The Thesis Committee for Travis Payton Hampton
Certifies that this is the approved version of the following thesis:

Seismic and Sparse Data Integration through the use of Direct Sampling

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor:

Sanjay Srinivasan

Leslie Wood

Seismic and Sparse Data Integration through the use of Direct Sampling

by

Travis Payton Hampton, B.S.M.E.

Thesis

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Engineering

The University of Texas at Austin

December 2013

Dedication

To my family and Lulu.

Acknowledgements

First, I would like to thank my supervisor, Dr. Srinivasan. He has provided me with guidance from my first interest in petroleum engineering to the publication of this thesis. His advice and support have been invaluable. I will always have a deep appreciation for his role in my life over the past three years. Additional thanks goes to Dr. Spike who assisted in my understanding of seismology which proved instrumental during the final stage of my research.

Additional thanks goes to all of my friends and colleagues in our research group. They have all had an impact on my time in this department. In particular, I would like to thank Aarti, Brandon, DJ, Harpreet, Hoon, Krupa, Kwang Jin, James, Mayuri, Morteza, Nnamdi, and Selin.

Abstract

Seismic and Sparse Data Integration through the use of Direct Sampling

Travis Payton Hampton, M.S.E.

The University of Texas at Austin, 2013

Supervisor: Sanjay Srinivasan

The integration of seismic attributes and well data is an important step in the development of reservoir models. These models draw upon large data sets including information from well logs, production history, seismic interpretation, and depositional models. Modern integration techniques use the extensive data sets to develop precise models using complex workflows at increased cost of time and computational power. However, a gap exists in which a geostatistically driven procedure could integrate pattern statistics inferred from seismic images and those integrated from analogous geologic systems in order to develop spatially accurate reservoir models.

Direct Sampling Seismic Integration Process, DSSIP, was first proposed by Henke and Srinivasan (2010) as an alternative to traditional seismic integration methods. The process provides a probabilistic mapping tool for fast reservoir analysis based on sparse conditioning data in a target reservoir and fully interpreted data from an analog field. DSSIP combines the structural information present in seismic data and facies patterns present in a training reservoir to create a fully realized output map for the target field. In this work, the basic DSSIP algorithm has been further optimized by performing a detailed parameter sensitivity study. The basic DSSIP algorithm has been demonstrated for a real field data set for a deepwater Gulf of Mexico reservoir. The basic DSSIP algorithm has also been analyzed to understand and model the effects of features such as salt canopy that can blur the seismic image. Finally, a modification to the basic algorithm is also presented that uses only a training model and the seismic data for the target reservoir in order to generate reservoir models for the target reservoir. This procedure eliminates the requirement to have a matching pair of training data sets for both the facies distribution and the corresponding seismic response.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

The integration of seismic attributes and well data is an important step in the development of reservoir models. These models draw upon large data sets including information from well logs, production history, seismic interpretation, and depositional models. Modern integration techniques use the extensive data sets to develop precise models using complex workflows at increased cost of time and computational power. However, a gap exists in which a geostatistically driven procedure could integrate pattern statistics inferred from seismic images and those integrated from analogous geologic systems in order to develop spatially accurate reservoir models.

Direct Sampling Seismic Integration Process, DSSIP, was first proposed by Henke and Srinivasan (2010) as an alternative to traditional seismic integration methods. The process provides a probabilistic mapping tool for fast reservoir analysis based on sparse conditioning data in a target reservoir and fully interpreted data from an analog field. DSSIP combines the structural information present in seismic data and facies patterns present in a training reservoir to create a fully realized output map for the target field.

To perform the base spatial recognition processes, the method adapts the Direct Sampling algorithm as developed by Mariethoz et al. (2010). In the first step of DSSIP, unknown facies locations in the target reservoir are filled using patterns developed by comparing seismic attribute maps from the training and target reservoirs. The first step captures the large structural patterns that can be extracted from the low resolution seismic data sets. The second step retrieves the high resolution facies relationships from the analog reservoir model and places them into the appropriate location in the target field map based on information gathered during the first step.

The following thesis presents a study of DSSIP and modifications that have to be performed to the base algorithm in order to implement the process on realistic data sets. Several data sets were used to advance the understanding of the parameters, data preparation, and importance of data quality to the results obtained by application of the method. A major advance reported in this thesis is the ability to model a reservoir using a single training reservoir model, target seismic information for a target reservoir and field specific conditioning data, rather than a matching pair of training reservoir model and seismic data that is required in the base algorithm to calibrate the relationship between the two. The base algorithm and the modifications are demonstrated on several cases – both synthetic as well as some based on actual field data.

This report is divided into eight chapters. Chapter 1 introduces the motivation for DSSIP. Chapter 2 provides a background on the processes and background literature that the algorithm draws upon. Chapter 3 states the order of operations for the algorithm and illustrates this using a basic synthetic case study. Chapter 4 is an optimization study for DSSIP and recommends specific parameter values. Chapter 5 applies DSSIP to Lobster Field and introduces possible modifications to the algorithm. Chapter 6 demonstrates the data preparation process using information from provided internal company reports. Chapter 7 examines the impact of low resolution or mixed resolution seimic data on the model characteristics. Chapter 8 concludes the report and summarizes the ideas explored in this thesis.

Chapter 2: Literature Review

Two focus areas of research have strongly contributed to the development of DSSIP, seismic integration and multiple-point statistics. Seismic integration techniques focus on the linking together of large seismic data sets and sparse geologic information to develop comprehensive reservoir models. Multi-point statistics based algorithms utilize patterns of reservoir objects described in terms of joint variability on a spatial template in order to develop complex reservoir relationships.

2.1 SEISMIC INTEGRATION TECHNIQUES

Current seismic integration techniques focus on developing extensive workflows that utilize large amounts of qualitative geologic information to guide facies interpretation (Lindseth, 1979, Riddiford and Goupillot, 1994, Narhari et al., 2008). Other methods use well and production data to constrain the seismic inversion process (King et al., 1993, Russell et al., 2001, Hampson et al., 2001, Andersen et al., 2006). These methods develop accurate reservoir models at the cost of high computation times. However, several authors have shown the added value of faster model creation using statistical techniques to evaluate the geologic information. One such set of methods use variograms and probability densities to capture spatial variance (Silva et al., 1998, Yang et al., 1995).

Calabrese et al. (2011) created probability maps using relationships between known lithology at the well locations and their relationships to inverted seismic attributes. The process is a supervised Bayesian classification using a set of conditional probability information for each lithofacies and seismic attribute pair. The probability of a selected facies (A) at an unknown location with a known seismic attribute (B) is then calculated using the probability functions developed at the known locations, yielding the

P(A|B) at the unknown location. Specific reservoir models were then created through the qualitative evaluation of the probability maps.

Al-Anezi et al. (2013) provided a detailed workflow on the statistical integration of a 3D seismic information and a facies model derived from well logs and depositional models. The core of the process is the truncated pluriguassian algorithm inspired by work done by Matheron et al. (1987). The algorithm truncates multiple Gaussian simulations and combines them to create categorical maps (Mariethoz et al., 2008). The workflow developed by Al-Anezi calculates facies proportions using data from seismic inversion and then simulates the location of facies based on variograms and conditioning data locations.

Quinto et al. (2013) developed a quantitative approach to integrate basin-scale geologic knowledge with locally known reservoir specific data in order to create a final static model of the reservoir. In addition to the traditional data - seismic and well data, Quinto also proposed a novel data integration scheme that takes into account the sequence stratigraphy information. The technique uses a method similar to that of Al-Anezi et al. in order to calculate a final output map using calibrated probability density functions. The locations of structures in the static model are not determined. Instead, the final volume consists of proportion of a cell volume occupied by channels. A notable aspect of the approach is a reliance on knowledge of several geologic parameters such as topography and sediment source.

The disadvantage of these papers is that variograms and PDFs fail to capture the multi-directional nature of anisotropy in reservoirs. They can only accurately recreate structures such as channels and fault systems. Multi-point statistical models recall patterns from specified training maps (analog reservoirs) and can better match geologic heterogeneities.

Caers (2002) utilized a multi-point statistical approach to develop reservoir models based on a set of production data and a geologic training image. His technique utilized the *snesim* algorithm to retrieve structural patterns from the training map. The method has been subsequently demonstrated for seismic integration case studies (Castro et al., 2006, Caers et al., 2006). The studies combine probability maps created during seismic inversion (Andersen et al., 2006) to develop an ensemble of facies models using *snesim*. The final reservoir model is chosen through history matching to prior production data.

Despite the inclusion of geostatistical procedures, the techniques discussed are still reliant on large amounts of qualitative interpretation and rely on workflows rather than algorithms. They also do not develop the reservoir model through the simultaneous comparison seismic and field data patterns. The DSSIP method discussed in this thesis provides a user with a powerful modelling tool given limited information about the target reservoir. The process differentiates itself through the use of pattern recognition in multi-point statistics without the requirement of extensive analysis and the building of reservoir models through simultaneous integration of seismic information.

2.2 MULTIPLE-POINT STATISTICAL MODELING

Multiple-point statistics addresses the spatial uncertainty present in geologic model development. As previously mentioned, the processes create reservoir models for an unstudied field from a set of conditioning data and a training image. Over the past two decades, several multi-point statistical methods have been developed for application in the geosciences (Hu and Chugunova, 2008).

Kriging is one of the earliest of the geostatistics methods, developed by Matheron (1976). The technique interpolates a sparse data map based on the proximity and value of neighboring locations. The following equation shows the linear predictor used to estimate the value of the random variable *Z* at an unknown location s_o using conditioning data at local known points $Z = (z(s_1), z(s_2), \ldots , z(s_n))$. The weights, λ_i, are not assigned as in typical distance driven estimations, rather they are based on a semivariogram.

$$\hat{z}(s_o) = \sum_{i=1}^{n} \lambda_i z(s_i)$$

Semivariograms are a function describing the spatial variance across a given data set. For a given lag or distance between two points h, the variance is calculated across the map at all points a distance of *h* apart and returned to the semivariogram (Bohling, 2005).

$$\gamma(h) = \frac{1}{2n(h)} \sum_{i=1}^{n(h)} [z(s_i + h) - z(s_i)]^2$$

The number of pairs separated by the lag h are designated as *n(h)*. The weights are then assigned such that they minimize the variance of the error of estimation as seen below:

$$\sigma_E^2 = Var\{\hat{z}(s) - z(s)\}$$

$$E\{\hat{z}(s) - z(s)\} = 0$$

The second condition ensures unbiasedness of the estimator.

If the random field value *z(s)* is broken down into a residual *R(s)* and trend component *m(s)* such that:

$$Z(s_o) = m(s_o) + R(s_o) \text{ and}$$

$$E\{R(s)\} = 0\ ,$$

$$Cov\{R(s), R(s+h)\} = E\{R(s) \cdot R(s+h)\} = C_R(h)$$

Then the kriged value $\hat{z}(s_o)$ can be identified with the mean $m(s_o)$ and to this a stochastic residue that has a mean 0 can be added. *The* covariance $C_R(h)$ is the residual covariance function derived from the semivariogram model calculated earlier.

$$C_R(h) = Sill - \gamma(h)$$

This becomes the basis for stochastic simulation. Secondary data such as seismic can be added to this procedure by extending the kriging expression to include data of other type. Co-kriging introduces the idea that instead of only one variable Z_1 multiple variables can be used $Z = (Z_1, \dots, Z_k)$ such that the estimator is now:

$$\hat{z}(s_o) = \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_{ij} z_j(s_i)$$

Semivariograms are still utilized to determine spatial similarity and the weights λ_{ij}.

Indicator kriging modifies the original algorithm to utilize a set of binary data which may be formed by setting a threshold, in the equation below, on the original data used in kriging (Journel, 1983). A set of semivariograms corresponding to different thresholds will be used.

$$I(z \leq q) = \begin{cases} 1, if\ z \leq q \\ 0, if\ z > q \end{cases}$$

Sequential indicator simulation (*sisim*) builds upon the structure of indicator kriging and introduces an algorithm for direct estimation at each location without the need of iteration (Guardiano and Srivastava, 1993). The process begins by defining a path through all unknown locations. Then, neighboring known data points are recovered at the next path location. Indicator kriging is performed at the given location to generate a conditional probability density function (cpdf) for the variable. A value is drawn from the cpdf and returned to the unknown location in the estimation path. The process is repeated at each location in the path until the sparse map is filled (Mu et al., 2011). The variable is discretized in the case of a continuous value. Indicator kriging and simulation does away with the Gaussian restriction of traditional kriging and stochastic simulation as described earlier.

The main disadvantage with the listed kriging processes and *sisim* is that they still utilize semivariograms. Variograms limit the functions to measures of linear similarity or similarity between two points. The following algorithms address this limitation and create measures of similitude using the patterns seen in the conditioning data *Z*.

Strebelle (2002) proposed an algorithm that utilized a training image to develop patterns without the need for variogram calculation, *snesim*. Single normal equation simulation (*snesim*) begins by moving a template over each pixel of the training image sequentially and storing the pattern of variability at each location within the template, resulting in a database of all of the pattern configurations for a given template size. The simulation then proceeds sequentially over the un-informed nodes of a simulation grid by acquiring conditioning data from the sparse grid, finding an exact match from the database, and designating the matched pattern to the simulation node. The creation of the database is intended to provide rapid recall of a pattern during the simulation process.

The major limitations of *snesim* are the inability to reproduce large image-scale relationships and the inability to process continuous data. In addition, the simulation procedure requires substantial computer storage for complex training images. Pattern reproduction is limited to the size of the template, and increasing the size of the template greatly increases computing costs. Efforts to streamline the data storage in *snesim* and to speed up the pattern recall process are currently underway (Straubahr, 2012).

Zhang et al. (2006) introduced a multi-point algorithm that operated similar to *snesim*, but pasted entire patterns from the training image into the sparse map. Rather than simulating one grid location at a time, *filtersim* scores patterns from the training image, places the pattern into a binned database, retrieves the pattern for each set of conditioning data during simulation, and places the entire pattern onto the simulation grid.

In this work a still newer algorithm for multiple point statistics called direct sampling (Mariethoz, 2010) is used for simulating complex patterns in the reservoir. This method is similar to the *snesim* algorithm described earlier, however, instead of scanning and saving the pattern statistics information and then retrieving those statistics during simulation, in direct sampling, the training image is directly scanned during simulation and the first match to the conditioning data pattern observed in the training image is exported back to the simulation grid. Details of this algorithm and its extension to reflect patterns exhibited by secondary data such as seismic is discussed in the next and subsequent chapters.

Chapter 3: Direct Sampling of Spatial Patterns and Seismic Integration

3.1 Direct Sampling

The Direct Sampling (DS) method was proposed by Gregoire Mariethoz (2010) as a multi-point geostatistical process that populates a sparse data field using patterns seen in a fully realized training image conditioned to sparse field data. The method differentiates itself from other geostatistical processes due to its very minimal use of computer memory storage and its application to continuous variables.

3.1.1 Development and Background of Direct Sampling

Direct sampling was initially developed to model geologic heterogeneity in hydrogeology and, therefore, shared inspiration from multi-point geostatistical methods seen in Chapter 2, such as *snesim* and *filtersim*. The direct sampling process differentiates itself from these methods by skipping the entire process of scanning and storing patterns observed on a training image. It instead simulates the outcome at the simulation node by directly finding the first match in the training image, rather than by scanning a database of prior patterns. The idea is grounded in the early work of Claude Shannon, more specifically his work on replicating English text using a Markov chain (Shannon, 1948). In *A Mathematical Theory of Communication*, Shannon demonstrated the ability to mathematically approximate English word and sentence structure using probabilities based on prior characters in a string.

The letter approximations used 27 characters (26 alphabetic letters and a space) and were categorized according to dependence on prior characters. Zero-order letter approximations estimated each character independently and with equal weight. First-order letter approximations again estimated each character independently but with a probability equal to the frequency of appearance in the English language. Second- and third-order letter approximations were calculated using the probability of a current character based on the prior two/three letters. The probability was not calculated prior to the approximation and retrieved from a database. Rather, Shannon chose a page out of a book at random and read until the prior combination of characters was found. The following character after the match was recorded and the process reiterated. The word approximation system follows an identical ordering nomenclature but replace characters with words as the basic unit. These methods exhibited the ability of algorithms to accurately estimate patterns using present neighbors (prior characters/words) and a training set (book), not a stored database of the probability of all possible combinations.

1. Zero-order letter approximation.
   XFOML RXKHRJFFJUJ ZLPWCFWKCYJ FFJEYVKCQSGHYD QPAAMKBZAACI
2. First-order letter approximation.
   OCRO HLI RGWR NMIELWIS EU LL NBNESEBYA TH EEI ALHENHTTPA
3. Second-order letter approximation.
   ON IE ANTSOUTINYS ARE T INCTORE ST BE S DEAMY ACHIN D ILONASIVE
4. Third-order letter approximation.
   IN NO IST LAT WHEY CRATICT FROURE BIRS GROCID OF DEMONSTRURES OF
5. First-order word approximation.
   RESPRESENTING AND SPEEDILY IS AN GOOD APT OR COME CAN DIFFERENT
6. Second-order word approximation.
   THE HEAD AND IN FRONTAL ATTACK ON AN ENGLISH WRITER THAT THE

Figure 3.1 Shannon's series of approximations to English (adapted from Shannon, 1948).

3.1.2 Direct Sampling Algorithm

The following algorithm has been adapted from The Direct Sampling method to perform multiple-point geostatistical simulations (Mariethoz et al., 2010). It provides the order of operations originally specified by Mariethoz.

1. Assign conditioning data to simulation grid (SG).
2. Define a random path through the remaining unknown nodes in SG.
3. For each successive point **x** in the path:
    a. Assign *n* number of closest known neighbors to unknown location **x**.
    b. Compute the lag vector **L** for the location **x**, containing *n* subsets. In Figure 3.2a, the neighborhood lag vector for the gray node consists of three subsets, **L** = {(1,2), (2,1), (-1,1)}.
    c. Specify a data event $\mathbf{d}_n$ corresponding to the values at each neighbor location. In Figure 3.2a, $\mathbf{d}_n$ = {0, 0, 1}.
    d. Define a search window in the training image (TI) such that all nodes in the event pattern (**L** and $\mathbf{d}_n$) will be within the TI.
    e. Define a random path through the TI search window.
    f. For each successive point in the search path:
        i. Retrieve the data event in the TI $\mathbf{d}_{TI}$ for the lag vector **L**. In Figure 3.2c, $\mathbf{d}_{TI}$ = {1, 0, 1}.
        ii. Calculate the difference *d* between $\mathbf{d}_n$ and $\mathbf{d}_{TI}$. The difference may be computed using various methods and for both continuous and categorical fields.
        iii. Store *d*, grid location, and value of the point in TI if *d* is the lowest calculated in the search path.
        iv. Return the stored value if d is less than the specified threshold t, signifying an appropriate match, Figure 3.2d. Proceed to next point **x**.

v. Return the stored value if the total number of iterations in steps i-iv exceed a specified fraction of the TI search path *f*. Proceed to next point **x**.

3.1.3 Discussion of Direct Sampling Algorithm

The direct sampling algorithm begins with the choice of an un-informed simulation node as a starting location in the sparse target field and storing the nearest data event surrounding that point. The stored data event is a patterned template containing the local coordinates and conditioning value of a specified number of known neighboring data points in the sparse field. The training image is then searched for a pattern best matching the data event previously retrieved. A limited search window and pattern matching tolerances may be specified to save computational time. Once a suitable match has been made, the value corresponding to the central node of the search template is obtained from the training image and transferred back into the original sparse data field. This value replaces the unknown value at the simulation node and a new un-informed location is chosen in the sparse field. The process repeats itself until the sparse field is fully realized.

One of the advantages of multi-point geostatistical algorithms based on conditioning data is the adaptive nature of the search template as the simulation process continues, seen in Figure 3.3. As data events are filled in to the simulation grid, more conditioning data become available for the next data event. The conditioning event template therefore become smaller since the specified maximum number of known neighboring locations can be found within a shorter distance away from the simulation node. These tighter patterns will enable reproduction of the small scale structures found in the training image.

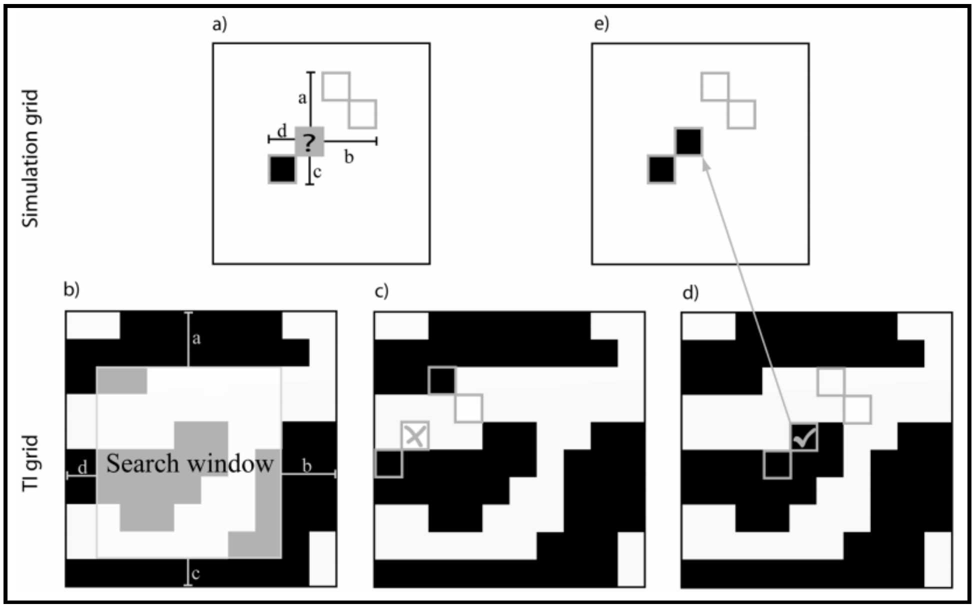


Figure 3.2 Visual order of operations for direct sampling (adapted from Mariethoz et al., 2010), a) Search window definition and acquisition of neighbors, b) Specifying the search window and path in the training image, c) Pattern does not match desired event pattern, d) Pattern match has been found and center node is stored, e) Stored value is returned to the original sparse map

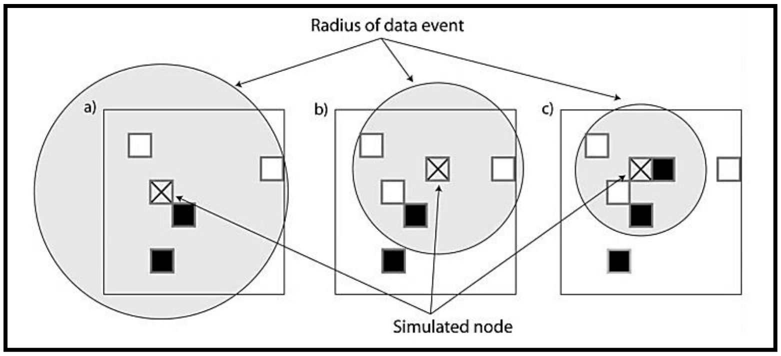


Figure 3.3 Search window reduction (adapted from Mariethoz et al., 2010).

3.2 Seismic Integration within the Direct Sampling Framework

Development on the Direct Sampling Seismic Integration Process (DSSIP) was first conceived by Henke and Srinivasan (2011). DSSIP synthesizes structural and geologic variability in a simulation grid using a training pair and of a facies map with its corresponding seismic data (the target pair). The relationship between the training image and the seismic map is established and subsequently imposed on the simulated image conditioned to the available data. A two-step simulation procedure is implemented to convert the sparse facies map into a fully realized geologic output image.

There are two main iteration loops in DSSIP. The first loop uses the coarse pattern related information seen in the seismic data to preserve the approximate location of the facies in the simulation domain. The second loop fills in small-scale geobody relationships that cannot be captured by the seismic data but may be contained in the training image. The two loops complement each other and at the end of the algorithm, yield a map that is both structurally and geologically accurate. Details of this implementation are presented in the next section.

3.2.1 DSSIP Input Requirements

The target reservoir is the subject of DSSIP analysis. The location and continuity of geologic facies bodies present in this reservoir are largely unknown as the conditioning data is only available at a few wells. Limited knowledge of the reservoir will make it difficult to infer facies characteristics without additional information in the training reservoir model. However, in order to pinpoint the location of the facies, an exhaustive seismic map of the target reservoir is necessary.

The training reservoir is the data set from which relationships between patterns from the seismic and facies information have to be developed. Therefore, it must be a geologic analog for the target reservoir in order to preserve the spatial relationships between facies. The interpretation of what defines an analog field will change depending on the situation. Analogs can include fields that share similar depositional environments or fields that neighbor one another in a common play.

