Mathematical Geology, Vol. 39, No. 2, February 2007 (© 2007)
DOI: 10.1007/s11004-006-9074-4

Simulating Sedimentary Successions Using Syntactic Pattern Recognition Techniques[1]

E. June Hill[2] and Cedric Griffiths[2]

Sections from a sedimentary succession can be simulated using a process which includes both probabilistic and deterministic information. The inclusion of both of these types of information allows the production of geologically realistic simulations which contain the required level of heterogeneity. The process uses syntactic pattern recognition techniques and is based on the formal description of a geological model using a grammar. The simulations can be conditioned on well data.

KEY WORDS: reservoir, stochastic, syntactic, pattern, grammar.

INTRODUCTION

In many regions of economic interest (such as oil fields), knowledge of a particular package of sedimentary rocks is restricted to the information provided by a few widely spaced drilled wells. The wells provide a very narrow, almost continuous, vertical succession of data and it is necessary to interpolate the data into the space between wells.

When manually reconstructing geology from limited data, geologists will typically produce the simplest model that honours the data. However, it has been found that oil reservoir predictions have failed because these models are overly simplistic and so stochastic techniques are widely used for reservoir modelling (e.g. Haldorsen and Damsleth, 1990; Damsleth and others, 1992; Srivastava, 1994). Stochastic models can control the size and density of sedimentary bodies and can therefore more accurately reflect the heterogeneity present in a sedimentary succession. This process of stochastic prediction is called conditional simulation. However, it does not produce geologically realistic models of sediment distribution and, for this reason, it is desirable to include deterministic information, based on geological observations, in the stochastic models (Damsleth and others, 1992).

[1]Received 5 August 2005; accepted 22 October 2006; Published online: 12 April 2007.
[2]Commonwealth Scientific and Industrial Research Organisation, Petroleum Resources, P.O. Box 1130, Bentley, WA 6102, Australia; e-mail: June.Hill@csiro.au

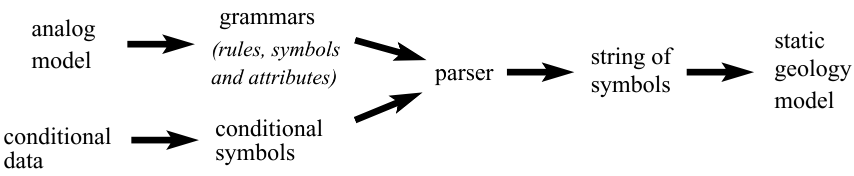


Figure 1. An analog model is described in the format of a grammar. The grammar and conditioning data are used by the parser program to generate stochastic geological models which are made up of vertically stacked strings of symbols.

A number of methods which can produce discrete object-based geological models have been developed, these all use marked point processes (e.g. Hatløy, 1994; Tyler, Henriquez, and Svanes, 1994; Deutsch and Tran, 2002). A completely different method is introduced here, based on syntactic pattern recognition. The syntactic method is not only object-based but it is also object-oriented. Object-oriented means that a geological object can be represented by a single symbol of variable size rather than a collection of rectangular cells, hence the simulation does not have a fixed resolution dependent on cell size. The syntactic method provides a holistic framework to describe a complete depositional environment in a single grammar and has the potential to describe sophisticated patterns in a simple format. Once the user has selected or designed a suitable grammar the generation of geology models is performed automatically by a computer program called the parser (Fig. 1).

If descriptions of outcrops of sedimentary rocks can be formalised then they can be used to predict, in a consistent way, sedimentological patterns in areas of no outcrop. The first step in the model-building process is to select a suitable analog model which best fits the known data from the reservoir. The analog model can be derived from a wide variety of sources including studies of modern day systems, outcrops of ancient systems, tank experiments, well data, seismic data, and theoretical studies of facies distributions. The analog model can then be encoded in the formal description and used to produce geological models of the reservoir. The advantage of this process is that different analog models can be tried and the results compared; any analog model which consistently produces geologically unrealistic results when the conditioning data from the reservoir is included can be rejected.

Syntactic pattern recognition, which is based on studies of linguistics and which has been used for a wide variety of pattern recognition problems (Fu, 1982), provides a useful structure for the description of geological rules. This idea has been tested by Griffiths (1990), Duan, Griffiths, and Johnsen (1996) and Duan, Griffiths, and Johnsen (1999) using phrase-structure grammars. A grammar comprises: symbols (the basic components), production rules and attributes (Fig. 1). Production rules describe the associations and structure of the components of a sedimentological system (syntax). Attributes add meaningful information (semantics), such as the geometry and size of the components.

One of the strengths of using grammars to describe geological models is that they can include information on the probability of occurrence of various sedimentological bodies, and on the probability distributions of the sizes of sedimentological bodies. These probabilities are essential to produce a realistic degree of heterogeneity in the final model.

