LLNL-TR-758521

LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Dynamic reduced order modelling (ROM) of chemical and mechanical processes in CO2-cement systems

X. Chen, J. Iyer, S. Carroll

September 19, 2018

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Dynamic reduced order modelling (ROM) of chemical and mechanical processes in CO_2-cement systems

Xiao Chen[1], Jaisree Iyer[1], Susan Carroll[1]

[1]Lawrence Livermore National Laboratory

Table of Contents

List of Figures

List of Tables

ABSTRACT OR EXECUTIVE SUMMARY

Damaged wells pose a significant risk of leakage of reservoir fluids stored in a geological CO_2 storage site. The leaking CO_2 can react with well cement and alter its chemical, mechanical, and hydraulic properties. Recently, we have developed an experimentally-calibrated model that couples two-phase flow, reactive transport of brine, cement geochemistry, and geomechanics to predict the leakage of reservoir fluids through a fractured pathway in a cemented well (Iyer, et al. 2018, Iyer, et al. 2017, Walsh, et al. 2013, Walsh, et al. 2014, Walsh, et al. 2014)

We are developing a reduced order model (ROM) to rapidly assess the evolution of leakage flux from a well for a wide range of CO_2 storage sites scenarios, because the coupled numerical model is computationally very expensive. The coupled numerical model was used to run simulations needed to train the ROM by applying quasi-Monte Carlo sampling of seven inputs parameters, with some physical restrictions, to ensure efficient and additive sampling. The ROM uses the reservoir overpressure and saturation, the fracture aperture, length, and width, the normal stress acting on the fracture, and the reservoir depth as inputs. To ensure a sensitive response the input variables and the output leakage rate were post-processed using a logarithmic transformation followed by normalization.

The coupled model solves several partial differential equations that describe the spatial and temporal evolution of pressure, velocity, concentrations, and extent of reaction. As a result, the solution at any time depends on the solution at previous times. To preserve this notion of memory the ROM was developed such that the leakage rate at any given time depends not only on the input parameters like pressure, saturation, etc., but also on the predicted leakage rate in the previous time steps. This process was found to perform significantly better than any approach that ignored this notion of memory.

The dynamic ROM is built upon time-dependent machine learning (ML) algorithms and predicts the evolution of the leakage flux of both CO_2 and brine based on a set of input variables for 100 days. To initiate the ROM, information about the leakage rate at 0 day and 0.1 day is required. This was obtained by developing a ROM using ML algorithms to predict the leakage rates at the first time step and second time step (0 and 0.1 day, respectively) using the seven parameters as input. Leakage rate at all subsequent time steps were predicted by using the last two leakage rates, in addition to the seven input parameters. The time step between subsequent prediction was fixed at 0.1 day.

In this report, we present the development, fidelity, and application of the dynamic ROM for well integrity that captures the complex chemical, mechanical and transport processes that are needed to assess leakage risk at CO_2 storage sites. We use root mean square error and correlation coefficients as metrics to check for the accuracy of the dynamic ROM. Cross-validation errors for the various kinds of ROMs are analysed to ensure that each ROM provides the correct trend with a zero-centred error distribution.

1. INTRODUCTION

Well integrity remains one of the biggest concerns regarding the safe operation of a geological CO_2 storage site. The reactive nature of both CO_2 and cement further complicates the evaluation of well integrity. Several experiments have shown that exposure of well cement to carbonated brine alters the chemical, mechanical, and hydraulic properties of cement. They have also shown that under certain conditions, these reactions can lead to chemical and/or mechanical sealing of fractured pathways in cement. As part of NRAP, a first of its kind modeling framework was developed that couples two-phase flow, cement-CO_2 geochemistry, and geomechanics to predict the permeability evolution of fractured pathways in damaged wells in CO_2 storage sites (Iyer, Walsh and Hao,, et al. 2018, Iyer, Walsh and Hao, et al. 2017, Walsh, Du Frane, et al. 2013, Walsh, Mason and Du Frane, et al. 2014, Walsh, Mason and Du frane, et al. 2014). The model can be used to identify conditions under which fracture pathways would self-seal due to chemical precipitation or mechanical deformation of reacted cement and stem the leakage of reservoir fluids.

