APPLICATION OF ARTIFICIAL INTELLIGENCE TO RESERVOIR CHARACTERIZATION: AN INTERDISCIPLINARY APPROACH

(DOE Contract No. DE-AC22-93BC14894)

Submitted by

The University of Tulsa
Tulsa, OK 74104

Contract Date: October 1, 1993
Anticipated Completion Date: September 30, 1996
Government Award: $240,540
Program Manager: B.G. Kelkar
R.F. Gamble
Principal Investigators: D.R. Kerr
L.G. Thompson
S. Shenoi
Reporting Period: Octover 1 - December 31, 1995

Contracting Officer's Representative

Mr. Robert E. Lemmon
Pittsburgh Energy Technology Center
P.O. Box 10940, M/S 141-L
Pittsburgh, PA 15236-0940

DISCLAIMER

Objectives

The basis of this research is to apply novel techniques from Artificial Intelligence and Expert Systems in capturing, integrating and articulating key knowledge from geology, geostatistics, and petroleum engineering to develop accurate descriptions of petroleum reservoirs. The ultimate goal is to design and implement a single powerful expert system for use by small producers and independents to efficiently exploit reservoirs.

The main challenge of the proposed research is to automate the generation of detailed reservoir descriptions honoring all the available "soft" and "hard" data that ranges from qualitative and semi-quantitative geological interpretations to numeric data obtained from cores, well tests, well logs and production statistics. In this sense, the proposed research project is truly multi-disciplinary. It involves significant amount of information exchange between researchers in geology, geostatistics, and petroleum engineering. Computer science (and artificial intelligence) provides the means to effectively acquire, integrate and automate the key expertise in the various disciplines in a reservoir characterization expert system. Additional challenges are the verification and validation of the expert system, since much of the interpretation of the experts is based on extended experience in reservoir characterization.

The overall project plan to design the system to create integrated reservoir descriptions begins by initially developing an AI-based methodology for producing large-scale reservoir descriptions generated interactively from geology and well test data. Parallel to this task is a second task that develops an AI-based methodology that uses facies-biased information to generate small-scale descriptions of reservoir properties such as permeability and porosity. The third task involves consolidation and integration of the large-scale and small-scale methodologies to produce reservoir descriptions honoring all the available data. The final task will be technology transfer. With this plan, we have carefully allocated and sequenced the activities involved in each of the tasks to promote concurrent progress towards the research objectives. Moreover, the project duties are divided among the faculty member participants. Graduate students will work in teams with faculty members.

The results of the integration are not merely limited to obtaining better characterizations of individual reservoirs. They have the potential to significantly impact and advance the discipline of reservoir characterization itself.

Summary of Technical Progress

1. Decomposition of System

We have decomposed the overall system development into smaller component parts to allow us to focus on the expert knowledge required for that component. In

addition, the decomposition will facilitate the implementation of the system and its validation and verification. The three component systems will be representative of how each of the experts in geology, geostatistics, and engineering characterizes the reservoir. Figure 1 describes a model for this breakdown. The concurrent development of these component systems fits into the development of the large and small scale aspects of the system as originally stated in the proposal.

The geostatistical system continues to be tested and updated. This sytem includes the use of wavelet transforms to determine the effect of compression to some part of the original data on the overall performance of the reservoir. Concentration on the geology system has been placed on upgrading the neural network output for log facies recognition. In addition, we continue to develop rules for correlation of zones among wells. The marker bed recognition system is considered complete at this time, though later enhancements may be added. The individual components (completion rules, type curve matching, and linear regression components) are currently being integrated to form a complete well test interpretation system. The graphical system is currently being designed for implementation to visualize correlations between wells. This system will be augmented as the other system components mature. The designing of the overall user interface to integrate all of the systems will begin in the following quarter.