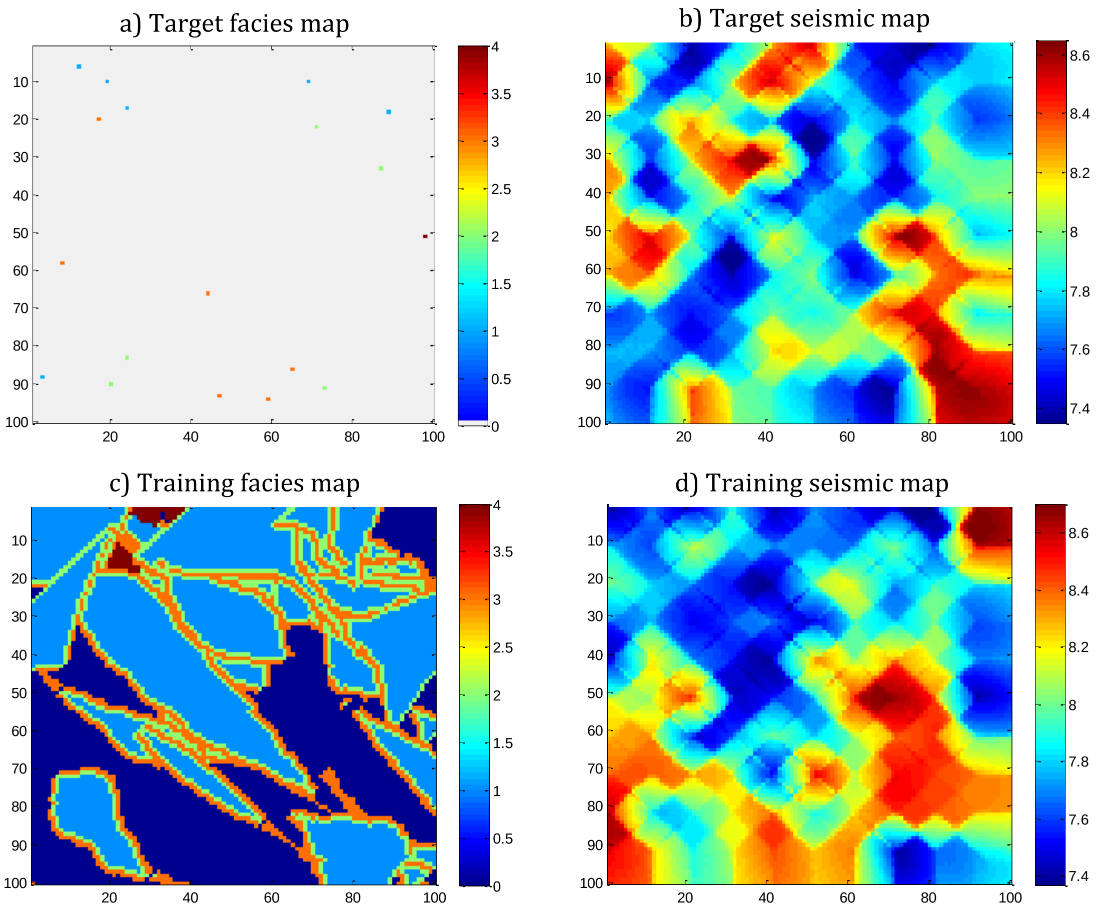
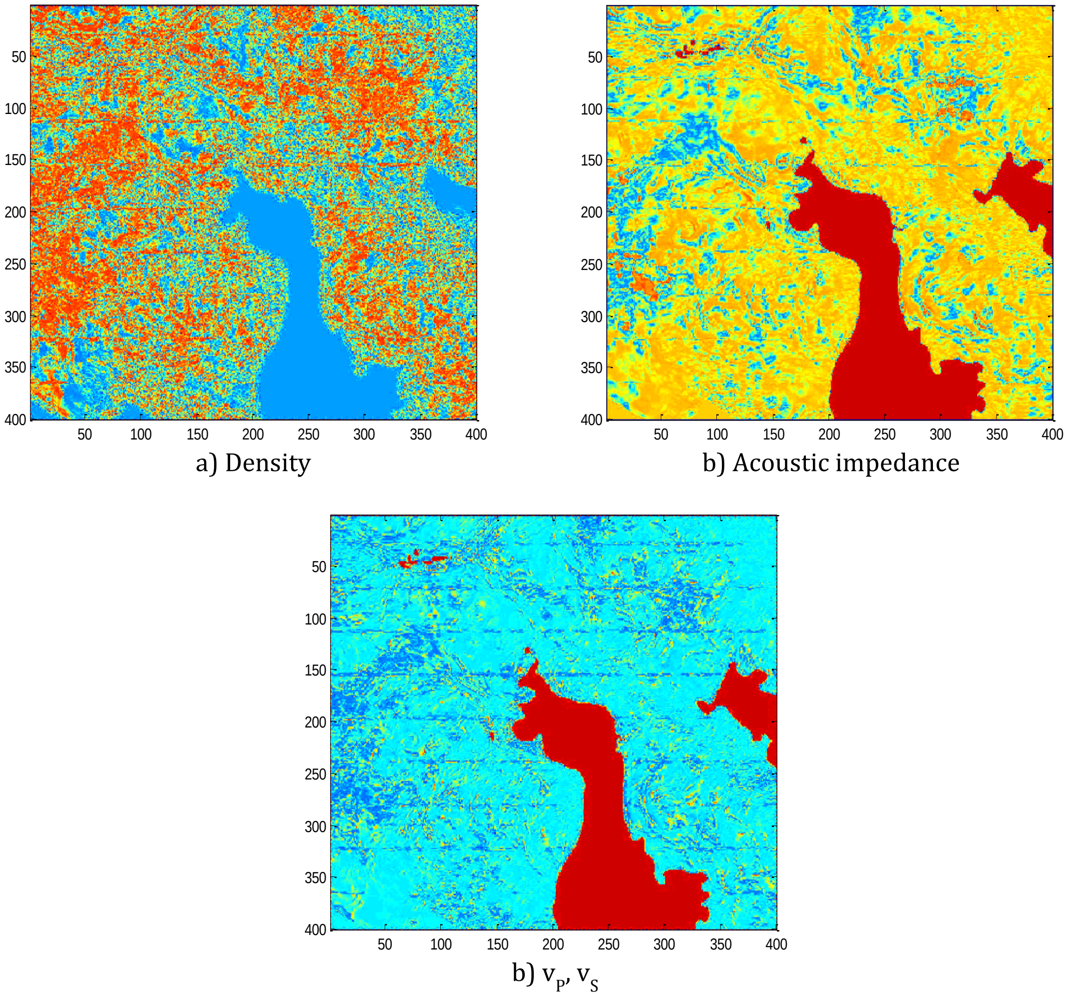


Figure 3.4 Data requirement for the DSSIP algorithm, a) Conditioning data for the target reservoir, b) Exhaustive seismic data for the target reservoir, c) Training reservoir model depicting the pattern of facies variability, d) Seismic response simulated on the training model in order to obtain the corresponding seismic training data.

As already discussed, the information regarding the reservoir is represented in map pairs composed of a facies map and a seismic map. Acoustic impedance, wave velocity, and density are all acceptable seismic attributes for developing the reservoir model. It is recommended though not necessary, that the seismic variable for the target and training reservoirs be the same so that a correspondence between the patterns can be established.



a) Density

b) Acoustic impedance

b) v_P, v_S

Figure 3.5 Differences in pattern information for different seismic data types, values have been converted to a neutral 0 to 1 scale.

3.2.2 The DSSIP Algorithm

The first section, Loop 1, of DSSIP focuses on information that can be gathered from the target seismic map. Figure 3.6 below illustrates the procedure inside the first loop. Starting with the target facies conditioning data, a random path through the un-informed nodes is first created. The random path defines the order in which each location in the simulation grid is visited. For the first location (**$\mathbf{u}_1$**), the nearest known neighbors are identified and their locations relative to **$\mathbf{u}_1$** are noted. This data subset creates the data event pattern. The pattern is then transferred to the accompanying target seismic map centered on the same location (**$\mathbf{u}_1$**). The seismic value at each location on the event pattern is stored. The event pattern now contains the facies data in the neighborhood of the simulation node as well as the corresponding seismic pattern event.

The seismic event pattern is carried into the training seismic map. Another random path is generated but, now, it is a search path through the training seismic map. The seismic event pattern is moved from location to location on the specified search path until a match in the training map is observed. A suitable tolerance for the matching process must be specified due to the continuous nature of the seismic data. Once a suitable match occurs, the location of center node of the matching location in the training pair is noted (**$\mathbf{u}_2$**). The corresponding facies at location **$\mathbf{u}_2$** in the training facies map is returned to the original simulation grid. This provides the simulated facies at the location **$\mathbf{u}_2$**. This process is repeated until all unknown facies locations in the sparse target reservoir model are filled.

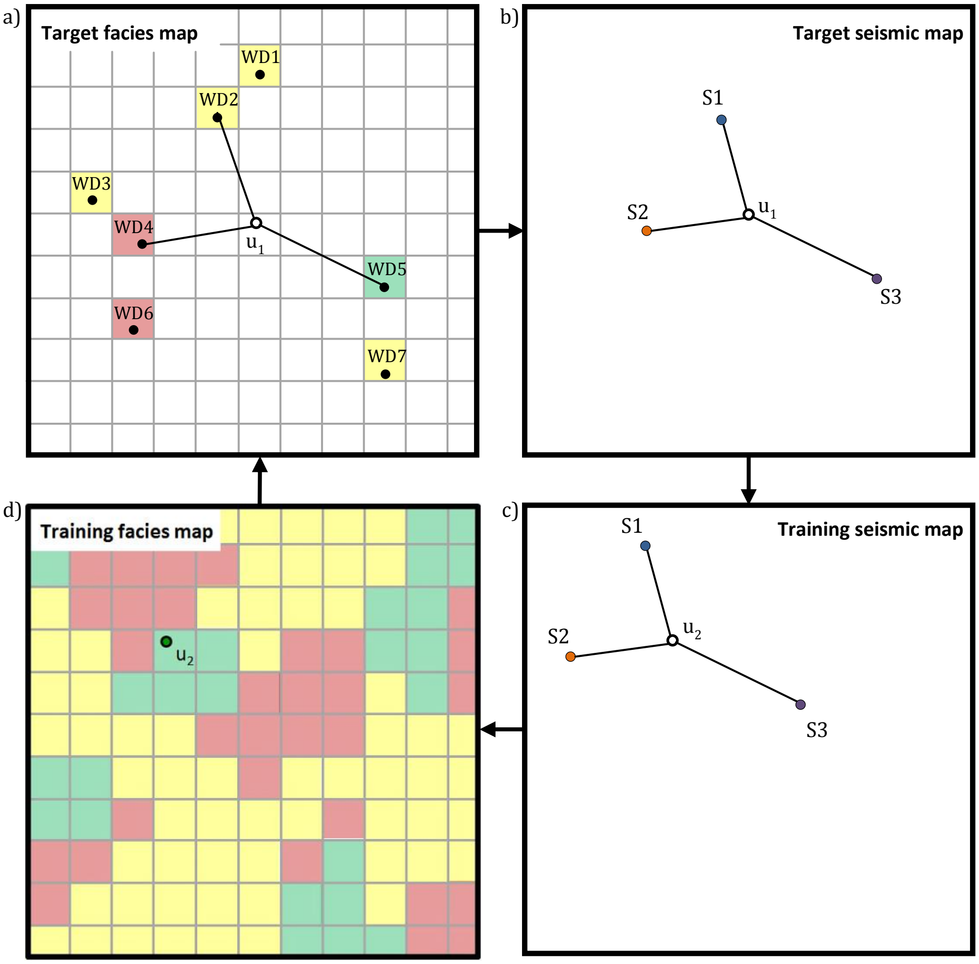


Figure 3.6 DSSIP pattern Loop 1 algorithm, a) Sparse facies map of unknown field, Field 1. An event pattern is created, storing the location of nearest known data points. b) Event pattern is transferred into seismic map of Field 1, and impedance is stored at pattern locations. c) Seismic map of geologic analog, Field 2, is searched for best match to updated event pattern. d) Best match location is transferred to Field 2 well data map and well data at location is stored. Well data value is transferred to event pattern center of Field 1 and process repeats.

At this point, the target facies map has been filled in completely using information from the relationship between the two seismic maps. The second loop begins by sampling a specified number of conditioning data from the target model generated to create a new simulation grid. The target facies map is sampled to allow for the next direct sampling step to insert higher resolution geologic patterns. These sampled conditioning data values can be considered as pilot points that are used to further develop the higher frequency structures observed in the training model. It is recommended that only 10% of the previously simulated facies values are carried into the new map. More discussion of the algorithm parameters is presented later in the thesis. Now, the sparse new simulation map and the training facies map are inserted into the direct sampling algorithm. The process fills in the gaps in the new simulation grid, creating the final output facies map for the target reservoir.

Figure 3.7 summarizes the DSSIP process using a basic example run. The target facies map uses far more conditioning data than will be typically used in a true application of DSSIP. Again, the first loop captures the larger structures as seen in the results of the first loop, the seismically-filled facies map that starts the second loop. The second loop fills in the details and finalizes the map.

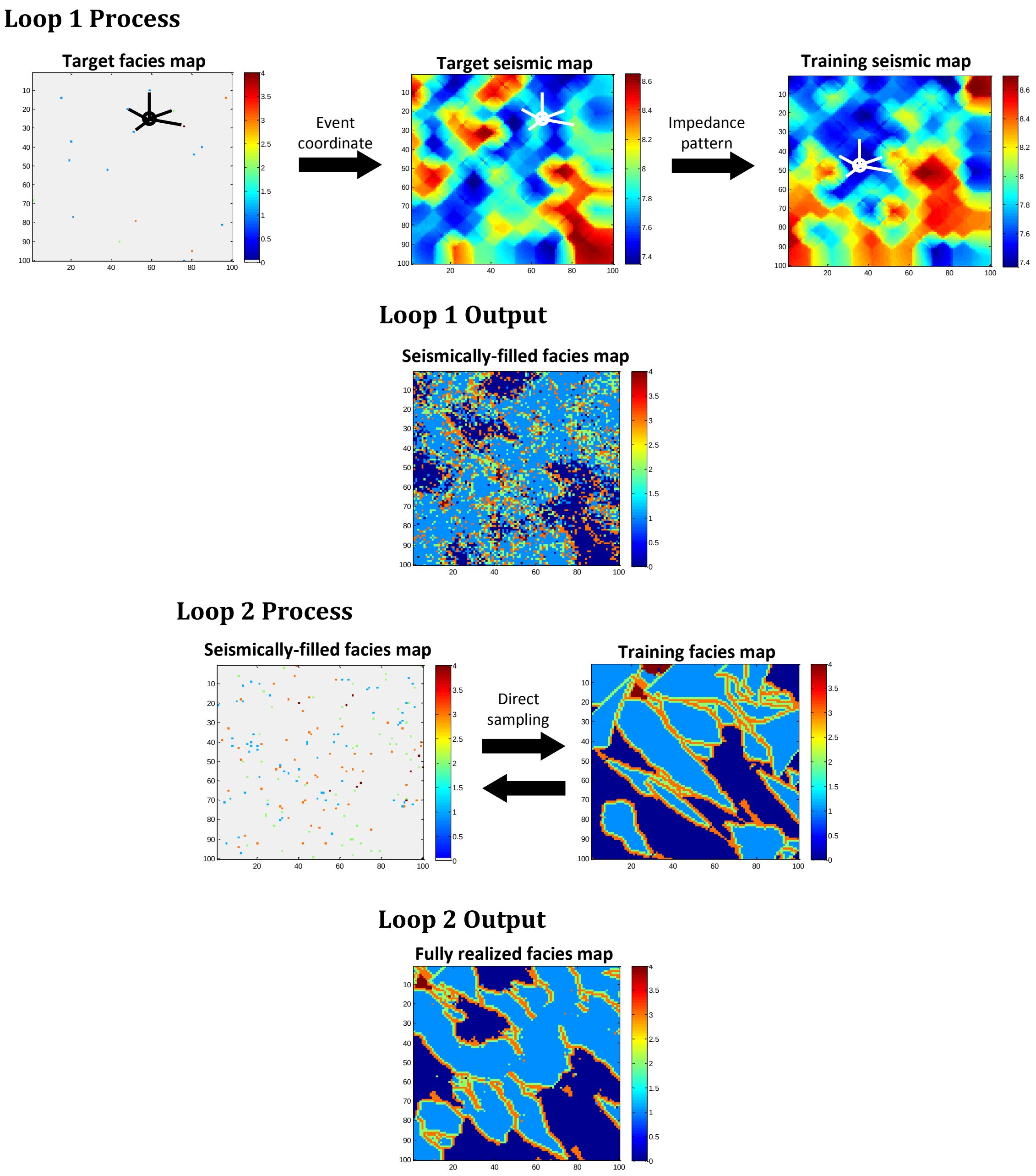


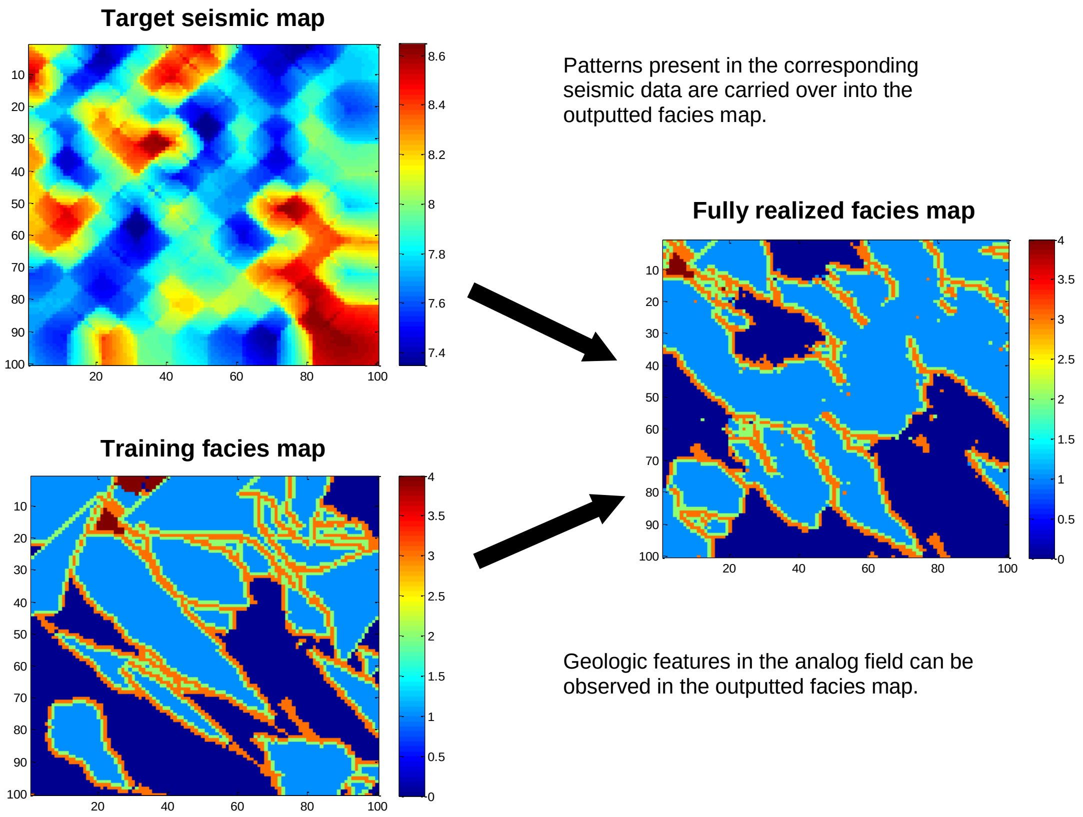
Figure 3.7 Implementation of the DSSIP algorithm.

3.2.3 DSSIP Output Maps

In Figure 3.8, the target seismic map and the training facies map are compared to the simulated facies realization. The previously discussed influences of the first and second loops become clear. The final simulated model shares the characteristics of the training model and yet retains the locations of the main facies as dictated by the seismic impedance. The higher impedance in red in the target seismic map roughly corresponds to the dark blue facies in the simulated realization. In the final simulated realization, four small regions of high impedance are present in the top left corner, one medium region in the bottom center, and one large object in the bottom right. These three features are also seen in the final simulated realization. Looking specifically at the facies object in the bottom right, the shape of the object has been preserved in addition to its location and size. A majority of the other objects have also been similarly preserved.

By comparing the training facies map to the fully realized simulated facies map, it is possible to see the high resolution details not captured by the seismic imaging. The dominant dark blue facies in the fully realized map is consistently outlined by the same pattern of facies (orange to green) as is present in the training image. The thin orange and green objects protruding from the dark blue facies are captured. At the top right of the fully realized map, the shape and connections of the red facies object has been copied over from the training facies map nearly identically. It is important to note that process also successfully replicates the frequency of small objects, such as the thin orange and green connections.

The application of the direct sampling algorithm has difficulty in connecting long thin objects as seen in the fully realized map along a corridor from the bottom left to the top right. Thin objects will begin at two separate locations and will have difficulty joining because each object lacks knowledge of the location of the other object.



Target seismic map

Training facies map

Fully realized facies map

Patterns present in the corresponding seismic data are carried over into the outputted facies map.

Geologic features in the analog field can be observed in the outputted facies map.

Figure 3.8 Information derived from the seismic map and the training image for the simulation of facies in the target reservoir model.

The following output maps reinforce the relationships seen in Figure 3.9 and discussed in the prior paragraphs. The same target and training reservoir models were used for each run and the following realizations were obtained by implementing the algorithm and the random path along which the nodes are visited. The realizations are compared against a reference facies model from which the initial conditioning data for Loop 1 were extracted. Large structural patterns are maintained in each of these output maps and the models share similar facies architecture in each instance.

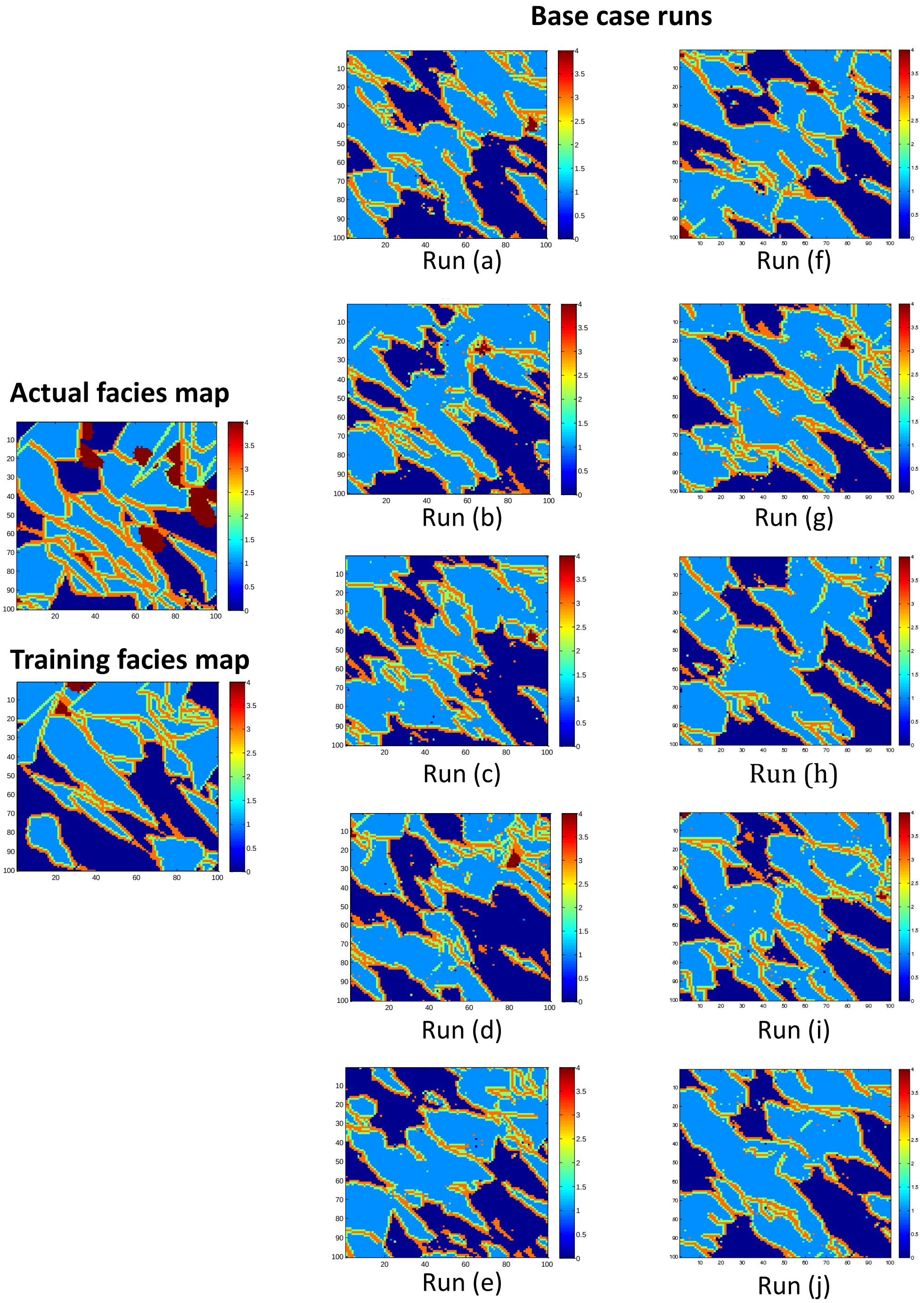


Figure 3.9 Ten realizations of the reservoir facies model obtained conditioned to the same "hard" data and the training image.

Figure 3.10 places the actual and training facies maps beside run (c) to further demonstrate the structures and relationships replicated. There are three main considerations to note between the actual map and the output map: location of structures, shape of structures, presence of thin structures, and proportion of facies. A majority of the large bodies were accurately placed in the simulated model, as evidenced by the replication of the dark blue structures in run (c). Two dark blue bodies are present in the bottom left quadrant of the reference facies map. The general location and shape of these bodies are also present in the output map. However, the algorithm does have difficulty in producing smooth transitions between facies, resulting in shapes that may capture the general outline but cannot be used to define facies boundaries.

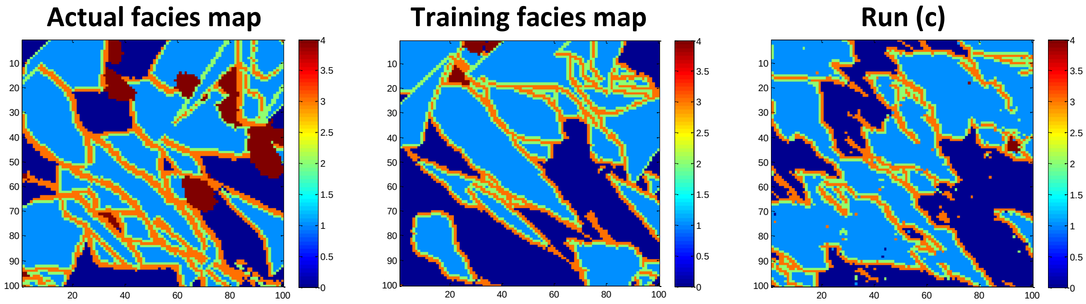


Figure 3.10 Comparison between the reference facies model, training facies model, and a simulated realization (Run (c) from Figure 3.9).

Thin bodies were propagated throughout the simulated model and in the correct direction of anisotropy. But they do not occur with the same frequency as those in the actual map. Instead, they appear to be more similar to the thin bodies in the training image. The similarity between the thin bodies in the training and output maps is expected. The reproduction of large structures relies on the conditioning influence of the seismic impedance data. Thin and high detail structures will be reproduced based on the characteristics of the training image in the second loop.

The proportion of facies is dependent on the proportion represented in the conditioning data for Loop 1 of the algorithm. A large number of known conditioning points at the start of Loop 1 will more accurately reproduce the desired facies proportions. A small number of conditioning locations where chosen in this simple example, to better represent a real field case for the implementation of the algorithm. Therefore, the facies proportions in run (c) are largely dependent on the facies present in the training image.

Chapter 4: Optimization of the DSSIP Algorithm

In the previous chapter a pattern search algorithm was presented for performing multiple point simulation of a reservoir model conditioned to seismic data. The algorithm uses the direct sampling algorithm for multiple point simulation. In this chapter, the seismic integration algorithm (DSSIP) is explored further in order to understand its sensitivity to model parameters and also to optimize the data integration process. A comprehensive sensitivity analysis for several internal parameters of DSSIP was conducted to optimize estimation quality and minimize computational cost. During the construction of the algorithm, small training models containing 10,000 nodes were used and search thresholds were held strict. However, once the process is scaled up to cases with greater than 25,000 simulation or training nodes, the base case parameters becomes extremely inefficient and optimum parameters must be specified. Search tolerances must be relaxed while maintaining an acceptable final facies model.

4.1 BASE CASE SURVEY

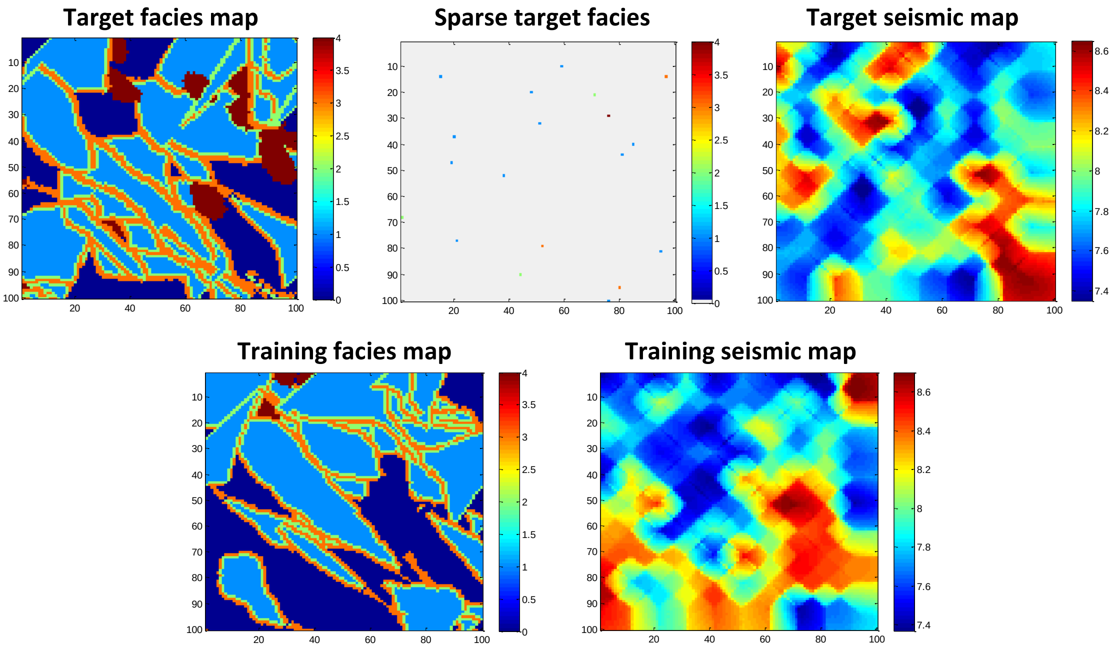
A base case simulation was performed to serve as a benchmark for further testing. Table 4.1 displays the main parameters defining the computational cost and simulation quality using DSSIP. The accompanying values are the initial values as specified by Henke (2011) and used in the base case. They are described in further detail in the following sections.