One of the goals of NRAP is to develop tools that provide a system level risk assessment of a CO_2 storage site. Assessment of leakage risk is obtained by coupling reduced order models (ROMs) for the storage reservoir, the wellbore, and the overlying aquifer/atmosphere in the Integrated Assessment Model (IAM). The current ROMs for the wellbore do not capture the impact of chemical reactions between the well cement and leaking CO_2 on the leakage rates from damaged wellbores. Given that laboratory experiments have shown that these reactions can, under certain conditions, significantly reduce the leakage rates, it is desirable to include a wellbore ROM in the IAM that can capture these effects.

A data-driven ROM generates data via Monte Carlo sampling procedures and maps model inputs to outputs via polynomials (Ghanem and Spanos 1991, Marzouk, Najm and Rahn 2007, Marzouk and Xiu 2009), radial basis functions (Bliznyuk, Ruppert and Shoemaker 2012, Joseph 2012), Gaussian processes (Rasmussen 2004), neural networks (Funahashi 1989, Hornik, Stinchcombe and White 1989) etc. The main idea behind these methods is that the model outputs are expressed as a function of the model inputs with pre-defined basis functions and the coefficients of the functions are learned by solving an optimization problem that minimizes the discrepancy between ROM predictions and actual data. The sampling cost can be reduced by optimizing the sampling process via sensitivity analysis approaches such as ANOVA, and model/control reduction approaches such as proper orthogonal decomposition (POD) (Berkooz, Holmes and Lumley 1993), or principal component analysis (PCA) and non-linear dimension reduction techniques (Van Der Maaten, Postma and Van den Herik 2009).

The aforementioned model couples two-phase flow, cement-CO_2 geochemistry, and geomechanics by solving a system of partial differential equations such that the solution at any time depends on the solution at previous times. ROMs constructed using the typical methods mentioned above ignore this important notion of memory. We have constructed a dynamic ROM that generates model outputs by building a map that marches from the previous state to the current state to account for the memory of the system. This model reflects the dynamic behaviour of the coupled numerical model as it expresses the dependent variable at any time as a function of the decision variables of the past time and along with other model parameters.

2. PROBLEM SETUP

The leakage rate through damaged wells depends on several operating variables, fluid properties, and cement properties. The ROM uses the reservoir overpressure and saturation, the fracture aperture, length, and width, the normal stress acting on the fracture, and the reservoir depth as inputs. The primary outputs are CO_2 and brine leakage fluxes over 100 days. Input ranges considered in the ROM are listed in Table 1.

Table 1: Input model parameters for ROM generation

Input	Min Value	Max Value
Pressure Drop	1 MPa	15 MPa
Brine Saturation	0.1	0.98
Fracture Aperture	10 μm	2000 μm
Fracture Length	10 m	400 m
Fracture Width	5 mm	120 mm
Normal Stress	10 MPa	42 MPa
Reservoir Depth	1100 m	2500 m

The input parameters were selected because they have a significant impact on the leakage rate and can be defined with reasonable upper and lower bounds. The sensitive input parameters can be individually selected or automatically chosen via sensitivity analysis. Other variables, for example, those describing hydraulic and mechanical properties of cement, also have a significant impact on the leakage rate. However, they are not included in the inputs to the ROM because currently we do not have a reasonable estimate for the range of these variables. The values of these variables are set equal to those listed in Table 2 (Iyer, Walsh and Hao, et al. 2017).

The simulations also assumed

- a linear relative permeability model to describe the flow of mixtures of CO_2 and brine
- the reactivity of cement is unaffected by the brine saturation
- the fracture is at the cement-caprock interface
- the horizontal compressive stress is constant along the length of the fracture
- the maximum effective stress on the fracture is lower than 12 MPa (we do not have experimental data to calibrate the model for higher effective stress)
- the bottom end of the fracture is at the reservoir depth where the pressure is equal to the sum of the hydrostatic pressure
- the imposed reservoir saturation and pressure boundary conditions do not vary over the 100-day simulation

We used LPτ sampling to ensure efficient and additive sampling of the input parameter space. LPτ sampling is based on the uniformly distributed sequences in space. LPτ gives a mechanism for generating deterministic sequence of points in n-dimensional space which is uniformly distributed. Important feature of LPτ sampling is that it provides a way to add more points to the initially sampled points with the same uniformity characteristics. We applied some physical restrictions to the samples generated by LPτ sampling and ensure that the maximum effective stress on the fracture of the generated samples is lower than 12 MPa.