This paper presents some preliminary results from a project which aims to use grammatical descriptions of analog models to produce conditional 3D simulations of petroleum reservoirs. The first stage of the project is to test the basic concepts in one dimension. The project is still in a too early stage to be able to test it against existing 3D marked point process models.

A Grammar for the Tanqua-Karoo Sub-Marine Fan Deposit

Published geological descriptions of deposits from a deep sea fan, now exposed in the Tanqua-Karoo basin, South Africa (Johnson and others, 2001; van der Werff and Johnson, 2003), have been used here for creating a grammar to test the application of syntactic pattern recognition to sedimentary sequences. The Tanqua-Karoo is described as a Permo-Triassic submarine fan complex that formed in a foreland basin. The complex is interpreted to contain 5 fine-grained turbidite systems alternating with thick shale intervals. Production rules have been constructed from the descriptions of turbidite deposits in the text and from patterns illustrated in the diagrams. The production rules are described in the next section.

We use the following description by Johnson and others (2001) to demonstrate with a simple example how the rules are constructed:

> . . . erosional channels are common in the mid-fan. . .overall geometry comprises a sand-rich axial zone with extensive heterolithic wings. These channels pass laterally into sandy overbank deposits that have a sheet geometry.

This information can be formalised as the following set of production rules:

$$\texttt{Midfan} \rightarrow C \mid C + A \tag{1}$$

$$A \rightarrow B \mid B + A \tag{2}$$

$$B \rightarrow \texttt{Channel} + \texttt{overbank} \tag{3}$$

$$C \rightarrow \texttt{overbank} + \texttt{Channel} + \texttt{overbank} \tag{4}$$

$$\texttt{Channel} \rightarrow \texttt{leftwing} + \texttt{axialzone} + \texttt{rightwing} \tag{5}$$

Each production rules contains an arrow (→, which means "is composed of"), a *head* (symbol to the left of the arrow) and one or more *tails* (string of

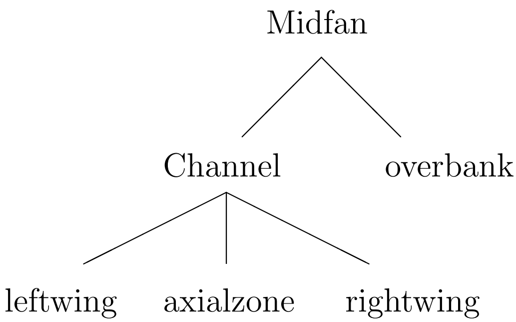


Figure 2. Illustrating the hierarchical nature of grouping of sedimentary bodies in the example given in the text.

symbol(s) to the right of the arrow). Where there is more than one tail, these are separated by the "or" symbol (|). To produce a pattern the start symbol (i.e. the highest level symbol, in this case Midfan) must be completely replaced by terminal symbols using the rules. The terminal symbols are the lowest level symbols, in this example they represent the basic geological elements: overbank, axialzone, leftwing, rightwing. Non-terminal symbols may represent higher level geological objects (e.g. Midfan and Channel) or simply represent convenient intermediate states (e.g. A, B, C). By convention, names of terminal and non-terminal symbols start with small and capital letters, respectively.

The second rule in the list above is recursive when the second tail is used (i.e. the symbol A occurs both in the head and the tail of the rule). The recursive rule allows the `Midfan` environment to contain any number of channel fills separated by overbank deposits. The hierarchical nature of the grouping of sedimentary bodies is effectively captured by the grammatical description (Fig. 2); i.e. that a `Channel` is composed of a `leftwing`, an `axialzone` and a `rightwing`; and, at a higher level, a `Midfan` is composed of a series of `Channels` separated by `overbanks`.

THE GRAMMAR

A grammar consists of a set of terminal symbols (to represent the basic elements of the pattern), a set of non-terminal symbols (to represent higher level elements of a pattern), a set of rules (to describe the valid patterns that can be formed by the terminal symbols) and, optionally, a set of attribute rules (to describe how to calculate attributes of the symbols, e.g. horizontal extent of a geological symbol). An ordered 1D series of terminal symbols which form a valid pattern is called a string. An algorithm which uses the grammar to recognise or produce strings of symbols is called a parser and will be described in more detail in the next section.

Sedimentary successions may be illustrated as 2D vertical sections which transect the paths of several wells (for examples see Figs. 6, 7 and 10 of van der Werff and Johnson, 2003). Duan, Griffiths, and Johnsen (1996) and Duan, Griffiths, and Johnsen (1999) created 2D vertical sections using syntactic methods by stacking a series of thin horizontal sections, where each horizontal section was represented by a string of symbols. The method introduced here also produces horizontal sections as strings of symbols; however, unlike the previous method it is object-based rather than cell-based.

