First Experiments in the application of Computational Mechanics to the analysis of seismic time series

Bertello G., Arduin P.J., Boschetti F., Weatherley D.

Abstract

We apply the Computational Mechanics approach to the analysis of seismic time series data, via the use of the Causal-State Splitting Reconstruction (CSSR) algorithm. We cast the choice of the input parameters for the CSSR algorithm, as well as of the time series symbolisation, into an optimisation problem, which seeks to maximise the predictability of large seismic events. When applied to synthetic data generated by a Cellular Automata, the reconstructed model is able to successfully predict more than 50% of large events. Further developments necessary to extend the approach to real data applications are also discussed.

Introduction

It is generally accepted that the Earth system is complex, and that many geoscientific measurements are the signature of the superimposition of different physical, chemical and mechanical dynamics, acting at very different temporal and spatial scales. In this context, the analysis and interpretation of geoscientific data faces two, somehow contradictory, requirements. From the one hand, we need to address this complexity, accounting for multiple phenomena interacting with one another, chaotic dynamics and large numbers of degrees of freedom. From the other hand, we need to simplify our analysis as much as possible. Simple models are required for human comprehension, in order to reason upon and communicate our understanding of the problem. Simple models are also necessary for computational purposes, since hardware and software improvements will never compensate for the 'curse of dimensionality' underlying most geoscientific problems.

This simplification can be achieved in different ways. A common approach is to simplify the model of the system 'a priori', that is before the data analysis. In this approach, we select 'a priori' the phenomena we want to model, by discriminating between the ones we expect to have a large impact on the process and the ones whose impact we believe to be 'secondary'. Such discrimination is carried out via the use of our 'a priori' physical understanding of the problem or our accumulated experience. In another approach, the simplification is achieved by directly analysing the data, looking for patterns which are most informative of the process we want to study. This approach includes several disciplines like data mining, unsupervised learning, time series analysis, etc. Among these, a number of theoretical developments have been achieved in the field of Computational Mechanics (Crutchfield and Young, 1889, Crutchfield, 1994, Shalizi and Crutchfield, 2001). In a nutshell, the purpose of Computational Mechanics (CM) is to identify patterns which are most informative about a certain process without specifying 'a priori' what the patterns look like, basically moving from pattern analysis to pattern discovery. Obviously, the requirement for 'a priori' assumption/information can not be fully eliminated, rather it is made more general and somehow more subtle. Specifically,

the search for informative patterns is based on two assumptions. First, if a particular pattern is often associated to a specific event, the two must be somehow correlated. Second, given multiple interpretations (models) of a process, we prefer the simplest (or minimum) one. Basically, these two assumptions are the cornerstones of scientific knowledge and consequently we accept them with only minor hesitation.

The purpose of this work is to explore the potential of the Computational Mechanics approach in the analysis of geophysical data, initially by analysing time series of seismic data, with the view of extending the analysis to spatio-temporal data in the future. The straightforward application of the CM algorithms to real data faces a number of practical hurdles, concerning data discretisation (or symbolization) and computational effort. Our main contribution lies in the heuristics we developed in dealing with these problems and in particular in casting the selection of the parameters behind these heuristics into an optimization problem. We test our idea on a synthetic data set of seismic time series data with the aim to optimise the predictability of large size events. However, our approach is not problem dependent and can be easily extended to other applications.

This should be considered as a first step along a possibly long path and we make no claim to have solved, nor significantly improved, earthquake predictability for real data monitoring. Nevertheless, the results are very encouraging and suggest that more work this direction is worthwhile. A number of issues, mostly computational, need to be improved for this work to progress to spatio-temporal data analysis and a later section of this paper contains a list of suggestions for future work.

Computational Mechanics

In this section we review the Computational Mechanics approach and in particular the Causal-State Splitting Reconstruction (CSSR) algorithm which we employ in our work. More details about the algorithm implementation, together with examples of simple binary processes can be found in Shalizi et al (2002), while a theoretical analysis, containing proof of several theorems related to the minimum properties of the reconstruction can be found in Shalizi and Crutchfield (2001).

Let's suppose we want to analyse a sequence of N discrete values $S_i, i = 1...N$, where S_i can take any of k values in an alphabet A, representing measurements taken at discrete time steps from a stochastic process. At any time i, we can divide the series S into two 'half'-series, $\overleftarrow{S}$ and $\vec{S}$, where $\overleftarrow{S} = ..S_{i-2}S_{i-1}S_i$, stepping backward in time, represents the 'past' and $\vec{S} = S_{i+1}S_{i+2}S_{i+3}...$, progressing forward in time, represents the 'future'. Following the same notation as in Shalizi et al (2002), we call $\overleftarrow{S}^L$ and $\vec{S}^L$ histories of length L symbols in the past and in the future, respectively. Also, we call s (and s^L) specific instances of histories belonging to S. Now, let's suppose we scan the series S, looking for occurrences of the history $\overleftarrow{s}$, and we store the symbol $\vec{S}^1$ seen as 'future' in each instance. We can calculate $P(\vec{S}^1 | \overleftarrow{s})$, that is, the probabilities of occurrence of any of

the k symbols in the alphabet A, given the history s, and we call the vector containing these probabilities the *morph* of $\overleftarrow{s}$. We can then define a *causal state* as the collection of all histories $\overleftarrow{s}$ with the same morph (i.e., histories which share the same probabilistic future). More formally, histories $\overleftarrow{s}_1$ and $\overleftarrow{s}_2$ belong to the same *causal state* if $P(\vec{S}^1|\overleftarrow{s}_1) = P(\vec{S}^1|\overleftarrow{s}_2)$.