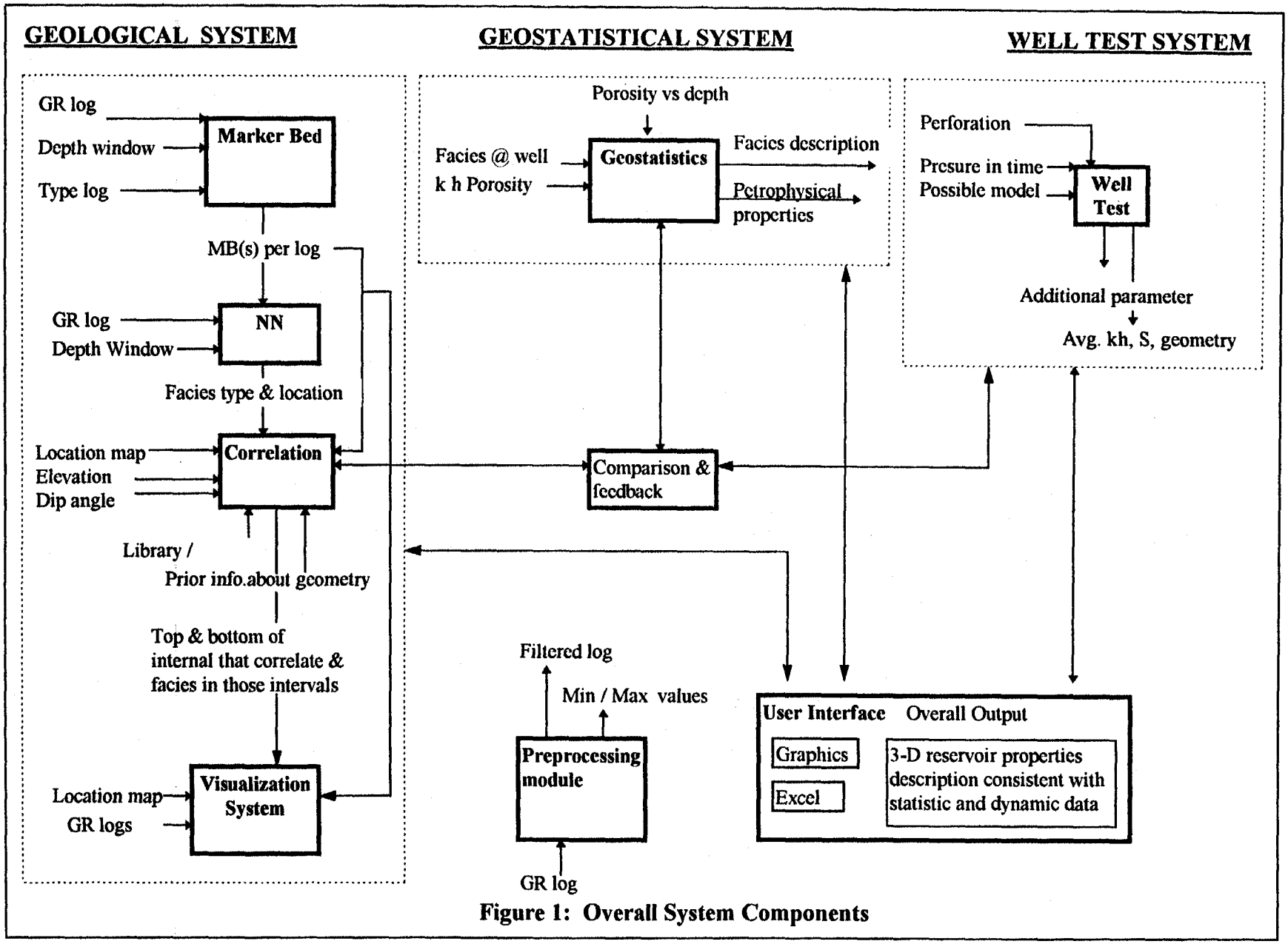


Figure 1: Overall System Components

2. Geostatistical System: Incorporation of Dynamic Constraints in a Reservoir Description Process

2.1 Summary of Progress

During this quarter, the following areas were studied: (1) Upscaling methods, (2) Use of *ECLIPSE* for the simulation part of the simulated annealing (SA) algorithm, (3) Convergence rate factor for the "temperature," (4) Weightings to apply to the components of the SA objective function.

2.2 Upscaling Methods

2.2.1 Modified Geometric Average Upscaling

As previously noted, the conventional upscaling techniques do not perform adequately in matching the pressure behavior between scales. As an example, Figure 1 shows the relative errors (defined below) when a 90x90 grid is upscaled to an 18x18 grid and the flowing BHPs from the fine scale and coarse scale grids are compared.

$$\text{Relative Error} = \frac{\Delta p_{wf_o} - \Delta p_{wf_c}}{\Delta p_{wf_o}} \qquad (1)$$

where Δp_{wf_o} = *fine scale flowing BHP change*

Δp_{wf_c} = *coarse scale flowing BHP change*

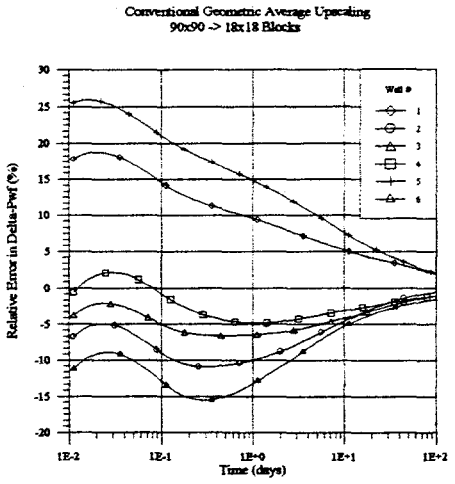


Figure 1

By modifying the upscaling procedure in the near-well region, we can obtain a much better match as Figure 2 shows.

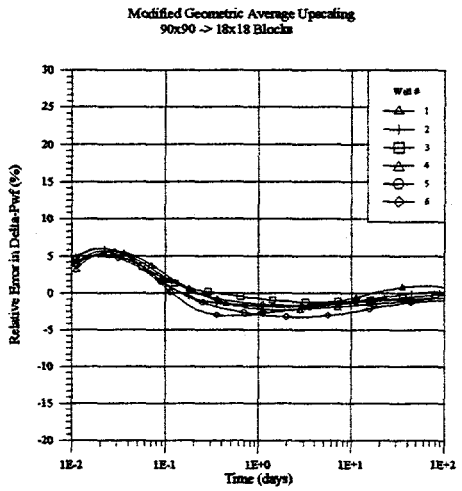


Figure 2

In this case, the relative errors lie between ±5% after 0.1 days. Note that Figures 1 and 2 are plotted on the same scale for purposes of comparison. The methodology for modifying the geometric averaging approach was outlined in a previous report.

It was subsequently observed however, that this modified approach did not give consistent results. As Figure 3 shows, in some cases the errors remained relatively high with this approach.

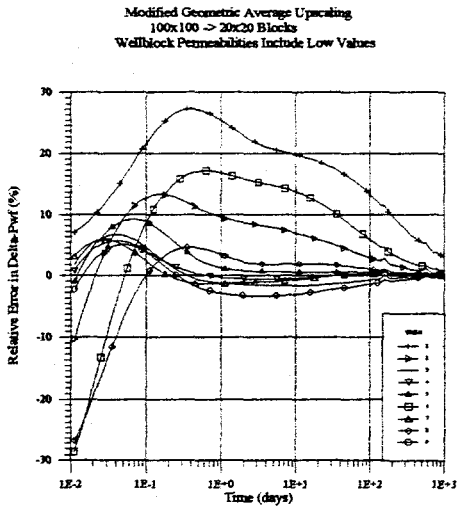


Figure 3

As can be seen however, some of the wells were "well-behaved" while others were not. For example, well 9 displays low errors, but those for well 1 are high. It was determined that the magnitude of the fine scale wellblock permeability determines how good the modified geometric average upscaling approach works. This may be explained by considering that, when the contrast between the wellblock permeability and those of the nearby gridblocks is large, upscaling will be dominated by the permeability of the other

blocks and so skewed away from the wellblock permeability. This will magnify the difference between the pressure responses observed at the fine and coarse scales.

well 1				
8.65	5.63	9.20	16.27	9.84
10.62	1.45	3.86	8.31	10.12
9.76	5.88	1.35	1.13	6.41
12.51	10.53	2.18	9.23	10.92
12.73	8.43	4.64	8.88	11.76

well 9				
42.31	35.22	23.32	24.98	26.26
43.69	39.81	31.56	22.96	22.23
48.46	42.00	30.59	25.90	21.59
58.83	36.66	31.65	18.71	24.28
47.94	44.75	37.20	28.91	14.37

The above figure shows the actual values of the 25 fine scale permeabilities in the near-well gridblocks for wells 1 and 9. The wellblock location and permeability value are denoted by the shaded value at the center. It can be seen that for well 1, the wellblock permeability value is relatively low, when compared to the surrounding values. In the case of well 9 however, this is not so. To verify whether this analysis was valid, another case was flow simulated in which all the fine scale wellblocks were set to permeability values comparable to those of the surrounding gridblocks. Figure 4 shows the results in this case, which supports our analysis.