Hierarchical System of Grammars

Most descriptions of large sedimentary deposits involve breaking down the deposit into a hierarchical series of depositional bodies or depositional boundaries (e.g. Miall, 1985; Pickering and others, 1995; Deutsch and Wang, 1996). van der Werff and Johnson (2003) have followed this tradition by dividing each fan into depositional units (time division) and then dividing the depositional units into depositional environments (spatial division, e.g., proximal fan, mid-fan and distal fan). The proximal fan is further sub-divided into overbank and axial channel system deposits (Fig. 3). The next level in the hierarchy is architectural elements; for example, channel fill or inter-channel high. At the lowest level of the hierarchy the deposits are described in terms of lithofacies. For example, the axial channel systems may contain fine to medium grained sandstones as the channel fill facies and laminated thick-bedded sandstones or amalgamated sandstones as the inter-channel facies.

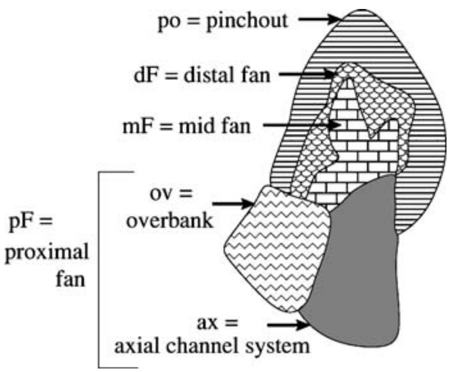


Figure 3. A simplified map illustrating the distribution of some architectural elements and depositional environments in the Tanqua fan (modified from van der Werff and Johnson, 2003).

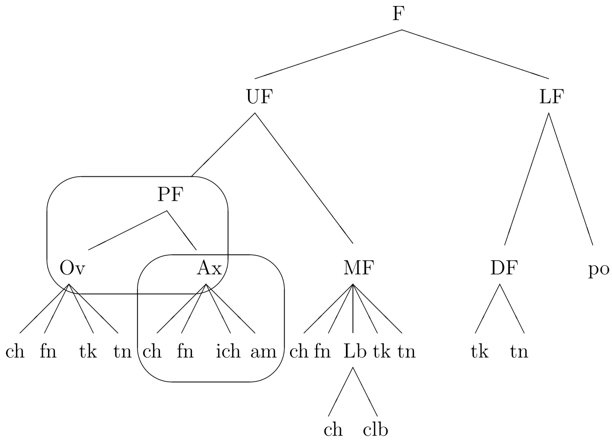


Figure 4. Hierarchical grammars for a submarine fan. The two rounded boxes indicate the levels of the grammars PF (proximal fan) and Ax (axial channel system). See Table 1 for other abbreviations.

Very complex systems which are inherently hierarchical in nature can be described using a hierarchical system of grammars (e.g. Stilman, 1993). Using this method a potentially very complex grammar can be broken down into a number of simpler grammars. The hierarchical system allows the horizontal patterns of sedimentary bodies to be described using regular grammars, this has the advantage of reducing computing complexity (Hopcraft, Motwani, and Ullman, 2001). By using the sub-divisions of van der Werff and Johnson (2003) the fan succession in the Tanqua-Karoo has been structured here into a hierarchical system of grammars each of which describes a coherent package of sediments which represents a recognisable geological entity (Fig. 4).

Two levels of grammars are described here as examples: (1) a high level description of a depositional environment, specifically the proximal fan which is composed of axial channel system and overbank environments; (2) a lower level description of architectural elements, specifically the axial channel system which is composed of channel fills separated by inter-channel highs, amalgamated sheet sandstones or fines (usually silt).

Stochastic Grammars

The sedimentary grammar must be stochastic as there is always more than one possible pattern that may fit the conditions. The are two types of stochastic variation in a simulation: (1) variation in the number and type of symbols; (2) variation in the extent of a symbol. The first is controlled by a probability value associated with each rule (this type of grammar is called a probabilistic grammar);

Table 1. Types of Start Symbols (S) and Terminal Symbols (T) and Their Meanings

S	T	Description	S	T	Description
F		fan	UF	uf	upper fan
LF	lf	lower fan	PF	pf	proximal fan
MF	mf	mid fan	DF	df	distal fan
Ov	ov	overbank	Ax	ax	axial channel system
Lb	lb	lobe		po	pinchout facies
	ch	channel fill		fn	fines
	th	thin bedded sandstone		tk	thick bedded sandstone
	am	amalgamated sandstones		ich	inter-channel high
	clb	channel lobe			

the second is controlled by a probability density function which is associated with each type of symbol. These are described in more detail in the relevant sections below.

