must also take into account the fluid production or injection. In this manner one obtains pressures and saturations for each of the grid-blocks. For details on mathematical modelling of oil reservoirs please refer to a standard text, for example, references 1 and 2.

Once a reservoir simulator is formulated, it can be used in many ways, e.g.:

1. Assist in making economic decisions for field operation, e.g., the investments to date at Prudhoe Bay exceed $9 billion.

2. Design of production strategy. The effect of changes in the number, location, spacing, or timing of wells can be studied.

3. Prediction of reservoir performance.

4. Matching of the production history.

When an oil field is developed, of course the most important objective is to maximize oil recovery. However, this objective is tempered by limitations, economic and physical, e.g., costs and capacities of various installations and devices.

The dashed curve in Figure 2 represents oil production when all wells flow at their maximum capacity. The area under this curve represents cumulative oil production. The ratio of cumulative oil production to in-place oil represented as a fraction or percentage is called the Oil Recovery Factor. If facilities were constructed for this production profile, they would have to be constructed to handle oil production at the maximum rate, q_{max}. Economic considerations give us a target oil rate, q_T, less than q_{max}, at which oil production can be sustained for a period of time. The solid curve in Figure 1 represents this strategy. Note that sometimes this can be achieved without appreciable sacrifice in cumulative oil production.

Well Scheduling Problem:

Once q_T is established, the problem of optimal scheduling, i.e., selecting for operation a given number of wells (say n) can be represented mathematically as follows:

Maximize,
$$\sum_{i=1}^{n} q_i \leqslant q_T$$

The maximum production rates of oil, gas and water are, however, limited to the capacity of the reservoir facilities. Thus, the field oil production is subject to constraints of the form:

$$\sum_{i} x_i q_i \leqslant L$$

where,

q_i is the oil production rate from well i,

q_T is the target oil production rate for the field,

x_i is either 1 or the gas-oil ratio or the water-oil ratio for well i,

$x_i q_i$ is then the oil or gas, or water production/injection rate.

and L is the oil or gas or water production of injection constraint.

Some examples of these limits are:

1. Fieldwide gas handling capacity,
2. Water injection limit,
3. Oil production limit at a station due to pipeline size,
4. Gas-lift capacity available.

In order to select wells for production, each well can be assigned a priority. In the penalty function approach priority assignment, is made with a function which becomes large as a particular constraint approaches violation.

Suppose (k-1) wells have been already chosen.

For choosing the k th well subject to a constraint of the form:

$$\sum x_i q_i \leq L,$$

a simple penalty function is:

$$p(k) = (\sum_{i=1}^{k-1} x_i q_i + x_k q_k)/ L$$

The penalty function p(k) has a value for each of the available wells, and arranges the set of available wells in order according to this particular constraint.

When there are several (say m) constraints, penalty functions $p_1(k)$, $p_2(k)$---$p_m(k)$ can be obtained similarly.

Since each constraint is individually fatal for well scheduling purposes, the violation of one constraint is as bad as any other.

Hence, an overall penalty function can be of the form:

$$p(k) = \max_{j=1 \ldots m} p_j$$

Results and Discussion:

The implementation of this scheme involves calculating for each available well, m different $p_i(k)$ and then obtaining an overall penalty, $p(k)$ as the maximum of these m values. Thereafter the well with the lowest value of $p(k)$ is selected. This procedure is repeated selecting one well at a time until the target rate q is achieved without violating any of the constraints. If the target rate cannot be achieved without violating one or more constraints, we are on the decline portion of the production curve.

This scheme was programmed into a three dimensional, three phase (oil, gas, water) simulator. The simulator originally used a simple prioritization scheme based on gas-oil ratios. When a scalar version of the penalty function algorithm was introduced, the simulator ran appreciably slower. It was therefore decided to vectorize the penalty function algorithm.

To calculate the penalty function in a case with n wells and m constraints declare an array p (n, m). Usually n is much greater than m.

For each of the m constraints vectorize the penalty calculation, e.g., for constraint i, store the values of $p_i(k)$ in the elements of p (n, m), starting with p (1, i) and ending at p (n, i).