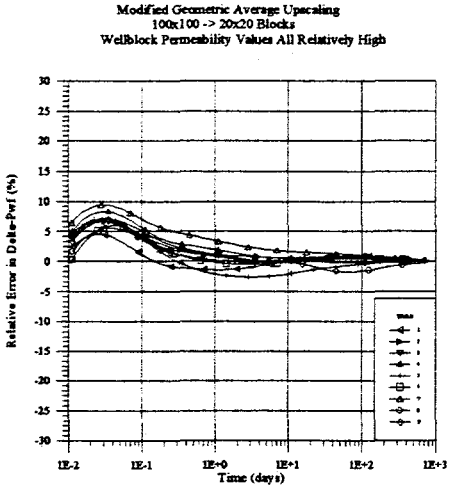


Figure 4

Here again Figures 3 and 4 are plotted on the same scale to facilitate comparison. We may thus conclude that the modified geometric average upscaling technique may be used when the wellblock permeability is not very much lower than the surrounding values.

2.2.2 Upscaling Approach Suggested by Ding[1]

Another procedure for upscaling in the vicinity of wellbores was presented by Yu Ding. In his approach the upscaled parameters consist of the transmissibilities and numerical productivity indices (PIs) around the well(s). He points out that this is a purely numerical problem, for which we need to determine the *equivalent* permeability on coarse grids from known values on a fine grid. This is different from the *effective* permeability defined in accordance with the spatial distribution or the correlation of the fine scale values. Because his procedure requires the numerical solution of a steady-state problem using a fine scale sub-grid around each well, the solution is somewhat more complex and time-consuming than the modified geometric average upscaling method; however, it consistently performs better than that approach. As Figures 5 and 6 below show, the errors in the pressure response are better-behaved (in that the convergence to zero is "smoother") and smaller than those from the modified geometric average upscaling method.

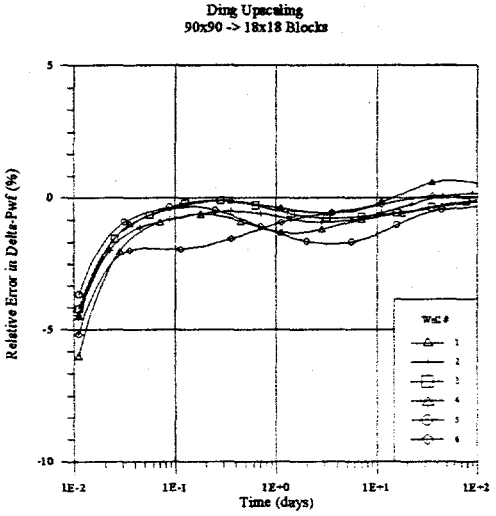


Figure 5

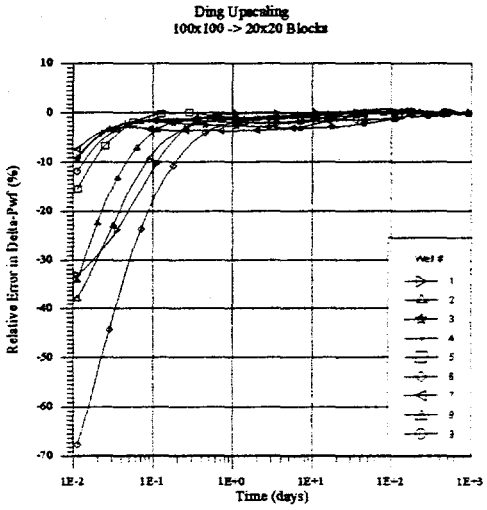


Figure 6

We use the pressure results from about one day for comparison purposes in the SA algorithm, thus both upscaling approaches may be used -- except that in the case of the modified geometric average upscaling method, the permeability contrast in the near-well region cannot be too great.

2.3 Use of *ECLIPSE* for the Simulation Part of the SA Algorithm

During the period under review, we also started to test the usage of the *ECLIPSE* black oil reservoir simulator for performing the flow simulation part of the SA algorithm. This required segmentation of the code into several executables (among which the calls to *ECLIPSE* were imbedded) which were then run under *UNIX* via a batch file. However, an extensive amount of input/output now becomes necessary since we have to read and write to the datafiles between calls from one executable to the next. This has made the algorithm

very inefficient. We are investigating however, procedures which may improve the efficiency of the code, and so make this approach more feasible.

2.4 Convergence Rate Factor for the "Temperature"

The default value being used for the convergence rate factor for the "cooling schedule" (i.e. the multiplicative factor by which the SA temperature is reduced) was 0.5, defined by Perez[2] as the "optimum" value. However, this value was optimum for a variogram-only objective function. As pointed out by Aarts and Korst,[3] a requirement for a finite-time implementation of the SA to result in an approximation of the optimal solution is that "quasi-equilibrium" is attained at each temperature level. Thus there is a "trade-off" between large decrements in the control parameter (temperature) and small homogeneous Markov chain lengths (the Markov chain length represents the number of perturbations or different distributions generated at a particular temperature). We experimented with values of the convergence rate factor larger than 0.5 and found that the fastest convergence was obtained for a factor of 0.1.

2.5 Weightings to Apply to the Components of the SA Objective Function

2.5.1 Sagar's Approach

To date, the default weight used for each component of the objective function was 0.5. Some testing was initiated on determining "appropriate" weights for our problem. We started with the 10,000-block case and, following Sagar[4] we defined the weights as:

$$\Psi_1 = \frac{\sum_{k=1}^{M} \frac{\left|E_2^k - E_2^0\right|}{E_2^0}}{\sum_{k=1}^{M} \frac{\left|E_1^k - E_1^0\right|}{E_1^0} + \sum_{k=1}^{M} \frac{\left|E_2^k - E_2^0\right|}{E_2^0}} \qquad (2a)$$

$$\Psi_2 = \frac{\sum_{k=1}^{M} \frac{\left|E_1^k - E_1^0\right|}{E_1^0}}{\sum_{k=1}^{M} \frac{\left|E_1^k - E_1^0\right|}{E_1^0} + \sum_{k=1}^{M} \frac{\left|E_2^k - E_2^0\right|}{E_2^0}} \qquad (2b)$$

where $E_i^0 = Initial\ Energy\ for\ Component_i$

$E_i^k = Perturbation_k\ Energy\ for\ Component_i$

$M = Number\ of\ Perturbations$

then $$E^k = \frac{\Psi_1}{E_1^0} E_1^k + \frac{\Psi_2}{E_2^0} E_2^k \quad (3)$$

where E^k = *Overall Objective Function*

Whereas Sagar found that 0.06 cycles of perturbations was adequate for obtaining stable weights, we found that more than 12 cycles were required for our case. Also the stable weights obtained from our analysis varied between 0.55-0.65 for weight1 and correspondingly 0.45-0.35 for weight2.