Terminal Symbols

Each terminal symbol represents a geological entity of interest. The text representation of a symbol is called its type, some symbol types are given in Table 1. Every terminal symbol has several attributes including position, which describes the geographical position of a geological entity, and extent, which describes the horizontal extent of the geological entity in the section. At this stage of the project we are only concerned with producing valid strings of symbols with horizontal extent, in the next stage of the project attributes such as shape and thickness will be included.

The Rules

The grammar is composed of a set of rules. The first rule always contains the start symbol to reduce the amount of searching required by the parser. The heads of the rules are unique; i.e. there is only one rule which can be applied to each non-terminal symbol. The rules used are of two forms, one which offers a choice of substitution and one which offers no choice; i.e.

$$\texttt{head} \quad \rightarrow \quad \texttt{tail} \;\; 1 \quad | \quad \texttt{tail} \;\; 2 \quad p = x_{\text{tail1}}$$

$$\texttt{head} \quad \rightarrow \quad \texttt{tail}$$

Where a choice of tail is offered a value, p, is provided which states the probability with which the first tail should be chosen over the second tail. There are 4 types of rules, types 0, 1 and 2 are of the first form. Type 3 rules are of the second form and are used to convert a non-terminal symbol into a string of terminal symbols. Rule

Table 2. Two Example Sets of Rules for *PF* and *Ax* Grammars

Head		tail 1		tail 2	p	t
PF	→	*PFS*	|	*ov* + *PFS*	0.35	2
PFS	→	*Axis*	|	*OvR* + *PFS*	1.0	1
Axis	→	*ax*	|	*OvR*	0.35	2
OvR	→	*ax* + *ov*			1.0	3
Ax	→	*ch*	|	*AxL* + *Ax*	1.0	1
AxL	→	*ch* + *ICh*			1.0	3
ICh	→	*ich*	|	*ISd*	0.7	2
ISd	→	*am*	|	*fn*	0.3	2

Note. Column p contains the probability that tail 1 is selected and column t contains the rule type. The terminal symbol ax in the *PF* grammar forms the start symbol Ax for applying the lower level grammar.

types 0 and 1 can be used by the parser to extend strings which have insufficient extent by adding extra symbols; rule types 0 and 2 can be used to find different types of symbols.

In each grammar, at least one rule must be recursive (rule type 1) to allow the string to be extended to the length required. Recursion is provided by the second tail:[3]

$$\texttt{head} \quad \rightarrow \quad \texttt{tail}\ 1 \quad | \quad \texttt{tail}\ 2 \quad + \quad \texttt{head} \quad p = 1.0$$

As p is always set to 1.0 the second tail is used only if the first tail fails, to ensure that strings are no longer than required.

Table 2 shows 2 examples of sets of rules devised for the Tanqua fan deposits. For example, the first set of rules describes a proximal fan (PF) consisting of zones of overbank deposits (ov) which alternate with axial channel systems (ax). The rules allow the series to begin and end with either ov or ax. The second set of rules provide several choices for the type of inter-channel material: inter-channel highs (ich), amalgamated sandstone sheets (am) or fines (fn). The published data for the fan is insufficiently detailed to provide the probability values, so for the purpose of this exercise, probabilities have been roughly estimated.

Conditional Symbols

Where data is available for a particular position in a section (e.g. from wells) a conditional symbol is created. The conditional symbols have type and position attributes but horizontal extent is unknown. Strings of terminal symbols

[3]The recursive rules are all right-recursive (Knuth, 1971).

produced by the parser must be consistent with the conditional symbols. Boundary conditional symbols are used to record the start and end positions of the section.

THE PARSER

A parser is an algorithm that uses a grammar to check that strings of symbols are valid or to produce new strings of symbols. The parser used in this study is a top-down parser; i.e. it works from the top of the hierarchy in the grammar downwards[4]. It is designed to produce new strings of symbols from the grammar. The parser produces new symbols in a left to right order.

Calculating the Extent of a Sedimentary Body

In this paper, *extent* is used to indicate the size of the sedimentary body(ies) represented by a symbol or string of symbols, and *length* is used to indicate the number of symbols in a string.

As Johnson and others (2001) and van der Werff and Johnson (2003) do not provide sufficient information to produce probability distributions for extents of sedimentary bodies, we have used the uniform probability distribution. The minimum and maximum values of the distribution have been determined for the extent of each sedimentological body, either from statements in the text or from measuring extents illustrated in maps in the articles.

The final extent of a body in a section may be smaller than the minimum measured value if (1) the section only partially cuts the body or (2) the entire sedimentary body is not be preserved in the sedimentary record due to subsequent erosion. For these reasons, even the shortest valid string that can be produced by the grammar may be too large to fit in the section. The parser can reduce the size of symbols if required, this process is described in a later section.

