Unsupervised Model Generation for Geological Events

Joaquim Assunção, Paulo Fernandes, Tiago Fischer, and Afonso Sales
PUCRS University - Porto Alegre - BRAZIL
{joaquim.assuncao, paulo.fernandes, tiago.fischer, afonso.sales}@pucrs.br

Keywords: Stochastic models, Simulation methods, Automatic generation, Geologic events prediction, Stochastic automata networks

Abstract
So far, large stochastic models require considerable amounts of time to be created. In fact, to simulate systems or events, there is a constant need to perform an analysis of the system and its variables. In this paper we propose a method to automatically generate Stochastic Automata Networks (SAN) models for geological events. Based on user-defined input data, the method creates a model in SAN formalism for the prediction of geological stratal stacking patterns through time. Although models automatically generated tend to be less accurate, we believe that the time saved compensates for the precision lost.

1. INTRODUCTION
Stochastic models are useful for many purposes. If we are able to perform an accurate representation of a given system, it is possible to retrieve interesting probabilities about the system behavior.

After years working with a Stochastic Automata Networks (SAN) formalism, we successfully reproduced a specific geological phenomenon described in Assunção et al. study [1]. Unfortunately, the final model has required a huge effort by the modelers; due both, to the understanding the phenomenon and the creation of the model itself.

The geological phenomena reproduced by Assunção et al. work are the stratal stacking patterns of a sedimentary basin in the south of Brazil. These stacking patterns provide important information since it impacts the geological formations in the continent margin, strongly influenced by the past conditions of local climate, relief, vegetation, etc.

These kind of data are concerned by paleo events, which need million of years to be covered. Furthermore, some natural paleo events, such as the eroded strata in the geological history, turns impossible to discover the exact information in some periods of time.

The proposed model uses these geological data, aiming to generate an accurate simulation from available data in the literature. Thus, the probabilities reached to the gaps of time, can be used as an indication of possible occurrences in a given period.

Despite the good values achieved, the model was generated specifically for one data set, i.e., one basin, one time scale etc. Now we focus on avoid the time spent developing the model through the automation of the handmade steps that construct the model. Also, the unsupervised generation allows the creation of models almost in real time without the intervention of a modeler specialist.

2. BACKGROUND KNOWLEDGE
This Section shows the basic concepts to understand the generic model; a necessary background to understand the creation of the unsupervised model generator.

2.1. Basins and Geological Phenomena
The filling of sedimentary basins is function of the amounts and types of sediments, depending on some factors such climate and relief. For this reason, sedimentary basins constitute essential records of the climate and tectonic history of the Earth.

Figure 1. Result of the relative sea level changes, in function of the combination between Eustasy and Subsidence.
The study of sedimentary basins is primarily based on drilling and seismic surveys, which provide information on the composition and arrangement of sedimentary rock strata. The configuration of strata results from the interplay between sediment supply and relative base level changes, which defines the accommodation space for those sediments. In marginal sedimentary basins, i.e., basins along a continental margin, the base level is determined by the Relative Sea Level, which, in turn, depends on the global sea level changes (eustasy), the rate of sediment supply and on the vertical movement of the underlying crust, being the rate of subsidence (movement downward) (Fig. 1). Along of the Relative Sea Level variation curve (sinusoid shaped), can be distinguished in four kinds, as summarized in Fig. 2. These are named Forced Regression (FR), Low-stand Normal Regression (LNR), High-stand Normal Regression (HNR), and Transgression (T).

Clearly, the simplified model in Fig. 1 does not represent the whole complexity of processes that may affect the configuration of sedimentary strata such as the variability of sediment supply and shelf gradient. Nevertheless, it emphasizes the dominant role of Relative Sea Level changes and provides a clear picture of the main processes and possible stratigraphic architectures. Contreras et al. [3] estimated subsidence rates and sediment influx using numerical modeling (Fig. 3). These estimates were obtained considering the global sea level (eustatic) curve proposed by Hardenbol et al. [5] re-calibrated to a more recent geological timescale [4].