Table 4.1 Base parameter specification

	Parameter	Description	Value
Loop 1	Full map fill	Loop 1 is completed prior to sampling into Loop 2.	Yes
	Fraction of TI to search	Defines the length of the search path in Loop 1.	90%
	Number of neighbors	Number of neighbors required to create a data event in Loop 1.	15
	Location threshold	Measure of similitude between matching patterns in Loop 1.	0.03
Loop 2	Conditioning points, Loop 2	Percentage of Loop 1 map sampled to start Loop 2.	2.50%
	Fraction of TI to search	Defines the length of the search path in Loop 2.	90%
	Number of neighbors	Number of neighbors required to create a data event in Loop 2.	15
	Location threshold	Measure of similitude between matching patterns in Loop 2.	0.03

4.1.1 Input Data

The input maps used for the base case are as shown in Figure 4.1. The reservoirs were synthetically created, and the seismic maps were similarly generated using the paired facies map. These maps demonstrate the dependency of the structural patterns in the seismic map on the location of facies in the facies map. However, the seismic impedance describes variations at a much coarser resolution. Values within the seismic map are continuous and intended to represent acoustic impedance. The target facies map was sampled to mimic sparse wells in a reservoir.


Target facies map

Sparse target facies

Target seismic map

Training facies map

Training seismic map

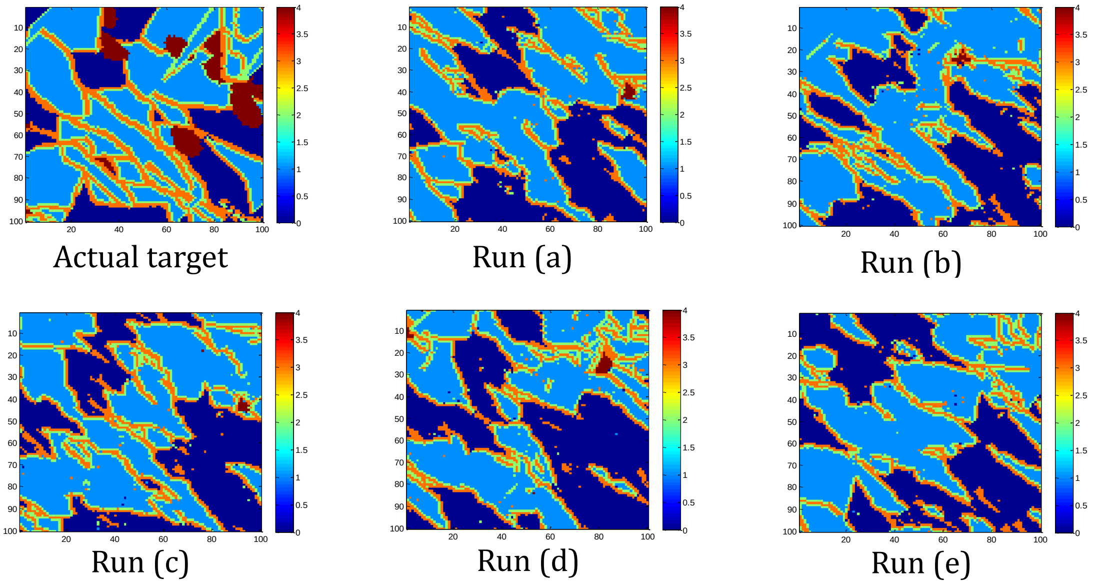
Figure 4.1 Input target and training maps utilized in parameter optimization.

4.1.2 Results and Discussion

The base case runs yielded results that matched the actual target reservoir as shown in Figure 4.2. As previously discussed in Chapter 2, DSSIP retains the direction of facies continuity and replicates the small-scale relationships between geologic bodies. This is evidenced in the output reservoir maps below. A summary of the computational times for each run is presented in Table 4.1. These times will increase dramatically once the process is applied to reservoirs any larger in size. Therefore, the following discussion of parameter optimization is warranted by a desire to reduce the computational time required to complete DSSIP.

Table 4.1 Loop completion time for base case runs.

Run	Completion Time (s)	
	Loop 1	Loop 2
a	2301	1679
b	2313	1708
c	2316	1682
d	2275	1465
e	2341	1665
Avg.	2309	1640



Actual target

Run (a)

Run (b)

Run (c)

Run (d)

Run (e)

Figure 4.2 Output fully-realized target facies map for base case runs with actual facies map for reference.

The slight variations in loop completion times between the runs are due to the different number of loop iterations that are terminated earlier in some realizations due to accurate matches. An accurate match prior to the end of the search path will end the loop and return that value to the sparse map. The completion times for Loop 1 are larger than that of Loop 2 due to the extra steps of seismic integration. In Loop 2, pure direct sampling will immediately search the training image for patterns and return a match to the sparse map. In Loop 1 on the other hand, the pattern search is conducted twice over the training pair. In addition, a larger number of conditioning points are present at the start of Loop 2, decreasing the total number of un-informed simulation nodes for iteration.

4.2 PARAMETER OPTIMIZATION

After the completion of the base case survey, five parameters were identified according to criticality and potential time improvement. They included the "full map fill" option, fraction of the training image to search, number of neighbors required to create an event pattern, event pattern shape/location tolerance, and the number of conditioning points transferred to Loop 2. The testing was conducted using the same input maps and reservoir pairs as the base case.

4.2.1 Full Map Fill

"Full map fill" indicates whether or not the first loop completely fills the sparse image with facies data prior to being sampled into the new map at the start of the second loop. If this option is not selected, the first loop will terminate after the required number of conditioning points for Loop 2 have been simulated, and those points will be used as conditioning data in the second loop. Skipping additional iterations to fill simulation nodes in Loop 1significantly decreases the calculation time for the first loop. However, this also limits the extent of information extracted from the seismic. As the sparse map is filled, the event patterns contain known locations closer to the unknown central node and, therefore, the matching process will more accurately return the details of small scale features. However, one could argue that the seismic only informs large scale reservoir structures and consequently, terminating the first loop earlier will be acceptable.

The results obtained by selecting or not-selecting this option is presented in Figure 4.3 where they are compared to the set of base case maps. In the “full map fill” case, all locations were simulated in Loop 1 and 250 points were sampled into Loop 2. In the non-“full map fill” cases, 250 conditioning points were calculated in the first loop and transferred directly into the second loop. All other parameters were held constant. Table 4.2 displays the completion time information for each run.

Table 4.2 Loop completion time for non-“full map fill” runs.

	Case B) 250 cond. pts.		**Case C) 1250 cond. pts.**	
Run	**Completion Time (s)**		**Completion Time (s)**	
	Loop 1	**Loop 2**	**Loop 1**	**Loop 2**
a	77	1662	472	1657
b	80	1680	459	1643
c	76	1761	471	1611
d	77	1624	463	1676
e	79	1698	484	1634
Avg.	78	1685	470	1644

In Figure 4.3, the first column contains Runs a-e for the base case algorithm, and the final two columns display the results of not completing Loop 1 and stopping with 250 and 1250 conditioning points (Case B and Case C, respectively). In Case B and C, the location of large structures was correctly reproduced and the presence of thin structures was accurately represented. However, the increased proportion of large dark blue structures in these two cases show the importance of full Loop 1 simulation.

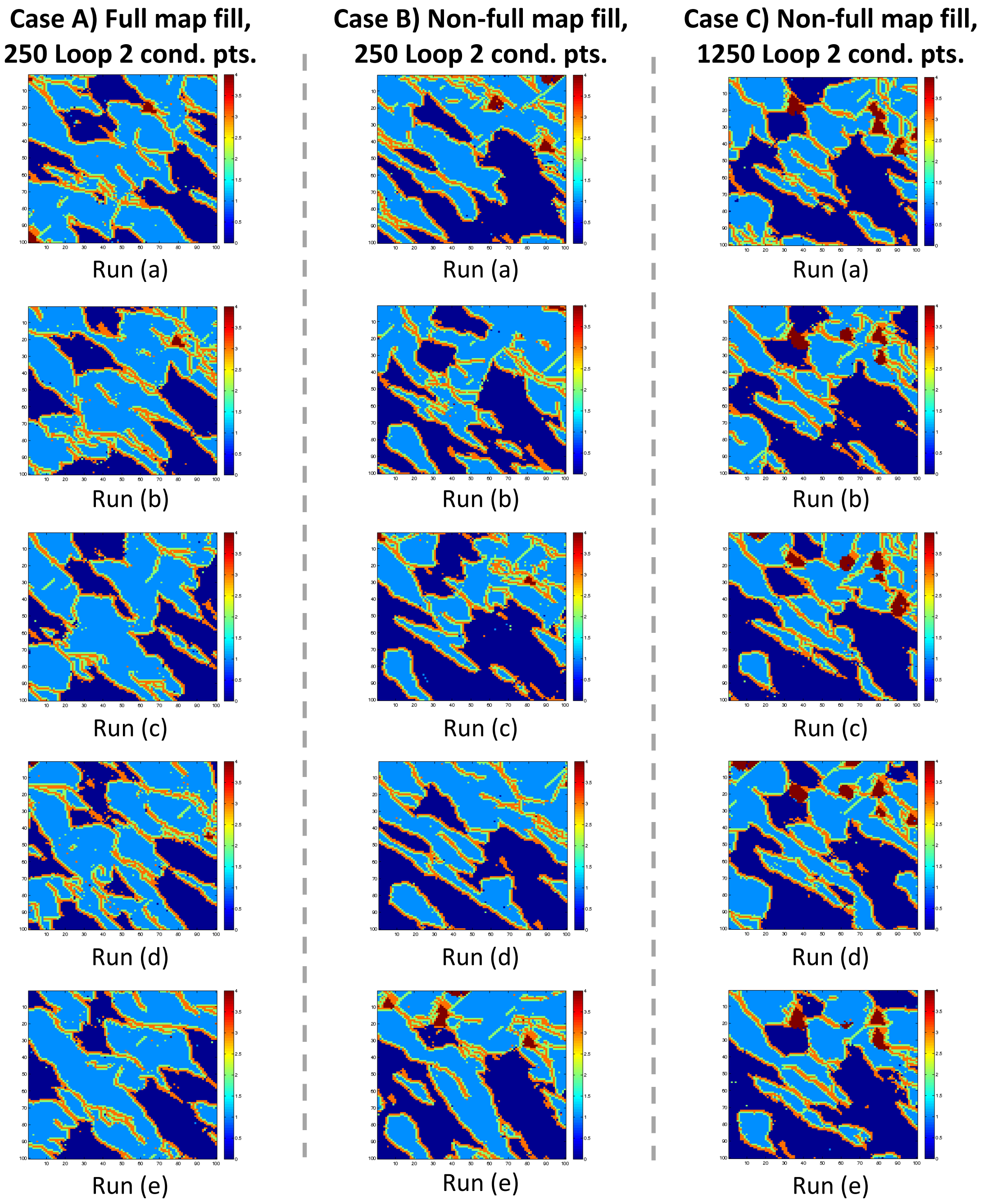


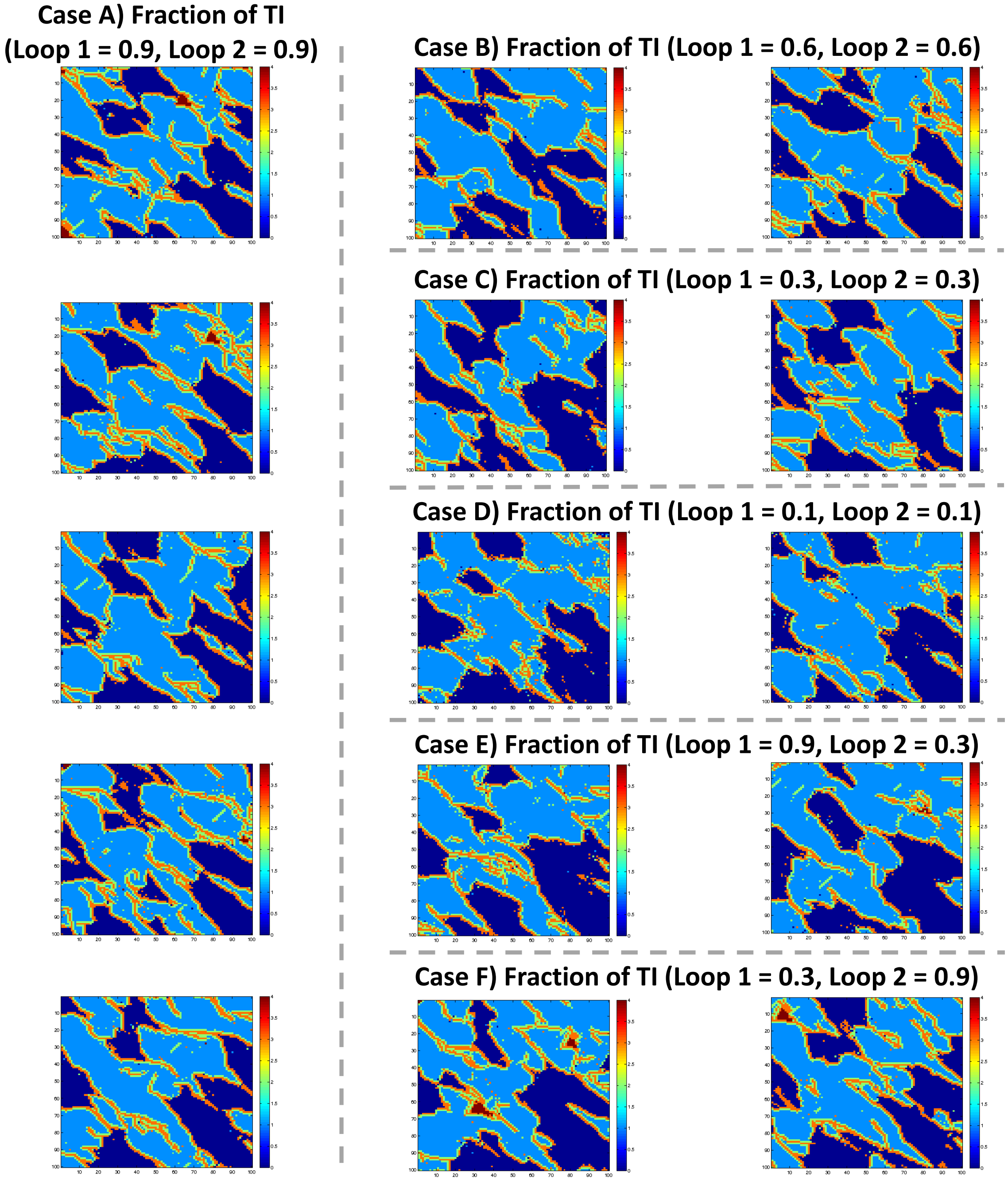
Figure 4.3 Non-"full map fill" and base case output target reservoirs.

Another difference between Cases B and C is the presence of the channel overbank facies indicated in red. Case C contains a large amount of this facies in the top right quadrant, while Case B has amounts similar to that seen in the base case. When compared to the actual facies map in Figure 4.2, Case C has correctly identified the facies proportion and the general location of the red facies. It marks an improvement over the base case in this aspect. The overbank structures are small objects that are not sampled by the original "hard" conditioning data and therefore have to be sampled at the end of Loop 1 in order to be represented in the second loop. The higher number of conditioning points transferred from the results of Loop 1 in Case C creates a higher number of seeded locations at the start of Loop 2.

4.2.2 Fraction of TI to Search

"Fraction of TI (training image) to search" defines the maximum number of training nodes that are visited for matching the conditioning data pattern within the training image. The best match is stored during each iteration loop and is carried into the target simulation grid if a match to the conditioning event pattern within the specified criterion is not obtained by the end of the search path. In Loop 1, the search path is calculated through the training reservoir's seismic map. In Loop 2, the search path is conducted through the training model's facies map.

By optimizing the length of the search path by minimizing the "fraction of the TI to search," the main computational cost of the first loop can be directly lowered. Once again, it comes at a cost of accuracy due to a limited search window. The results in Figure 4.4 below show that a minimal difference exists between the large and small search windows, but an optimal value can still be chosen. Each iteration loop contains a separate "fraction of TI to search" modifier requiring a more complex analysis procedure. Table 4.3 specifies the parameters for each case and displays the average run times. The images shown in Figure 4.4 are representative for each case.



Case A) Fraction of TI (Loop 1 = 0.9, Loop 2 = 0.9)

Case B) Fraction of TI (Loop 1 = 0.6, Loop 2 = 0.6)

Case C) Fraction of TI (Loop 1 = 0.3, Loop 2 = 0.3)

Case D) Fraction of TI (Loop 1 = 0.1, Loop 2 = 0.1)

Case E) Fraction of TI (Loop 1 = 0.9, Loop 2 = 0.3)

Case F) Fraction of TI (Loop 1 = 0.3, Loop 2 = 0.9)

Figure 4.4 "Fraction of the TI to search" output target reservoirs. The models in the left column are realizations of the base case. Two realizations corresponding to each value of the parameter are shown on the right.

The continuity of features in the output maps decrease as the "fraction of TI to search" decreases, but the structure locations are maintained. Case D is at the lowest extreme and searches only 10% of the training image. The mudstone (dark blue) facies bodies in Case D are too large and begin to merge together unlike in the reference. The minimal difference between Case A and Case B supports setting the "fraction of TI to search" to 60%.

Cases E and F explore the sensitivity to the search parameter in Loop 1 and Loop 2 separately. Case E shortens the search length in Loop 2 and produces poorer quality output realizations than in Case F. Loop 2 was found to be more sensitive to reductions in search pattern length. Therefore, it is recommended that the "fraction of the TI to search" for Loop 1 be set at 0.6 and 0.9 for Loop 2.

Table 4.3 Average loop completion time for "fraction of the TI to search" cases.

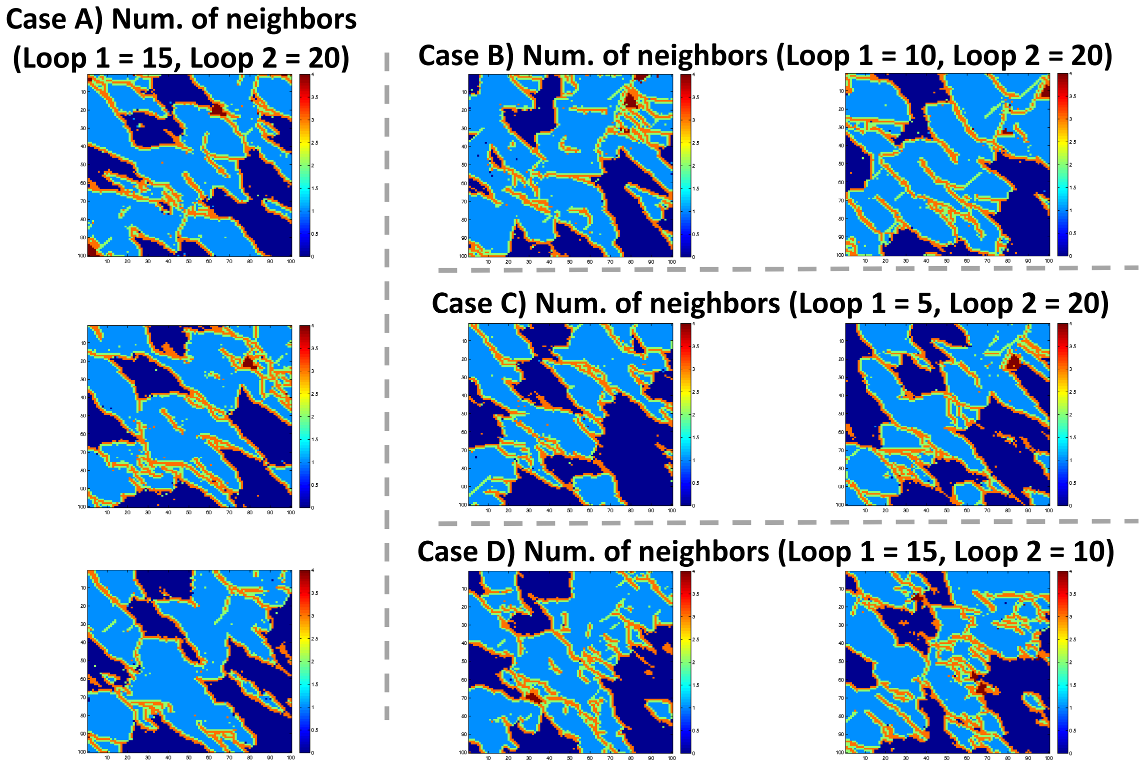
Case	Frac. of TI to search		Completion Time (s)	
	Loop 1	Loop 2	Loop 1	Loop 2
a	0.9	0.9	2301	1679
b	0.6	0.6	1727	1214
c	0.3	0.3	1034	717
d	0.1	0.1	592	455
e	0.9	0.3	2290	722
f	0.3	0.9	1050	1748

4.2.3 Number of Neighbors

Neighbors are the closest known conditioning data in the vicinity of the simulation node. "Number of neighbors" identifies the required number of nearby known data points to create a data event pattern. These event patterns are created on the sparse target maps where few conditioning data values are known. As the target map begins to fill in with new data, the neighbors become closer and more related to the central simulation node. Increasing the number of neighbors will increase the amount of information required to estimate an unknown node, increasing both the accuracy but at the expense of computational time.

The "number of neighbors" surrounding a simulation node will define the total amount of information within an event pattern. It is a parameter defined within both iteration loops and is set to a default value of 15 in Loop 1 and 20 in Loop 2. Table 4.4 shows the decrease in completion time corresponding to the decrease in the number of neighbors.

Case B produced results with structures similar to that of the base case. The large mudstone (blue) structures are in the correct locations and the smaller overbank (red) facies are correctly located in the top right quadrant. In Case C, the limited number of neighbors used for simulation causes the geobodies in the simulated image to be broken up. Further, the proportion of facies favors the larger blue structures. Case D decreased the number of neighbors in Loop 2, resulting in inaccurately placed red facies and an increased number of thin structures. Of the three modified cases, only Case B produced results that capture the variability in the reference facies model.



Case A) Num. of neighbors (Loop 1 = 15, Loop 2 = 20)

Case B) Num. of neighbors (Loop 1 = 10, Loop 2 = 20)

Case C) Num. of neighbors (Loop 1 = 5, Loop 2 = 20)

Case D) Num. of neighbors (Loop 1 = 15, Loop 2 = 10)

Figure 4.5 “Neighbor of neighbors” output target reservoirs.

Table 4.4 Loop completion time for “number of neighbor” cases.

Case	Num. of neighbors		Completion Time (s)	
	Loop 1	Loop 2	Loop 1	Loop 2
a	15	20	2301	1679
b	10	20	1562	1775
c	5	20	583	1617
d	10	10	2336	832

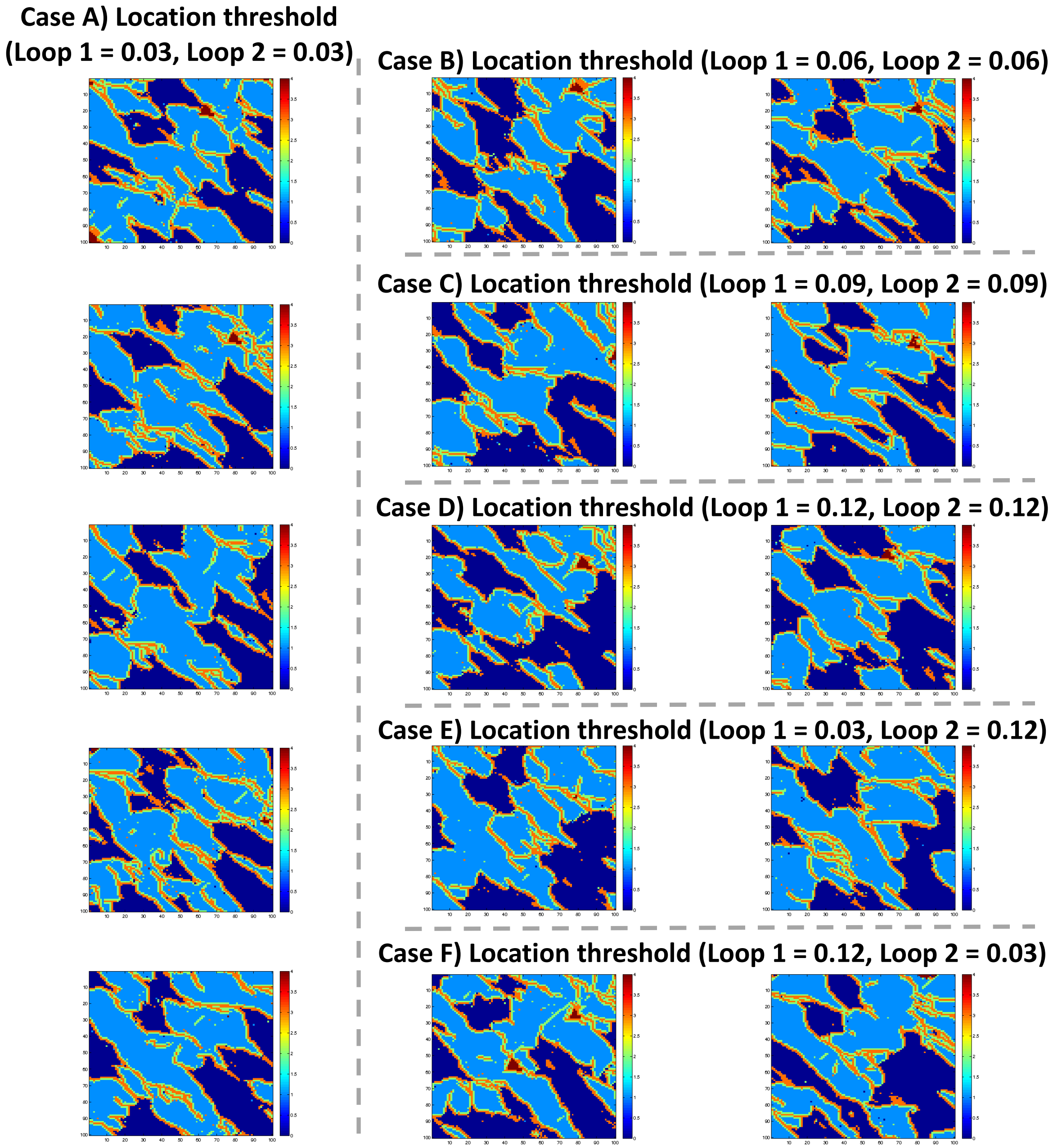
4.2.4 Location Threshold

Once an event pattern is created from neighbors in the simulation grid, the training map is searched for a matching pattern. The “location threshold” is an allowance given for any difference in the exact location of matching data points defining a data event. Flexibility in the exactitude of matched patterns allows for faster simulation and more diversity in the simulated facies maps.