The ROM reported here is based on 376 simulations run for up to 100 days. Due to the dynamic nature of the ROM, it isn't clear how many samples should be generated to develop a robust ROM. Therefore, we chose LPτ sampling, a quasi-Monte Carlo method which allows for additive samples. Quasi-Monte Carlo methods use low-discrepancy sequences (also called quasi-random sequences or sub-random sequences). This is in contrast to regular Monte Carlo methods, which is based on sequences of pseudorandom numbers. Since low discrepancy sequences are not random, but deterministic, quasi-Monte Carlo method can be seen as a deterministic algorithm or derandomized algorithm. Therefore, if the training dataset has to be expanded the old subset of samples will be included in the larger subset of samples generated by quasi-Monte Carlo sampling. This ensures that the results are repeatable.

Table 2: Parameters used in the simulations

Description	Value
Viscosity	1×10^{-3} Pa s
Diffusion coefficient	2×10^{-9} m^2/s
Porosity of the unreacted cement	0.14
Porosity of the portlandite depleted layer	0.31
Porosity of the calcite layer	0.11
Porosity of the amorphous layer	0.70
Porosity of the precipitate in fracture	0.005
Tortuosity of the portlandite depleted layer	11.4
Tortuosity of the calcite layer	11.4
Tortuosity of the amorphous layer	1.00
Tortuosity of the precipitate in the fracture	100.0
Difference in calcium density across the portlandite dissolution front	5.18×10^{-3} mol/cm^3
Difference in calcium density across the calcite precipitation front	1.04×10^{-2} mol/cm^3
Difference in calcium density across the calcite dissolution front	2.36×10^{-2} mol/cm^3
Calcium density in precipitate in the fracture	1.08 g/cm^3
Hydraulic aperture stiffness for the unreacted cement	60 MPa
Hydraulic aperture stiffness for the portlandite depleted layer	60 MPa
Hydraulic aperture stiffness for the amorphous layer	60 MPa
Inelastic hydraulic aperture stiffness for the portlandite depleted layer	18 MPa
Inelastic hydraulic aperture stiffness for the amorphous layer	13.2 MPa
Effective yield of the portlandite depleted layer	3 MPa
Effective yield of the amorphous layer	2 MPa

The high-fidelity model, on which this ROM is being developed, solves several partial differential equations that describe the spatial and temporal evolution of pressure, velocity, concentrations, and extent of reaction. The solution at any time depends on the solution at previous times. The equations below show a simple case to illustrate this concept. Here $\bar{s}$ denotes the current state of the system and p denotes the input parameters

$$\frac{\partial \bar{s}}{\partial t} = f(p, \bar{s}(t,p)) \qquad 1.$$

$$\bar{s}(t,p) \approx \bar{s}(t-\Delta t, p) + F(\bar{s}(t-\Delta t, p), p) \qquad 2.$$

Typical ways of developing a ROM do not include this notion of memory. However, we found that the performance of the ROM improves significantly by allowing the leakage rate at any given time to depend not only on the input parameters listed in Table 1, but also on the predicted leakage rate in the previous time step. The accuracy of the ROM improved further when leakage rate from two previous time steps were included. Moreover, the dynamic behaviour of the coupled numerical model is reflected by the fact that a dependent variable (a modelled output) at any time is traceable not only to the presently applied decision variables but also to those applied in the past.

2.1 REDUCED ORDER MODEL COMPONENTS

The ROM developed here includes two static ROMs that account for the dependence of the leakage rate at the initial two time steps on the input parameters and a dynamic ROM that uses the leakage rates for the previous two-time steps and the seven input parameters to predict the leakage rate for the next time step (Equation 2). Combination of static and dynamic ROM yield more robust estimates of wellbore leakage rate over time.

For the preparation of the synthetic data used for both static and dynamic ROM generation we used LLNL in-house software: Problem-Solving environment for Uncertainty Analysis and Design Exploration (PSUADE). PSUADE is useful for providing a standard way for ROM model selection, verification and validation, sample generation, sample number specification, and visualization. It provides a convenient way to modify the sampling method, the range of parameters, and the number samples.

```
<Parameter Temperature = "333.15"
           Width="FractureWidth.Tmplt"
           Length="FractureLength.Tmplt"
           Depth="10.0"
           pressureOutlet="PressureOutlet.Tmplt MPa"
           pressureInlet="PressureInlet.Tmplt MPa"
           initialPressure="$:pressureOutlet"
           aperture="FractureAperture.Tmplt um"
           initialSaturation="BrineSaturation.Tmplt"
           saturationInlet="BrineSaturation.Tmplt"
```

Figure 1: Input template needed for sample generation using PSUADE.