2.2. SAN

Stochastic Automata Networks (SAN) [6] is a structured Markovian formalism that represents a whole system by the composition of “small” subsystems. In other words, SAN defines a modular way to describe continuous-time Markovian models. Therefore, it is possible to obtain a continuous-time stochastic process related to the SAN model, i.e., the SAN formalism has exactly the same application scope as Markov Chains formalism [8].

Each subsystem in a SAN model is represented in particular by an automaton, i.e., a finite-state machine, where the interaction between automata is expressed by some particular transition rules relate to the automata internal states [2]. The state of a SAN model, known as global state, it is obtained by the combination of the local states of all automata. Figure 4 shows a very simple SAN model and its equivalent Markov chain. In this example, there are two automata, where automaton A has three states ($a_0$, $a_1$, $a_2$) and automaton B has two states ($b_0$, $b_1$).

Moreover, in a SAN model, there are two types of events that change the global state of a model: local events and synchronizing events. Local events change the SAN global state moving from a global state to another that differs only by one local state. Synchronizing events can move simultaneously more than one local state, i.e., two or more automata can si-
multaneously move their local states. Specifically, the occurrence of a synchronizing event forces all concerned automata to fire a transition corresponding to this event. In our SAN model example (Fig. 4), there are three local events (\(e_1, e_2, e_3\)) and one synchronizing event (\(s_1\)).

Each event has an associated rate of occurrence, which describes how often a given event will occur. Each transition between states may be fired as consequence of the occurrence of any number of events. In the model of Fig. 4, the rates of the events \(e_1, e_3\) and \(s_1\) are equal to the constant values \(x_1, x_3\), and \(x_4\), respectively. However, the occurrence rate associated to event \(e_2\) is not a constant value, but a functional rate that is defined in function of the states of other automata. In this example, event \(e_2\) will occur with a rate equal to \(f\) which is equal to \(x_2\) if automaton \(A\) is in state \(a_0\), otherwise this event rate is equal to zero, i.e., the event will not occur.

As the SAN model is a modular description of an equivalent Markov chain, it is possible to obtain this equivalent model by the successive firing of events given an initial state of the structured model. In our example (Fig. 4), assuming as initial state the global state \(a_0b_0\), we can easily find the four states that represent the equivalent Markov chain of this model by the firing of events \(e_1, e_2, e_3,\) and \(s_1\).

3. UNSUPERVISED GENERATION

Our model generation is based on SAN formalism, and applied to a specific nature phenomenon, using a specific input information. The trick here is to keep the model structure by fixing the number of input parameters. Any model generated will hold the same structure, i.e., composed by the same number of automata that always have the same representation.

Our model is limited to generate seven automata. One to control the time passage (called \(Ch\) as in Chronos), i.e., each \(Ch\) state represents a time slot pre-defined according to the input data. Three for the geological events called: \(E\) (for eustasy), \(Su\) for subsidence, and \(SS\) for sedimentary supply. Also, for each geological event, there is a memory automaton to control the number of changes allowed in each time slot. Memories automata, respectively called \(M_E, M_Su,\) and \(M_SS\), are created using parameters collected from the same data that are used to create the others automata.

Although the structure remains the same, due to the input parameters, the generated model tends to have a limited number of reachable states; nevertheless, still there is a need to handle the problem of space state explosion. Therefore, we limited \(Ch\) to 36 states and each of the other six automata to nine states each. In consequence, the larger model we can handle has 19,131,876 states.

Each memory automaton controls its counterpart automaton. It assures that the change of states will not pass more than one state at time and it indicates the number of steps that should be take at the current \(Ch\) state. This process continues until every memory reach the \(DM\) (do not move) state. When a memory automata is in \(DM\) it stops its counterpart geological event automaton. When all memory automata reach \(DM\) state, the \(Ch\) automaton can change his own state to the next time slot. For example, Fig. 4 illustrates this process by using a memory automaton \(M\) with three states (\(DM, M_1\) and \(M_2\)), a \(Ch\) automaton with four states (\(T_0, T_1, T_2\) and \(T_3\)) and one geological event automaton \(G\) with three states (\(H, A\) and \(L\)).