Figure 4.6 summarizes the effect of changing the "location threshold" on the simulation results. Similar to decreasing the number of neighbors, relaxing the location threshold (increasing the value) in either loop causes an increase in the proportion of the large structures. The most extreme case, Case D, showed structures that begin to merge together. Case B increased the threshold value the least, and, even at that increment, the geobodies begin to form incorrect shapes with noticeably inaccurate facies proportions. The computational time savings realized by relaxing the location threshold parameters do not justify the inaccuracies seen in Cases B - F in Figure 4.6.

Table 4.5 Loop completion time for "location threshold" cases.

Case	Loc. threshold		Completion Time (s)	
	Loop 1	Loop 2	Loop 1	Loop 2
a	0.03	0.03	2301	1679
b	0.06	0.06	1616	1295
c	0.09	0.09	989	1030
d	0.12	0.12	398	737
e	0.03	0.12	2294	842
f	0.12	0.03	432	1389



Case A) Location threshold (Loop 1 = 0.03, Loop 2 = 0.03)

Case B) Location threshold (Loop 1 = 0.06, Loop 2 = 0.06)

Case C) Location threshold (Loop 1 = 0.09, Loop 2 = 0.09)

Case D) Location threshold (Loop 1 = 0.12, Loop 2 = 0.12)

Case E) Location threshold (Loop 1 = 0.03, Loop 2 = 0.12)

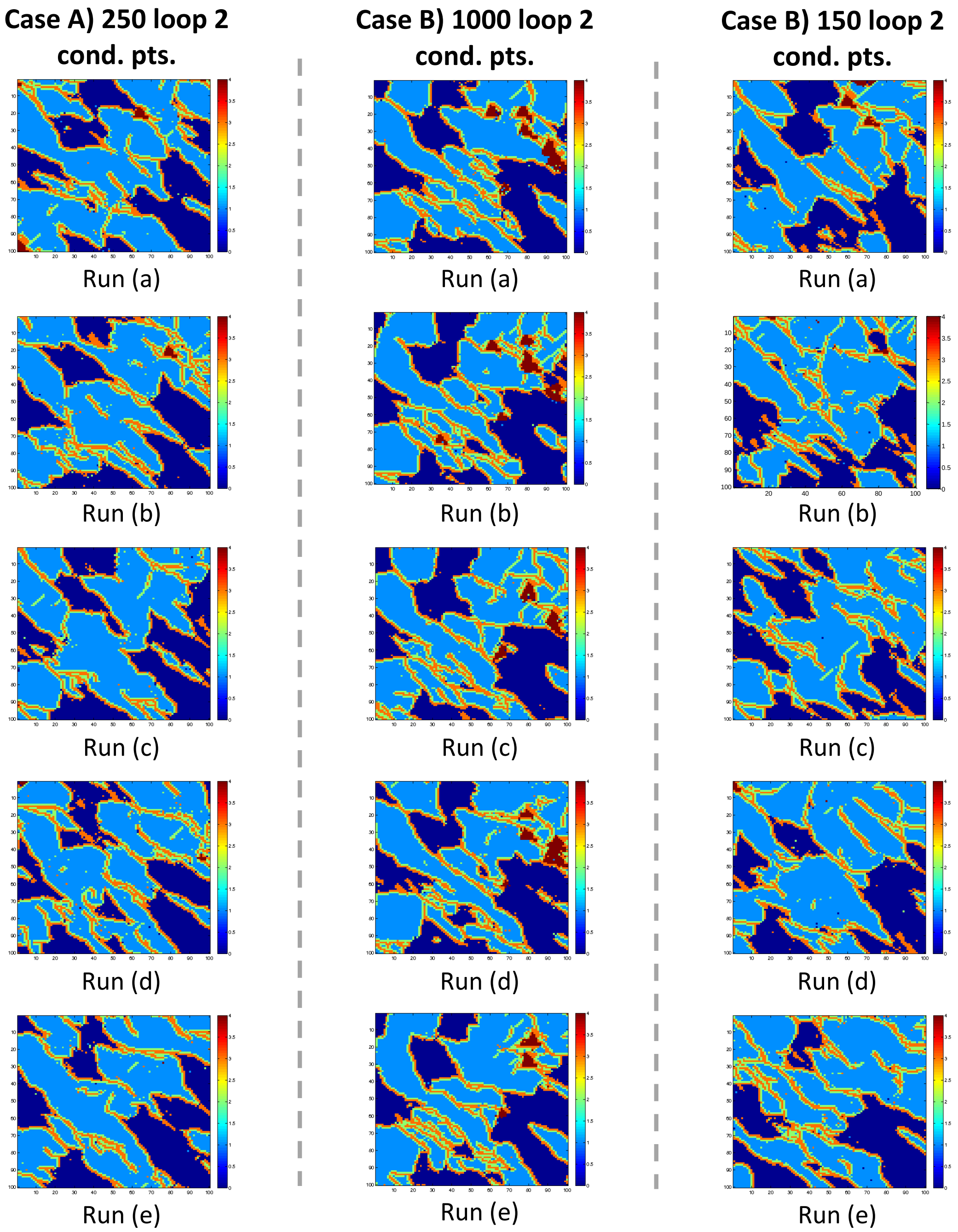
Case F) Location threshold (Loop 1 = 0.12, Loop 2 = 0.03)

Figure 4.6 Sensitivity of simulated facies maps to “location threshold.”

4.2.5 Loop 2 Conditioning Points

Loop 1 utilizes patterns in the seismic dataset to create a basic facies configuration map. The transitioning to the second loop utilizes a specified number of conditioning values sampled from this fully realized structural map that are used together with the original "hard" conditioning data in order to populate the simulation grid. The following exercise uses the "full map fill" option as seen in the base case. Increasing the number of transferred conditioning points will cause the final simulated map to reflect closely the structural information gained from the seismic. Decreasing the number of Loop 2 conditioning points will decrease the impact of the field-scale structural data and, instead, cause short-scale facies relationships observable in the training model to be pre-dominant in the simulated model. In the following results, the number of conditioning points does not impact the total iteration time significantly, because the first loop is allowed to finish and the conditioning points are sampled directly from the full map produced by Loop 1. As seen earlier, the computation time for Loop 2 is a smaller fraction of the total computational cost.

The mudstone (dark blue) structures' locations and facies proportions are consistent across Cases A, B, and C. In addition, the thin border facies (orange/green bodies) occur at a similar frequency and in the correct direction. The main difference is the presence of the overbank (red) facies in the top right quadrant for Case B. In this case, a large increase in conditioning points at the start of Loop 2 increased the final output's reliance on the results of the first loop and the actual seismic information. The increased number of sampled locations from Loop 1 increased the number of red facies seed points, leading to an increased presence of the red facies. It was found that any further increase in conditioning points limited the effectiveness of the second loop.



Case A) 250 loop 2 cond. pts.

Case B) 1000 loop 2 cond. pts.

Case B) 150 loop 2 cond. pts.

Run (a) Run (a) Run (a)

Run (b) Run (b) Run (b)

Run (c) Run (c) Run (c)

Run (d) Run (d) Run (d)

Run (e) Run (e) Run (e)

Figure 4.7 Sensitivity of the simulation models to the number of conditioning data at the end of the first loop.

Table 4.6 The computation time for different number of conditioning points at the end of the first loop.

Case	Cond. pts., Loop 2	Completion Time (s)	
		Loop 1	Loop 2
a	250	2301	1679
b	1000	2369	1661
c	150	2380	1672

4.3 FINAL RECOMMENDATION

Three changes to the parameters of the base simulation are recommended as a result of the optimization analysis. The first is a decrease in the number of search locations in the first loop. Limiting the fraction of the training image to search did not decrease the quality of the facies organization developed during Loop 1. The Loop 1 search path is through the target seismic map, and as a result of the decrease potential search locations are spread evenly throughout the image but at a lower frequency. This lower frequency is acceptable for searching and simulating features from the low resolution seismic map.

The second parameter modification decreases the number of neighbors required to create a data event in the first loop. The low resolution of seismic images implies that a coarser event pattern is sufficient to capture the large structures present. The increased detail introduced by additional neighbors does not justify the increase in computational time. Both changes decreased the amount of information extracted during Loop 1 in favor of decreased completion time.

The last recommended modification is to increase the number of conditioning points transferred from the Loop 1 output into the start of Loop 2. Structure location and facies proportion more closely resemble the actual facies map after increasing the number of Loop 2 conditioning points to 10% of the total grid locations. The change does not increase total computational time and creates a better balance between fine and large scale facies reproduction.

Table 4.7Base and final recommendation parameter specifications.

	Parameter	Base	Final
Loop 1	Full map fill	Yes	Yes
	Fraction of TI to search	90%	60%
	Number of neighbors	15	10
	Location threshold	0.03	0.03
Loop 2	Conditioning points, Loop 2	250 (2.5%)	1000 (10%)
	Fraction of TI to search	90%	90%
	Number of neighbors	20	20
	Location threshold	0.03	0.03

The results of the final recommendation runs are located in Figure 4.8. The accuracy of both the large and thin structures did not change when relaxing the Loop 1 parameters from Case A to Case B. The increased number of conditioning points at the end of Loop 1 created a more accurate representation of the overbank (red) facies locations corresponding to the recommended parameters. In Table 4.8, the Loop 1 computational time showed a 46.6% decrease as a result of the recommendations. Across both iteration loops, Case A (base parameters) had an average run time of 3949 seconds and Case B (final recommendation) finished in 2898 seconds, a decrease of 26.6%.

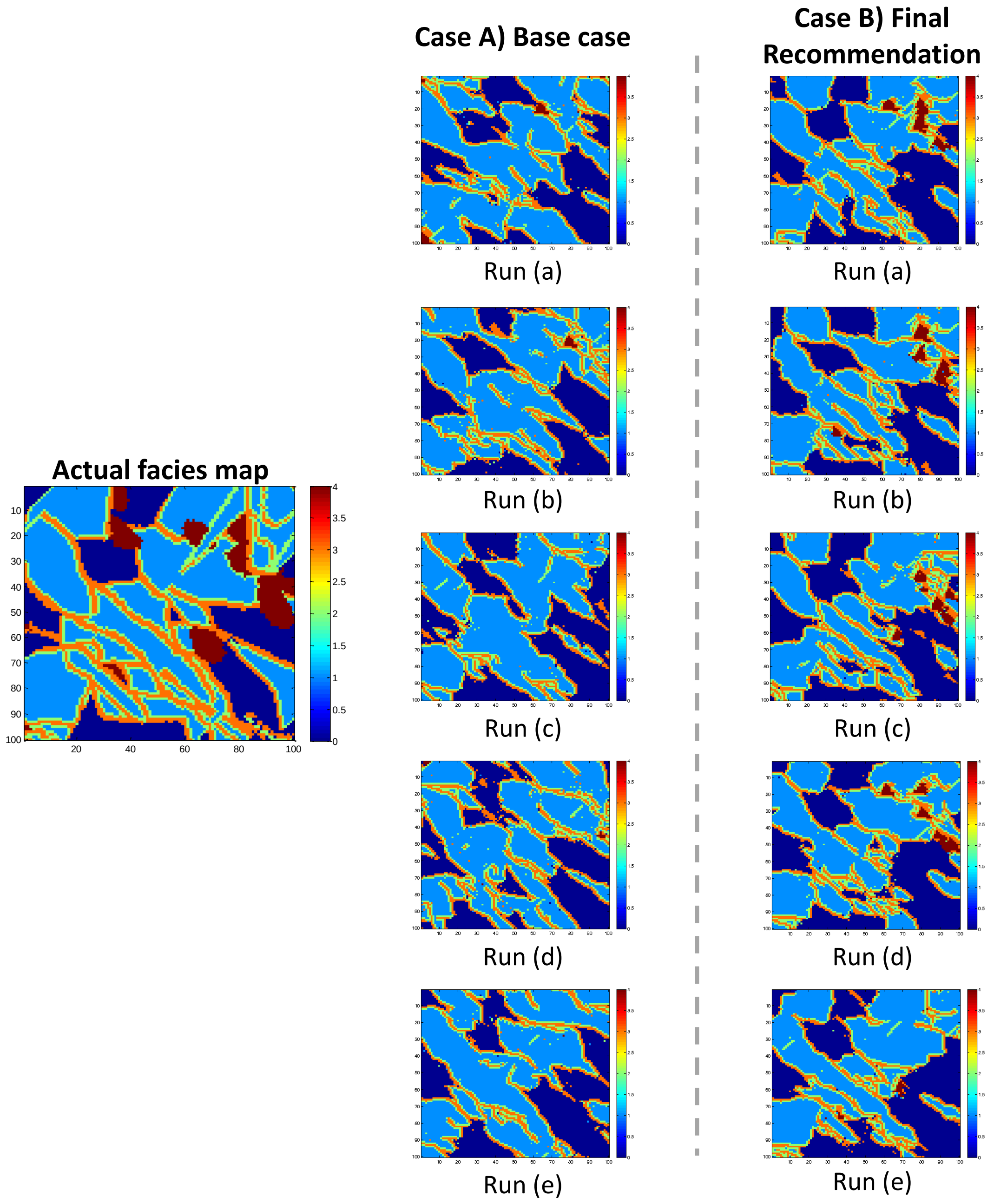


Figure 4.8 Base and final recommendation case output maps.

Table 4.8 Comparison of base and final recommendation case completion times.

	Base case		Final recommendation	
Run	Completion Time (s)		Completion Time (s)	
	Loop 1	Loop 2	Loop 1	Loop 2
a	2301	1679	1244	1700
b	2313	1708	1207	1634
c	2316	1682	1220	1596
d	2275	1465	1237	1712
e	2341	1665	1258	1683
Avg.	2309	1640	1233	1665

The parameter optimization exercise has resulted in the algorithm representing the observed features more accurately and trimming the computation time. The recommendations are a general set of parameters that will provide users a starting point for DSSIP application. However, each application will differ in resolution, size, and desired accuracy. To that end, the optimization exercise has revealed that deep reductions in time can be offered for a minimal reduction in accuracy.

Chapter 5: Implementation of DSSIP to Lobster Field

The DSSIP algorithm described in the previous chapters requires a training reservoir model and the corresponding seismic in order to learn the relationship between the reservoir heterogeneity and the corresponding seismic response. Then, when the seismic response for the target reservoir is known, the "hard" conditioning data for the target reservoir together with the seismic response are used to develop models for the reservoir. These models incorporate the relationship between seismic response and the target reservoir characteristics such as facies distribution learned from the training pair. However, often it is difficult to obtain the forward seismic response corresponding to a training reservoir model. Representative synthetic seismic response is often difficult to compute because of the limiting assumptions incorporated in those models. Therefore, it is beneficial to develop a method of determining facies information given a single volume of seismic response for the target reservoir. The following exercise explores two different applications of DSSIP to handle this scenario.

DSSIP is capable of adapting to these circumstances without major modification to the base algorithm by utilizing the conditioning data and the training reservoir model to create a suite of reservoir models using direct sampling. A single seismic map is then used to match the data event in the vicinity of the simulation node to a similar pattern at another location in the same map. The matching facies pattern is then searched among the suite of reservoir models and the facies outcome at the simulation node corresponding to the closest match is pasted in the simulation grid. If more than one reservoir model has the same facies pattern, the facies outcome with the highest frequency of occurrence at the simulation node is returned.

In the following Lobster field example, a slight variation to the above was implemented. A coarse visual interpretation of the seismic response was performed in order to delineate channel-like features. This coarse interpretation was used as the training image and the seismic impedance values were used as the training seismic data. The training pair will thus consist of the partial interpretation facies map and the seismic map, and the target pair will contain the sparse well data and the same seismic map. The objective is to fill in more details in the target reservoir model based on an understanding of the relationship between the seismic response and the channel interpretation. The identical seismic input maps causes the event patterns created in the seismic image to be searched within the same seismic image for matching locations. The search path should be limited to ensure that the same location on the partial map is not directly transferred to the simulation grid.

We start first with a description of the Lobster field and the accompanying data for this field. The data was afforded to us by Marathon Corporation.

5.1 DESCRIPTION OF LOBSTER FIELD

Lobster field is a turbidite complex in the Gulf of Mexico that has been extensively studied since its discovery in 1991. The seismic amplitude volume for this field as well as a partial suite of well logs and production information was made available to the University as part of an ultra-deepwater reservoir characterization project funded by RPSEA.

5.1.1 Overview of Turbidite Geology

Turbidite complexes are formed from clastic deposits on the deep ocean floor transported by gravity flows. At deposition, the complexes contain silt, mud, and sand and form in overlapping fan patterns. The sediments are carried downslope and onto the basin plain in turbidity currents, high density currents containing sediments. Turbidity currents travel faster than the lower density surrounding fluids enabling the sediments to maintain suspension within the turbulent flow. High density particles fall out of the turbidity current as the flow slows down, leading to lower densities as the velocity decreases.

Figure 5.1, below, shows the general pattern of deposition for a turbidite complex. Sediments from a delta are fed along incised submarine canyons across the continental shelf at high velocities. Once the turbidity currents reach the basin floor, they begin to lose velocity and are transported within leveed channels to the location of deposition. Distributary channels may diverge from the main leveed channels creating larger and more complex systems. Deposit fans, or lobes, occur at the end of the distributary or leveed channels and are the final resting place for the sediments. The geometry, composition, and architecture of the fans vary according to deposit location and source material. Fan widths range from 10 – 300 km with lengths reaching as far as 3000 km. The fans are deposited in bathymetric lows, gradually raising the lowest elevations. As more sediment is deposited over time, the fans begin to overlap and shift deposit locations.

Reservoir architecture and stratigraphy of the turbidite complexes is defined by the nature of the sediment supply and the slope gradients in the receiving basin (Lomas and Joseph, 2004). Lomas and Joseph make the case that the bounds and contours of the ocean floor play a critical role in the development of turbidite formations. The traditional model of radial fans growing uniformly across the basin is inadequate for a vast majority of complexes. The two large categories advocated by Lomas and Joseph have been further divided by Arnold Bouma into four key controls: tectonics, climate, sea-level fluctuations, and sedimentary characteristics (Bouma, 2004). Bouma identifies tectonics as the most influential factor on the formation of a turbidite system. Tectonics includes faulting, slopes, trenches, diapirism, and any other physical bound on the turbidity current. After deposition, tectonic activity and salt diapirs often break and separate the turbidite formations leading to compartmentalization (Edman and Burk, 1998). Sea-level fluctuations and climate alter the characteristics (velocity, sediment, and location) of the fluvial run-off and source. Sediment make-up is dependent on the three prior controls and will be discussed further in the following paragraph. It is the complex interaction between all four controls that make each turbidite system unique, requiring a case-by-case approach to analysis.

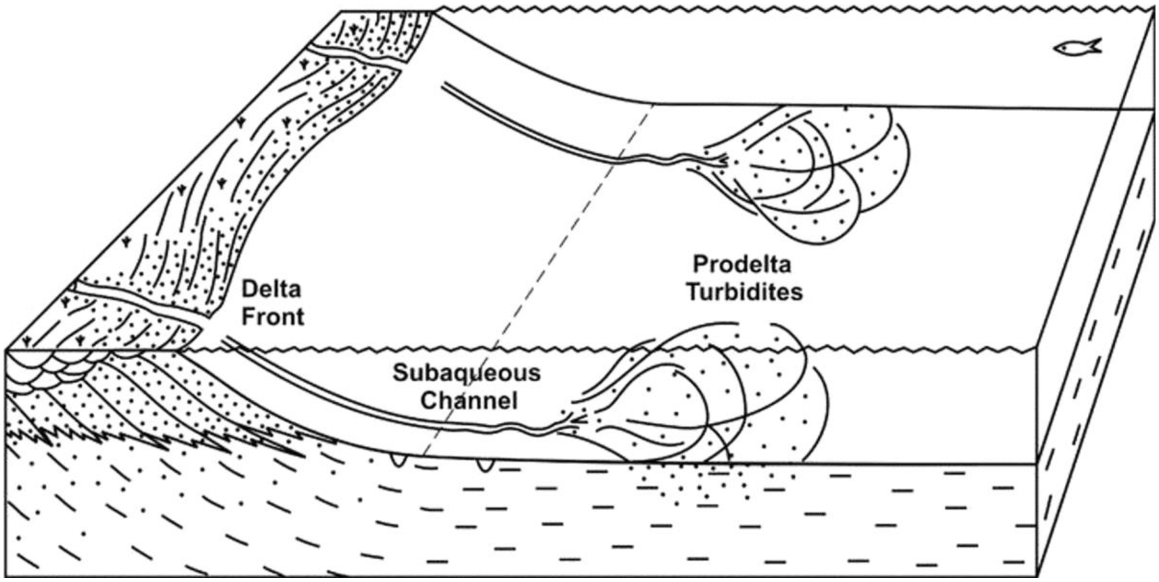


Figure 5.1 General overview of a turbidite complex (adapted from Pattison, 2005).

5.1.2 Lobster Field Geology

Lobster Field, or Ewing Bank 873, is located approximately 200 miles south of New Orleans in the Gulf of Mexico. It contains turbiditic structures and has been chosen for further study based on the completeness of the data set. The following work focuses upon the Bulminella 1 reservoir within the larger Lobster Field. This reservoir is composed of six stacked and overlapping Pliocene turbidite lobes comprising three separate compartments. The system was formed by deltaic sediment settling on a salt canopy and has received multiple hydrocarbon charges since deposition. The boundaries of the reservoir are defined by an updip pinch out of the turbidite sands and faults on the east and west sides.

Figures 5.2 and 5.3 contain a conceptual overview of the location of reservoir compartments in Lobster field. A7 is the smallest of the three compartments (Edman and Burk, 1998). It contains the oldest lobe (Lobe 5) and the lowest quality/most biodegraded oil of the reservoir. Lobe 5 is located on the western portion of the system, below Lobes 10, 20, and 30. The "West" compartment is comprised of Lobes 10, 20, and 30 and is characterized by massive sand-rich amalgamated channels and high porosity sheet sands. The "East" compartment is composed of the youngest lobes, Lobes 70 and 80. It contains notable amounts of channel and overbank deposits. The three compartments are separated by shales and do not communicate.

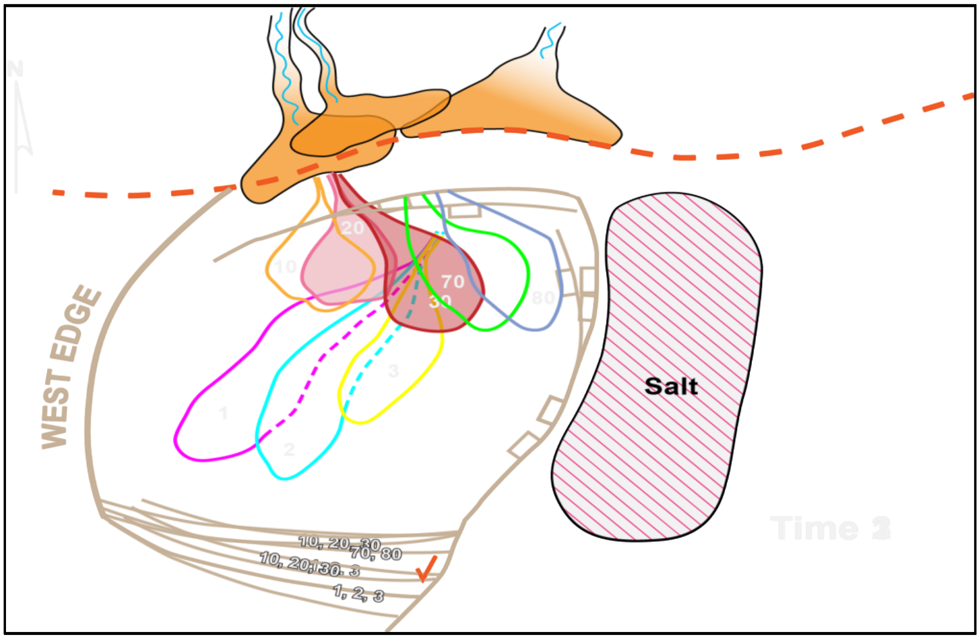


Figure 5.2 Overview of lobe deposition in Lobster Field.

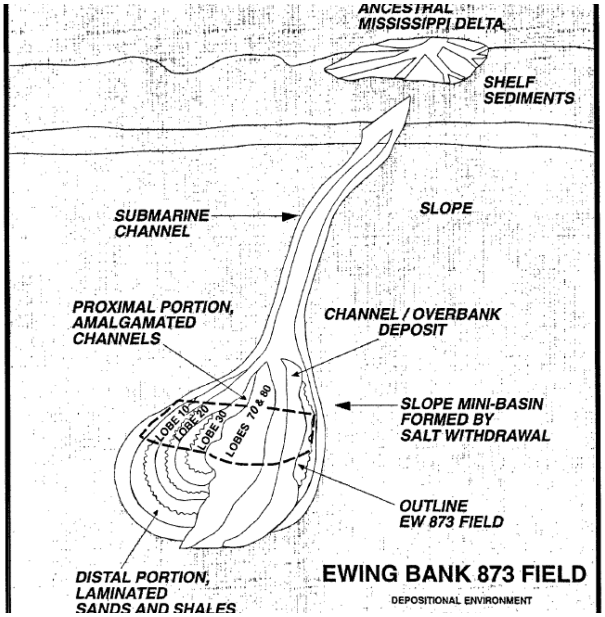

ANCESTRAL MISSISSIPPI DELTA
SHELF SEDIMENTS
SLOPE
SUBMARINE CHANNEL
PROXIMAL PORTION, AMALGAMATED CHANNELS
CHANNEL / OVERBANK DEPOSIT
SLOPE MINI-BASIN FORMED BY SALT WITHDRAWAL
LOBE 10
LOBE 20
LOBE 30
LOBES 70 & 80
OUTLINE EW 873 FIELD
DISTAL PORTION, LAMINATED SANDS AND SHALES
EWING BANK 873 FIELD
DEPOSITIONAL ENVIRONMENT

Figure 5.3 Overview of lobe deposition in Lobster Field (adapted from Edman and Burk, 1998)

5.2 Preparation of Data

Model and map creation is a vital step prior to the application of DSSIP. Seismic data and facies interpretations must be acquired and condensed into the corresponding data maps for use in the algorithm. Seismic data in the form of RMS velocity were provided for Lobes 10, 20, 30, 70, and 80. Root mean square (RMS) velocity is a form of average velocity across multiple flat layers and can be calculated using the equation below. It is typically used in CMP (Common Midpoint Gather) stacking and in the application of NMO (Normal Moveout) correction. Seismic wave velocity of a rock can be tied back to the rock properties, such as bulk modulus and density, and provides a good connection to facies location for use in DSSIP. Figure 5.4 displays the overlapping sets of RMS velocity data provided. The data was taken from a depth slice near the base of the lobes. Lobe 30 is not visible in Figure 5.4, because it is overshadowed by data from Lobes 80 and 70.

$$v_{RMS} = \sqrt{\frac{\sum_{i=1}^{n} v_i^2 * t_i}{\sum_{i=1}^{n} t_i}}$$

where:

v_i = seismic wave velocity of the ith layer

t_i = vertical two-way travel time within the ith layer

Heterogeneities within the lobes, seen in Figure 5.4, motivate the application of DSSIP at the lobe scale. Lobe 10 was chosen as the focus of the study due to the large extent of the seismic map and also because a coarse interpretation of channels based on the velocity variations is available for that lobe. Figure 5.5 shows the interpretation of channels based on the observed velocity variations. Based on this interpretation, the description of reservoir heterogeneity was separated into three categories: channel, splay, and background. At the time of deposition, the channels transported sediments within the lobe and were located throughout the center region. Splays are located near the edges of the lobe and were confined locations where sediments pooled. The two attributes are differentiated by shape and their location within the lobes rather than by a clear contrast in the RMS velocity value. Figure 5.6 shows the categorized facies interpretation map.