Given the above definition, the purpose of the CSSR algorithm is to reconstruct the set of the *causal states* of the process and the transition probabilities between the causal states. Following the nomenclature used in Shalizi et al (2002), the combination of causal states and their transition probabilities is called a ε-machine.

The CSSR algorithm can be divided into a number of steps:

1) we start from the null hypothesis that the process is independent and identically distributed. In this case each of the k symbols $a \in A$ is equally likely at each time step and only one causal state is necessary to model the process: the morph of the state is the k-length vector of components $1/k$.
2) we select a maximum history length *max_L* for our analysis. This is the length of the longest history with which we scan the series *S.* For histories of length = 1… *max_L,* we scan the series S, storing both the histories found and their futures. Given an history $\overleftarrow{s}$, its morph is trivially obtained by calculating $P(a|\overleftarrow{s}\ ) = v(a,\overleftarrow{s})/v(\overleftarrow{s})$, for each $a \in A$, where $v(\overleftarrow{s})$ is the number of occurrences of the history $\overleftarrow{s}$ and $v(a,\overleftarrow{s})$ is the number of occurrences of the symbol a given the history $\overleftarrow{s}$.
3) We group histories with similar morphs into the same causal states. This involves three steps: a) first, we need a measure for morph similarity. Real time series are characterised by both the presence of noise and by finite data extent. Consequently we need to relax the requirement of exactly matching morphs $P(\vec{S}^1|\overleftarrow{s}_1) = P(\vec{S}^1|\overleftarrow{s}_2)$ to an approximation $P(\vec{S}^1|\overleftarrow{s}_1) \approx P(\vec{S}^1|\overleftarrow{s}_2)$. In particular we accept $\left|P(\vec{S}^1|\overleftarrow{s}_1) - P(\vec{S}^1|\overleftarrow{s}_2)\right| < \varepsilon$, where ε is a user defined parameter; b) Second, we define the morph for a state as the average of the morph of all histories in that state; c) finally, in order to ensure the reconstruction of a minimum number of states, new states are created only when a history is found which can not match any existent causal state. That is, for each history, we look for an existent state with similar morph and we create a new state only when we can not find any. After these steps, we have a collection of states, grouping all histories found in the time series S according to the similarity between their morphs.
   4) As a last step, we want to make sure that transitions between states, on a given symbol, are unique. That is, we want to make sure that, given any *history* in a state, and a next symbol $a \in A$, the next *state* is uniquely determined. Notice the difference between the occurrence of the *next symbol*, which is stochastic and measured by the morph, and the transition to the *next state*, *given a next symbol*, which we want to be deterministic. In order to do this, for each state, we store the

next state transitions for each history, that is, we store into what state a history goes after seeing a certain symbol. This is also represented by a vector of length *k*, containing, as elements, the next state on each symbol. If a state has two histories whose next state transition vectors are different, we split the state and create a new one.

Once the ε-machine is reconstructed, we can use an approach proposed by Crutchfield and Young (1989) and define as *statistical complexity* of the process the entropy of the ε-machine itself.

The causal states so reconstructed have a number of important features which are discussed in Shalizi et al (2002) and formally proved in Shalizi and Crutchfield (2001). First, we have a means to measure the complexity of a process (for a discussion about the use of statistical complexity versus other information theoretic measures like Kolmogorov complexity, see Crutchfield, 1994). Second, the causal states are minimal, in the sense that they represent the minimum model able to statistically reconstruct the original data and they have minimum statistical complexity. Finally, as discussed above, the ε-machine is deterministic in an automata sense: the probabilities of seeing a certain symbol *a* at time step *i,* is stochastic, but, given the past history and that symbol, the next state of the model is uniquely determined.

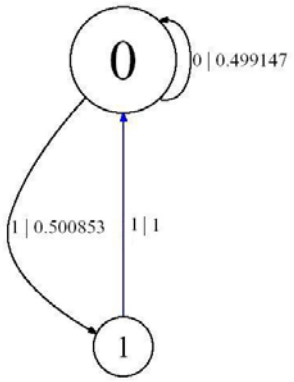

Figure 1. Example of ε-machine reconstruction, obtained by applying the CSSR algorithm to the even process. The nodes of the graph represent states, while the edges represent state transitions. Edge labels represent the morphs.

In Figure 1 we see an example of ε-machine reconstruction, and its representation in terms of a directed graph. It is the reconstruction of the 'even' process, which is described in detail in Shalizi et al (2001). The even process generates a stochastic time series of binary digits, in which odd repetitions of the symbol '1' are forbidden, that is, the symbol '1' can only occur in sequences of even repetitions, like '11', or '1111'. In the graph in Figure 1, states are represented as nodes and transitions as directed edges. Each edge shows the symbol emitted by the state and the probability of its emission. The collection of edges arising from a state represents its morph. Also, the edges point to the next state given the symbol emitted. Figure 1 shows two states, A and B. State A can emit symbol '0' or symbol '1' with even probability. Emission of symbol '0' brings back into the same state A, while emission of symbol '1' leads to state B. State B always emits

‘1’ and goes to A. The process always starts from state A. It is clear from the graph that the CSSR algorithm has reconstructed the correct model of the even process. Symbols ‘0’ and ‘1’ are stochastically generated, thanks to the random probabilities of emitting ‘0’ and ‘1’ from state A. However, odd occurrences of the symbol ‘1’ are forbidden, since every time A emits a ‘1’, another ‘1’ is generated by state B. The visual analysis of Figure 1 allows an easy comprehension of the process underlying the data generation. Unfortunately, as we see next, on complex data sets, the graph generated by the ε-machine can be very complicated and an easy visual interpretation may be impossible. In such case a different kind of analysis needs to be attempted.