Next, using a WHERE comparison statement pick out the largest of the m values for each well. We now have the priority p (k) for each well. Use the Q8SMINI call to pick out the minimum value. If this value exceeds 1., no well can be chosen without violating a constraint.

TABLE 1.

	Case 1	Case 2
No. of wells	119	343
No. of constraints	9	15
Average Well Selection Time (sees) Scalar:	.14	1.6
Vector:	.001245	.00287
Scalar: Vector Ratio	112	560

A summary of results for two cases is presented in Table 1. For a reservoir with 119 wells and nine constraints, the vector algorithm was on the average 112 times faster than the scalar version. For a larger example, Case 2 in Table 1, 343 wells with 15 constraints, the vector algorithm achieved even more spectacular results, an average acceleration factor of 560.

The details of Case 1 are represented graphically in Figure 3. In the scalar algorithm, the time required for selection of wells increases monotonically for each subsequent selection. The selection of the first well required only .005 secs while the selection of the 65th well required .226 secs. However, in the vector algorithm, each well selection required .001244 secs, except for the first, which required .00155 secs.

Similarly, for Case 2, the vector algorithm took .00287 secs for each well selection, except for the first well, for which it took .00447 secs. The scalar algorithm had a monotonic increase from .0185 secs for the first well, to 2.641 secs for the 220th well. This means that the selection of the 220th well was some 920 times faster in the vector algorithm as compared to the scalar version.

Conclusions:

Clearly as the number of wells and the number of constraints increase, the advantage of the vectorized version over the scalar version becomes greater.

The reservoir simulator with the vectorized well selection scheme, including the more complicated penalty function scheme, ran faster than the original version with the simpler scalar well selection scheme.

In short, judicious use of vectorization can make feasible highly desirable enhancements to large simulators.

References:

1. D. W. Peacemain, Fundamentals of Numerical Reservoir Simulation, Elsevier Scientific Publishing Company, N.Y., 1977

2. K. Aziz and A. Settari, Petroleum Reservoir Simulation, Applied Science Publishers Ltd., London, 1979
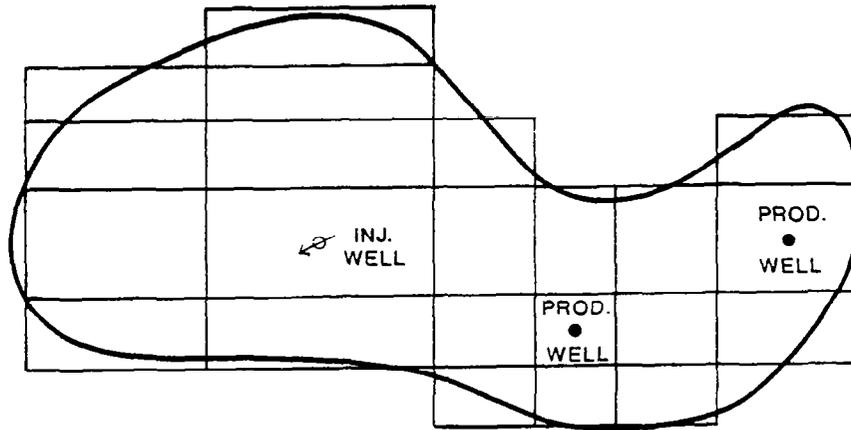
FIGURE 1A.
RECTANGULAR GRID TO REPRESENT
A RESERVOIR. EACH BLOCK MAY
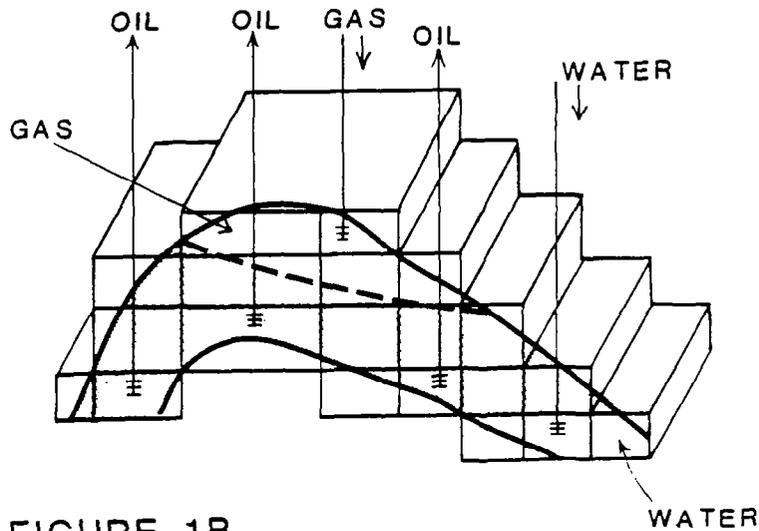HAVE DIFFERENT THICKNESS AND
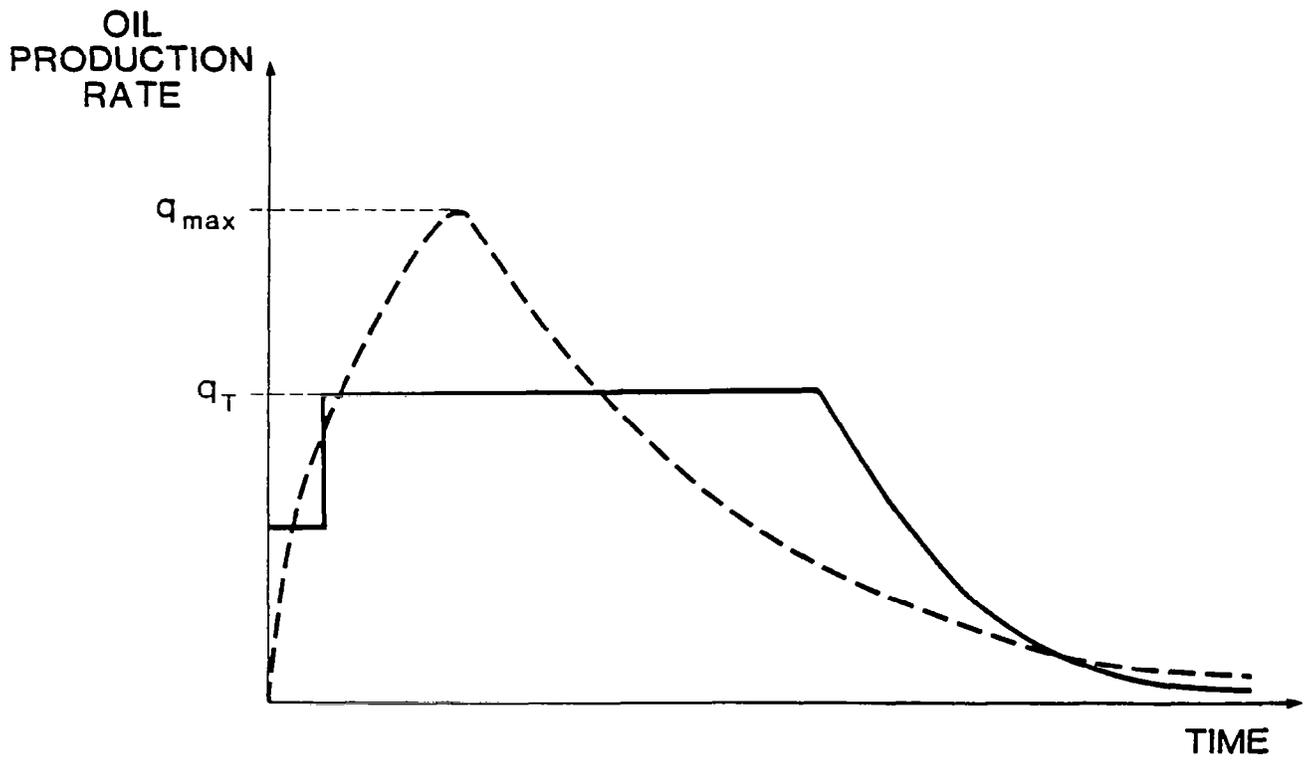POROSITY.