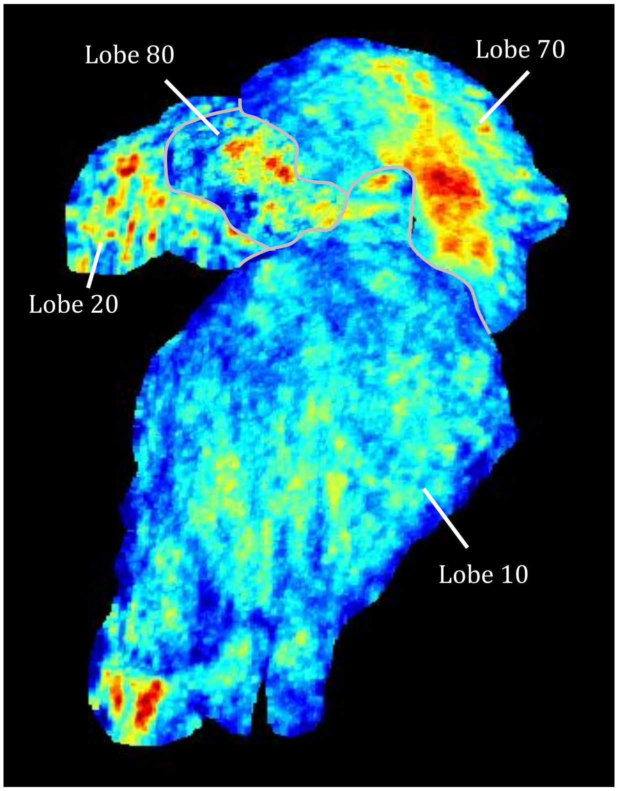


Figure 5.4 RMS velocity variations in Lobes 10, 20, 70, and 80.

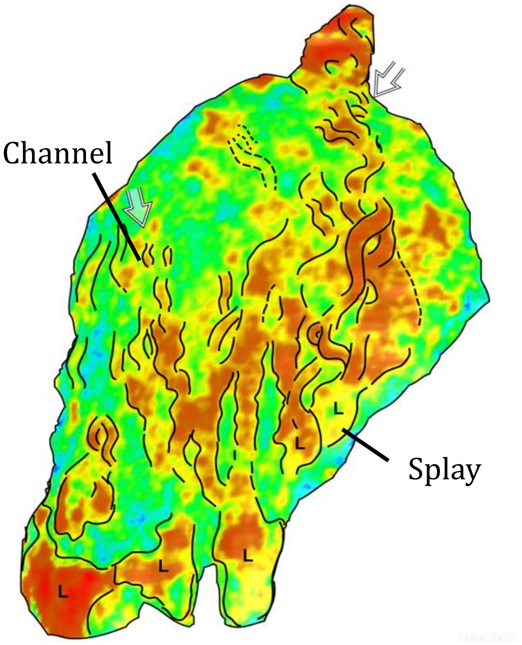


Figure 5.5 Original facies interpretation map for Lobe 10.

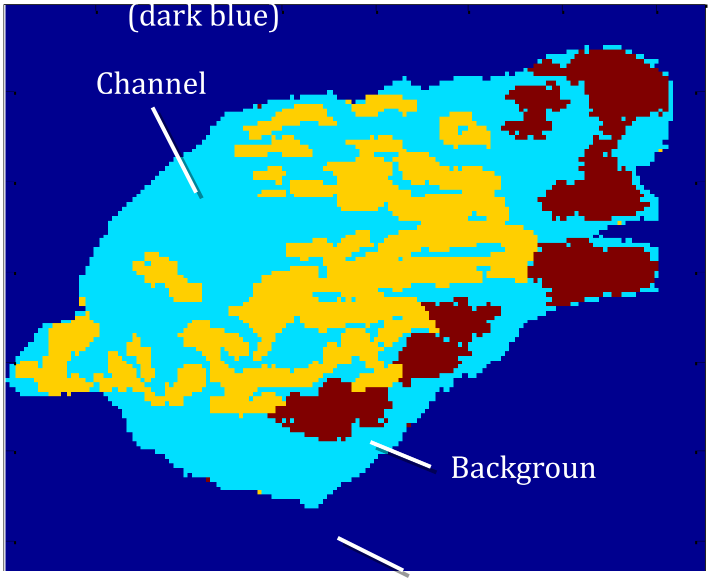


Figure 5.6 Categorized facies map for Lobe 10.

5.3 Application of DSSIP Using the Lobster Field Dataset

Figure 5.6 represents the reference facies model for the reservoir. Starting with a limited interpretation of channels based on RMS velocities, the strategy is to use it as a training model to model more detailed facies variations in the reservoir. Two modifications were implemented. In the first, conditioning data were chosen at random from the reference model in order to perform the simulation. In the second, the conditioning data in Loop 2 of the simulation was sampled so as to match the known facies proportions of analog reservoirs. The following section discusses the results of these modifications and makes recommendations for future development.

5.3.1 Adaptation of Input Data

As mentioned above, based on the channels and other facies observed in the reference model, a second interpretation was created to mimic partial knowledge of the facies locations. The data for the target reservoir model consisted of the conditioning data map (sampled from the fully categorized facies map in Figure 5.6) and the reservoir's seismic image. The training reservoir pair was composed of the partial interpretation of the original facies map and the same seismic image.

The maps in Figure 5.7 have been prepared for use in DSSIP. They have been realigned to overlap and resized to 150x150 pixel grids for efficient processing. The seismic map was imported from the data files and left unaltered after resizing.

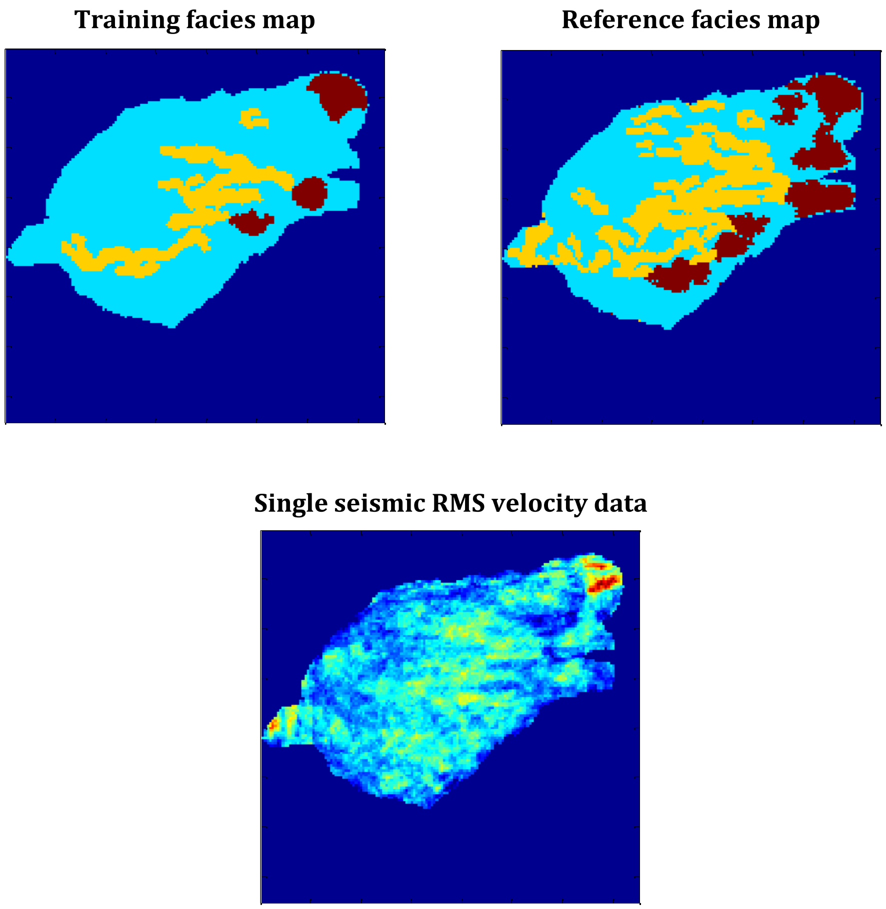


Figure 5.7 Lobe 10 seismic and facies input maps for DSSIP.

5.3.2 Results and Discussion

The DSSIP algorithm with the final set of recommended set of simulation parameters discussed in the previous chapter was implemented. The only change was to reduce the Loop 1 search path to limit the number of exact seismic location matches. The conditioning points for Loop 1 were sampled at random, but held constant across all runs. Case 2 was identical to the first except that the Loop 1 conditioning points were not randomly sampled. The conditioning points were chosen to reflect the proportions within the reference facies map. Both cases used the input maps in Figure 5.7 and started with 25 conditioning points in Loop 1.

Case 1

Case 1 results are located in Figure 5.8. The output maps do not resemble the reference facies map (target/actual) from which the initial conditioning locations were sampled. Instead, they are more similar to the training facies map. The inaccuracy is due to the inability of the first loop to identify new locations away from the established pattern locations in the partially analyzed training facies map. Our assumption was that the event pattern matching process between the identical target and training seismic maps would produce channel or splay facies matches in locations that have only been identified as background in the training map. If the algorithm starts at a "hard" data location which in the training map would be in the background facies, the subsequent steps anticipated are:

- Event pattern is transferred into the target seismic map and a matching location is searched for in the training seismic image
- After a match is made, the corresponding channel facies value is introduced into the unknown location in the sparse map.

Figure 5.9 contains a representative output map from the end of Loop 1 prior to sampling into Loop 2. It is a near match to the partially interpreted training map. The image verifies the inability of Loop 1 to reproduce the desired full facies interpretation map. An increased number of conditioning points at the start of Loop 1 would resolve this issue, because a larger number of know channel and splay locations encourages the development of these particular facies. It is necessary for the sampled "hard" data to at least reflect the target facies proportion.

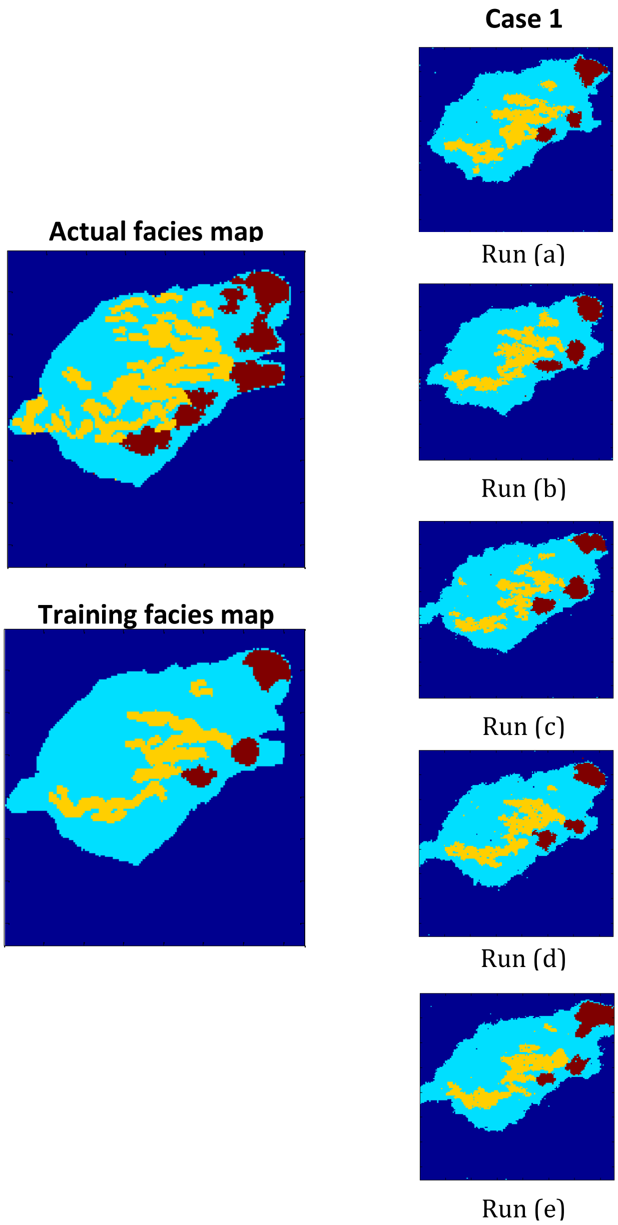


Figure 5.8 Reservoir models obtained by application of the DSSIP process. The “hard” conditioning data at the start of Loop 1 does not reflect the facies proportion in the reference reservoir model (Case 1).

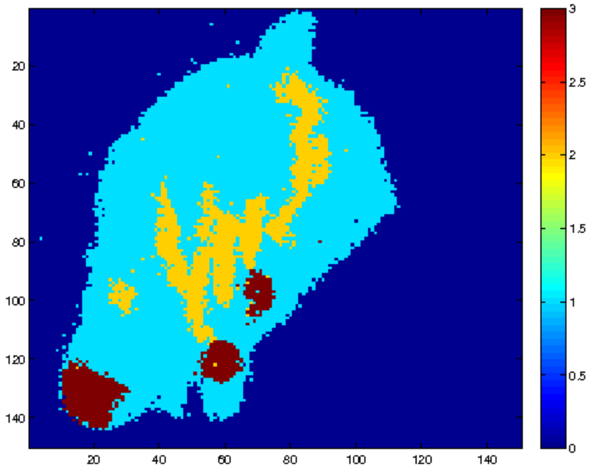


Figure 5.9 Output map (Case 1) at the end of Loop 1 prior to sampling and starting Loop 2.

Case 2

Case 2 addresses the issue of the conditioning data not reflecting the target proportions. In this case, the Loop 1 conditioning points were randomly sampled according to the facies proportion reflected in the target facies map. Facies proportions in the Loop 1 conditioning data and target facies map were identical across runs. Case 2 results are located in Figure 5.10. Again, the output maps resemble the partial interpretation and do not correct the issue of incorrect facies proportion in the final output map.

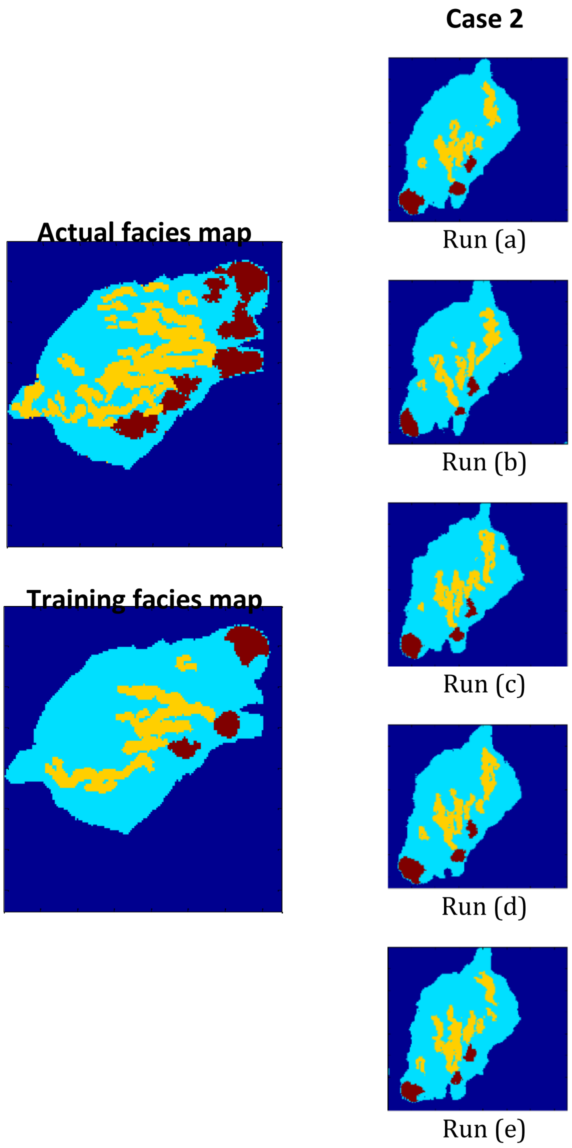


Figure 5.10 Reservoir models obtained by implementing DSSIP using "hard" data that reflect the facies proportion in the reference reservoir model (Case 2).

5.3.3 Recommended Modifications

The results from Cases 1 and 2 show that while the DSSIP algorithm is able to model reservoir objects that exhibit complex connectivity and place them in the correct locations as reflected by the seismic, in this current case, because both the training and simulation are performed using the same data set, the final simulated image resembles the training model too closely. However, modifications to the algorithm may provide the proper changes to correct these issues. Currently, there is no way to force the target facies proportion in the reservoir models and consequently, the simulation ends up bearing too much resemblance to the training model. A weight system could be added into the algorithm using which the simulation at a node would favor the selection of certain facies, such as the channels and splays based on the observed proportions.

Another change would be to specify the location of the search path rather than just limiting the length. Limiting the search window to a certain region of the training reservoir that has been fully evaluated would produce facies patterns similar to those seen in the training image. The search path order should still be random after each iteration to ensure unbiased pattern matching.

5.4 ENSEMBLE DIRECT SAMPLING SEISMIC INTEGRATION PROCESS

One of the problems with the DSSIP algorithm described in the previous section is that it requires a correspondence between the training reservoir model and the seismic data set. As mentioned previously, it is very difficult to have a matching pair between the training model and corresponding seismic maps. In the following section, a modification to the DSSIP algorithm is presented that takes into consideration the available "hard" data for the reservoir and a generic training image in order to obtain an ensemble of reservoir models. The available seismic data for the reservoir is then used to refine the prior ensemble in order to find a posterior set of models that exhibit the patterns similar to that observed in the seismic. This modified algorithm is described next.

5.4.1 Description of Ensemble DSSIP

A suite of training images is created from the original partial interpretation map and the conditioning data locations using direct sampling. Next, an event pattern is created at an unknown location in the sparse map containing the conditioning data. The corresponding seismic values are recorded and the same seismic image is searched for a matching location. Once a matching location is found, the new center node location is transferred to each facies training image in the suite and the corresponding facies values are recorded. The highest frequency facies value is calculated and returned to the original unknown location in the sparse map. This process is repeated for each unknown location until the sparse map is filled.

5.4.2 Results and Discussion

Figures 5.11 and 5.12 contain the results of Ensemble DSSIP. Figure 5.11 displays the suite of training maps generated during the first loop of the process. As previously discussed, they are created using the main training image and the conditioning data sampled from the target field. The final output of the process is the set of probability maps located in Figure 5.12.

The process showed results very similar to that of the original modifications to DSSIP seen in Section 5.3. The probability maps do not show an increased presence of the channel or splay facies. Again, they reproduce the training facies map rather the target reservoir that was conditioned. This is because, in the direct sampling procedure there is no way to dial in a required facies proportion in the algorithm. The reproduced facies proportion in the ensemble of initial models reflects the facies proportion of the training model even though the facies proportion indicated by the well conditioning data sampled from the reference are quite different. This problem can be rectified by using another algorithm such as *filtersim* to generate the initial ensemble. The algorithm filtersim has a mechanism for imposing the target facies proportions.

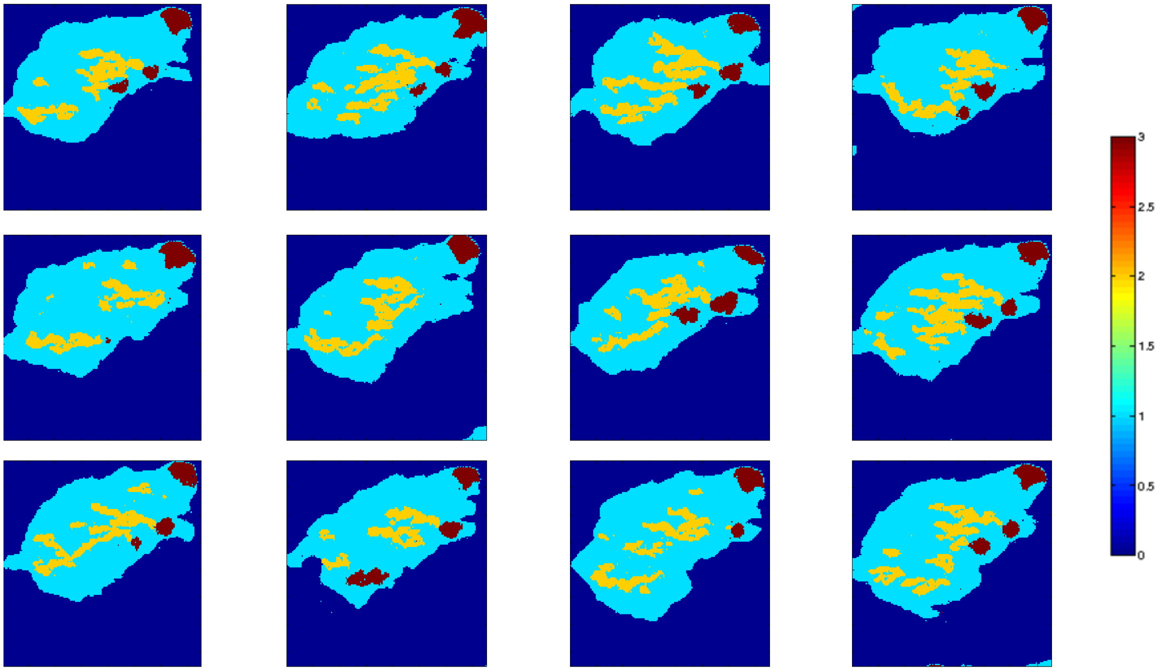
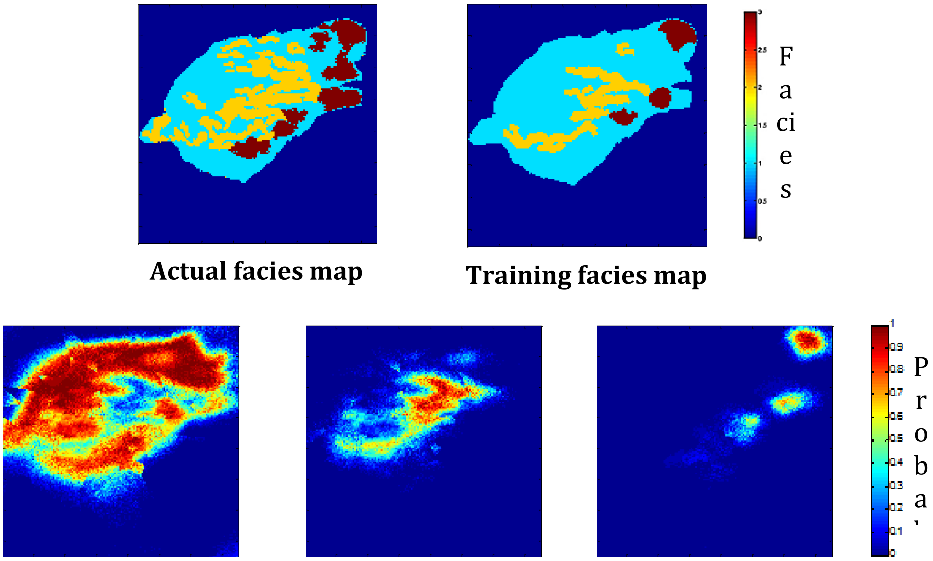


Figure 5.11 Selected training images developed during first loop of Ensemble DSSIP.



Actual facies map

Training facies map

Background

Channel

Splay

Figure 5.12 Probability maps produced as an output of Ensemble DSSIP.

Chapter 6: Data Preparation for DSSIP

DSSIP requires extensive knowledge of reservoir geology and subsurface attributes prior to applying the algorithm. An opportunity was presented to applying DSSIP to model the spatial variations of a reservoir in the Gulf of Mexico. However, prior to application of the DSSIP algorithm it was necessary to analyze the data in order to identify the main reservoir facies. In the following study, data preparation techniques were applied to the real data set in order to produce maps and conditioning data for application of the DSSIP algorithm. Internal company reports have provided an exhaustive data set for the two fields to be used for the study. The training reservoir will be referred to as Field A, and the target reservoir will be referred to as Field B. The data set contains log data, core analysis, seismic volumes, and petrophysical data. In this chapter, we present the data analysis that was performed prior to application of the DSSIP algorithm.

6.1 Geology in Field A and Field B

The evaluation focused on the two fields located in the southern portion of the Gulf of Mexico. Both fields are part of a large complex of oolitic structures that comprise the Kimmeridgian play in the Mexican part of the Gulf of Mexico. The play has been manipulated by salt diapirism and dolomitization due to subaerial diagenesis, which has led to the formation of a large anticline and the entrapment of hydrocarbons. The fields are adjacent and are both categorized in the same geologic subsection of the oolitic structure. Field A has been more extensively studied and has been in production for a longer period of time than Field B.

6.1.1 Classification of the Local Region Based on Oolitic Structure

The Kimmeridgian play is composed of a large oolitic structure comprehensively studied in internal sedimentology reports provided. Oolites are sedimentary rocks comprised of ooids, which originate as small spherical grains of concentric layers of calcite. The calcite can later be replaced by other minerals such as dolomite, as is the case in parts of the Kimmeridgian play. It has been reported that the productivity in this region is strongly determined by the extent of dolomitization exhibited by the carbonate facies. Therefore, mapping spatial variations in dolomitic facies was an added focus during facies classification. Oolitic banks form in tropical climates and highly energetic, shallow environments. The structures in oolitic banks are not homogeneous and require multiple sections to define the geology.

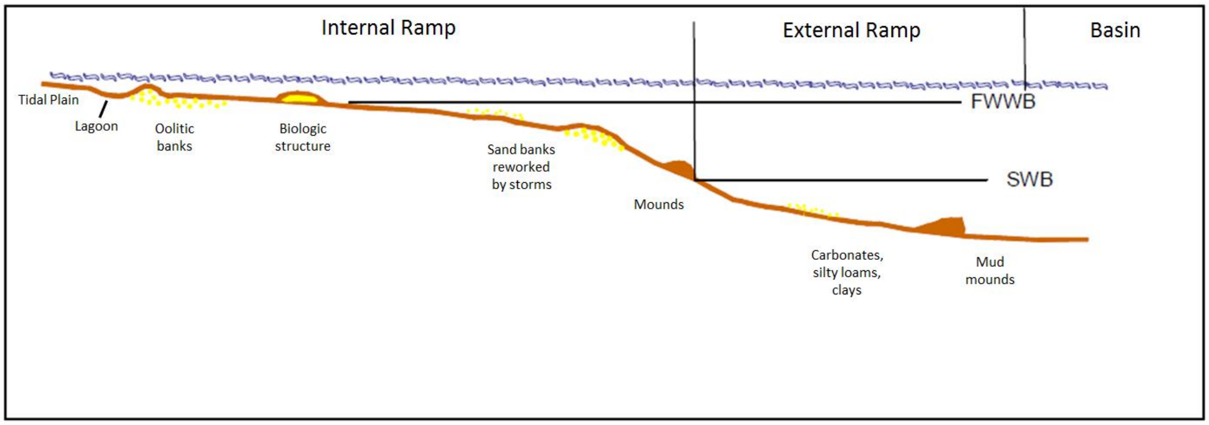


Figure 6.1 Diagram of an oolitic bank complex at the time of deposition.

The oolitic banks of the Kimmeridgian Play have been divided into four large sections that describe the paleoenvironments: external ramp, internal ramp, lagoon, and restricted lagoon. These categories are based on the position on the ramp where they are formed and are differentiated by facies present and formation mechanism. The lagoon and restricted lagoon are in the eastern half of the formation and are composed of a wide range of facies including dolomite, slightly calcareous shale, mudstone, and sandy siltstone. The external ramp is on the western most edge of the play and was formed in the deepest part of the carbonate ramp. This environment contains argillaceous mudstone with concentrations of calcareous shale. The internal ramp is located between the lagoon and the external ramp and is the main area of interest for the Kimmeridgian Play in the Field A-Field B complex. It is the most productive area of the formation and contains the most oolitic bank development.