Before we proceed to the applications it is important to say a few words about the computational complexity of the CSSR algorithm. The exact run time of the algorithm depends on the data itself and on the complexity of the underlying ε-machine, but its worse case scenario is given by $O(k^{2\max\_L+1})+O(N)$ (see Shalizi at al, 2002), which shows its linear dependence on the number of data samples (N), and its exponential dependence on the alphabet size k, with the exponent depending on the maximum history length. The importance of this formula in our analysis is discussed below.

Application of the CSSR algorithm to real data

In applying the CSSR algorithm (and, similarly, many other time series analysis algorithms) to real data we are faced with a number of hurdles:

1) as we have seen, the algorithm requires symbolic data, that is, each datum has to take one of k values in an alphabet A. However, many geoscientific measurements, ideally, represent values on a continuous range. Even accounting for finite instrument resolution, the number for values allowed by most geophysical instruments defies the concept of a limited alphabet (most symbolic time series analysis applies to binary series). A means to discretise the real valued measurements is thus needed. This requires two decisions: first, how many symbols to use, and, second, how to assign symbols to numerical ranges in the data. No standard method is available in the literature to tackle either problem. For a nice review of symbolisation methods and their application we refer the reader to Daw et al, (2003). The most widely used method is to ensure that the k symbols occur evenly in the symbolised time series. For a binary discretisation, this amounts to choosing the median on the data as the separation criterion and to bin the data accordingly. However Bollt et al (2001) have shown that non optimal symbolisations are often obtained as result of this approach. A more sophisticated approach has been proposed in Kennel and Buhl (2003), whereby consistency in the delayed coordinate representations of the original and the symbolised series is sought. However, the approach applies only to binary symbolisation and requires a numerical optimisation to generate the symbolisation. It should be kept in mind that, as mentioned above, the CSSR algorithm has a computational complexity exponential on the number of symbols k used. It is thus important to limit the number of symbols used in the analysis.
2) the CSSR algorithm computational performance also depends crucially on *max_L,* that is on the length of the longest history with which we scan the data. Usually

max_L is kept below 10. Clearly this affects considerably the kind of analysis we can carry out since this limits the correlation range we can analyse. Since with a low *max_L* we can not access the full amount of memory stored in a process, the process itself will appear more 'random' (see Crutchfield and Feldman, 2003, for a discussion on the subject). However, because of the finite amount of data available for analysis (and the exponentially lower probability of occurrence for longer histories), a very large *max_L* will inevitably explore histories with very low occurrence and consequently not well defined statistics (especially when noisy data and potential symbolisation errors are present). An optimal choice of *max_L* is thus problem dependent and not straightforward.

3) Also, measure of ε, that is, a criterion to access the similarity between two morphs needs to be given. In particular, the CSSR algorithm uses the Kolmogorov-Smirnov (KS) statistical test (Press et al, 2002, page 617) to determine the statistical significance of the morph differences (in which case ε can be seen as the significance level in the KS test). Very low ε suggest more accurate results, but, in real data, may also result in assigning two histories which belong to the same state to two different states only as a result of finite number of occurrences of the histories, noise in the data or inaccurate symbolisation. Of course, large ε results in potentially the opposite error, that is to assign to the same state two histories which actually belong to different states, simply because we confuse real difference in the process with errors due to noise.

It is clear that choices on how to tackle the above points depend on the data at hand as well as the computational resources available. More important though, they also depend on the specific problem we want to address. Our approach is to make this explicit by casting it into a numerical optimisation problem, in which choice of the symbolisation, *max_L* and significance level of the statistical test are chosen depending on what kind of information we want to extract from the ε-machine and, more specifically, on the maximisation of some sort of measure of such information.

In the rest of the paper we describe our approach in the analysis of synthetic seismic time series with the aim to maximise the predictability of seismic events of large size.

Modelling of seismic processes and earthquake prediction

In this section we describe the numerical model used to generate the synthetic seismic time series and discuss the reasons for selection of this model. The numerical model is a cellular automaton earthquake model based upon the Bak et al. (1987) sandpile automaton, with some significant refinements to model the seismic activity of fault systems more precisely. Weatherley et al. (2002) demonstrated that this model produces quasi-periodic cycles of activity culminating in large earthquakes. Large earthquakes are preceded by intervals in which the rate of energy released in smaller earthquakes accelerates. Accelerating energy release is associated with the progressive formation of long-range spatial correlations in the stress field, providing the conditions necessary for the large mainshock. The large earthquakes destroy the long-range correlations, reseting the model to a state in which only smaller earthquakes occur. Accelerating energy release prior to large historic earthquakes has been detected in seismic catalogues from a variety

of different seismic regions of the world (see Jaume and Sykes, 1999 for a review). Bufe and Varnes (1993) demonstrated a method to forecast the time of large mainshocks by fitting time-series of cumulative energy release to a power-law time-to-failure relation. Models in which the predictability of large mainshocks has already been demonstrated, were selected for analysis in this paper so that we could focus upon refinement of the analytical method without the additional complication of using datasets that contain no obvious predictability of large earthquakes.