2.5.2 Deutsch's Approach[5]

In this approach, the weights are determined as:

$$\Psi_i = \frac{1}{\overline{|\Delta E_i|}} = \frac{1}{\frac{1}{M}\sum_{k=1}^{M} |E_i^k - E_i^0|} \quad (4)$$

then $$E^k = \frac{1}{E^0} \sum_{i=1}^{N} \Psi_i E_i^k \quad (5)$$

where $$E^0 = \sum_{i=1}^{N} \Psi_i E_i^0 \quad (6)$$

Here N is the number of components of the objective function and M is a large number of independent perturbations, say 1,000. Analysis of this approach is continuing.

3. Integrated Lithofacies and Petrophysical Properties Simulation

In this section we present the new procedure developed to generate reservoir characterization by simultaneously simulating the lithofacies and petrophysical properties, i.e., porosity and permeability. The technique used is the conditional simulation method which is capable of honoring the original distribution of the data and the associated spatial relationship. The method is capable of predicting several equiprobable images of the reservoir. The procedure used in simulating the lithofacies is the indicator simulation whereas the porosity is simulated using sequential Gaussian simulation. The permeability distribution is simulated using conditional distribution technique.

The program is developed using the C++ language. We intend to incorporate some pre-and-post processing tools in the program to help the user in applying the program. At present time, important classes required for this program have been completed. The future work is to develop the main driver and to test the program with the available data.

3.1 Conditional Simulation Methods

Conditional simulation is a geostatistical method to generate description of reservoir properties which uses the available quantitative and qualitative data. This method is a stochastic approach because reservoir properties are represented by random variables. The description of the properties generated by this method are conditional since the available data are honored at the sampled locations. And, the method simulates several equiprobable descriptions of the actual distribution of a property in the reservoir. In constructing the possible reservoir descriptions, the constraints imposed on the simulation process may include prior distribution of the simulated variables, spatial relationships in various directions and geometry of geological shapes and sizes. As more constraints are incorporated in a conditional simulation process, more similar would be the equiprobable images.

3.2 Co-Simulation of Rock type and Petrophysical Properties

Common practice in generating reservoir description in the industry is the two stage approach where at the first stage the rock type or the geological facies is simulated followed by the simulation of the petrophysical properties at the second stage. The process at the second stage requires a lot of computation time and computer storage to hold the temporary results which will be discarded after combining with the results of the first stage through filtering process. Therefore, if we can combine these two processes in one, an efficient simulation will be obtained.

To eliminate the two stage approach and to reduce the computation time, the grid block is visited only once. Using the same search neighborhood, the geological facies is estimated first, followed by porosity and permeability. The method accounts for correlations among these variables as well as the spatial relationships. This reduces the storage requirements and makes the process computationally efficient while maintaining the consistency between the generated petrophysical properties with the underlying geology.

A co-simulation program to eliminate the two stage processes described in the previous paragraph is being developed using the C++ language. The program is the translation and modification of the original COSIM program which was written in Fortran. In the Fortran version, both facies and porosity simulation are conducted using the sequential Gaussian technique. It is believed that an indicator variable such as lithofacies will better be simulated using indicator simulation rather than a Gaussian technique which will be more suitable for continuous variables such as porosity. In addition, a modification is also being made to account for the uncertainty of the data.

During this quarter, the creation of the required classes has been completed. Table-1 presents the description of each class. The main driver of this program is now being developed. Upon completion, this program will be tested using the available sandstone data of Glenn Pool field and carbonate data of North Robertson Unit - Texas. The future

work will also include the creation of the pre-and-post processing tools to make this program user friendly.

Class Name	Description
Grid	Provide the grid block network of the simulation system that includes the neighborhood searching technique such as super block search.
Variogram	Provide the calculation of variogram and/or covariance value between any two points in 3D for a given Variogram model.
CovTab	Provide the calculation and storage for the covariance table. This class is inherited from Variogram Class.
Kriging	Provide the procedure to estimate the node value either by Simple kriging or Ordinary kriging technique.
CondDist	Provide the procedure to perform the conditional distribution technique in generating the permeability distribution and the storage of the related correlation among variables
Simulation	Provide the procedure to perform the simulation either Gaussian or Indicator techniques.
Point3D	Provide the structure to represents a 3D point. This class is generated using a template that can accept any data-type.
Utility	Provide several utility functions that are common in geostatistical simulation technique such as random number generator, inverse of Gaussian data, normal transformation, etc.
listClass	Provide the link list of the data to store variable with unknown size. This class is also templated to accept any data-type.
Application	Provide the main driver of the program.

Table 1 **C++ Class Description of the COSIM program.**

4. Geological System: Sand Body Identification

In order to analyze well log data, we solve the following two problems sequentially:

- *Well log segmentation problem*
- *Log facies identification problem*

Well log segmentation. Given a well log data file the system determines the endpoints, called *cuts*, of every sand body present in the log file. This is needed to divide the well log (gamma ray) into discrete stratigraphic units. Such segmentation is for log facies identification and well-to-well correlation. A rule-based system is applied to the original

data file to determine the cuts or segments. The resulting file is then fed to the neural network to solving the log facies identification problem.

Log facies identification. Given a well log data file and the predetermined cuts, the system determines which kind of facie or sand body is between any two cuts. A neural network is used to solve this problem. The input to the network is an intermediate file generated by the rule-based system.

Our neural network was previously trained with expert-classified well logs to recognize the following set of fundamental shapes:

bell, funnel, blocky, symmetrical, linear

?.? Well Log Segmentation

Well logs have to be scaled and normalized in order to set a common ground on which the problem can be solved. In consequence every log file is scaled in such a way that:

- *maximum gamma ray value maps to 1*
- *minimum gamma ray value maps to 0*

As a result of this process all the gamma ray values will be within this range (0-1). This is done before attempting to solve either the log segmentation or the facies identification problems.