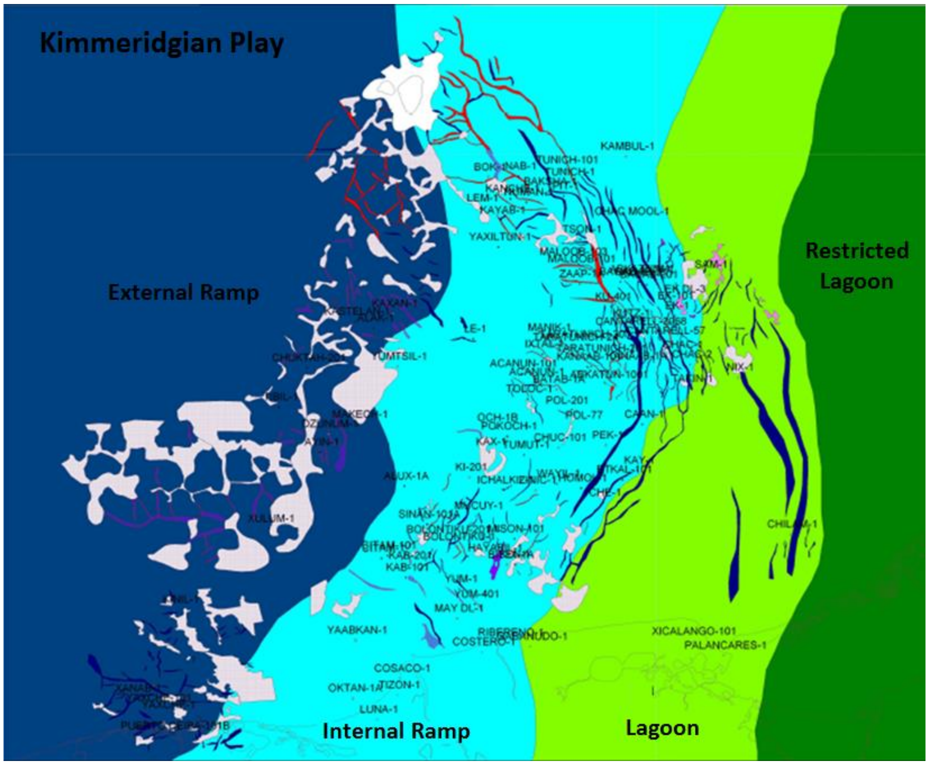


Figure 6.2 Paleoenvironments of the Kimmeridgian Play.

The internal ramp has been subdivided into separate lithofacies to provide further geologic clarification: front bank, oolitic bank, and inside bank. All three lithofacies contain oolitic deposits but can differ greatly in composition and productivity. The front bank is nearest the external ramp and is composed of wackestone-packstone sediments containing slightly dolomitized peloids with interbedded shales. The oolitic bank is in the middle of the internal ramp. It contains packstone-grainstone with ooids and minor amounts of dolomitized ooids. The inside bank is generally on the border between the internal ramp and lagoon and is largely composed of dolomites with a small presence of ooids and interbedded shales.

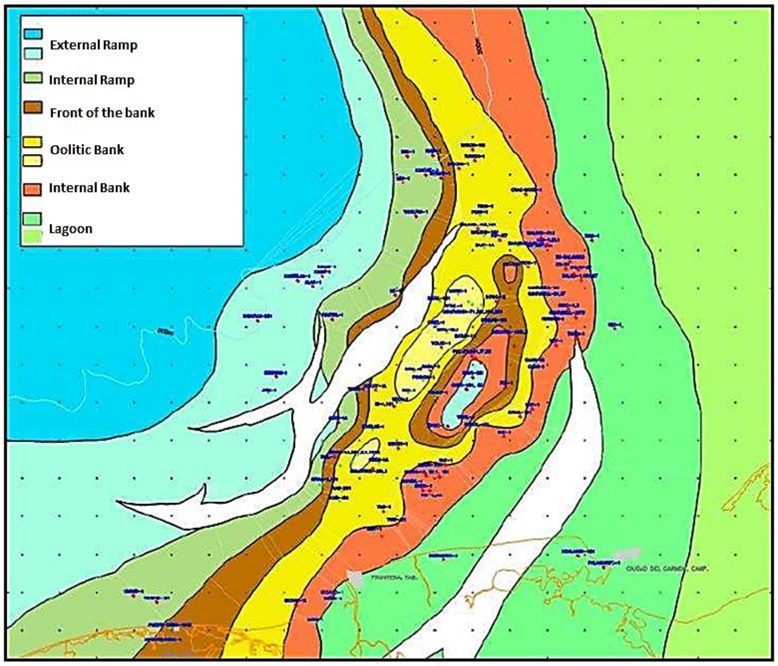


Figure 6.3 Lithofacies present within the Kimmeridgian Play.

6.1.2 Field A and B Classification

Fields A and B are located in the internal ramp, more specifically within the inside bank lithofacies. As described above, dolomitization is an important feature of these internal bank lithofacies. The interval of interest in the reservoirs is within the Jurassic Kimmeridgian (JSK) geologic zone. This zone is defined by an Oxfordian base and Tithonian ceiling. The size of the Kimmeridgian producing section in the Field A is roughly 12.9 km^2.

In Field A, five distinct facies have been identified within the desired Jurassic horizon, from F1 at the top to F5 at the bottom. F3 is the most productive subinterval and is composed of dolomitized oolites and dolomites with a small amount of dolomitized mudstone. F1 and F2 contain partially dolomitized mudstone and packstone, the only difference being a larger clay content in F2. F4 is composed of mudstone with interspersed shale layers. F5 consists of interbedded mudstone, wackestone, and packstone facies with a small amount of grainstone of oolites and, to a lesser extent, shale intervals. Lateral continuity between the fifteen wells in Field A has been shown and studied. Field B has not been studied as thoroughly as Field A and has only nine wells from which information can be gathered.

6.2 Well Log Preparation and Analysis

The goal of the well log analysis was to define the internal facies variations within Fields A and B and to locate the desired facies. Extensive log information was coupled with sparse core facies descriptions to generate facies proportion and dolomitization estimations. The initial work focused upon using established log cross plots to identify variations in lithology. This approach was followed by multivariate statistical analysis of the well logs through the application of principal component analysis (PCA) and neural networks.

6.2.1 Cross-plot Analysis

A common method in well log analysis is to cross-plot log properties in order to determine the lithology and porosity of a sample. Lithology is identified by comparing to cross-plot curves for known geologies (Zimmerle, 1995). Two such cross-plots were used to evaluate geologies in Fields A and B: the neutron-bulk density cross-plot and the sonic-neutron porosity cross-plot.

Figures 6.4 and 6.5 exhibit the neutron-bulk density and sonic-neutron porosity cross plots at the F3 interval for several wells in Field A. Only six locations were plotted for Field A due to a limited number of wells containing the appropriate log measurements. In both cross-plots, the log values for the Field A – Well 58, Well 38, Well 7, and Well 2 wells track the dolomite curve indicating that the F3 interval at these locations is more dolomitized. All the plots suggest a carbonate facies with very little clastic remnants, agreeing with prior geologic knowledge of the reservoir.

Figures 6.6 and 6.7 display the neutron-bulk density and sonic-neutron porosity cross plots for the Field B – Well 1 and Field B – Well 12 wells within the F3 interval. Again, available log measurements limited the scope of cross-plot application to only a fraction of the total wells. The plots show that the two wells are not as heavily dolomitized as those in Field A.

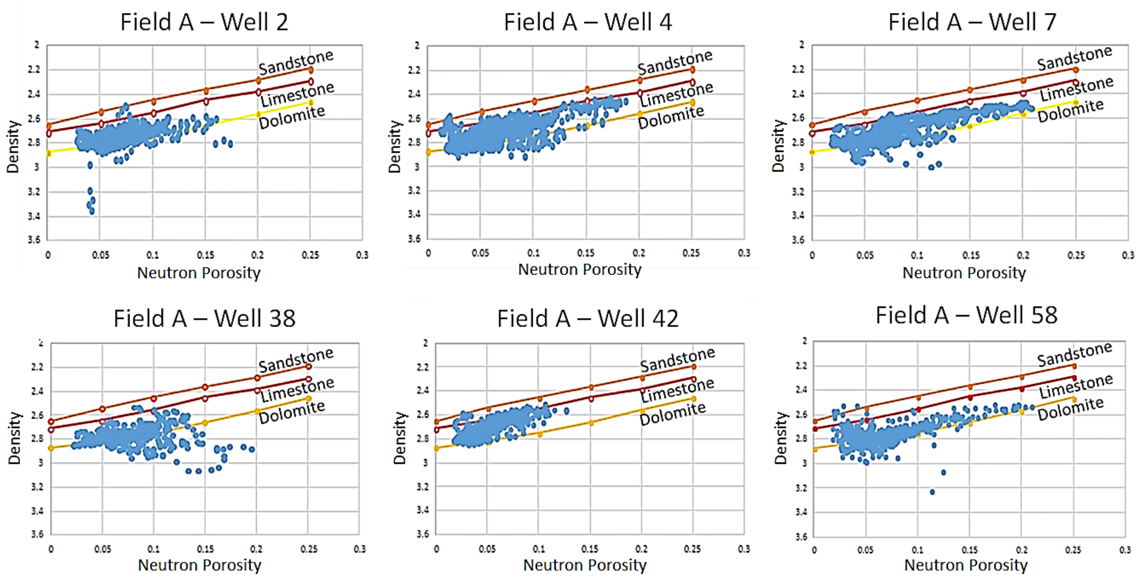


Figure 6.4 Neutron-density cross-plots within F3 interval in Field A.

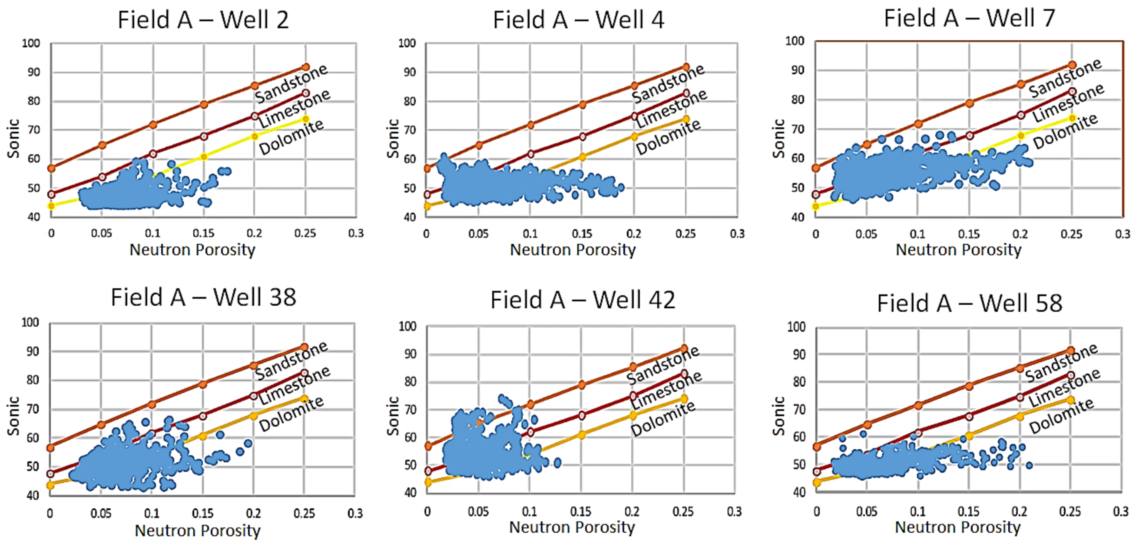


Figure 6.5 Sonic-neutron porosity cross-plots within F3 interval in Field A.

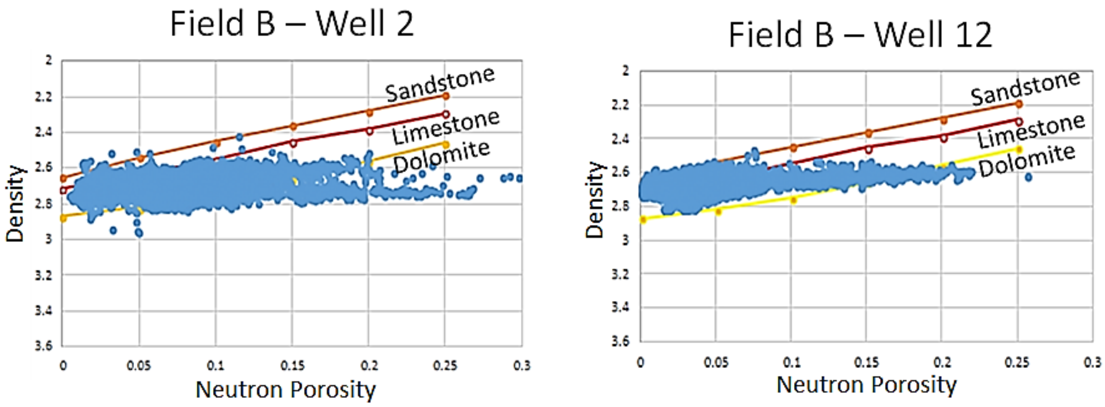


Figure 6.6 Neutron-density cross-plots within F3 interval in Field B.

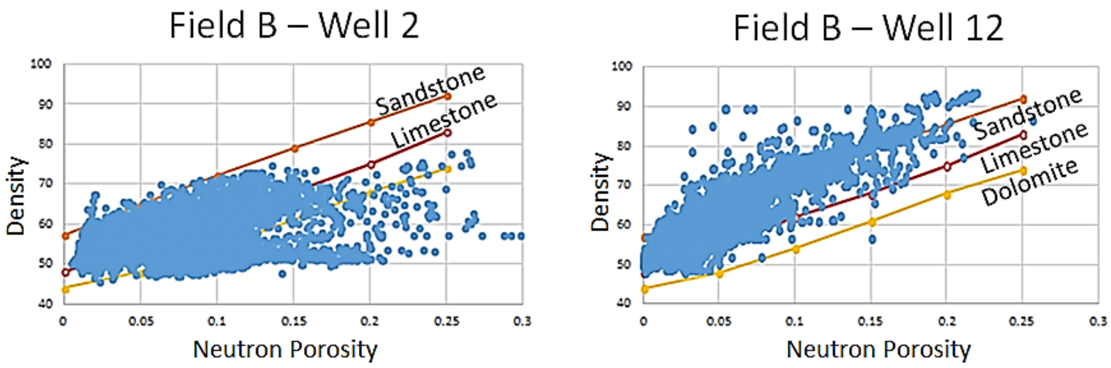


Figure 6.7 Sonic-neutron porosity cross-plots within F3 interval in Field B.

6.2.2 Principal Component Analysis

Cross plot analyses provide a broad indication of whether a dolomitized rock facies is present. More details about the facies variations indicated by logs may be obtained by performing multivariate statistical analysis. One of these methods, principal component analysis (PCA), reduces the complexity of a problem using the inherent interdependencies in the data. Classification performed after PCA can then be compared to data from core analysis.

The process is based on sample statistics and to that extent, blind to geologic interpretation and requires that the log attributes used for PCA be available at all wells. PCA was applied to the well logs to obtain a clustering of data locations and then compared against the geologic interpretations from core analysis to develop lithological correlations. Facies categorization can be assigned to each cluster once these correlations have been created.

Background on Principal Component Analysis

Principal component analysis calculates the normalized covariance matrix of a data set and reduces the matrix into a set of eigenvalues and corresponding eigenvectors (Joliffe, 2002). The vectors corresponding to the highest eigenvalues are noted as the principal components, and the vectors corresponding to the lowest eigenvalues are discarded. The higher the amount of redundancy in the data set, fewer will be the number of principal components required to describe the data. In this fashion, a large number of attributes can be condensed down to a core number of eigenvectors with minimal loss of information.

Transforming a new data set into the principal component space yields a set of principal scores. The principal scores are a projection of this data onto the principal components. After the principal scores are calculated, k-means clustering is utilized to categorize the data into separate groupings. In this case, our conjecture is that the clusters correspond to facies groupings that can then be compared to actual core evaluations for validation.

Application of PCA on the F3 Facies within Field A

An attempt was made to obtain facies indicators using PCA for the F3 facies interval across all wells in Field A. Principal component analyses were conducted on each well separately. In the Field A, seven wells contained the F3 facies, had adequate core data, and the proper number of log attributes to conduct the analysis (Field A – Well 2, Well 3, Well 4, Well 7, Well 38, Well 42, and Well 58). An example of an output principal score plot for Field A – Well 58 prior to the clustering process is located in Figure 6.8. The final output of the process for Field A – 58 after clustering has been presented in Figure 6.9.

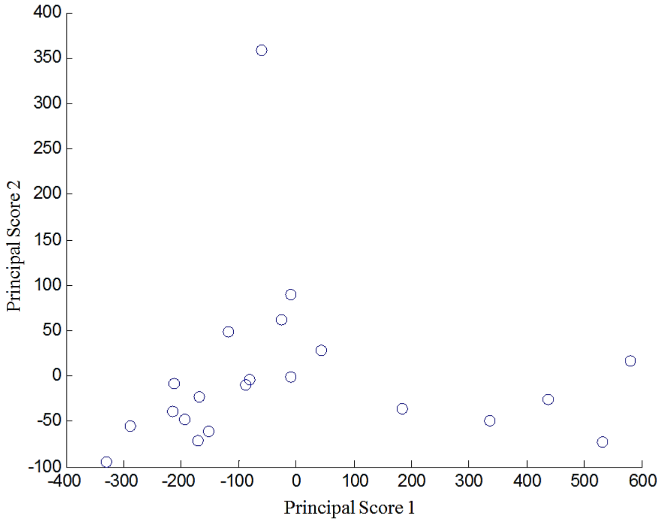


Figure 6.8 Principal scores prior to clustering process for the F3 interval within Field A – Well 58.

Clustering is a method to create groups of objects such that the profiles of objects in the same cluster are very similar. K-means clustering was utilized to group the scores seen in Figure 6.8. K-means clustering treats each data point as a location in space and finds a partition between clusters such that objects within each cluster are as close as possible. Silhouette values are calculated to determine the appropriate number of clusters after k-means clustering. The silhouette value is a measure of how close a point in one cluster is to points in neighboring clusters. Higher silhouette values correspond to tightly grouped clusters. The silhouette scores produced during the analysis of Field A suggested four facies clusters. The color bands in Figure 6.9 correspond to the four clusters specified during PCA. At this point, they have not been correlated to core data.

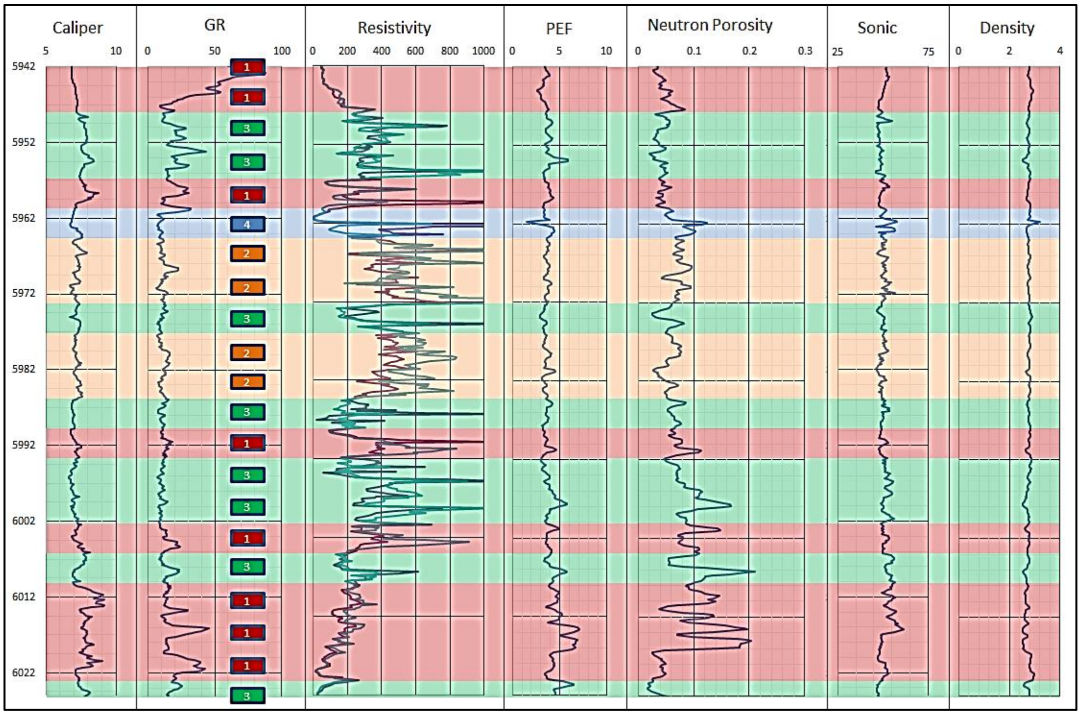


Figure 6.9 PCA and cluster results overlaying multiple log attributes for the F3 interval within Field A – Well 58.

Application to Core N2 in Field A – Well 2

After conducting PCA application on only the F3 interval, PCA was then applied across all known geologic points in Field A. Again, the goal of the analysis was to develop a clustering model that accurately matched known lithology groupings so that it could be applied to points with unknown geologies.

The interval of Field A – Well 2 – Core N2 (6187.04 – 6193.23m) was used to create a standard set of criteria prior to conducting the joint principal component analysis. The core was chosen because it contained three representative lithofacies, minor dolomitization, and extensive log attributes. Table 6.2 contains the results of PCA alongside the core geology interpretations. The team conducted the analysis using the gamma ray, deep resistivity, neutron porosity, density, and sonic log attributes. The eigenvalue threshold was set at 95% to determine the appropriate number of principal scores to use. In this case, the first two principal scores reached the threshold as seen in Table 6.1.

The results indicate that it is very difficult to capture the actual transitions in geology present in the core sample. Similar results were obtained for two other core samples Field A – Well 3 – N3 and Field A – Well 4 – N3C. PCA was unsuccessful over the regions tested. It was unable to adequately predict the geologic facies.

Table 6.1 Eigenvalues for PCA in Field A – Well 2 – Core N2.

Eigen values	**% of total EV value**
13414.47	0.999292
7.044588	0.999816
2.464705	1
0.001029	1
3.23E-05	1

Table 6.2 Facies comparison for the PCA process in Field A – Well 2 – Core N2.

Depth	GR	NP	Density	Resistivity (deep)	Sonic	PS 1	PS 2	Cluster Value	Geology
6187.04	16.0	0.0494	2.7953	181.776	45.4	8.98507	2.28571	3	Grainstone
6187.25	14.173	0.0628	2.7912	188.615	47.6355	6.30694	3.36253	3	Grainstone
6187.50	10.35	0.0639	2.7953	216.439	48.44	2.56989	2.22203	1	Grainstone
6187.75	5.8620	0.0658	2.7801	286.751	49.7224	-1.97969	1.17960	2	Grainstone
6188.27	5.6009	0.0665	2.7391	383.198	47.3578	-1.06791	-1.01748	1	Mudstone
6188.48	5.6168	0.067	2.7432	433.029	46.7297	-0.75088	-1.55994	1	Mudstone
6188.75	5.0901	0.0673	2.7114	453.323	48.0595	-1.85418	-0.64937	2	Mudstone
6189.01	4.7677	0.0723	2.6569	448.464	47.4941	-1.86439	-1.30017	2	Grainstone
6189.25	4.3477	0.0844	2.6099	453.322	49.4655	-3.18403	0.22387	2	Grainstone
6189.50	3.9209	0.0911	2.6161	457.994	52.0300	-4.79515	2.26410	2	Grainstone
6189.75	3.9776	0.0834	2.6719	476.618	50.8620	-4.18125	1.26845	2	Grainstone
6190.12	5.3467	0.0728	2.7199	452.979	49.3834	-2.26815	0.63403	2	Grainstone
6190.25	5.3467	0.0728	2.7199	452.979	49.3834	-2.26815	0.63403	2	Grainstone
6190.76	6.66	0.0727	2.7262	472.372	45.53	0.74140	-2.10733	1	Grainstone
6190.99	6.36	0.0774	2.7302	485.001	46.18	0.16506	-1.68276	1	Grainstone
6191.25	6.0823	0.0825	2.7397	504.662	47.5232	-0.72615	-0.64028	1	Grainstone
6191.73	6.6135	0.0723	2.7495	416.543	46.3657	0.29772	-1.39779	1	Floatstone
6192.04	7.4467	0.0604	2.768	319.382	44.4072	1.97269	-2.71115	1	Grainstone
6192.26	8.575	0.0444	2.7805	250.09	44.140	3.08998	-2.40077	1	Grainstone
6192.51	8.58	0.0428	2.7804	240.315	44.06	3.13296	-2.46843	1	Grainstone
6193.23	6.8564	0.0878	2.7653	184.254	52.2360	-2.32184	3.86108	2	Mudstone

6.2.3 Neural Network Application

Artificial neural networks utilize non-linear regression methods to create a training network. The training network learns from a set of input data with known outputs and can be used to predict outcomes from another set of input data. Neural networks can be generated using differing configurations and functions, but, at their most basic, they can be thought of as consisting of three layers: input, transform, and output. Training data sets shape the learning transform (neurons) over a series of repeating tries until a satisfying rule set is created. In this application, the training phase develops a relationship between the multiple log attributes and the corresponding facies at known locations. The trained neural network is then applied to a full set of log attributes at all locations, and the resulting outcomes are the predicted facies. The validation step is important to ensure that the network is not overfitted.

Neural Network Training Algorithm

Ten wells in Field A were evaluated using neural networks. The networks were trained using the data available for selected intervals from the seven wells used during PCA (Field A – Well 2, Well 3, Well 4, Well 7, Well 38, Well 42, and Well 58). The size of the training data set was limited by the amount of geologically evaluated locations. They were trained using the Neural Network Toolbox within the Matlab software suite. Several different neural network models were created:

1. Detection of dolomite
2. Detection of partial dolomitization
3. Detection of mudstone
4. Detection of grainstone,
5. General network for simultaneous detection of packstone, wackestone, mudstone and grainstone

All the networks, except for the general network, used a single training network with only one indicator output. They all used ten hidden nodes and the Levenberg-Marquardt method (Pujol, 2007), a widely accepted training function and the default training algorithm within the Neural Network Toolbox. The training data set consisted of 100 locations that have both known geology and the required log attributes. The input attributes required are the sonic, gamma ray, deep resistivity, neutron porosity, PEF, and density log values.

The dolomite, mudstone, and grainstone networks identify only the locations of pure dolomite, mudstone, or grainstone, respectively, within the data set. The partial dolomitization filter can separate sources that have only been partially dolomitized from fully dolomitized facies. The general neural network is a multi-layered network created by combining the results of multiple neural networks to identify all types of geologies. The steps of the algorithm are specified below in Figure 6.10. The general neural network delineates between seven geologies: dolomite, mudstone, wackestone, packstone, grainstone, dolomitized mudstone, and dolomitized packstone. It uses four separately trained networks to filter out different geologies in a hierarchical fashion.