In detail, the numerical model consists of a rectangular grid of cells, each of which is assigned a failure threshold and a scalar variable presenting shear stress at a fault segment. The failure thresholds of cells are selected from a statistical fractal distribution (Turcotte, 1997) and remain constant throughout a simulation. Initially the stress of each cell is set to zero. Tectonic loading is modelled by gradually increasing the stress of all cells at a uniform rate, until the stress of one cell equals its failure threshold. The cell then fails (representing earthquake rupture of a fault segment). The stress of this cell is reduced to a residual value, a portion of the stress is dissipated from the model (to model loss of energy due to seismic radiation) and the rest of the released stress is redistributed amongst neighbouring cells (the method for redistribution of stress is described below). Stress redistribution typically causes neighbouring cells to reach their failure thresholds. These cells also fail and redistribute stress. Cascades of cell failures represent an earthquake that ruptures multiple fault segments, as typically occurs in nature. Upon completion of a failure cascade, tectonic loading recommences until yet another earthquake is triggered. The size-distribution of simulated earthquakes is found to follow a power-law distribution in agreement with the well-known Gutenberg-Richter (1954) distribution of natural earthquakes.

The features described above are common to a suite of cellular automaton earthquake models. The differences between such models lie in the method for redistribution of stress from failed cells. In the numerical model employed in the present study, we employ a long-range stress redistribution method with dissipative healing of failed cells during failure cascades. The stress redistributed from a failed cell is shared amongst all cells within a rectangular region surrounding the failed cell. The fraction of stress transferred to an individual cell is given by a proxy stress Green's function. Let T_{ij} be the fraction of stress transferred to site i due to failure of site j and let r_{ij} be the distance between site i and site j. The stress Green's Function is given by:

$$T_{ij} = \frac{r_{ij}^{-p}}{\sum_i r_{ij}^{-p}} \qquad (1)$$

where p is the interaction exponent, a model parameter governing the interaction range of the model.

In addition to long-range stress redistribution, we employ dissipative healing of failed cells. A cell that has already failed during a given failure cascade will dissipate any stress transferred to it, until the cascade has completed (i.e. no more cells fail in response to stress redistribution). This dissipative healing mechanism is intended as a rudimentary method for simulating the effect of slip-weakening friction along fault segments. The stress drops to a lower residual level and remains at that level for the duration of earthquake rupture.

The synthetic seismic time-series selected for analysis in the following sections were obtained from models with $p=1.5$ and a relatively large neighbourhood of 39^2 cells, within a model containing a total of 128^2 cells. Weatherley et al. (2002) demonstrated that for these model parameters, large earthquakes may be forecast by examining cumulative energy release time-series. For an interaction exponent p=1, the seismic time-series is dominated by almost periodic system-spanning earthquakes (an obviously predictable time-series) however for p=1.5, the time series of earthquakes is considerably more irregular with quasi-periodic cycles of large earthquakes of variable size. The more irregular time-series was selected so that the ε-machine constructed was not trivial yet the predictability of large mainshocks was still not in question.

ε-machine optimisation

We can now describe in detail our optimisation approach. We start with the 'a priori' choice of an alphabet of 5 symbols, ('0','1','2','3','4'). This is the only 'a priori' defined parameter and in a later section we describe how this parameter could also be included as an unknown in the optimisation. We then decided to assign the last symbol ('4') to seismic events of large size, that is, to earthquakes. The exact definition of 'large size' is one unknown in our optimisation, as we see next.

Given this, the purpose of our optimisation is to reconstruct the ε-machine which maximises the predictability of 'large events' (earthquakes), that is, the ε-machine with maximum predictability in the emission of the symbol '4'. Notice the difference between maximising the predictability of the process itself and maximising the predictability of the production of a specific symbol. In earthquake prediction we are not interested in predicting the very frequent, and mostly unnoticeable, small events which occur daily in large areas on the globe. What we aim at is to predict events of extreme size, which can affect populated areas. For this reason we do not aim at maximising the general predictability of the ε-machine applied to our data set, rather we aim specifically at maximising the predictability of generating an earthquake, which, with our symbol choice, corresponds to generating the symbol '4'.

In our optimisation scheme we employ the CSSR algorithm to generate the ε-machine from the seismic time series and then measure the earthquake predictability, as we describe below. Consequently, the unknowns in our optimisation problem (the parameters we want to reconstruct) are the input to the CSSR. Since the CSSR algorithm can be applied to real data only after symbolisation, the iterative process also needs to include the symbolisation, and the parameters controlling the symbolisation are further unknowns in our problem.

The list of the unknowns we want to recover via optimisation is as follows:

1) we have 5 symbols and consequently we need to bin our seismic measures into 5 bins. We thus require 6 bin boundaries. We assume that the lowest limit of the first bin is $-\infty$ and the upper limit of the last bin is $+\infty$. Thus we are left with 4 boundaries to determine, that is with 4 unknowns. Notice that the last boundary

(between symbol '3' and '4') is what effectively defines a 'large event'. This means that the definition of 'large event' is itself heuristic, and it represents a compromise between the requirement to describe events of large size and our need to maximise their predictability;

2) the fifth unknown is *max_L,* the longest history we analyse;
3) the sixth unknown is the significance level in the KS statistical test in the CSSR algorithm, which accounts for morph similarity;
4) finally, we have seen above that ε-machine reconstructions are most often carried out with *max_L* shorter than 10. Experience with seismic modelling reveals that histories of this length are far too short to account for useful correlation in the data. Rather, histories of a few hundreds samples are needed. Since this can not be achieved computationally, we decided to subsample the time series. We thus a) subdivide the time series into non overlapping windows, b) for each window, we assign a value equal to the sum of the seismic events within the window, c) we then account for the difference between such value and the value at the previous window position (finite difference between the integral values). This is the (real valued) series which is then symbolized according to point (1) above. The size of the window used in this down-sampling is the last unknown in the optimisation.