Conditional Simulation using the Parser

To start, the parser is given the conditional symbols and the start symbol and it searches for an appropriate grammar to match the start symbol. Then it repeatedly searches the list of rules from top to bottom to find a rule whose head matches the left-most non-terminal symbol in the string (initially the start symbol) and performs the replacement specified by the rule. The parser stops when the string consists entirely of terminal symbols. For example, the replacements in the

[4]It should be possible to solve the problem of predicting strings using either a top-down or a bottom-up parser. Initially, a top-down parser was tested and this proved successful.

Ax grammar (Table 2) may occur as follows:

$$Ax \rightarrow AxL + Ax \rightarrow ch + ICh + Ax \rightarrow ch + ISd + Ax$$
$$\rightarrow ch + am + Ax \rightarrow ch + am + ch$$

The produced string is then checked against the conditional symbols. The leftmost conditional symbol is checked first. It the type of conditional symbol does not occur in the string then backtracking occurs repeatedly until a type match is found in the string. Backtracking involves going back through the history of rules used and trying different tails. Once a matching type is found in the string then the extent of the symbols must be made to fit (described in more detail below). As each conditional symbol is matched and fitted the parser moves on to the next conditional symbol to the right, until all conditional symbols are fitted. Finally, the string is fitted to the right boundary conditional symbol. The parser stores the full history of the derivation for purposes of backtracking. Additionally, the parser must monitor the conditional symbols that have already been matched to ensure that they are not changed during backtracking.

Fitting of the conditional data to the simulation is not an iterative process, therefore this method is very fast and expected to be more computationally efficient than the methods using marked point processes. Providing that the grammar rules following the general structure of the examples given, including a recursive rule, then the grammar will always be able to fit the conditional data if the symbols in the conditional data match those in the grammar (this is checked by the parser before it begins the simulation). The successful fitting of the data always occurs because: (1) all the rules are checked during backtracking so the rule which contains the matching symbol will eventually be found although it may take several attempts depending on how many choices are available; (2) if the simulated string (or the relevant part of the string) is of insufficient length then the string can always be increased in length by backtracking to the recursive rule; (3) if the simulated string (or the relevant part of the string) is too long then the parser can truncate the length of the symbols as required to fit.

Fitting Symbol Extent to Match Conditional Terminal Symbols

Each terminal symbol has a position and the extent of the symbol is divided into a left extent and right extent either side of the position. These extents are used to calculate whether the symbol fills the space between conditional symbols, see Figure 5. The left extent is initially set at 95% of the total extent (helps reduce the total number of symbols required to fill the space).

Once a match between conditional and terminal symbols has been found, the extent of each terminal symbol is estimated from the probability distribution.

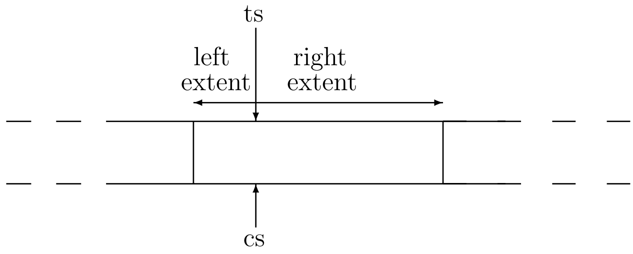


Figure 5. Visual representation of a symbol (rectangle) in a string showing the horizontal position of the terminal symbol (vertical arrow labeled ts) aligned to the position of the matching conditional symbol (vertical arrow labeled cs).

The extents between the current matching symbol's position and the previous matching symbol's position (initially the left boundary) are added. If the space required by the conditional symbols is not filled, then the parser backtracks to a type 0 or type 1 rule which has not had its second tail tried. A new string of terminal symbols is produced. Backtracking is repeated until the space is filled. Two or more consecutive conditional symbols of the same type can match a single terminal symbol if the extent of the terminal symbol is sufficient.

If the space required by the conditional symbols is over-filled, then the symbol size is reduced by repeatedly and randomly selecting a symbol and reducing it by a fixed percentage (or less if less reduction is required). If the extent selected is the left extent of the terminal symbol which matches the current conditional symbol then the amount reduced on the left is added to the right, so that the symbol is not unnecessarily reduced in total extent. Only symbols which lie to the left of this currently matching symbol will have their total extent reduced if required.

TESTING THE PARSER

The 3 potential sources of error in a simulation are: (1) the geological model does not adequately reflect the true geology; (2) the grammar is not a good interpretation of the geological model; and (3) the conditional simulation does not work well. It is not feasible to test this simulation procedure against actual geology because there are no sedimentary successions that are completely documented. However, assuming that the geological model is realistic and the grammar is a good representation of the geological model, it is feasible to test the effects of the conditional symbols on the output of the parser.