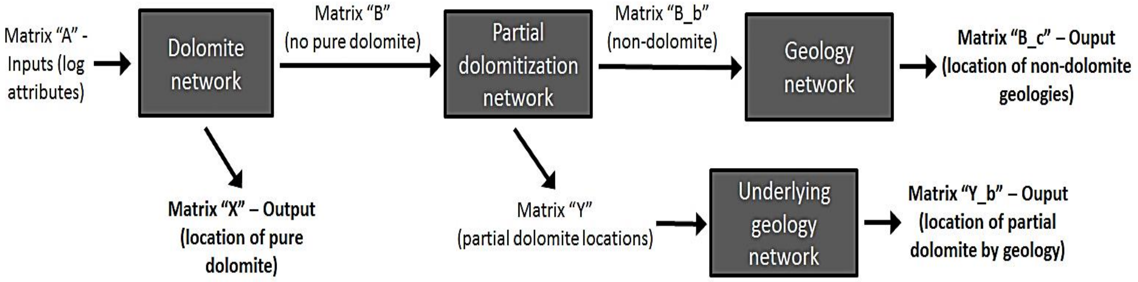

Matrix "A" - Inputs (log attributes)
Dolomite network
Matrix "B" (no pure dolomite)
Partial dolomitization network
Matrix "B_b" (non-dolomite)
Geology network
Matrix "B_c" – Ouput (location of non-dolomite geologies)
Matrix "X" – Output (location of pure dolomite)
Matrix "Y" (partial dolomite locations)
Underlying geology network
Matrix "Y_b" – Ouput (location of partial dolomite by geology)

Figure 6.10 Flow diagram of the general neural network algorithm.

Network Training Results

All four trained neural networks have a high degree of accuracy in facies estimation. Table 6.3 summarizes the results for the final neural networks. In each case, several iterations were required before resolving the most representative neural network.

The most accurate networks are the four that identify the lithologies - dolomite, partial dolomite, grainstone, and mudstone.

The results for the general network have been broken down into the four different component networks. As a whole, it has the lowest accuracy as compared to the previous neural networks. Therefore, when only knowledge on the location of dolomite is required, it is faster and more accurate to use the binary networks. However, the general network will still provide a good estimation of the overall geology in instances when facies pattern studies is desired.

Table 6.3 Summary of trained neural network accuracy.

Network Type	**Accuracy**	
	Training	**Validation**
Dolomite	94.0%	93.0%
Grainstone	83.3%	95.0%
Mudstone	89.7%	100.0%
General	-	-
Identify pure dolomite	94.0%	93.0%
Partial dolomitization identification	96.0%	90.0%
Identify geology under partial dolo.	100.0%	100.0%
Identify non-dolomite facies	89.0%	80.0%

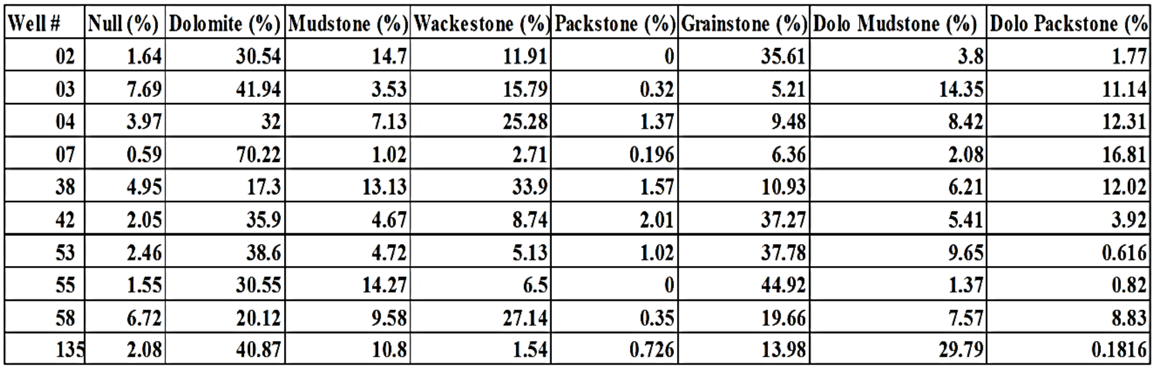
Neural Network Based Prediction of Facies Variation

After training, the networks were applied across all data points in Field A containing the required suite of log data. This new data set was composed of information from ten wells with measurements at approximately 14,000 unique depths (observations).

The output data was used to conduct the interpretation and analysis in the following section.

The size of the training data set was significantly smaller than the prediction data set to which the neural networks were applied. This problem could not be avoided due to the limited number of data points with detailed geologic classification based on core data. A limited number of training data points could potentially create a lack of geologic diversity and, consequently, prediction inaccuracy. However, the wide range of locations and geologies present in the current training set counteract some of these concerns. Table 6.4 summarizes the percentages of the lithologies in each well obtained after application of the trained neural network model.

Table 6.4 Neural network estimated facies proportion by well.



Well #	Null (%)	Dolomite (%)	Mudstone (%)	Wackestone (%)	Packstone (%)	Grainstone (%)	Dolo Mudstone (%)	Dolo Packstone (%
02	1.64	30.54	14.7	11.91	0	35.61	3.8	1.77
03	7.69	41.94	3.53	15.79	0.32	5.21	14.35	11.14
04	3.97	32	7.13	25.28	1.37	9.48	8.42	12.31
07	0.59	70.22	1.02	2.71	0.196	6.36	2.08	16.81
38	4.95	17.3	13.13	33.9	1.57	10.93	6.21	12.02
42	2.05	35.9	4.67	8.74	2.01	37.27	5.41	3.92
53	2.46	38.6	4.72	5.13	1.02	37.78	9.65	0.616
55	1.55	30.55	14.27	6.5	0	44.92	1.37	0.82
58	6.72	20.12	9.58	27.14	0.35	19.66	7.57	8.83
135	2.08	40.87	10.8	1.54	0.726	13.98	29.79	0.1816

6.3 DATA EXTRACTION FROM FIELD B SEISMIC VOLUME

The next step was to investigate the seismic data sets for the reservoirs and delineate the information in that data. Six attribute volumes were provided for Field B. The attribute volumes contain inversion results for dolomitization, acoustic impedance, amount of limestone, density, amount of clay, and the ratio of compressional to shear velocity. The following section details some of the analysis that has been done using the data.

6.3.1 Covariance Matrices and Correlation Coefficients

The similarity between the attribute volumes was assessed by computing the covariance matrix between pairs of attributes. Covariance is a measure of similarity between multiple objects, seen in the equation below. The final output of the equation is a measurement of the similarity or dissimilarity of two attribute arrays, *X* & *Y*.

$$Cov\ (X, Y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n - 1}$$

where:

X & Y = two independent variables (vectors)
x_i & y_i = individual components at location i
$\bar{x}$ & $\overline{y}$ = means of X & Y
n = number of components in X & Y

In order to perform this computation, a time slice was extracted from each volume at a specified time. Figures 6.11 – 6.13 demonstrate the extraction process from the seismic volume to the final input matrix. Figure 6.11 shows the 3D visualization of the acoustic impedance attribute volume. The horizontal slice in Figure 6.11 is the time slice seen in Figure 6.12 prior to any intensity value manipulation. Figure 6.12 displays the time slice after the values have been normalized for comparison to other attribute volumes. A smaller portion of the time slice was used to reduce computational time. Figure 6.13 contains this specific portion of the time slice used in the final similarity calculations. Petrel was used for visualization of the seismic volumes and Matlab was utilized to process calculations and intensity manipulation.

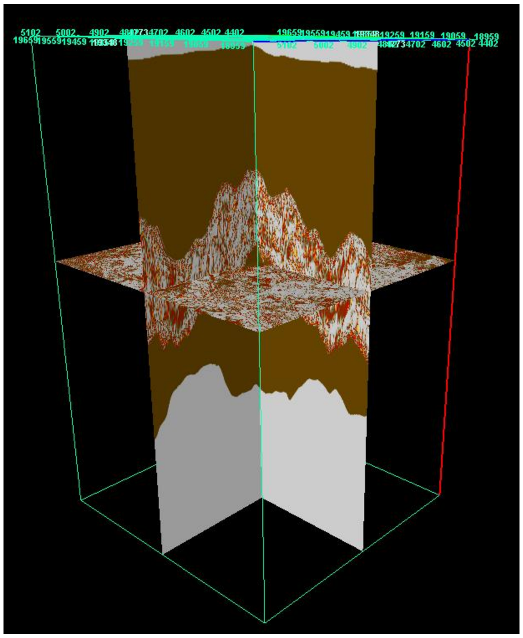


Figure 6.11 3D visualization of the acoustic impedance attribute volume.

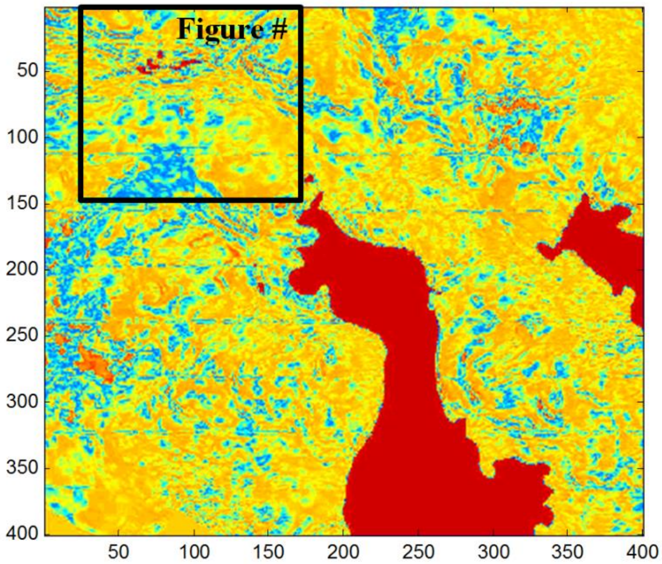


Figure 6.12 Acoustic impedance time slice after intensity manipulation.

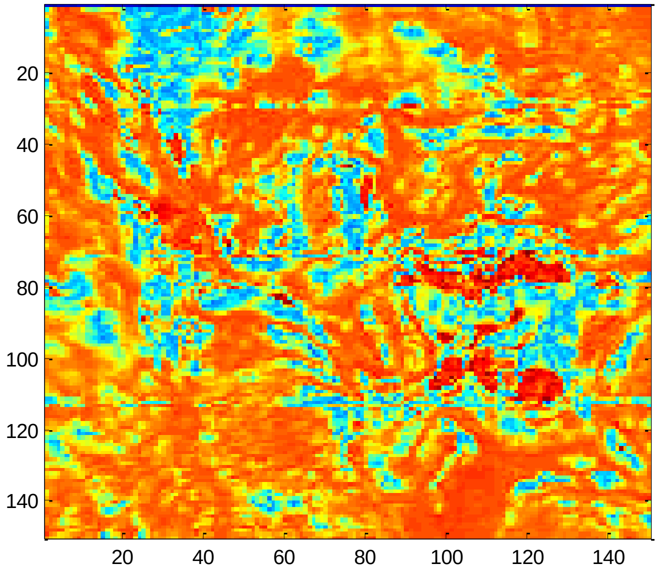
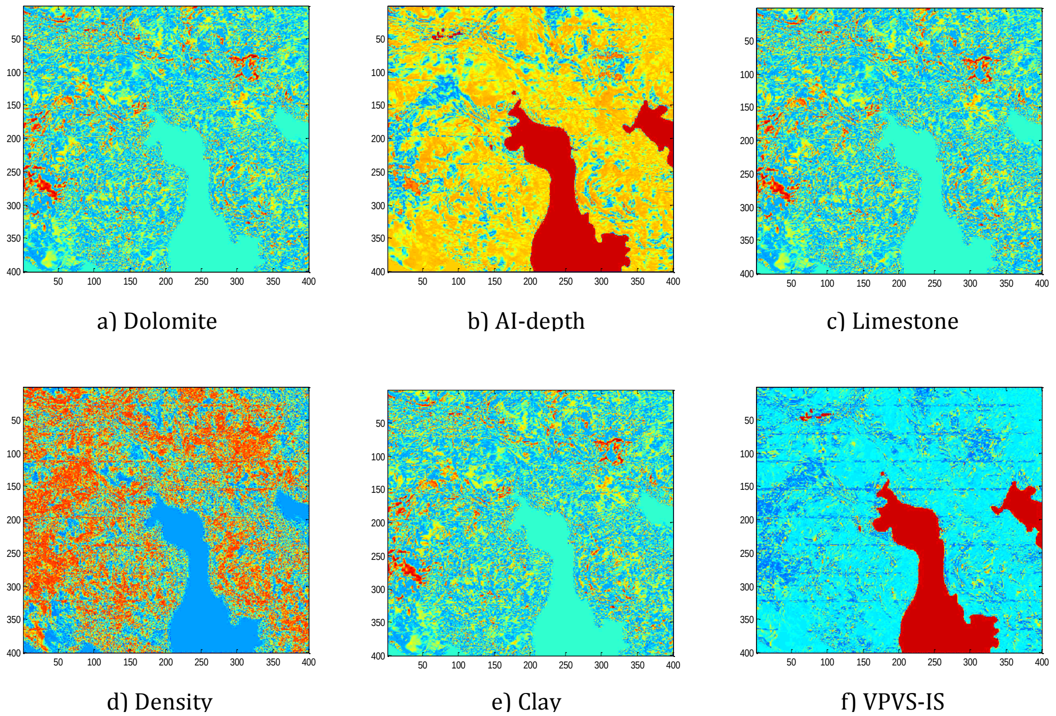


Figure 6.13 Portion of the time slice in Figure 6.12 utilized in covariance calculations.

6.3.2 Seismic Attribute Similarity

The following figure contains the time slices of the attribute data extracted from their respective volumes after intensity correction had been applied. Images a, c, and e are all apparently identical. These lithology-based seismic attributes are likely based on linear variations in the other seismic attributes and, for that reason, the data is exactly identical. This exact similarity between all three lithology attribute volumes was confirmed during the covariance matrix calculations. The results are located in Table 6.5 below.



a) Dolomite

b) AI-depth

c) Limestone

d) Density

e) Clay

f) VPVS-IS

Figure 6.14 Time slices utilized in similarity measurement.

The following table contains all of the normalized covariance values for each possible pairing of attribute volumes. The value in each cell corresponds to the similarity between the attributes on the intersecting row and column. For example, to find the similarity between the limestone and density attribute volumes, one would go to the column specifying “c) Limest.” and go down until the intersecting “d)” row was reached. A normalized covariance value of one corresponds to a pair of identical attribute volumes. As the values approach negative one, the volumes become more dissimilar and entirely independent.

Table 6.5 Covariance matrix displaying similarity between attribute volumes.

	a) Dolo.	**b) AI-dep**	**c) Limest.**	**d) Dens.**	**e) Clay**	**f) V_PV_S**
a)	1	-0.03117	1	-0.01775	1	-0.15099
b)	-0.03117	1	-0.03117	-0.34126	-0.03117	-0.01123
c)	1	-0.03117	1	-0.01775	1	-0.15099
d)	-0.01775	-0.34126	-0.01775	1	-0.01775	0.08203
e)	1	-0.03117	1	-0.01775	1	-0.15099
f)	-0.15099	-0.01123	-0.15099	0.08203	-0.15099	1

The computed covariance values confirm that the lithology indicators for limestone, clay, and dolomite are based on the same combination of underlying seismic attribute and, consequently, exhibit perfect correlation between them. The results also indicate weak correlation between v_pv_s, density, and acoustic impedance. This indicates that each of these seismic attributes bring independent information that can be used to model spatial variations in properties such as variation in lithology, porosity, and dolomitization in future work. Therefore, subsequent analysis will only consider the directly inverted seismic attributes such as acoustic impedance, density, and v_pv_s ratio.

6.4 Discussion on the Preparation of Data Prior to DSSIP

The log property study highlighted the large amounts of data preparation to create the facies interpretation maps required in DSSIP. Cross-plot analysis gave a broad overview of the local geology and confirmed prior background research. Multivariate non-linear regression based algorithms provided a more accurate estimation of the facies present in each interval. The neural network study showed dolomitization was maximum in wells located in the thickest parts of the reservoir and near major faults. Cross comparison with production data for wells in the Field A revealed that the productivity of wells was directly related to the extent of dolomitization.

The seismic exercise showed the importance in reducing data sets prior to map extraction. Covariance calculations determined interdependency between multiple attribute volumes. The next step in the analysis is to generate suitable training image for oolitic bank reservoirs. This aspect is discussed in the next chapter. That chapter also investigates the impact of seismic data quality on the modeling results obtained using the DSSIP algorithm.

Chapter 7: Modeling Reservoirs with Low Seismic Resolution

The quality of information recovered during DSSIP is directly related to the amount of information present in the seismic maps. Connected geobodies identifiable in the seismic images dictate the structural accuracy of the final output map. Therefore, it is important to collect the highest quality seismic maps. However, seismic imaging may not always generate a clear rendition of the important geologic structures that influence reservoir connectivity. Factors such as the salt canopy in the Gulf of Mexico and surface noise near high population areas will inhibit seismic capture and resolution (Farmer et al., 1994). This chapter provides an overview of how the quality models obtained by DSSIP are dependent on the quality of the seismic inputs. This work was initiated in the context of the Field B reservoir data set, which is impacted by the presence of a salt canopy. However, the study was performed prior to embarking on modeling the facies distribution in those reservoirs, as an exploratory investigation to delineate what information might be extractable from the seismic data set for those reservoirs. The facies and seismic models used were synthetically created based on field examples of oolite reservoirs.

7.1 Synthetic Oolitic Facies Models

Oolite reservoirs are composed of ooids, small spherical grains, and often are oil-bearing reservoirs (Middleton, 2003). Modern depositional locations include the Bahamas Great Bank in areas of lagoons, tidal flats, and beaches (Kindler and Hine, 2012). The models used in the following exercise are based on the geology seen in the St. Louis Limestone in southwest Kansas. A more thorough discussion of the subsections and characteristics of oolite reservoirs is located in Section 6.1.

7.1.1 Modeled Geology

The St. Louis Limestone is an ancient oolite deposit dating to the Mississippian time period. The reservoir is defined by porous oolitic bodies largely composed of intergranular porosity of up to 20% and permeabilities greater 600 md (Carr and Lundgren, 1994). An extensive study conducted by Qi et al. on the Big Bow and Sand Arroyo Creek Fields within the St. Louis Limestone play will form the backbone of the lithologic breakdown (Qi et al., 2007).

Qi et al. divided the geology into six lithofacies: quartz-rich carbonate grainstone, argillaceous limestone, skeletal wackestone, skeletal grainstone-packstone, ooid grainstone, and cemented ooid grainstone. A diagram containing the representative depositional environment for each of these facies is located in Figure 7.1. Quartz-rich grainstones are deposited in eolianite sand dunes and consist of well-sorted grains. Argillaceous limestones and skeletal wackestones are deposited in the deepest settings and are less porous than most neighboring material. The skeletal grainstone-packstone matrix generally has a lower porosity than the ooid lobes but is very well sorted with fine skeletal grains. Ooid grainstones are the target facies and contain the highest reservoir potential. Cemented ooids contain little to no pore space and will have suffered heavily from cementation and compaction.

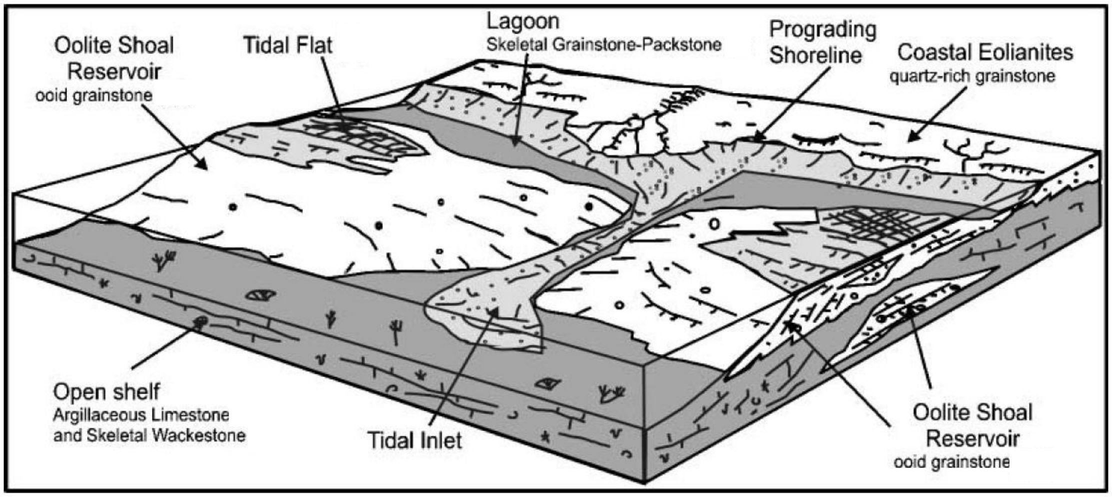


Figure 7.1 Depositional environment associated with oolitic bank reservoirs.

Table 7.1 Dimensions and proportion of oolitic objects used to generate training images.

				Length (m)			Width (m)			Orientation (deg)		
	Facies	**#**	**Vol.**	**Min**	**Avg**	**Max**	**Min**	**Avg**	**Max**	**Min**	**Avg**	**Max**
Set 3	**Oolite**	1	0.42	7750	9000	10250	5500	6250	6750	67.5	90	112.5
	Oolite (cem)	2	0.18	5000	5500	6000	2750	3000	3250	67.5	90	112.5
	Eolianite	3	0.08	13500	17000	20500	6500	9000	11500	62.5	85	107.5
	Tidal-flat	4	0.25	7750	9000	10250	6000	10000	14000	72.5	95	117.5

7.1.2 Facies Map Creation

Synthetic geologic models were created using the relationships developed during the study of oolite banks and summarized in the table above. The training image generator within Stanford Geostatistical Modeling Software (SGeMS) was utilized to create the images (Remy, 2004). SGeMS is an open-source platform built for fast computing of spatial statistics, specifically problems in the geosciences.

The parameters in Table 7.1 were input into the training image generator as ellipses and several models were created. The erosion rules were set as follows, from no erosion to most eroded: cemented oolites, oolites, tidal flats, eolian sands, and marine deposits. In addition, the cemented oolites were forced to overlap, or neighbor, oolite geobodies. The final facies maps were created using the *sisim* algorithm and 4000 conditioning points from the training images to retain the structure and volume proportions specified in Table 7.1. By introducing *sisim* into the creation process, the geobodies within the final facies maps were more misshapen, better representing the fuzzy transition between facies in any geologic deposit.

7.2 SYNTHETIC SEISMIC MODEL

7.2.1 Mimicking the Blurring of Seismic Surveys

After the training and final facies maps were created, a blurring mechanism was used to create synthetic seismic images similar to how the processed seismic data would look like in the presence of high density features such as salt. Forward seismic modeling is generally used in conjunction with seismic acquisition to develop appropriate acquisition parameters given the expected geology and to assist in interpretation (Anderson and Cardimona, 2002). The process uses an impedance map based on the anticipated facies with depth and transforms the data into a reflection amplitude map with two-way travel time. Forward seismic algorithms mimic the actual seismic acquisition process and calculate travel times and amplitudes based on the changes in acoustic impedance and source/receiver location.

In addition to a general loss of data resolution during seismic surveys, salt bodies will further obscure the visualization of underlying layers. Salt has a sharply higher p-wave velocity than any neighboring facies leading to an unrecoverable loss of information (Raymer et al., 2000, Jardin et al., 2011). The blurring mechanism that we implemented mimics this influence of a salt canopy in the form of a loss of information. The blurring algorithm used in this research is a highly simplified recreation of the output of forward seismic processes. A large amount of information is lost due to the low resolution of the seismic capturing process, and, as a result, the seismic image looks hazy.

The seismic blurring algorithm begins by replacing the indicator facies values in the geologic maps created in Section 7.1 with impedance values. Acoustic impedance is the product of wave velocity and density of the rock, and it governs the speed at which a wave will travel through a body. The impedance values used in this exercise are typical to the respective facies.

At this point, a Gaussian filter is applied to the indicator impedance map. Gaussian filters are commonly used in single-dimensional signal processing and two-dimensional image editing applications. The two dimensional Gaussian function, as used here, applies weights to all locations surrounding a given point and replaces the original point's value with a weighted average of the neighbors. The filter more heavily weighs locations closer to the starting point according to the parameters that define the Gaussian curve, seen in the equation below. The main advantages of the Gaussian filter as compared to other convolution matrices are the lack of an overshoot into the negative weight domain and the heavy weighting of the central value. These attributes allow the blurring mechanism to preserve the edges and location of structures in the image while applying a loss of information evenly.

$$g(x,y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}}$$

where x and y are the distances from the original point and σ is the standard deviation.

7.2.2 Synthetic Seismic Map Results

The results obtained by applying the Gaussian filter are located in Fig. 7.2 below. The filter grid sizes in Fig. 7.2*b*, *c*, and *d* govern the standard deviation for each run and define the location of the third standard deviation. Points at distances further than three standard deviations from the original location are effectively weighted zero due to the shape of a Gaussian curve. Therefore, the standard deviation was defined as one third of the grid size to create an effective cutoff at the edge of the filter grid.

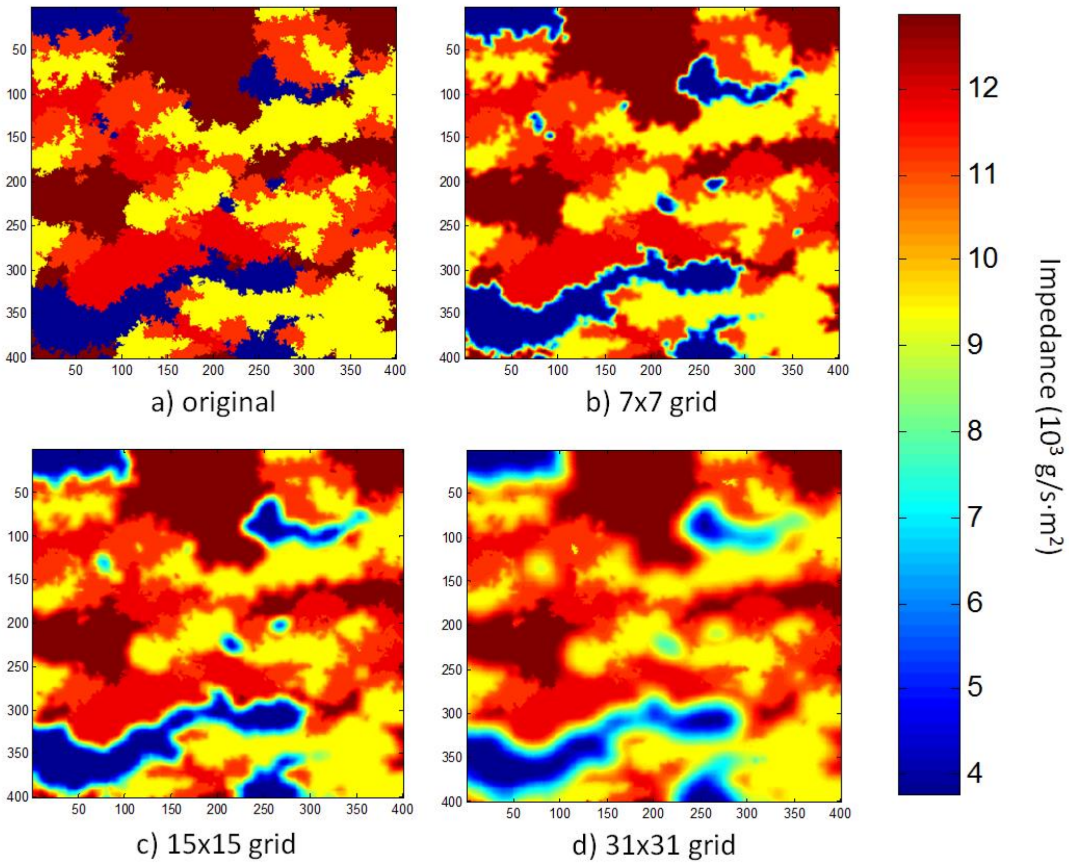


Figure 7.2 Results of Gaussian filter application to a 400x400 impedance map.