This results in a 7 dimensional problem. Because of its non-linearity we decided to address the optimisation via the use of a real coded Genetic Algorithm (GA). Specific details of the GA used in this problem can be found in Boschetti et al (1995).

ε-machines Performance Measure

The last component of the method is the measure of performance for the ε-machine reconstruction. As explained above we want to maximise the predictability of generating the symbol '4'. This means that, among many possible ε-machines resulting from different combinations of input parameters, we look for one(s) for which the possibility of emitting the symbol '4' is limited to fewer states, and within these states, the probability is very high. A ε-machine with this property would be preferred to one for which, for example, the possibility of emitting the symbol '4' is spread over many more states with lower probability.

In our calculation we also need to account for the fact that our data set is finite and that probabilities are calculated empirically. That is, we want to account for the fact that probabilities calculated for events which occurred only a few times are less reliable than those of events with high occurrence.

Let's call E the set of states e which can produce an earthquake (that is, that can emit the symbol '4'). For each e, let's define as $p_{e\to 4}$ the probability of emitting the symbol '4' and $\eta_{e\to 4}$ the number of times the transition has occurred. Our optimisation criterion is to maximise

$$\chi = \frac{\sum_{E} \eta_{e\to 4} p_{e\to 4}}{N} \qquad (2)$$

where N is the total number of earthquakes occurring in the process.

Since $N = \sum_E \eta_{e\to 4}$ we have $\chi \le 1$.

By calling p_e the probability of occurrence of the state *e* we notice that

$\eta_{e\to 4} = L p_e p_{e\to 4}$, where L is the length of the date series. Also

$$N = \sum_E \eta_{e\to 4} = L \sum_E p_e p_{e\to 4}$$

We can thus re-arrange equation 2 to obtain

$$\chi = \frac{\sum_E p_e p^2{}_{e\to 4}}{\sum_E p_e p_{e\to 4}} \qquad (3)$$

Finally, the overall algorithm can be summarised as follows:

1) initialise the Genetic Algorithm, that is, generate a number of individuals, each characterised by an initially random array of 7 numbers, representing the 7 unknown parameters in the problems;
2) for each GA individual:
   a) down-sample the time series using a window size defined by parameter 7,
   b) symbolise the time series by using the bins defined by the parameters 1-4,
   c) on the symbolised sequence, run the CSSR algorithm, with *max_L* defined by parameter 5 and the statistical accuracy determined by parameter 6 and
   d) on the resulting ε-machine, calculate the predictability of large events, as defined in equation (3); this is the measure of fitness for the GA individual;
3) once all the individuals in the GA generation have been run, use the fitness of the GA individuals to produce the new GA generation;
4) iterate 2-5 until a predefined number of function evaluation has been reached.

Results

Optimisation. We ran the GA with a population of 20 individuals for 20 generations, for a total of 400 function evaluations. Since the GA produces stochastic results, we performed four optimisation runs. The results are fairly consistent, giving us some sort of mild reassurance that the problem is not strongly non unique (to assert this with more confidence a more exhaustive analysis is needed).

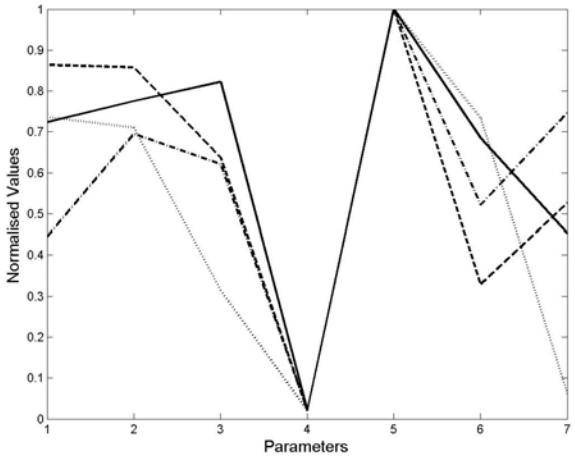

Figure 2. Parallel axis representation of the results from four optimisation runs. Each line represents a solution, that is a point in 7 dimensional space. The horizontal axis represents the 7 parameters while the vertical axis gives their values, normalised between the minimum and maximum allowed range.

Figure 2 displays a parallel axis view of the 4 solutions obtained as output of the GA optimisation. The horizontal axis shows the 7 parameters and the vertical axis shows their values, normalised to account for the maximum and lower bounds (for the choice of the bounds for the inversion we refer the reader to the Discussion section below). Points joined by a line represent individual GA solutions. A few conclusions can be drawn from this figure. First, the solutions give a quite clear indication of what is the 'optimal' definition of 'large event' for prediction application. Parameter 4, which represents the boundary between symbols '3' and '4', is consistent and quite low. The algorithm finds it easier to predict when a combination of smaller and larger size earthquakes (i.e. smaller 'large events') will happen, than to selectively predict very large earthquakes (i.e., very large 'large events'). Second, the search tends to prefer large *max_L,* and actually selects the maximum value allowable. This is not surprising and suggests that long range correlations exist in the data which would enhance our prediction exercise. In other words, computational limitations do not allow us to exploit the full amount of memory stored in the system.