3.3.1 Applying Well Log Segmentation Rules

Figure 7 shows how this rules are applied to a section of a log:

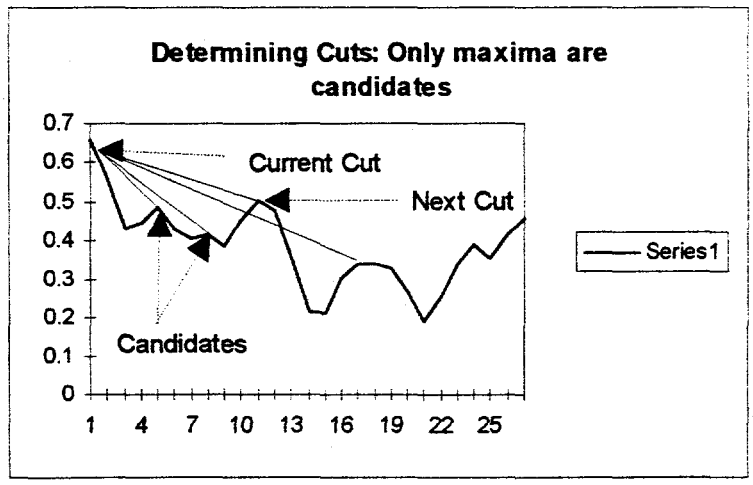


Figure 7

3.3.2 Segmentation Results

Currently, the neural network has problems recognizing the cuts between facies. It only recognizes about 70 percent of the actual cuts. We believe that this recognition problem is due to the neural network using the high frequency information improperly. As a result, the facies identification rate is lower than expected. Thus, it was necessary to create a new approach to improve the percentage of correct facies recognition. This approach consists in a digital filter that cuts off the high frequency.

The digital filtering process can be represented by the block diagram in Figure 8.

Figure 8

where $x(nT)$ is the unfiltered data or the excitation and $y(nT)$ is the filtered data or the response of the filter. The response is related to the excitation by some rule of correspondence. This fact can be indicated notationally as

$$y(nT) = Rx(nT)$$

where R is an operator.

The type of filter used to filter the well log is *time-invariant*, *linear* and *nonrecursive*. *Time-invariant* means that the operator R does not depend on the time of the application of the excitation. *Linear* means that R satisfies the next conditions:

$$R\alpha x(nT) = \alpha Rx(nT)$$
$$R[x_1(nT) + x_2(nT)] = Rx_1(nT) + Rx_2(nT)$$

for all possible values of α and all possible excitations $x_1(nT)$ *and* $x_2(nT)$. *Nonrecursive* means that the response to the filter at instant nT is the form

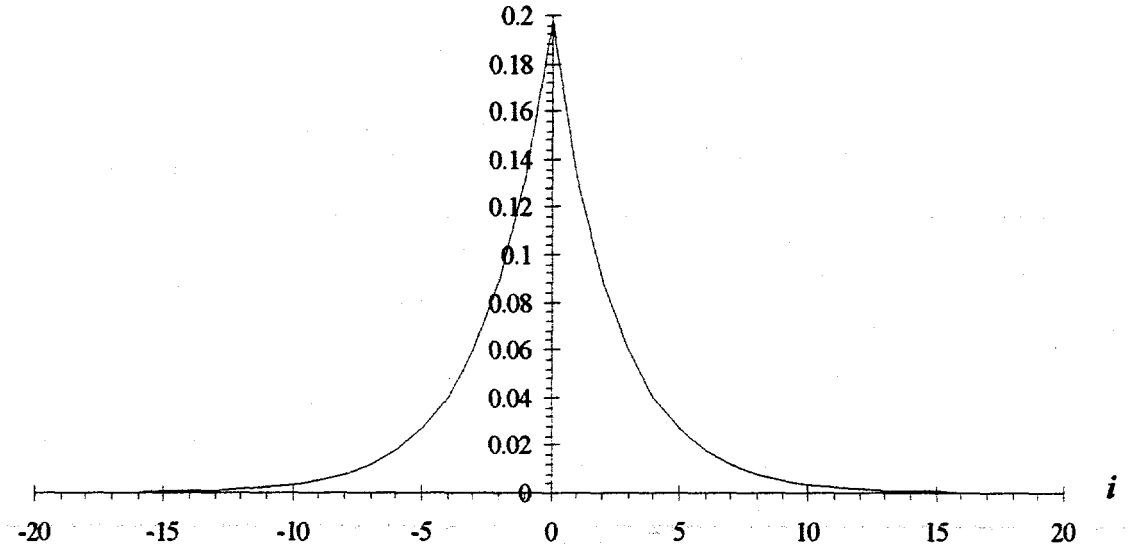
$$y(nT) = f\{\ldots, x(nT-2T), x(nT-T), x(nT), x(nT+T), x(nT+2T), \ldots\}$$

Because the filter used is *linear* and *time-invariant*, $y(nT)$ can be expressed as

$$y(nT) = \sum_{i=-\infty}^{\infty} a_i x(nT - iT)$$

where a_i represents constants.

The a used to filter the well logs is the following:



The main characteristics of this are the sum of all a_i is equal to 1, the maximum is at $i = 0$ and it is symmetric respect to the maximum. So that, the gain of the filtering process is 1 and there is not phase delay between the unfiltered and filtered well logs. The result of the filtering process is shown in Figure 9.

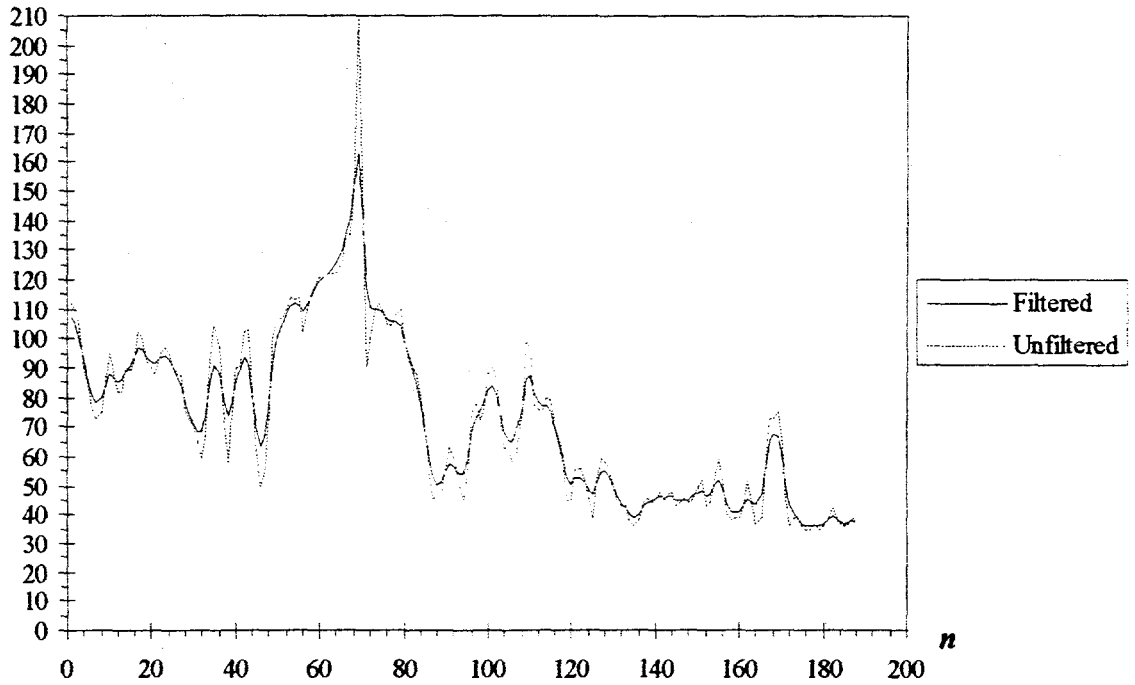


Figure 9

We expect improvement from using this low-pass filter before to using the neural network to find the cuts because it filters out the high frequency component that normally