7.3 IMPLEMENTATION IN DSSIP

The impact of quality of seismic information in DSSIP was tested using two separate data sets. The first exercise decreased the resolution of both the training and target seismic images. The second exercise held training seismic data at a higher resolution while the seismic map for the target reservoir was decreased in resolution.

7.3.1 Decreased Resolution in Both Seismic Maps

As mentioned, the first exercise studied a general decrease in seismic quality. The cases within this study are differentiated by the size of the Guassian blur: a) 5x5 blur, b) 9x9 blur, c) 15x15 blur, and d) 31x31 blur. The training facies maps were held constant across all cases and the final recommendation set of parameters from Chapter 3 were used. Note that the original implementation of the DSSIP algorithm using two sets of seismic data was used for in these cases. Because the objective is to investigate the impact of the quality of seismic data on the results of the seismic integration algorithm, the most straightforward implementation of the algorithm using matching pairs of training and target reservoir data was used.

Figures 7.4 and 7.4 display the seismic input maps. The size of the different filters ranged from no filter to a filter that averaged half of the map for a single point. The black box in each of the images corresponds to the size of blurring filter. As the filter size increased, the extremes of the impedance map began to disappear. In Seismic Map *(a)* within Figure 7.3, a large series of low impedance bodies can be observed in the right side of the image, but, as the blurring mechanism increases in intensity, the borders of the objects become unclear and the impedance values in the objects increase.

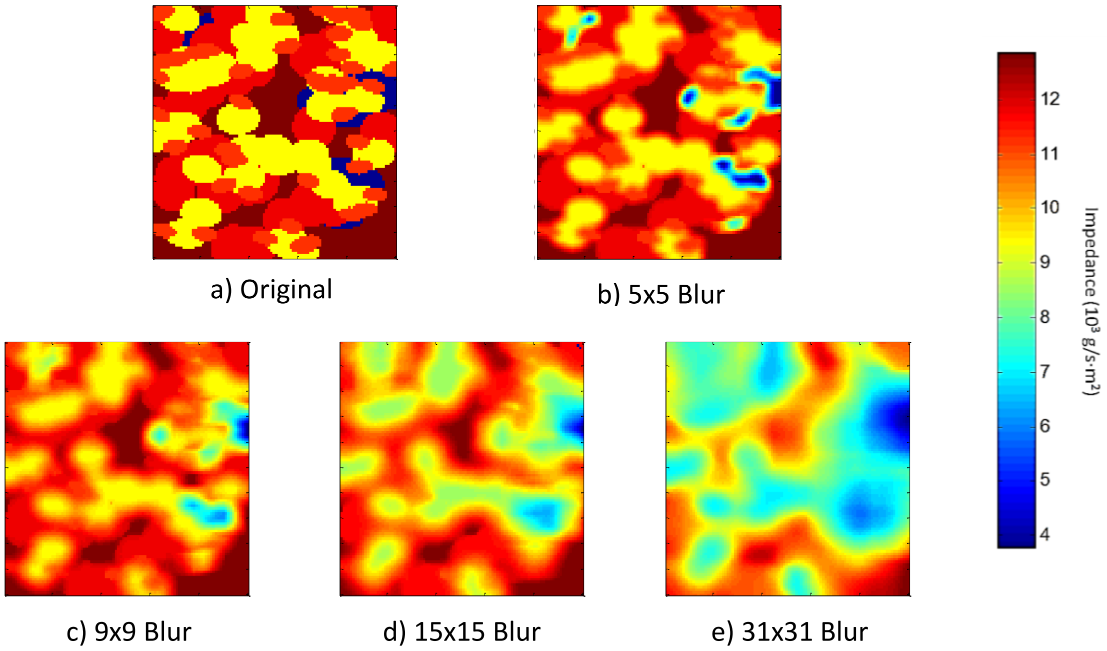


Figure 7.3 Application of Gaussian blur on the training reservoir impedance data.

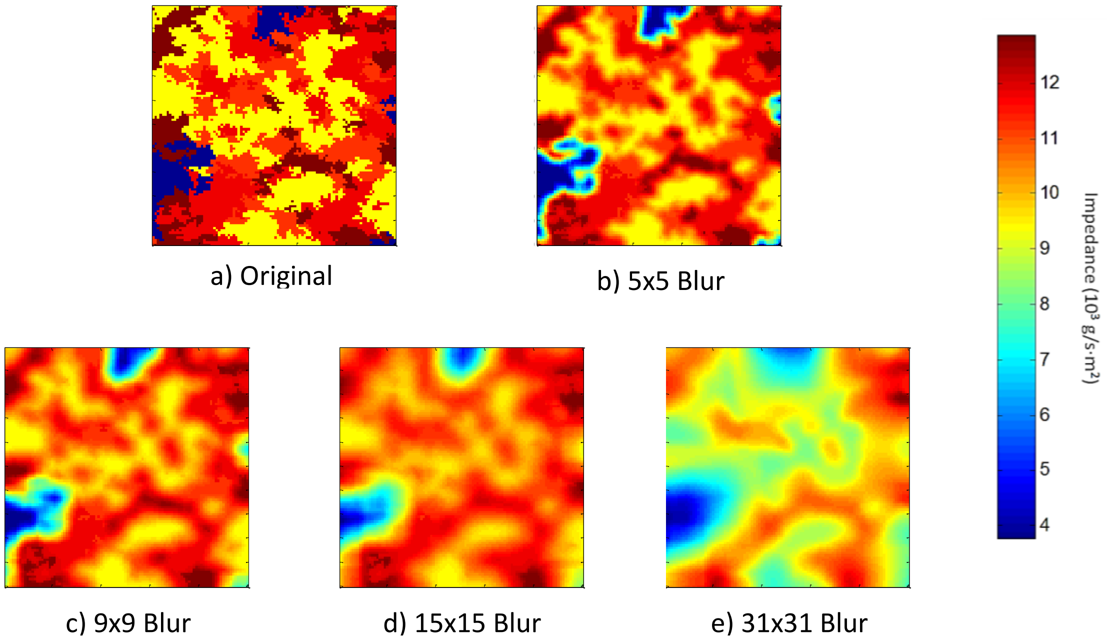


Figure 7.4 Application of Gaussian blur on the target reservoir impedance data.

The results obtained by the DSSIP algorithm are presented in Figure 7.6. The output maps are difficult to differentiate because the impact of the training image in the second loop. However, one can note the increase in oolites from Case A to Case D. Figure 7.6 contains the results after only the first loop iterations have completed. This provides a much better representation of the impact of resolution on the structures identified using the seismic. The continuity of the geobodies is much better represented using the higher resolution seismic maps, but the overall structure locations are still maintained even in Case D.

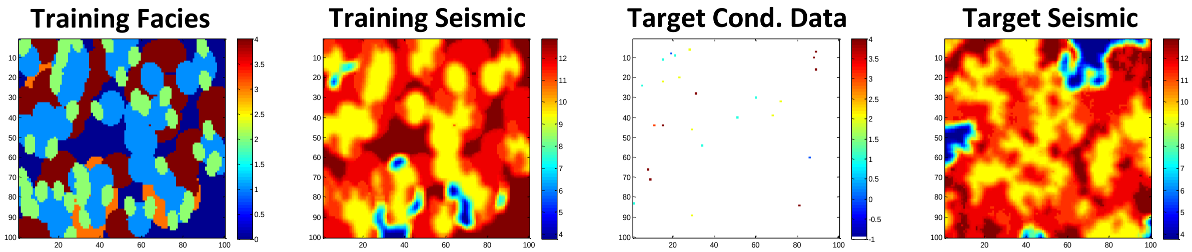


Figure 7.5 Training and target reservoir pair inputs for Case A. The facies maps were used in all following cases. The seismic maps were changed accordingly.

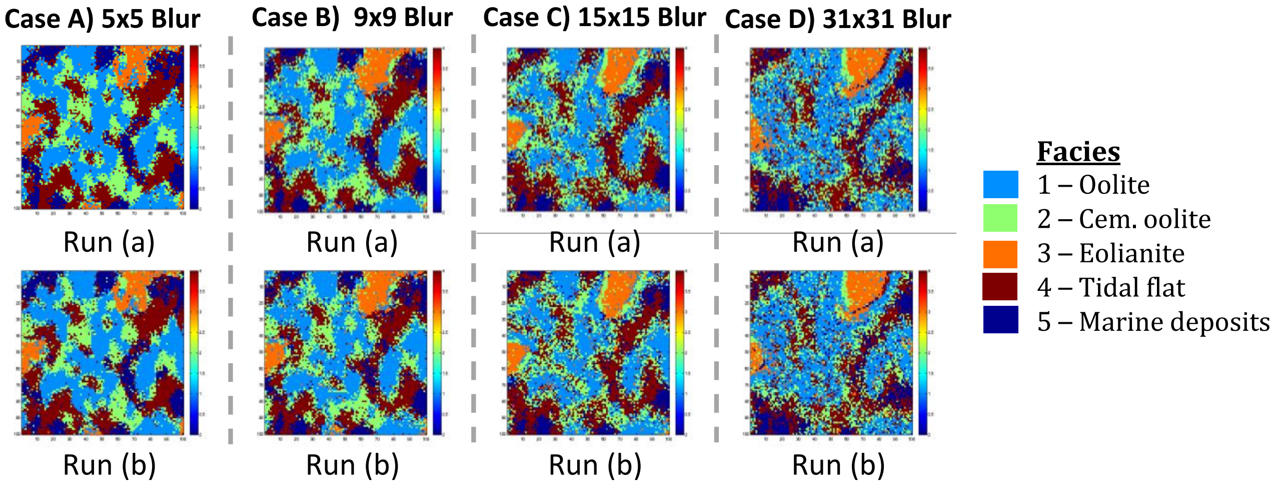


Figure 7.6 Results after completion of Loop 1 from study of decreased resolution in both seismic images.

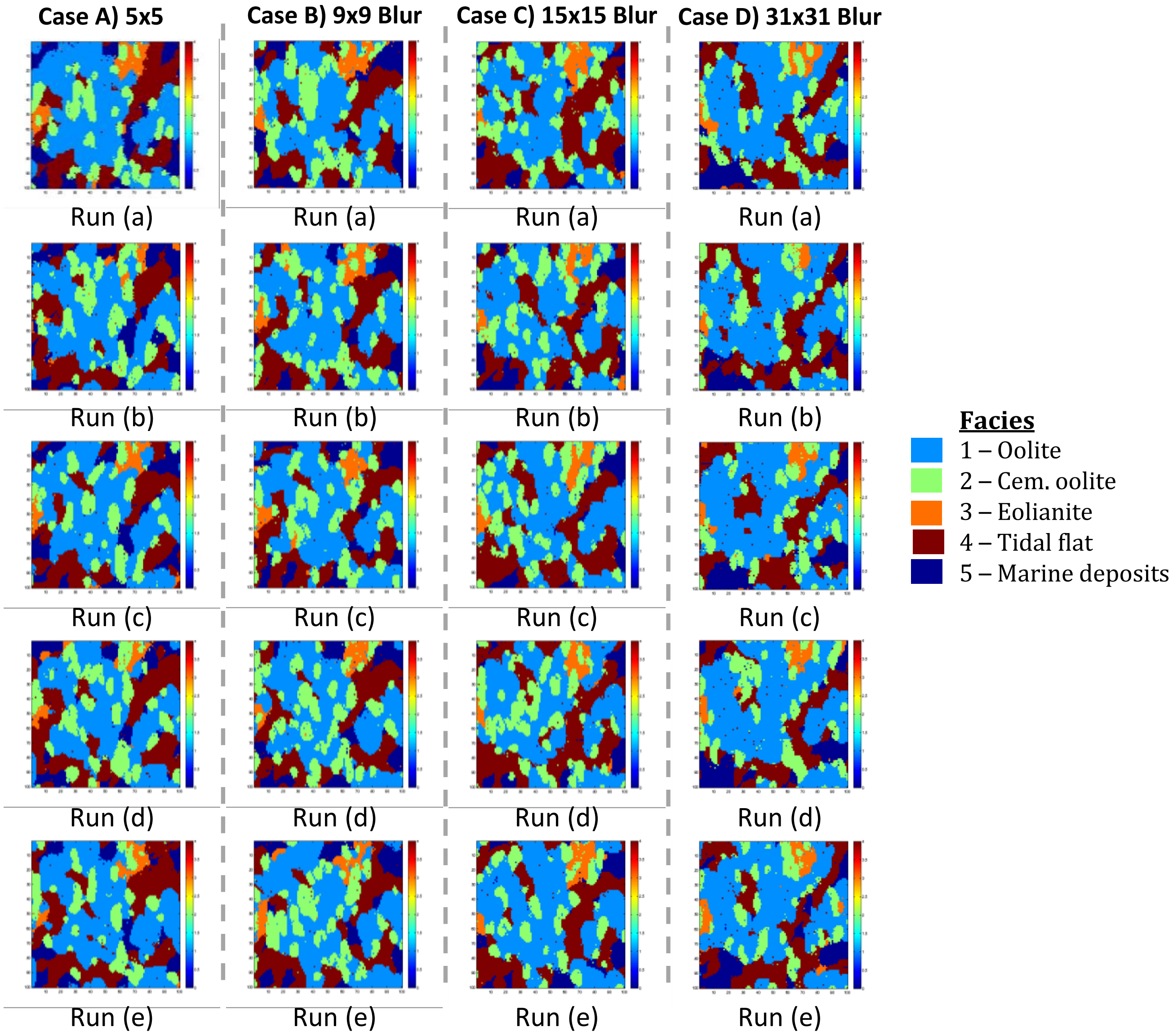


Figure 7.7 Results obtained using the DSSIP algorithm with training and target seismic images containing different amounts of Gaussian blur.

7.3.2 Decreased Resolution in Only One Seismic Map

The second exercise studied the impact of differing quality seismic data for the target reservoir. Cases A and B were simulated with higher resolution training image but using seismic data for the target reservoir that was of poor quality, while Cases C and D tested the opposite situation. Table 7.2 summarizes the dimensions of the blur filter used for each case. This case used the same facies maps, seismic images, and algorithm parameters as in the first study.

Table 7.2 Summary of filter dimensions used in Figure 7.8

Case	**Train.**	**Target**
A	5x5	31x31
B	5x5	15x15
C	31x31	5x5
D	15x15	5x5

The results after Loop 1 are located in Figure 7.8. Again, increasing the resolution created a better interpretation as can be seen when comparing Case A to Case B and Case C to Case D. More important to note is that the difference in resolution between the training and target seismic maps had a large impact on the overall image quality. Due to this difference, the first DSSIP loop was unable to create accurate pattern matches. Decreasing the resolution of the target reservoir seismic map decreased the accuracy of results at the end of Loop 1 more than decreasing the resolution of the training seismic image.

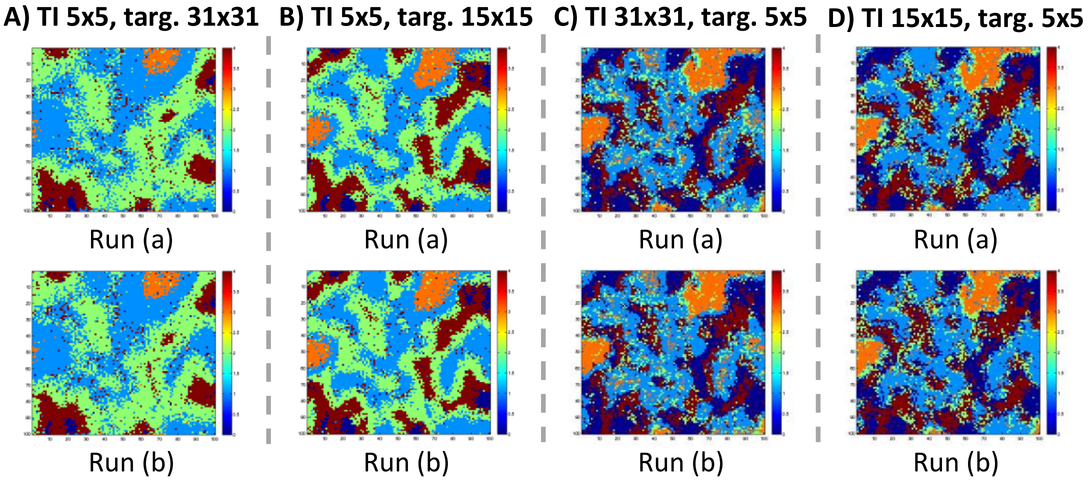


Figure 7.8 Results after Loop 1 when seismic data of variable quality was used to train and simulate the reservoir model.

7.3.3 Discussion of Results

The effect of decreased seismic information may not be obvious, as seen in Figure 7.7, due to the ability of the second loop to match the training image. However, the loss of structural data at the end of Loop 1 of the algorithm will limit the accuracy of placement of geobodies and result in simulated reservoir models that bear close resemblance to the training reservoir model. The two studies showed that the decrease in seismic quality does have a large impact on the information collected during the first loop in DSSIP. This lack of structural information affects the results shown in Figures 7.6 and 7.8. Decreasing the resolution of the target seismic map decreases the accuracy of the final simulated model more than decreasing the resolution of the training reservoir model. This is important because in most practical cases the seismic data corresponding to the training reservoir model will be obtained by running a forward seismic algorithm and that algorithm might yield a very fuzzy indication of facies variation in the reservoir.

Chapter 8: Conclusions

This thesis presents the application of Direct Sampling Seismic Integration Process on several synthetic and real data sets. The results show that the current version of DSSIP can be successfully applied in a wide context, but further modifications to the algorithm may lead to even broader application. Based on the results in previous chapters:

- Direct sampling provides a solid basis for the pattern matching process within DSSIP. It has been shown to accurately reconstruct and identify patterns in both continuous and categorical variables.
- The user possesses a large set of controls on the speed and accuracy of DSSIP by adjusting the internal parameters of the algorithm. The parameter optimization study showed steep decreases in computational time with only minor loss in output quality.
- A final set of optimum simulation parameters for DSSIP has been recommended.
- Using the current version of DSSIP in a self-iterating function reproduces the training image without adding knowledge from the seismic map. However, the addition of an ensemble of produced training images shows promise. Future work in this area includes using *filtersim* in the second loop.
- Careful data processing is necessary prior to application of the DSSIP algorithm. Robust techniques for determining facies based on well log values and combining that information with the available seismic information are necessary. DSSIP can address this second issue but only after reliable classification of facies information are available.

- A sharp decrease in seismic image resolution, such as when under a salt canopy, reduces the overall quality of information gathered during the first loop of DSSIP. Decreased seismic resolution in the target map will decrease the output quality more than similarly decreased resolution in the training map.

References

Al-Anezi, K., Kumar, S., and Ebaid, A., 2013. Geostatistical Modeling with Seismic Characterization of Wara/Burgan sands, Minagish Field, West Kuwait. Paper SPE 166046 presented at SPE Reservoir Characterisation and Simulation Conference and Exhibition held in Abu Dhabi, UAE, 16-18 September 2013.

Andersen, T. et al, 2006. Method for Conditioning the Reservoir Model on 3D and 4D Elastic Inversion Data Applied to a Fluvial Reservoir in the North Sea. Paper SPE 100190 presented at the SPE Europec/EAGE Annual Conference and Exhibition held in Vienna, Austria, 12-15 June 2006.

Anderson, N. and Cardimona, S., 2002. Forward seismic modeling: The key to understanding reflection seismic and ground penetrating radar (GPR) techniques. Presented at the 2nd Annual Conference on the Application of Geophysical and NDT Methodologies to Transportation Facilities and Infrastructure held in Los Angeles, CA, U.S.A., 15-19 April 2002.

Bouhling, G., 2005. Introduction to Geostatistics and Variogram Analysis. Retrieved from *people.ku.edu/~gbohling*

Bouma, A.H., 2004. Key controls on the characteristics of turbidite systems. *Geological Society, London, Special Publications* **222**: 9-22.

Calabrese, M., Pirera, F., Rossi, M., Scaglioni, P., Tosoratti, F., 2011. Integration of Seismic Lithology Data in the Reservoir Model. Paper SPE 143126 presented at the SPE EUROPEC/EAGE Annual Conference and Exhibition held in Vienna, Austria, 23-26 May 2011.

Carr, T.R. and Lundgren, C.E., 1994. Use of Gamma Ray Spectral Log to recognize exposure surfaces at the Reservoir Scale: Big Bow Field (St. Louis Limestone), Kansas.

Castro, S., Caers, J., Otterlei, C., Hoye, T., Andersen, T., Gomel, P., 2006. A Probabilistic Integration of Well Log, Geological Information, 3D/4D Seismic and Production Data: Application to the Oseberg Field. Paper SPE 103152 presented at the SPE Annual Technical Conference and Exhibition held in San Antonio, TX, U.S.A., 24-27 September 2006.

Edman, J.D. and Burk, M.K., 1998. An Integrated Study of Reservoir Compartmentalization at Ewing Bank 873, Offshore Gulf of Mexico. Paper SPE 49246 presented at 1998 SPE Annual Technical Conference and Exhibition held in New Orleans, LA, U.S.A., 27-30 September 1998.

Farmer, P. et al., 1994. Exploring the Subsalt.

Guardiano, F.B., Srivastava, R.M., Multivariate Geostatistics: Beyond Bivariate Moments. *Geostatistics Troia '92* **1**: 133-144.

Hampson, D.P., Schuelke, J.S., Quirein, J.A., 2001. Use of multiattribute transforms to predict log properties from seismic data. *GEOPHYSICS* **66** (1): 220-236.

Hu, L.Y. and Chugunova, T., 2008. Multiple-point geostatistics for modeling subsurface heterogeneity: A comprehensive review. *Water Resources Research* **44**.

Jardin, A., Roure, F., Nikolla, L., 2011. Subsalt Depth Seismic Imaging and Structural Interpretation in Dumre Area, Albania. *Oil & Gas Science and Technology* **66** (6): 911-929.

Jolliffe, I.T., Principal Component Analysis: 2nd Edition. *Springer Series in Statistics*.

Journel, A.G., 1983. Nonparametric estimation of spatial distributions. *Mathematical Geology* **15** (3): 445-468.

Kindler, P. and Hine, A.C., 2012. The Paradoxical Occurrence of Oolitic Limstone on the Eastern Islands of Great Bahama Bank: Where Do the Ooids Come From?

King, G.A., Archer, S.H., Seymour, R.H., 1993. The Integration of Seismic and Well Data for Improved Reservoir Models. Paper SPE 25635 presented at the SPE Middle East Oil Technical Conference & Exhibition held in Bahrain, 3-6 April 1993.

Lindseth, R.O., 1979. Synthetic sonic logs – a process for stratigraphic interpretation. *GEOPHYSICS* **44** (1): 3-26.

Lomas, S.A. and Joseph, P., 2004. Confined Turbidite Systems. *Geological Society, London, Special Publications*.

Mariethoz, G., Renard, P., Straubhaar, J., 2010. The Direct Sampling method to perform multiple-point geostatistical simulations. *Water Resources Research* **46**.

Mariethoz, G., Renard, P., Cornaton, F., Jaquet, O., 2008. Truncated Plurigaussian Simulations to Characterize Aquifer Heterogeneity. *Groundwater* **47** (1): 13-24.

Matheron, G., 1976. A Simple Substitute for Conditional Expectation: The Disjunctive Kriging. *Advanced Geostatistics in the Mining Industry*: 221-236

Matheron, G. et al, 1987. Conditional Simulation of the Geometry of Fluvio-Deltaic Reservoirs. Paper SPE 16753 presented at the 62nd Annual Technical Conference and Exhibition of the Society of Petroleum Engineers held in Dallas, TX, U.S.A., 27-30 September, 1987.

Middleton, V., 2003. Encyclopedia of Sediments and Sedimentary Rocks.

Mu, C., Shi, B., Zhang, T., 2011. Application of SISIM in reconstructing porous media. Presented at the Computer Research and Development (ICCRD), 2011 3rd International Conference held in Shanghai, China, 11-13 March 2011.

Narhari, S.R., Al-Ajmi, M.D., Tanoli, S.K., and Al-Qadeeri, B., 2008. Integration of seismic attribute analysis and well data to identify depositional trends: A case study from Kuwait. *SEG Technical Program Expanded Abstracts 2008*: 453-457.

Pattison, S. A.J., Storm-influenced Prodelta Turbidite Complex in the Lower Kenilworth Member at Hatch Mesa, Book Cliffs, Utah, U.S.A.: Implications for Shallow Marine Facies Models. *Journal of Sedimentary Research* **75** (3): 420-439.

Pujol, J., 2007. The solution of nonlinear inverse problems and the Levenberg-Marquardt method. *GEOPHYSICS* **72** (4): W1-W16.

Qi, L., Carr, R., Goldstein, R.H., 2006. Geostatistical three-dimensional modeling of oolite shoals, St. Louis Limestone, southwest Kansas. *AAPG Bulletin* **91** (1): 69-96.

Quinto, S., Gert Jan, W., Eloisa, S.B., 2013. How to Integrate Basin-scale Information in Reservoir Models. Paper SPE 164830 presented at the EAGE Annual Conference & Exhibition Incorporating SPE Europec held in London, UK, 10-13 June 2013.

Raymer, D.G., Tommasi, A., Kendall, J.M., 2000. Predicting the seismic implications of salt anisotropy using numerical simulations of halite deformation. *GEOPHYSICS* **65** (4): 1272-1280.

Remy, N., 2004. Geostatistical Earth Modeling Software: User's Manual.

Riddiford, F.A., Goupillot, M., 1994. Geotechnical Integration and its Impact on Field Management, The IRMA Methodology. Paper SPE 28932 presented at the SPE 69th Annual Technical Conference and Exhibition held in New Orleans, LA, U.S.A., 25-28 September 1994.

Russell, B., Hampson, D., Todorov, T., 2001. Combining geostatistics and multiattribute transforms – A channel sand case study. Presented at SEG International Exposition and Annual Meeting at San Antonio, TX, U.S.A., 9-14 September 2001.

Shannon, C.E., 1948. A Mathematical Theory of Communication. *The Bell System Technical Journal* **27**: 379-423, 623-656.

Silva, A.C., Guerreiro, L., Maciel, C., Ramos, L., Guerra, R., 1996. Integration of 3D Seismic Data into Reservoir Models Using Geostatistics. Paper SPE 36217 presented at the 7th Abu Dhabi International Petroleum Exhibition & Conference held in Abu Dhabi, UAE, 13-16 October 1996.

Strebelle, S., 2002. Conditional Simulation of Complex Geological Structures Using Multiple-Point Statistics. *Mathematical Geology* **34** (1): 1-21

Wang, Y., Liu, J., Zhang, T., Comparison of IK SISIM and SNESIM in Reconstructing 2D Porous Media. Paper Proc. SPIE 8285 presented at International Conference on Graphic and Image Processing (ICGIP 2011) held in Cairo, Egypt, 1 October 2011.

Yang, C. T., Chopra, A.K., Chu, J., Huang, X., Kelkar, M.G., 1995. Integrated Geostatistical Reservoir Description Using Petrophysical, Geological, and Seismic Data for Yacheng 13-1 Gas Field. Paper SPE 30566 presented at SPE

Annual Technical Conference and Exhibition held in Dallas, TX, U.S.A, 22-25 October 1995.

Zhang, T., Switzer, P., Journel, A., 2006. Filter-Based Classification of Training Image Patterns for Spatial Simulation. *Mathematical Geology* **38** (1): 63-80.

Zimmerle, W., 1995. Petroleum Sedimentology. *Kluwer Academic Publishers*: The Netherlands.