ε-machine	Large Event Predictability	Statistical Complexity
GA solution 1	0.55	10.68
GA solution 2	0.47	10.03
GA solution 3	0.51	11.24
GA solution 4	0.53	9.97
Random Machine 1	0.07	7.03
Random Machine 1	0.14	8.48
Random Machine 1	0.04	7.68
Random Machine 1	0.20	5.44

The first 4 rows in Table 1 show the numerical values of the predictability (given by Equation 3 above) for the 4 solutions together with the statistical complexity of their ε-machines. In all cases the predictability is close to .5. This means that, once we see a state which can lead to an earthquake, we have 50% probability to successfully predict the event. This information alone does not allow us to evaluate the quality of the optimisation result. A rough idea of the optimisation performance can be obtained by calculating the predictability and statistical complexity of ε-machines arising from random input parameters chosen within the same allowable ranges used in the optimisation. Four of such choices are also shown in the last 4 rows in Table 1. Three features can be noticed:

1) the large event predictability is considerably larger for the 'optimised' ε-machines;
2) the non optimised ε-machines show a considerable variability in predictability covering almost one order of magnitude;
3) the statistical complexity of the optimised machine is consistently larger than of the non-optimised one, suggesting that a machine of large statistical complexity is required in order to improve predictability.

Emergent Features. As we mentioned in the Introduction, one of the purposes of Computational Mechanics is the unbiased detection of predictive patterns in the data. These patterns carry most predictive information about the process under analysis. Crutchfield (1994) defines these features as emergent, while Shalizi (2001) gives a formal definition and measure of their emergent properties. In the context of our application, emergent features are the ones which enable us to better predict the occurrence of large seismic events. These are the features which, in a hypothetical real time monitoring system, would warn the analyst of an increasing risk of earthquake occurrence.

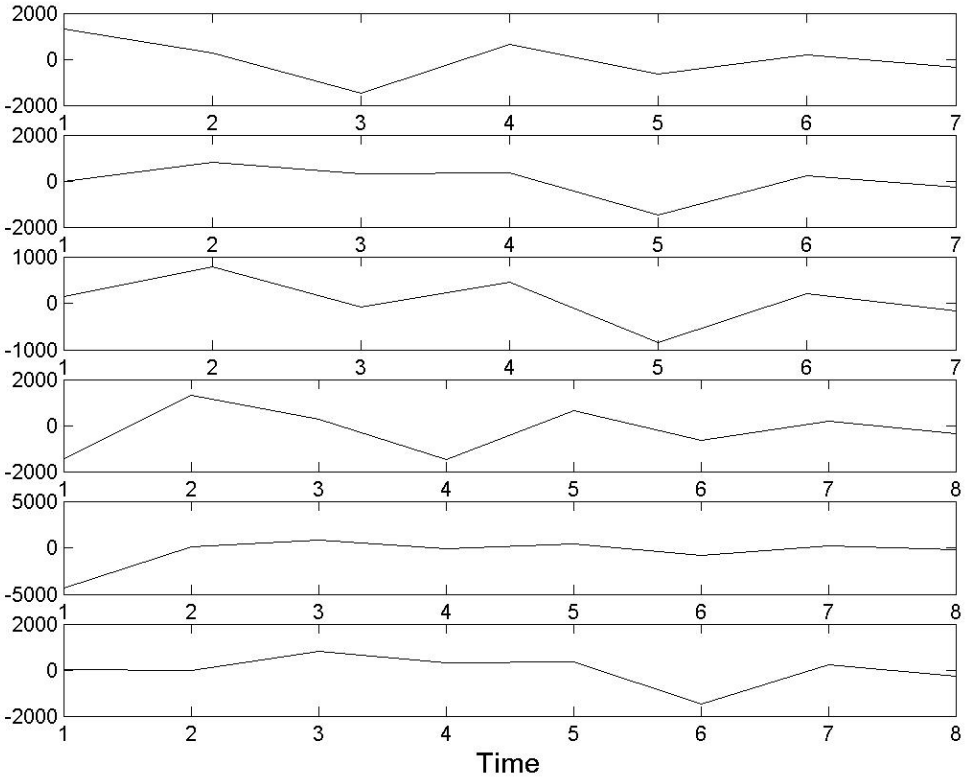

Figure 4. Example of emergent features detected by the optimisation process. Once these features are seen in the time series, the probability of seeing a large event at the next window position is 1.

An example of such features is given in Figure 4. These are the histories belonging to one of the states reconstructed by the ε-machine. This state has a probability one of emitting the symbol '4', which means that, every time we see this state we are certain (within computation errors) to see an earthquake at the next time step.

Even more interesting are *sequences* of states with deterministic symbol emission. These sequences start with a state which can emit a single symbol, proceeds to another state which also can emit a single symbol and so on. The sequence ends only when a state with non deterministic symbol emission is found. Basically the machine enters a cycle of several time steps. We can see these sequences as a sort of 'meta states'. One example is given in Figure 5. Once we enter the first state, we know the sequence will be followed until the final state is reached, after which the dynamics is again stochastic. In the case shown in Figure 5a the symbol '4' is emitted five times, which means that six earthquakes are produced. In this case, not only we can predict the occurrence of an earthquake at the next time step, but also of a pattern of earthquake happening a different time steps in the future.