We tested the output using the Ax grammar for the axial channel system depositional environment. We need real geological sections as reference sections to compare the simulated sections with. However, because the papers used in this study did not describe any geological sections in sufficient detail to use as reference sections we used the grammar itself to produce a pseudo-real section. We did this by creating a simulated section using only boundary conditional symbols and no other conditional symbols. Then conditional symbols were selected at random positions from the pseudo-real sections for use in the simulations. Thirty simulations were performed using 5 conditional symbols and another 30 simulations were performed using 10 conditional symbols.

The information provided to the parser to produce the sections is (1) the name of the grammar; (2) the total number of conditional symbols including the boundary symbols; (3) the position of the boundary symbols; (4) the name and position of each non-boundary conditional symbol; (5) the probability distribution for the extent of each terminal symbol (uniform probability distributions were used for extent); (6) the probability of selecting tail 1 of a rule.

RESULTS

An example derivation is shown here in detail in order to demonstrate how the parser works in practice. Then the results of the simulations are presented. Two aspects of the results are discussed: the average number of symbols produced in the simulations (this indicates the level of heterogeneity in a simulation); and the total extent and proportion of each type of symbol produced in the simulations.

Derivation History

The derivation history of a single simulation is given in Table 3 and the changing extents of the symbols during this derivation are illustrated in Table 4. These tables show that the parser first produces the simplest valid pattern which is a single channel (ch). However, as this failed to fill the space required by the simulation the parser was forced to backtrack. A total of 4 channels and associated inter-channel deposits were produced before all the conditions were fulfilled. The final parse tree is shown in Figure 6.

Number of Symbols Produced

The first section in Figure 7 illustrates the pseudo-real section from which the conditional symbols for the simulations were extracted. The rest of the sections are some examples of the sections generated using the randomly selected conditional symbols.

When 30 simulations of the Ax grammar were produced using 0, 5 and 10 conditional symbols, the average number of symbols in a section was 7.47, 10.13

Table 3. The Derivation History of a Simulated Section Using Grammar Ax with 5 Conditional Symbols (cs)

Step	String	Rule	Tail
0	Ax	1	1
1	ch	fitted cs 1 and 2, insufficient extent for cs 3, backtrack	
2	Ax	1	2
3	AxL Ax	2	1
4	ch ICh Ax	3	1
5	ch ich Ax	1	1
6	ch ich ch	fitted cs 3, insufficient extent for cs 4, backtrack	
7	ch ich Ax	1	2
8	ch ich AxL Ax	2	1
9	ch ich ch ICh Ax	3	1
10	ch ich ch ich Ax	1	1
11	ch ich ch ich ch	fitted cs 4, insufficient extent for cs 5, backtrack	
12	ch ich ch ich Ax	1	2
13	ch ich ch ich AxL Ax	2	1
14	ch ich ch ich ch ICh Ax	3	2
15	ch ich ch ich ch ISd Ax	4	1
16	ch ich ch ich ch am Ax	1	1
17	ch ich ch ich ch am ch	all cs fitted, end	

Note. The *rule* and *tail* columns indicate which rule is applied to the string next. Rule numbers refer to the order of rules given in Table 2.

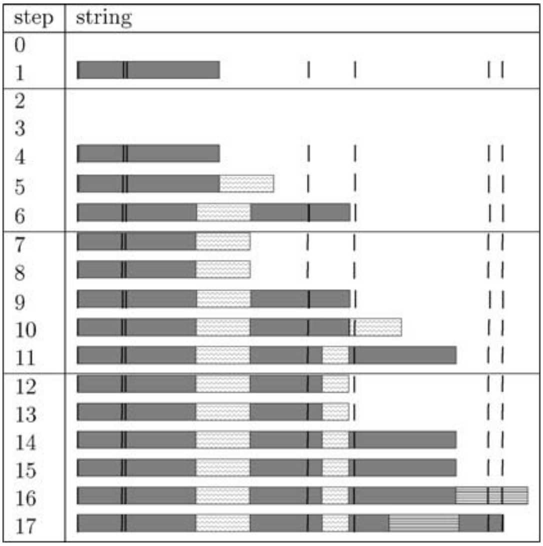
and 9.80 symbols, respectively. The inclusion of conditional symbols in the simulation results in more symbols on average than in simulations with no conditional symbols. This result is not surprising because, in some cases, shortening of the extent of the produced symbols will be required to allow the terminal symbols to match the position of the conditional symbols. The number of conditional symbols used in the simulation had very little effect on the average number of terminal symbols produced.

Relative Proportions of Symbols

Table 5 summarises the extents and proportions of the symbols produced by the simulations. The values given are average values for 30 simulations. Where there is more than one symbol of the same type in a section, the extents have been added.