The static ROMs are 4^{th} order polynomials mapping the flow rates at time t = 0 and t = 0.1 day to the input variables. The dynamic ROMs take 4^{th} order polynomial static ROM results as inputs for the initial conditions then taking 3^{rd} order polynomial for dynamic integration. Both static ROMs and dynamic ROM are written in Python so that these ROMs can be seamlessly integrated into IAM. The inputs and the output leakage rate over time were post-processed using a logarithmic transformation followed by normalization to ensure a sensitive response, because the output leakage rate spans several orders of magnitude (see section 3.1 for details on the logarithmic transformation and normalization). For the static ROMs, as shown in Fig .1, we give an example of input template for sample generation and model selection using PSUADE for validation purpose.

In this work, we only consider leakage flux rate as a scalar output. The input parameters are time invariant in our test case. In general, the methods used in this report can be readily extended to time-varying input parameters and is being pursued as ongoing work. The LPτ sampling can be specified as LPTAU sampling method for quasi-Monte Carlo sampling. The driver *psuadeInterface.py* file, as shown in Fig. 2, is an interface Python code linking PSUADE and numerical model. This file is used to generate the input files needed for running the coupled chemical, mechanical, transport model for cement fractures to calculate the leakage flux rate.

```
PSUADE
INPUT
  dimension = 7
  variable 1 PressureDrop      =   1    15.0
  variable 2 BrineSaturation   =   0.1  0.98
  variable 3 FractureAperture  =   10   2000
  variable 4 FractureLength    =   10   400
  variable 5 FractureWidth     =   5    120
  variable 6 NormalStress      =   10   42
  variable 7 ReservoirDepth    =   1100 2500
END
OUTPUT
  dimension = 1
  variable 1 LeakageFlux
END
METHOD
  sampling = LPTAU
  num_samples = 100000
#  num_replications = 1
END
APPLICATION
  driver = psuadeInterface.py
  max_parallel_jobs = 1
END
ANALYSIS
#  analyzer method = Moment
END
END
```

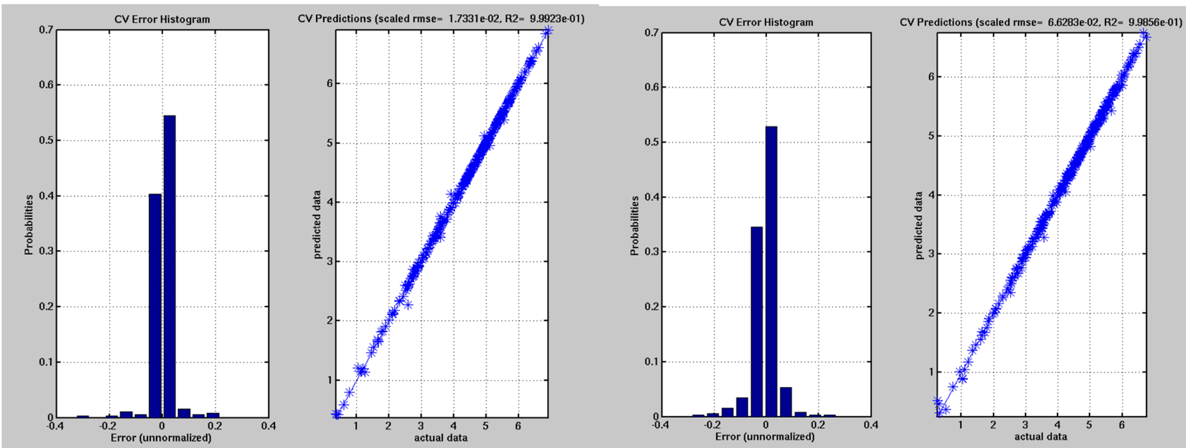
Figure 2: The PSUADE interface file to generate the input files needed for running the coupled chemical, mechanical, transport model for cement fractures to calculate the leakage flux rate.

Specially, the *psuadeInterface.py* file is a Python script that will be used to link the 1000 effective samples that was generated by PSUADE to the input XML files required for running the coupled numerical model *GEOS*.