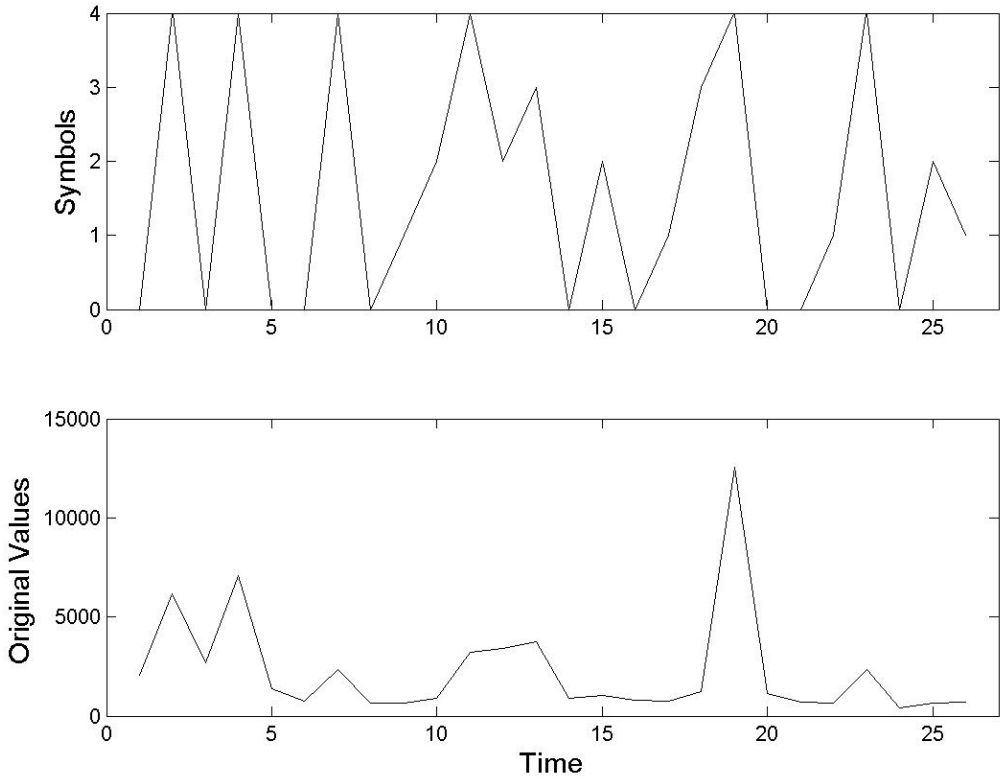

Figure 5. Example of a sequence of states with deterministic symbol emission. The top Figure shows that 5 symbols '4' are emitted during the sequence, which corresponds to 6 'large events'.

Discussion

Although we have applied ε-machine reconstruction to synthetic rather than real data, to our knowledge this work represents one of the more challenging tests of the Computational Mechanics approach thus far. First, we did not limit ourselves to binary symbolisation. This considerably increased the computational effort of the machine itself, but also, we believe, gave us the ability to keep more structure, and consequently information, in the symbolised sequence. Second, by using an optimisation approach, we limited the user intervention in the determination of both the symbolisation and the ε-machine input parameters. We believe this is in line with the unbiased approach underlying Computational Mechanics, in which we aim to detect structure and patterns with the minimum amount of 'a priori' intervention.

The fact that our ε-machine reconstruction is achieved via global optimisation, and that several optimisation runs give fairly consistent ε-machine reconstructions give us a certain level of confidence that ε-machine itself successfully captures the dynamic of the process we studied. This seems to be confirmed also by the fact that ε-machine performance is fairly consistent when applied to different data sets.

It is however also clear that this work has to be seen as a first investigation into the potential in the application of Computational Mechanics and that a number of limitations need to be address before real, large geophysical problems can be tackled. First, the limitation in the maximum history length we can analyse forced us to down sample the original time series. We decided to use the finite difference between the integrals of the event size between adjacent windows as a pre-processed time series. This inevitably smooths the information in the original data. While this considerably improved the ε-machine reconstruction, it also makes its prediction less precise. What we predict is not the occurrence of a large event as an exact time step, but over a time window. With current computational resources, a considerably longer history length could be used only at the cost of reducing the number of symbols used in the time series discretisation. Whether better results could be obtained in this fashion is an avenue worth exploring and we envisage testing this in our future work.

An option is to make the number of symbols a further variable in the optimisation. This would require the choice of a different optimisation routine, since in this case the number of unknowns in the optimisation problem would be variable. In line with the Computational Mechanics aim to make as little 'a priori' decisions on the algorithm implementation as possible, we could be even more audacious and include as unknowns in the optimisation other criteria for the down sampling operation, that is, different criteria for choosing how to represent the data within a time window. We have used the sum of the events size, but measures like the extrema could also be a reasonable choice. Providing the optimisation with several of these choices could make the algorithm more flexible in dealing with different kinds of data sets.

Another inevitable 'a priori' choice enters in the definition of the ranges for the parameters in the optimisation, that is, the maximum and minimum value an unknown is allowed to take. On the one hand we want to make these ranges as broad as possible, in order to avoid removing a useful part of the parameter space. In practise, this may make the optimisation hopelessly slow, and a clever choice of the ranges can be a crucial factor in obtaining successful results. A natural choice could be a multi stage approach, in which a first broad search on a large parametric space is performed and then refined into a smaller subdomain of the search space once favourable solutions are found. A final local optimisation with a local steepest descent algorithm could finally further refine the global search solution. All these options currently face the challenge of the heavy computational cost of the CSSR algorithm, which, on a standard PC, can take from 30 seconds to a few minutes for a single ε-machine reconstruction, depending on the input parameters. Such reconstructions becomes more and more time consuming for large ε-machines of high statistical complexity. These are the ones which also provide good measure of predictability, which results in further refinement of the solution being increasingly costly.