![Figure 5. Execution example.](image)

The example in Fig. 5 depicts the firing of four events affecting the three automata \(G, M\) and \(Ch\). In this figure we start with the initial state in the left hand side, i.e., the global state (\(H, DM, T_0\)), then the synchronizing event \(C_1\) advances to the first time slot (\(T_1\) in \(Ch\)) and at the same time changes \(M\) automaton to state \(M_2\) meaning that \(G\) automaton will have to change, in the future, to two states below, i.e., going from the current \(H\) state to \(L\).

The second event to be fired is \(Dn\), which begins to perform the change needed according to automaton \(M\). After, \(Dn\) is fired again leading to the fourth global state and making \(C_2\) event able to fire and change to the second time slot (\(T_2\) in \(Ch\)). A similar sequence of events continues until reaching the last time slot (\(T_3\)) when event \(rst\) (as for reset) brings back the system to the initial global state.

Fig. 5 example is a simplification considering just one geological event, but our actual model has three geological event represented (\(E, Su\) and \(SS\)). This brings more complexity to \(Ch\) events, but the basic firing sequences remain similar.

The proposed automatic generation consists in receiving three curves describing the evolution of Eustatic sea level; Subsidence; and Sedimentary supply during a time period and to compute adequate automata. This is done in three major
steps: (i) choosing granularity for values in order to determine automata states; (ii) defining the synchronizing events binding the Ch automaton to the memory automata \((M_i)\); and (iii) defining the synchronizing events and functional rates to bind each memory automata to its counterpart geological event automaton.

The first step starts defining a value granularity to each input curves curve, \(i.e.,\) to reduce the dimensionality of each input curve to a set of discrete states. This will define the states of automata \(E, Su\) and \(SS,\) but the alignment of these discretized curves in time slots will also define the granularity of time, \(i.e.,\) the states of automata \(Ch.\) In fact, once the three curves are aligned, every time at least one of them change values a new time slot will be defined. Finally, the last task of this first step is to observe the number of states each geological event automata jumps every time the slot changes. These values will define the number of states needed by each memory automata.

Even thou, this first step is quite straightforward, the granularity decisions made here have the major effect both in the model accuracy and the SAN model state space. Therefore, the choice of granularity brings a classical trade-off decision between model size and quality.

The second step has less decisions to take, but it has much more relations to establish, since it must associate as many synchronizing events as necessary to assure that each change in Chronos automata \((Ch)\) performs the correct change in the memory automata \((M_E, M_{Su}\) and \(M_{SS}).\) This task demands the creation of \(C_i\) events (as in Fig. 5), but the existence of three geological events demands that each \(C_i\) event must synchronize the four concerned automata: \(Ch, M_E, M_{Su}\) and \(M_{SS}.\)

The last step is quid similar to the second one, being simpler by the fact that each memory automaton synchronizes to only one geological event automaton. However, it is more complicated by the fact that each event must take the current time slot into account to memorize if the geological automaton must go up or down, \(i.e.,\) it requires functional rates according to the state of \(Ch.\)

Nevertheless, we are investing a lot of work to analyze different situations and to produce a wider set of models. Consequently, we are also investing in the development of a web service to make our automatic model generation and a corresponding SAN model solver [7] to encourage geologists to produce and analyze our model results.

4. FINAL REMARKS

As a work in progress, our SAN model generator is able to create models for virtually any geological basin in any period of time, considering any hypothesis for the geological events. We are currently experimenting the generation of models for some basins to whom we have some geological previous information.

We started with the one presented in Assunção et al.’s work [1] which is a very detailed handmade SAN model. Although the number of resulting local states of the automatically generated model was significantly smaller than the one in the handmade model, the accuracy was not changed. In fact, the probabilities were different but the outcome, \(i.e.,\) the prediction of FR, LNR, HNR and T stratal patterns in each time slot was not changed. This result is encouraging, but we know that, since Assunção et al.’s work was our starting point, a larger number of experiments must be conducted to improve the confidence in our automatic tool.

REFERENCES


