

Imperial College London
Department of Earth Science and Engineering
MSc in Applied Computational Science and Engineering

Independent Research Project
Project Plan

CoNuS-Viz: A Visualization Package for the Concurrent Library for Numerical Simulations

by
Fazal Khan

Supervisor:
Professor Cédric John

Email:
fk4517@ic.ac.uk

GitHub login:
acse-fk4517

June 2020

1 Introduction

1.1 Forward Modelling

Many real-world scientific processes may be formulated as a model that evolves with time. Formulating a process this way allows it to be simulated using computational techniques. Suppose \vec{y} are the variables of interest stacked in a vector, then a forward model describing their evolution with respect to time may be written as:

$$\vec{y}_{t+1} = f(t, \vec{y}_t) \quad (1)$$

1.2 An Example Use Case: Stratigraphic Forward Modelling

Forward modelling has widespread applicability in many different scientific fields ranging from finance to physics. In order to make this discussion more concrete, I will focus on a particular example known as Stratigraphy. Stratigraphy is concerned with the processes which lead to the organization of rocks in space and time. Stratigraphic Forward Models (SFM) are forward models which have been used to simulate realistic stratigraphic processes based on a set of reasonable geologic parameter values, such as tectonic movement and erosion [1].

When studying stratigraphic observational data, it is often important to understand what set of initial geological conditions may have caused the stratigraphic patterns in the observations. Problems of this type can be generally classified as inverse problems [2]. SFMs can be used to generate a set of outputs which in turn are compared to observations. Outputs are generated systematically for a range of initial parameter values, and those outputs which are close to the observations within a pre-defined bound are then taken as solutions to this inverse problem. Used in this way SFMs are an integral part of solving inversion problems in stratigraphy. More generally, solving inverse problems in this way can be useful for many other domains apart from stratigraphy.

1.2.1 Computational Intensity of large Forward Models

Large forward models are computationally intensive methods which may require many outputs to find a satisfactory solution. Coming back to the example of using SFMs to solve inverse problems; in addition to generating many outputs, the set of geological parameters, \vec{y}_i is often very large too [2]. For these reasons computational techniques that can deal with large models efficiently are very important. One way to address this is to use concurrency and parallelism when performing calculations on the underlying data. Such techniques have been used in scientific modelling for increasing performance and reducing computation time. Domains ranging from biophysical modelling to computational fluid dynamics have benefited from technologies such as GPUs that allows these techniques to be more readily exploited [3].

1.3 The Concurrent Library for Numerical Simulations (CoNuS)

CoNuS is an experimental open-source library developed by Professor Cédric John from the Carbonate Research Group at Imperial College London. The library is written in the Scala programming language and has two general aims:

1. **Abstraction:** Abstract away unnecessary implementation details from the user so they can focus on the modelling
2. **Performance:** Be performant enough to run large, concurrent models in a reasonable amount of time

1.3.1 Abstraction

Conventional forward models generally require users to define a grid of variables, some type of mathematical equation that defines how variables evolve with time, and a loop to actually carry out calculations. CoNuS abstracts away these common mechanics of forward models so that users from any discipline, regardless of their programming experience, can compose models by focusing on the mathematical basis for how variables evolve. Users essentially define equation (1), and let the model run.

1.3.2 Performance

CoNuS leverages the Scala programming language and functional programming techniques. Functional programming techniques are excellent at creating data recursion schemes that can exploit concurrency and parallelism; an example is Google’s famous MapReduce algorithm which processes large datasets in parallel [4].

Since it is written in Scala, CoNuS benefits from several language features that make writing concurrent code easier. Scala is a statically typed, compiled language that targets the Java Virtual Machine (JVM) [5]. A strong type system can be very useful in designing software as it allows a programmer to discover variants and trigger generalizations that simplify design [4].