is noise. Therefore, the probability of error finding correct cuts between facies is lower. This means the probability of recognizing the type of facies is higher.

There are several steps we need to perform to determine the extent of improvemen the filtering process has on the percentage of correctly identified facies. First, we must retrain the neural network using filtered and unfiltered well logs and evaluate the results. Next, we need to create an additional pre-processing module that performs a dynamic normalization of well logs before or after to use the filter. The reason for creating this new module is because in some well logs the information has less magnitude.

5. Geological System Components: Correlation of Log Curves

In this section, we detail the current approach to the correlation of log curves. Implementation is underway of an initial rule set to generate a matrix of compatibilities of zones in wells. Expert rules are being developed to analyze the matrix to determine the appropriate correlation of the zones.

5.1 Overview of Approach

The approach has been to develop a set of rules for correlation of two log curves. The rules are based on similarities in well log trace shapes, thickness and vertical position of the zones. The segmentation of the well logs and log-facies identification by the neural network and depths of identified marker beds will be given as input.

With this approach, initially for correlating two zones from two different wells the following four criteria have been chosen:

1. position of the zones with respect to the length of the logs;
2. distance of the zones from the marker beds;
3. thickness of the zones;
4. log-facies of the zones;

Considering the basic rules for each of these criteria as described in our last Annual Report, we formulated a comprehensive rule set of 54 rules which were also presented in our 1995 Annual Report. With these rules, correlation between two synthetic wells and two wells from Glenn Pool field (Self 81 and Self 82) was tested. These two Glenn Pool logs with identified zones and corresponding log facies and a tentative correlation between the two wells have been presented as Attachment A. The logs were manually zoned and the log facies of each zone was identified which were used as input for testing the correlation rules. The "correlation rank matrix" derived using the two Glenn Pool logs is presented as Table 2. As can be seen from Table 2, the correlation of zones a1, a9 of well Self 82 with zones b1 and b7 of well Self 81 respectively has been ranked higher than any other possible correlation combination and hence can be correlated. With some additional

heuristics about how these ranks need to be interpreted, we feel that the correlation of other remaining zones can be done with reasonable confidence.

Correlation Rank Matrix								
	Self 81							
	Zone	b1	b2	b3	b4	b5	b6	b7
	a1	54	21	27	27	6	3	3
	a2	52	47	51	27	6	3	1
	a3	48	47	51	51	6	3	3
	a4	27	39	54	54	47	6	4
	a5	6	6	47	47	47	39	4
	a6	4	27	9	47	47	39	4
	a7	6	27	9	47	47	39	4
	a8	6	6	9	9	54	51	49
	a9	3	3	6	6	27	48	54
	a10	3	2	6	6	6	9	9

Table 2

While carrying out these tests with different test data, both synthetic and real, we noted that the comparison of log facies of two zones has been grouped into two categories: 'same' and 'different'. The logfacies are identified as belonging to one of the five categories: Bell(1), Funnel(2), Blocky(3), Symmetrical(4), Linear(5). We recognize that these logfacies represent change in depositional evvironments within a fluvial setting. Hence while comparing two different logfacies we need to rank the 'difference' beteween them instead of grouping all 'different' comparisions into one category. This will allow to incorporate lateral facies change in our correlation. When the above concept is incorporated to formulate a comprehensive rule set, we formed a rule set with 209 rules. The above two wells (Self 81 and Self 82) and two other wells (11-75 and 11-86) were tested using the expanded rule set.

5.2 Implementation

The rules developed for the correlation are being prototyped in CLIPS to allow for fast development and flexible changes prior to porting to C++. CLIPS (C Language Integrated Production System) is an expert system tool that is designed to facilitate the development of software to model human knowledge or expertise. There are three ways to represent knowledge in CLIPS:

- Rules which are intended for heuristic knowledge based on experience,
- Functions which are intended for procedural knowledge and
- Object-oriented programming.

CLIPS has a design that allows for full integration with other languages such as C. In addition to being used as a stand-alone tool, CLIPS can be called from a procedural language, perform its function, and then return control back to the calling program. The

CLIPS shell (which performs inferences and reasoning) provides the basic elements of an expert system.

Fact-list and instance-list : Global memory for data

Knowledge-base: Contains all the rules.

Inference engine: Controls overall execution of rules.

CLIPS also provides a good user interface. Prototyping the rules in CLIPS is useful because we get quick feedback. Also, the development of the rules and changes to them can be performed easily in this framework. It is in keeping with the above features that the well correlation module was decided to be implemented first in CLIPS

An example of a rule in CLIPS

```
(defrule rank176
        (zones ?z1)
        (zones ?z2)
        (zones ?z3)
        (log1 ?l1)
        (log2 ?l2)
        (test (and (eq ?z1 diff) (eq ?z2 same) (eq ?z3 close)))
        (test (and (= ?l1 2) (= ?l2 4)))
=>
        (assert (rank 176)))
```

The above rule gives the rank 176 if the zones have same thickness, if their respective distances from the marker beds are close, and the log facies are different and they are of type 2 and 4. The rules first classify the zones being compared based on their distance from the marker bed, thickness and log facies as indicated in the previous reports. Then a set of rules similar to the one above are implemented to get a rank.