2.2 REDUCED ORDER MODEL PERFORMANCE

In this section, we will discuss the fidelity of the two static ROMs that are polynomial ROMs of order 4 for the prediction of leakage rate at first two-time steps; and the dynamic ROM for the prediction of leakage rate up to 100 days.

Figure 3 shows that the two static ROMs perform very well on the training dataset with about 99 % of the samples having less that 1 % error. For the generation these static ROMS, as shown in Fig. 3, we have applied logarithmic transformation on 7 parameters and leakage rate output.



(1) 0 days (a) error distribution (b) q-q plot (2) 0 days (a) error distribution (b) q-q plot

Figure 3: The performance of the static ROMs to predict leakage rate at (1) 0 days and (2) 0.1 days. Panel (a) plots the error distribution while panel (b) compares the ROM predictions with the simulated leakage rate

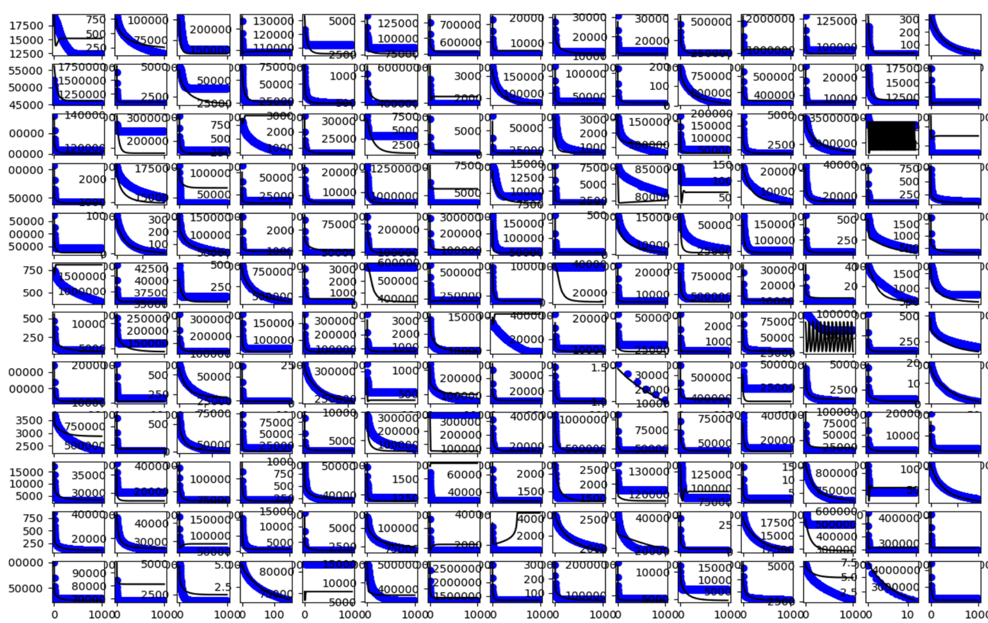


Figure 4: Dynamic ROM prediction of leakage rates till 100 days for 180 test samples.

As shown in Fig. 4, we are able to use polynomial-based dynamic ROM of degree 3 to predict the leakage rate by evolving the rate from the previous two rates. The initial flow rates are provided by the two static ROMs. In this test case, the flow rates are scaled back to the actual scale from the normalized scale.