Scala also combines both object-orientated and functional programming concepts to create highly expressive programs that can support many levels of abstraction over the underlying computation. One such abstraction is the Scala actor model which allows concurrent code to be written ergonomically in a type-safe manner. Originally the actor model was implemented to deal with the JVMs shared-memory thread model, which suffered from high memory consumption and context switching costs [6]. Making use of functional programming techniques available in Scala, such as partial functions, the original actor model implementation was able to run on unmodified JVMs, requiring no special compiler support. The Scala actors library has since been deprecated and replaced by the Akka library, which is built on the same concept and allows code to be run on clusters [7]. Making use of actors on clusters is on the roadmap for the CoNuS library.

1.4 Comparison to Other Approaches

Currently there are not many open-source SFM tools available, especially those making use of Scala; most tools are closed-source commercial software. Examples of commercial software include DionisosFlow [8], which the Carbonate Research Lab has recently used to model carbonate systems in Northern Oman [9]. Although these tools are easier to use vs. coding up a forward model from scratch, they generally lack flexibility and are written for a narrow problem domain. CoNuS in contrast aims to be a general forward modelling library that is also easy to use.

Software Package	Type	Scope
CoNuS	Open-source	General
DionisosFlow	Commercial	Geophysical
Schlumberger GPM	Commercial	Geophysical
pyGIMLi	Open-source	Geophysical

Table 1: Comparison of CoNus to alternative modelling software.

1.5 Project Aims

As CoNuS is very early stage and experimental, it lacks an integrated visualization library. The aim of this independent research project will therefore be to build the visualization package, CoNuS-Viz, that provides integrated charting functionality to the CoNuS library. This will be an original contribution to the open-source community and will allow users of the library to visualize models built not only with CoNuS.

References

- [1] Timothy A. Cross and Margaret A. Lessenger. “Construction and Application of a Stratigraphic Inverse Model”. In: *Numerical Experiments in Stratigraphy: Recent Advances in Stratigraphic and Sedimentologic Computer Simulations*. SEPM Society for Sedimentary Geology, Jan. 1999. ISBN: 9781565761841. DOI: 10.2110/pec.99.62.0069. URL: <https://doi.org/10.2110/pec.99.62.0069>.
- [2] Ian Lerche. “Inverse Methods and Uncertainties in Stratigraphy and Basin Analysis”. In: *Energy Exploration & Exploitation* 19.4 (2001), pp. 397–462. DOI: 10.1260/0144598011492363. eprint: <https://doi.org/10.1260/0144598011492363>. URL: <https://doi.org/10.1260/0144598011492363>.
- [3] J. D. Owens et al. “GPU Computing”. In: *Proceedings of the IEEE* 96.5 (2008), pp. 879–899.
- [4] Ralf Lämmel. “Google’s MapReduce programming model — Revisited”. In: *Science of Computer Programming* 70.1 (2008), pp. 1–30. ISSN: 0167-6423. DOI: <https://doi.org/10.1016/j.scico.2007.07.001>. URL: <http://www.sciencedirect.com/science/article/pii/S0167642307001281>.
- [5] *Scala*. <https://www.scala-lang.org/>.
- [6] Philipp Haller and Martin Odersky. “Scala Actors: Unifying thread-based and event-based programming”. In: *Theoretical Computer Science* 410.2-3 (2009), pp. 202–220. ISSN: 0304-3975. DOI: 10.1016/j.tcs.2008.09.019. URL: <https://dx.doi.org/10.1016/j.tcs.2008.09.019>.
- [7] *Akka*. <https://akka.io/>.
- [8] *dionisosflow*. <https://www.beicip.asia/software/explorations/dionisos-flow/>.
- [9] Marya Al-Salmi, Cédric M. John, and Nicolas Hawie. “Quantitative controls on the regional geometries and heterogeneities of the Rayda to Shu’aiba formations (Northern Oman) using forward stratigraphic modelling”. In: *Marine and Petroleum Geology* 99 (2019), pp. 45–60. ISSN: 0264-8172. DOI: <https://doi.org/10.1016/j.marpetgeo.2018.09.030>. URL: <http://www.sciencedirect.com/science/article/pii/S0264817218303945>.