FIGURE 1B.
CROSS-SECTION OR A GRID WITH
DIFFERENT TYPES OF WELLS.
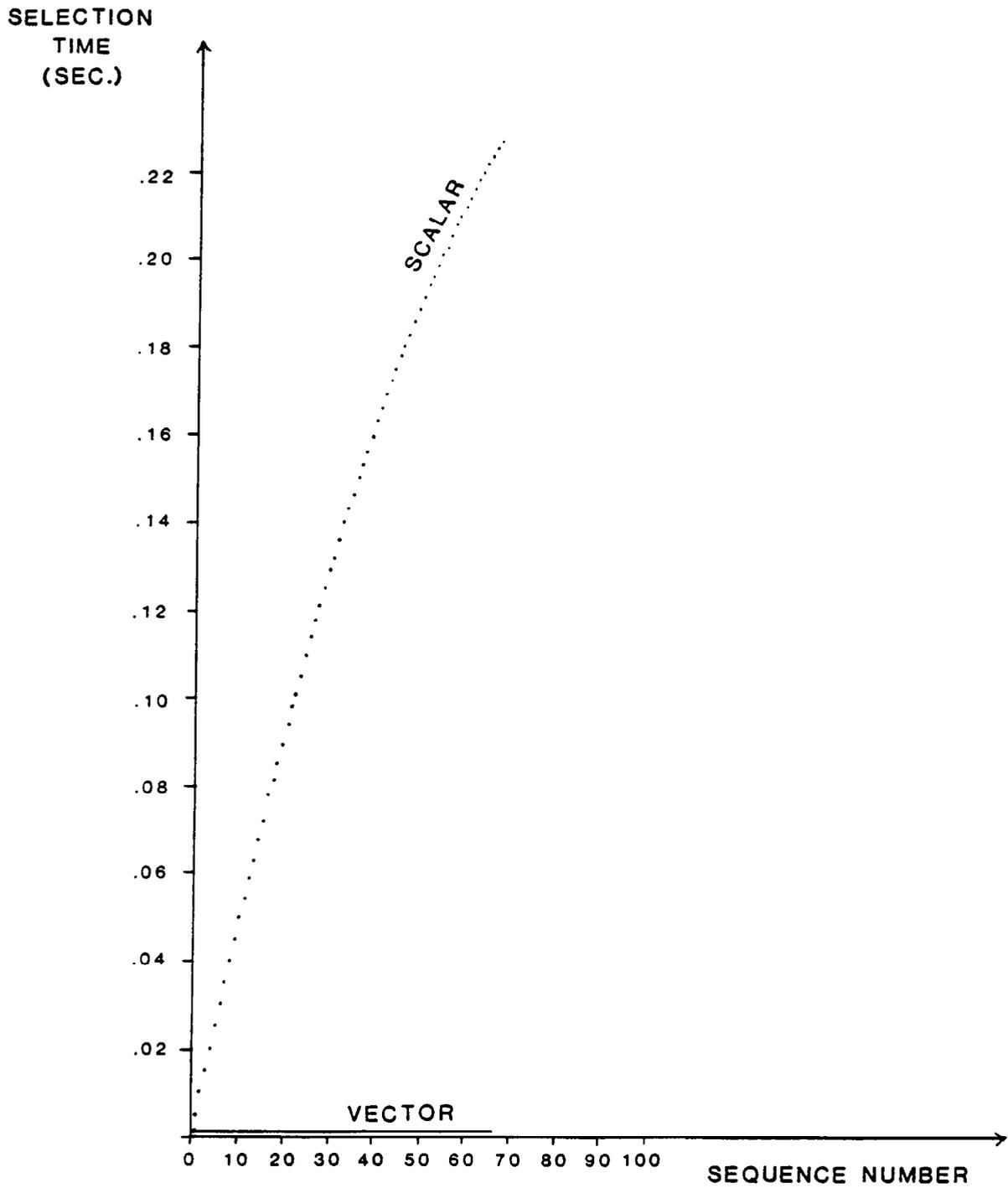
FIGURE 2.

PRODUCTION PROFILE FOR AN OIL FIELD.

FIGURE 3.

370