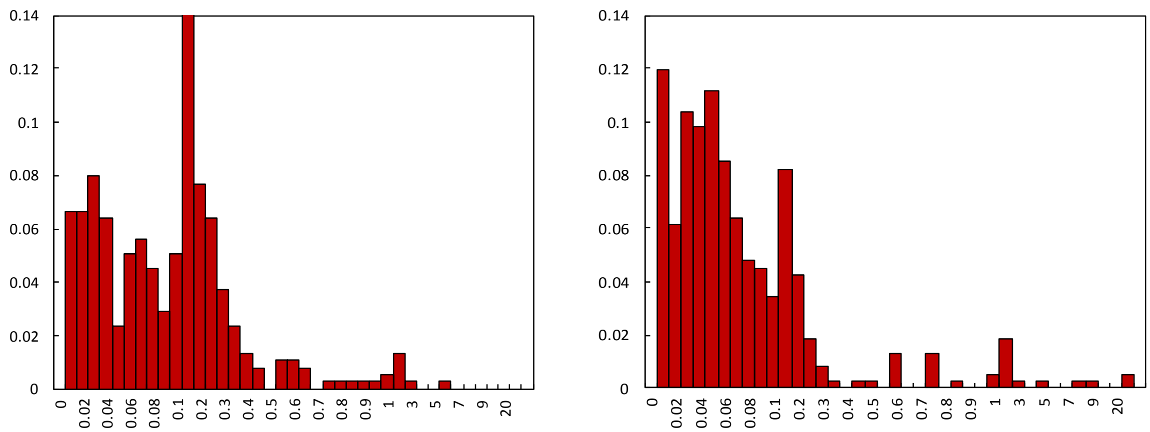


Figure 5: Relative root mean square error distribution using 3rd order (a) and 4th order (b) polynomial in the dynamic ROM

We used the relative root mean square error to evaluate the ROM's performance against the training data set. Figure 5 shows that:

- For 89% of the training examples, the ROM using 3rd order polynomial in the dynamic ROM has less than 30% prediction error
- For 2% of the training examples, the ROM using 3rd order polynomial in the dynamic ROM has prediction error greater than 100%
- For 92% of the training examples, the ROM using 4th order polynomial in the dynamic ROM has less than 30% prediction error
- For 3% of the training examples, the ROM using 4th order polynomial in the dynamic ROM has prediction error greater than 100%

It should be noted that dynamic ROMs in terms of polynomial format with higher polynomial degree generally have less approximation errors, but it may incur more overfitting errors.

3. POSTPROCESSING FOR ROMS

3.1 LOGARITHMIC TRANSFORMATION

Mass flow rate, $\dot{m}$, of a single-phase fluid flowing through a smooth fracture is given by:

$$\dot{m} = \frac{\rho b^3 w \Delta P}{12 \mu L} \Rightarrow \ln(\dot{m}) = \ln(\rho) + 3\ln(b) + \ln(w) + \ln(\Delta P) - \ln(12) - \ln(\mu) - \ln(L) \qquad 3.$$

where ρ is the density, b is the aperture, w is the width, ΔP is the pressure drop, μ is the viscosity, and L is the length of the fracture. Even though the physics included in the high-fidelity model used here includes two-phase flow, time-varying aperture, and pressure dependent density and viscosity, the equation above shows that it is advantageous to work in the logarithmic space to reduce the non-linearity of the ROM. This is illustrated in Figure 6a and Figure 6b which show the performance of a multivariate adaptive regression splines (MARS)-based ROM generated to predict the leakage rate at the end of 10 days.

In Figure 6a, the MARS-based ROM was generated using the realizations of input parameters and the corresponding leakage rate data as is. This resulted in large ROM prediction errors including unrealistic negative values for leakage rate. This means that the relationship between leakage flux data and input parameters is not well approximated by the MARS generated ROM. In other words, there exists a non-linear relationship between raw leakage flux data and the raw samples of input parameters. In fact, the initial flow rate can be represented by an analytical functional format that contains a logarithmic relationship described as the following set of expressions. In Figure 6 Figure 6b the MARS-based ROM was created after logarithmically transforming the inputs and outputs. The ROM also uses time as an input and predicts the leakage till 10 days. The ROM performs significantly better than the ROM in Figure 6a but still has large prediction errors. The initial flow rate can be represented by the following expressions. It can be shown that the initial flow rate is somewhat linear if logarithmic transformation is taken on both sides.

$$Q_{brine} = \frac{1024.987452}{0.0005154} \times \frac{Aperture^3 \times PressureDrop \times Width \times \left(\frac{BrineSaturation - 0.01}{0.99 - 0.01}\right)}{12 \times FractureLength}$$