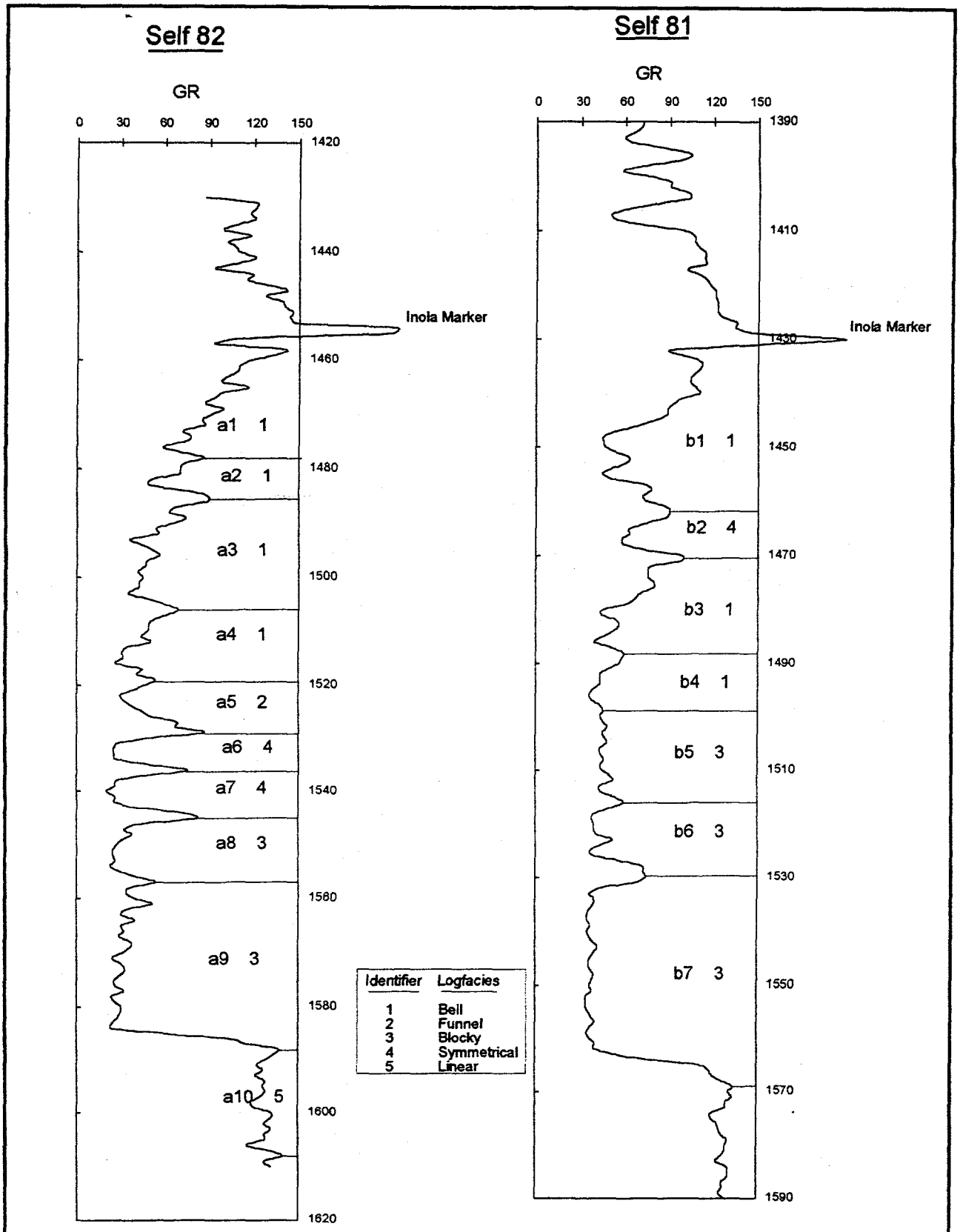
5.3 Future Work:

The correlation rank matrix resulting from the use of the expanded rule set will be analysed to look for any improvement in correlation. We have felt that the 'Relative Position' criterion may seem to be redundant in view of the criterion 'Distance from Marker Bed'. We may have to delete this criterion and reformulate the rules accordingly. More testing will be done to evaluate the results.