In stochastic reservoir simulation it is conventional to reproduce the proportion of facies calculated from the vertical well sections in the 3D simulation. If a large number of wells is present in the model area and the wells were drilled in random locations then this is a valid method of controlling the output of the simulation. However, it there are too few wells to adequately sample the heterogeneity

Table 4. The Extents of Terminal Symbols During the Derivation Described in Table 3



Note. Terminal symbols are shown as boxes with length proportional to extent: grey=ch, zigzag=ich, lines=am. Non-terminal symbols are not shown as they have zero extent. Positions of the 5 conditional symbols and begin and end points in the string are shown as short vertical bars.

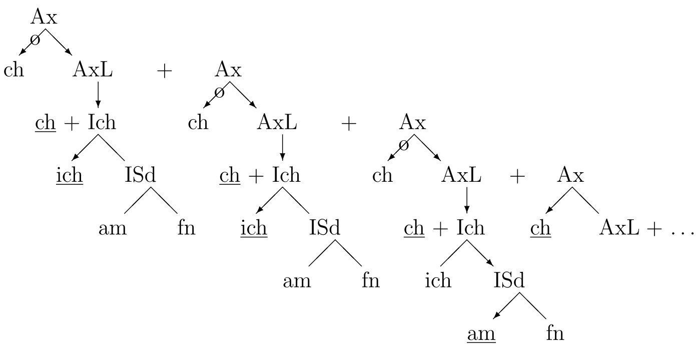


Figure 6. Parse tree for Ax grammar simulation in Table 3. Branches with arrows indicate the branches tried by the parser. Branches with arrows and circles indicate branches tried and then backtracked on. The terminal symbols in the final derived string are underlined.

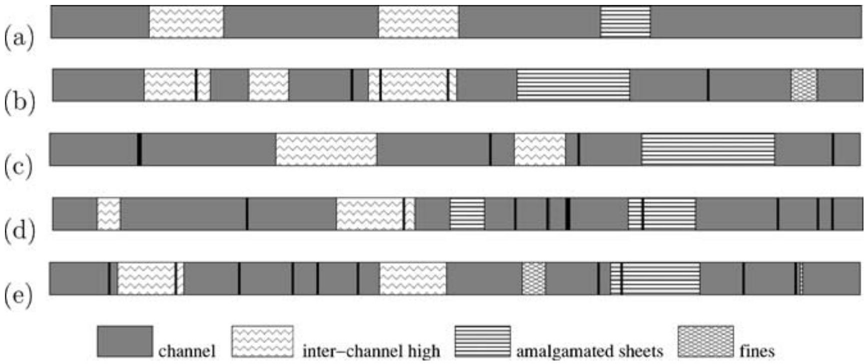


Figure 7. Examples of simulations produced by the Ax grammar. (a) 0 conditional symbols, the pseudo-real section from which the conditional symbols were extracted to create the rest of the sections. (b) and (c) simulations with 5 conditional symbols; (d) and (e) simulations with 10 conditional symbols. The position of conditional symbols is indicated by thick vertical bars.

of a region or if the wells were targeted to areas which were though to contain specific geological features then the method could result in models which are incorrectly biased in their facies proportions.

In the syntactical method the proportions of facies are controlled by the grammar. In the 1D case these proportions are controlled by the probability density functions for the horizontal extent of the units. If the analog model on which the grammar is based is a good model for the reservoir then the proportion of facies should be correct. If the simulation is run without using the well data as conditional symbols, then the facies proportions will only be dependant on the well data in so far as the geologist uses the well data to select an appropriate analog model for the reservoir.

Table 5. Average Extents of Symbol Types and Proportions of Facies Produced by Simulations of the Ax Grammar

Terminal symbol	am	ax	fn	ich	total
original extents	919	11209	0	2871	15000
original proportions	0.06	0.75	0.00	0.19	1.00
5 cs extents	1521	10826	282	2371	15000
5 cs proportions	0.10	0.72	0.02	0.16	1.00
10 cs extents	1212	10925	124	2738	15000
10 cs proportions	0.08	0.73	0.01	0.18	1.00

Note. The proportions for the 5 and 10 conditional symbols (cs) are the averages of 30 simulations.

Table 5 indicates that the introduction of the conditional symbols affected the results of the experiments by up to a 4% difference in the proportions. However, when more conditional symbols were included the difference in proportion of each facies in the simulation compared to the original was 2% or less.

CONCLUSIONS

The grammars are capable of producing simulated sedimentary successions. When no conditional symbols are available for the simulation then the simulation will tend to produce less terminal symbols than a simulation which does use conditional symbols. It is desirable to improve the method used by the parser for matching conditional symbols to remove this effect.

Varying the number of conditional symbols has only a small effect on the number of symbols produced in the simulated section. These results indicate that the number of conditional symbols used should not adversely affect the heterogeneity of the sections produced. The relative proportions of sedimentary facies produced in the simulations improves (i.e. gets closer to the original) when the number of conditional symbols is increased.