All the limitations we listed so far are related, one way or another, to computational issues, and could be explored if large computational resources were available. The most obvious approach seems to be distributed computing, since several optimisation algorithms for global optimisation (including the GA we used) can be easily parallelised. Here we imagine a scenario in which several copies of the CSSR run in parallel in a distributed environment, and the results are sent to the central optimisation code which

produces the new generation. Shalizi et al. (2004) also envisage a Computational Mechanics approach for spatio-temporal data in which time series of 2D or 3D data are analysed. This is naturally attractive for geoscientific studies since it would allow analysis of the time evolution of 2D images or even 3D geological models. Now the histories to analyse in the CSSR algorithm become 2D or 3D, which further increases the computational burden. In these cases a parallelisation of the CSSR algorithm itself may be needed even for simple (non-optimised) ε-machine reconstructions.

Widening the scope of Computational Mechanics to geoscientific applications needs to address fundamental problems which go beyond simply computational power. The first is that geoscientific data rarely come in large quantities, and, most important, rarely come sampled regularly in time, or space. Irregularly spaced (in time or space) data can not for the moment, be analysed via the CSSR algorithm. We believe this will be the most crucial hurdle to address in the extension of the approach to real data. A hint of a way to proceed may be provided in Kennel and Mees (2002) where non contiguous templates are used for time delay time series analysis, although the templates used in their work are regular along the time series and this is also a luxury which may not be available in real data applications.

Another challenge is represented by the amount of data usually necessary for ε-machine reconstruction, which usually is in the order of several thousands. This also is not always available in geoscientific studies. Combining data sets collected in different areas which we expect underwent similar originating processes may be an option but would first require we have learnt how to combine different, non regular data sets as mentioned above.

Conclusions

We have applied the Causal-State Splitting Reconstruction (CSSR) algorithm to synthetic seismic time series data obtained via a CA-like model. By using a Genetic Algorithm we have optimised the input parameter both of the CSSR algorithm and of the time series symbolisation in order to maximise the predicability of large seismic events. We have obtained models which are able to successfully predict occurrences of large events in more than 50% of cases. The method also allows to calculate the statistical complexity of the model and well as to detect the emergent structures in the time series, that is the structure with maximum predictability capabilities. Further challenges to the extension of the method to monitor real seismic events have also been outlined.

Acknowledgements

Weatherley acknowledges the support of the ARC and the University of Queensland. Cellular Automaton simulations were performed using the Australian Computational Earth System Simulator, a 1.1TFlop SGI Origin 3800 supercomputer.

References

Bak, P., Tang, C. and Weisenfeld, K. (1987) Self-organised Criticality: an Explanation of 1/f Noise, *Phys. Rev. Lett.*, 59(4): 381-384.

Bollt, E.M., Stanford, T., Lai, Y.C. and Zyczkowski, K. (2001). What Symbolic Dynamics Do We Get with a Misplaced Partition? On the Validity of Threshold Crossings Analysis of Chaotic Time-Series. *Physica D* **154**: 259-286.

Bufe, C.G. and Varnes, D.J. (1993) Predictive Modelling of the Seismic Cycle of the Greater San Francisco Bay Region. *J. Geophys. Res.*, **98**(B6): 9871-9883.

Crutchfield, J. P. (1994) The Calculi of Emergence: Computation, Dynamics, and Induction. *Physica D* **75**: 11-54.

Crutchfield, J.P. and Feldman, D.P. (2003). Regularities Unseen, Randomness Observed: Levels of Entropy Convergence. *CHAOS* **13**(1): 25-54.

Crutchfield, J. P. and Young, K. (1989). Inferring Statistical Complexity. *Physical Review Letters* **63**: 105-108.

Daw, C.S., Finney, C.E.A., Tracy, E.R. (2003). A review of symbolic analysis of experimental data. *Review of Scientific Instruments* **74**: 916-930.

Gutenberg, B. and Richter, C.F. (1954). Seismicity of the Earth and Associated Phenomena, Princeton University Press, Princeton, New Jersey.

Jaume, S.C. and Sykes, L.R. (1999). Evolving Towards a Critical Point: a Review of Accelerating Seismic Moment/Energy Release Prior to Large and Great Earthquakes. *Pure Appl. Geophys.*, **155**: 279-305.

Kennel, M.B. and Buhl, M. (2003). Estimating good discrete partitions from observed data: symbolic false nearest neighbors. *Physical Review Letters* **91**: 084102.

Kennel, M.B. and Mees, A.I. (2002). Context-tree modeling of observed symbolic dynamics. *Phys. Rev. E*, **66**: 056209.

Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T. (2002) Numerical Recipes in Fortran 77: The Art of Scientific Computing, Cambridge University Press, Cambridge.

Shalizi, C. (2001). Causal Architecture, Complexity and Self-Organization in Time Series and Cellular Automata. PhD Thesis, http://www.cscs.umich.edu/~crshalizi/thesis/.

Shalizi, C. R. and Crutchfield, J. P. (2001). Computational Mechanics: Pattern and Prediction, Structure and Simplicity. *Journal of Statistical Physics* **104**: 819--881.

Shalizi, C.R., Shalizi, K.L. and Crutchfield, J. (2003). An algorithm for pattern discovery in time series. *Santa Fe Insitute Working Paper* 02-10-060. arXiv.org/abs/cs.LG/0210025.

Shalizi, C., Shalizi, K. and Haslinger, R. (YEAR) Quantifying Self-Organization with Optimal Predictors, Physical Review Letters, 93(11): PAGES

Turcotte, D.L. (1997). Fractals in Geology and Geophysics. Cambridge University Press, Cambridge.

Weatherley, D., Mora, P. and Xia, M.F. (2002). Long-Range Automaton Models of Earthquakes: Power-law Accelerations, Correlation Evolution and Mode-Switching. *Pure Appl. Geophys.*, **159**: 2469-2490.