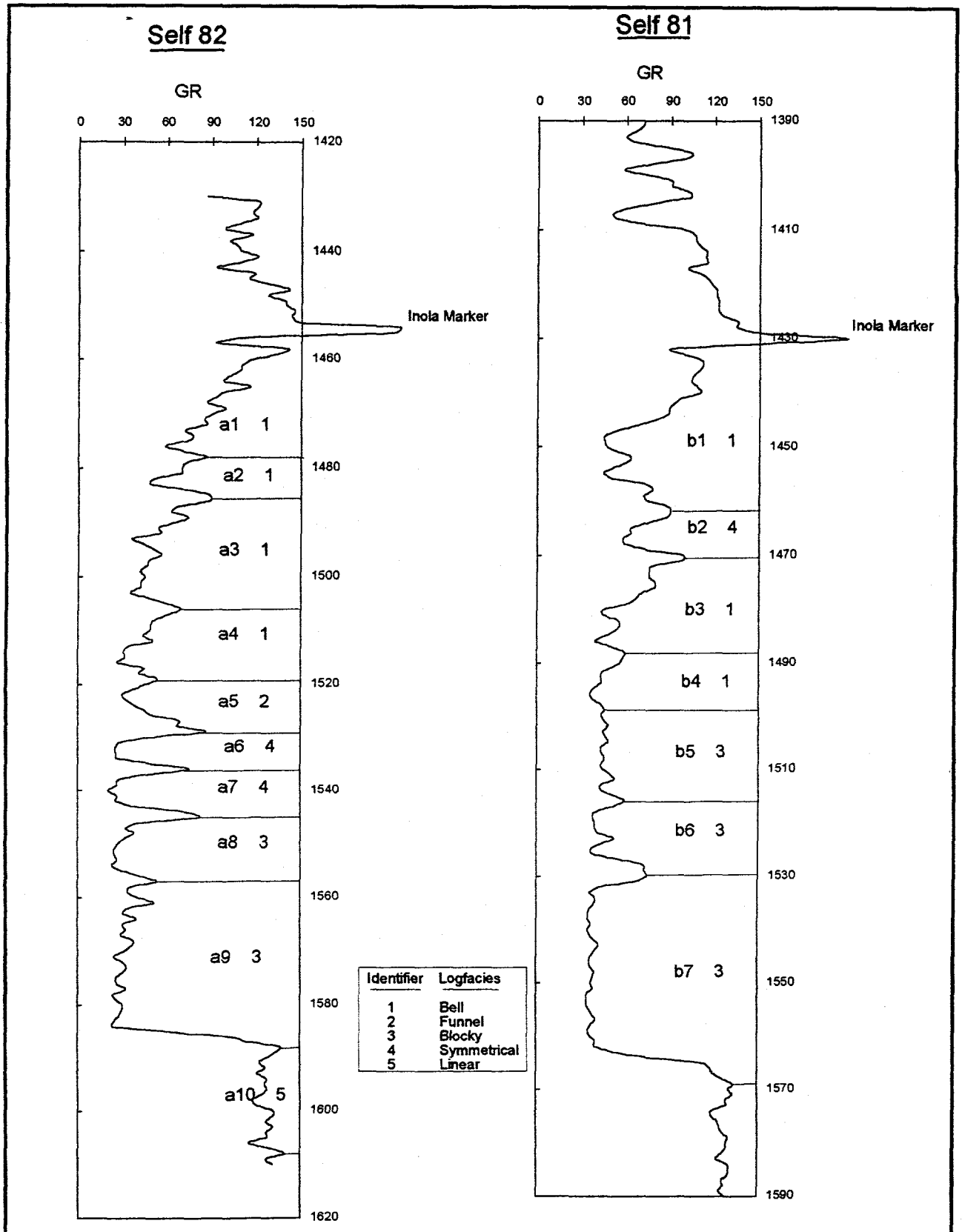
References

1. Ding, Yu: "Scaling-up in the Vicinity of Wells in Heterogeneous Field", SPE 29137 presented at the 13th SPE Symposium on Reservoir Simulation held San Antonio, TX, U.S.A., 12-15 February 1995.
2. Perez, Godofredo: "Stochastic Conditional Simulation for Description of Reservoir Properties", Ph.D. Dissertation, The University of Tulsa, Tulsa, OK (1991).
3. Aarts, Emile and Korst, Jan: *Simulated Annealing and Boltzmann Machines A Stochastic Approach to Combinatorial Optimization and Neural Computing*, John Wiley and Sons, New York (1989).
4. Sagar, R.K.: "Reservoir Description of Well Test Data and Spatial Statistics", Ph.D. Dissertation, The University of Tulsa, Tulsa, OK (1993).
5. Deutsch, C.V. and Cockerham, P.W.: "Practical Considerations in the Application of Simulated Annealing to Stochastic Simulation", Mathematical Geology, Vol. 26, No. 1, 1994.

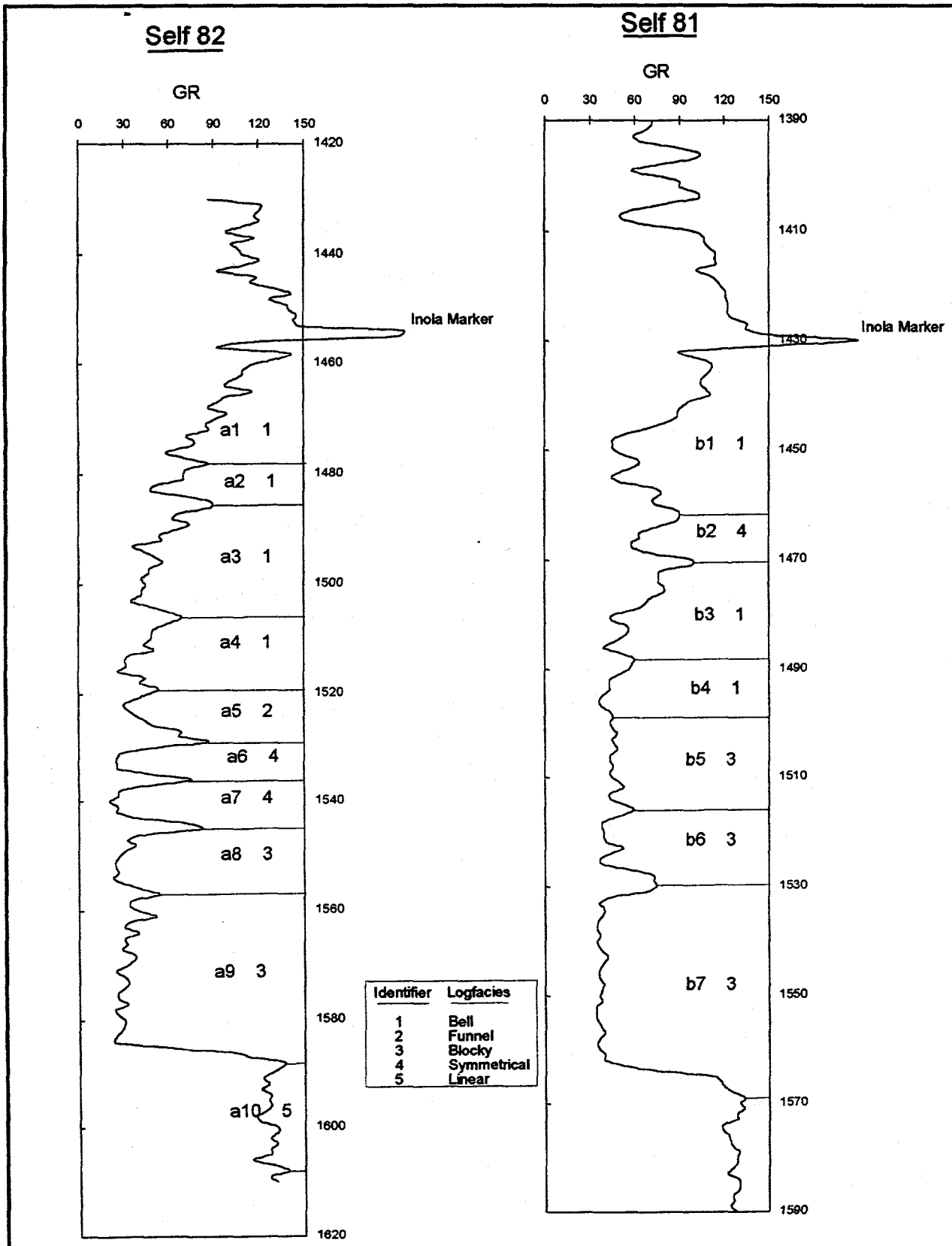
Attachment A



Gamma ray logs from the Self Nos. 81 and 82 wells used for testing the correlation module. Each stratigraphic interval is indicated by a well designator (Self 82 = a; Self 81 = b) and a level designator progressing downward from the Inola marker (1to 10, in this case).



Gamma ray logs from the Self Nos. 81 and 82 wells used for testing the correlation module. Each stratigraphic interval is indicated by a well designator (Self 82 = a; Self 81 = b) and a level designator progressing downward from the Inola marker (1to 10, in this case).



Gamma ray logs from the Self Nos. 81 and 82 wells used for testing the correlation module. Each stratigraphic interval is indicated by a well designator (Self 82 = a; Self 81 = b) and a level designator progressing downward from the Inola marker (1to 10, in this case).