This paper introduces 1D conditional simulations for sedimentary successions. It is the ultimate aim of this project to build on this idea to produce 3D conditional simulations. In order to achieve this, first, thickness information will need to be included in the 1D sections to produce 1.5D sections. A vertically stacked succession of 1.5D sections can be used to produce 2D sections. And a series of vertical 2D sections can be used to form a 3D geological model.

ACKNOWLEDGEMENTS

This project is being developed with Woodside Energy Ltd through the R2D3 consortium (a Western Australian oil and gas research collaboration).

REFERENCES

Damsleth, E., Tjølsen, C. B., Omre, H., and Haldorsen, H. H., 1992, A two-stage stochastic model applied to the North Sea reservoir: Soc. Petrol. Engr., v. 44, p. 402–408, 486.

Deutsch, C. V., and Wang, L., 1996, Hierarchical object-based stochastic modeling of fluvial reservoirs: Math. Geol., v. 28, no. 7, p. 857–880.

Deutsch, C. V., and Tran, T. T., 2002, FLUVSIM: a program for object-based stochastic modeling of fluvial depositional systems: Comput. Geosci., v. 28, p. 525–535.

Duan, T., Griffiths, C. M., and Johnsen, S. O., 1996, Syntactic simulation of 1-D sedimentary sequences in a coal bearing succession using a stochastic context-free grammar: Jour. Sediment. Res., v. 66, no. 6, p. 1091–1101.

Duan, T., Griffiths, C. M., and Johnsen, S. O., 1999, A new approach to reservoir heterogeneity modelling: conditional simulation of 2-D parasequences in shallow marine depositional sequences using an attributed controlled grammar: Comput. Geosci., v. 25, no. 6, p. 667–681.

Fu, K. S., 1982, Syntactic pattern recognition and applications. Prentice-Hall, New Jersey, 596 p.

Griffiths, C. M., 1990, The language of rocks–an example of the use of syntactic analysis in the interpretation of sedimentary environments from wireline logs, *in* Hurst, A., Lovell, M. A., and Morton, A. C., eds., Geological Applications of Wireline Logs I: Geol. Soc. Spec. Publ., No. 48, p. 77–94.

Haldorsen, H. H., and Damsleth, E., 1990, Stochastic modelling: Soc. Petrol. Engr., v. 42, p. 404–411, 501.

Hopcraft, J. E., Motwani, R., and Ullman, J. D., 2001, Introduction to automata theory, languages and computation, 2nd Ed: Addison-Wesley, Boston, 521 p.

Johnson, S. D., Flint, S., Hinds, D., and Wickens, H. DeV., 2001, Anatomy, geometry and sequence stratigraphy of basin floor to slope turbidite systems, Tanqua Karoo, South Africa: Sedimentology, v. 48, p. 987–1023.

Hatløy, A. S., 1994, Numerical facies modeling combining deterministic and stochastic methods, *in* Yarus, J. M., and Chambers, R. L., eds, Stochastic modeling and geostatistics: AAPG Computer Applications in Geology, No. 3, p. 109–120.

Knuth, D. E., 1971, Top-down syntax analysis: Acta Informatica, v. 1, p. 79–110.

Miall, A. D., 1985, Architectural-element analysis: a new method of facies analysis applied to fluvial deposits: Earth Science Reviews, v. 22, p. 261–308.

Pickering, K. T., Clark, J. D., Smith, R. D. A., Hiscott, R. N., Ricci Lucchi, F., and Kenyon, N. H., 1995, Architectural element analysis of turbidite systems, and selected topical problems for sand-prone deep water systems *in* Pickering, K. T., Hiscott, R. N., Kenyon, N. H., Ricci Lucchi, F., and Smith, R. D. A., eds., Atlas of deep water environments: Chapman and Hall, p. 1–10.

Srivastava, R. M., 1994, An overview of stochastic methods for reservoir characterisation, *in* Yarus, J. M., and Chambers, R. L., eds., Stochastic modelling and geostatistics: AAPG Computer Applications in Geology, No. 3, p. 3–16.

Stilman, B., 1993, A linguistic approach to geometric reasoning: Comput. Math. Appl., v. 26, no. 7, p. 29–57.

Tyler, K., Henriquez, A., and Svanes, T., 1994, Modeling heterogeneities in fluvial domains: a review of the influence on production profiles, *in* Yarus, J. M., and Chambers, R. L., eds., Stochastic modeling and geostatistics: AAPG Comput. Appl. Geol., No. 3, p. 77–90.

van der Werff, W., and Johnson, S., 2003, High resolution stratigraphic analysis of a turbidite system, Tanqua Karoo Basin, South Africa: Marine Petrol. Geol., v. 20, p. 45–69.