CO_2:

$$p = \frac{(ReservoirDepth - FractureLength) \times 1023 \times 9.81}{40000000}$$

$$\rho_{CO_2} = -1071.3 + 8727p - 15794p^2 + 13091p^3 - 4061.7 \times p^4$$

$$\beta = \ln\left(\frac{60 + 273.15}{251.196}\right)$$

$$v_{limit} = 1.00697\sqrt{60 + 273.15} \times e^{-(0.235156+\beta(-0.491266+\beta(0.05211155+\beta(0.05347906-0.01537102\beta))))}$$

$$v_{excess} = \rho_{CO_2}\left(0.004071119 + \rho_{CO_2}\left(0.00007198037 + \rho_{CO_2}^4\left(\frac{2.411697 \times 10^{-17}}{\left(\frac{60 + 273.15}{251.196}\right)^3} + \rho_{CO_2}^2\left(2.971072 \times 10^{-23} - \frac{1.627888 \times 10^{-23}}{\left(\frac{60 + 273.15}{251.196}\right)}\right)\right)\right)\right)$$

$$\mu_{CO_2} = 0.000001(v_{limit} + v_{excess})$$

$$Q_{CO_2} = \frac{\rho_{CO_2} \times Aperture^3 \left(PressureDrop + (1023 - \rho_{CO_2}) \times 9.81 \times FractureLength\right) \times Width \times \left(\frac{1 - BrineSaturation - 0.01}{0.99 - 0.01}\right)}{12\mu_{CO_2} \times FractureLength} \quad 4.$$

3.2 INCOPORATING TIME DIMENSION FOR ROM PREDICTION

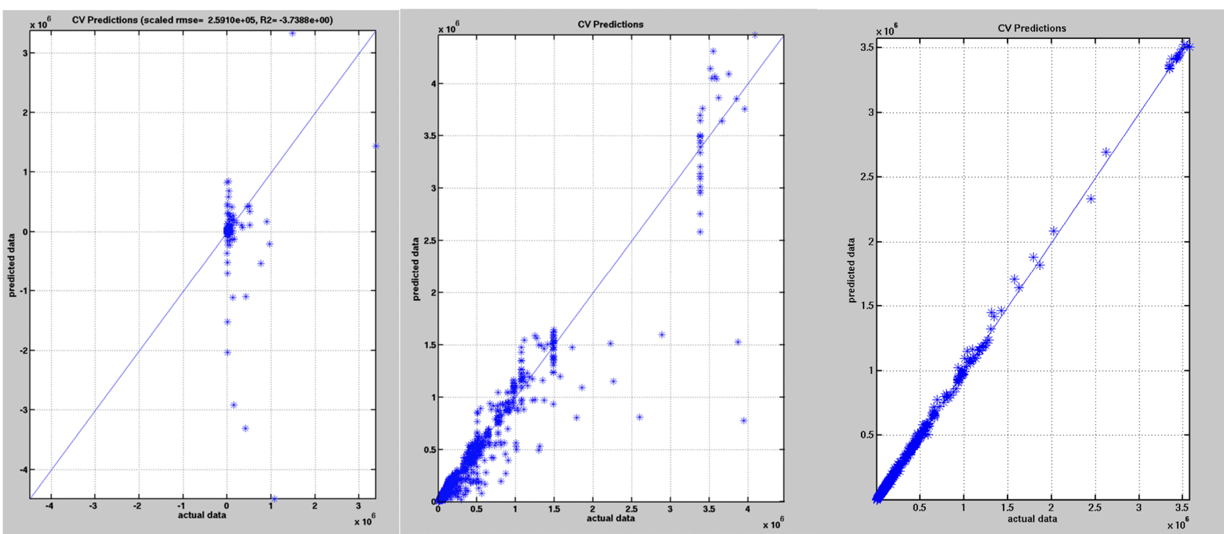


Figure 6: Performance of different MARS-based ROMs. (a) ROM developed using the raw input parameters and output leakage rate (b) ROM developed using logarithmic transformation and using time as the 8th input variable. (c) ROM developed using logarithmic transformation and using the previous leakage rate as the 8th variable.

Even though we apply the logarithmic transformation of input parameters and leakage rate data for the generation of MARS, there is still a large prediction error Therefore, even though logarithmic postprocessing is required for data postprocessing for ROM generation, there are still some other issues that need to be resolved to improve the accuracy of the resulting ROM for the given time window. One of the reasons responsible for inadequate ROM accuracy is due to the following.

The decision variables do change with time in reality. A dynamic phenomenon possesses a 'memory' in which the effect of past applied decision variables is stored. While system parameters can be assumed to be uncertain and time-independent, decision variables are often time dependent. In the next test case, we apply MARs -based ROM using 2000 samples (i.e., 100 samples per time step multiplied by 20-time steps) generated by quasi Monte Carlo sampling. The only thing we change in this experiment is that, instead of using time dimension as the 8th input variable, we use the previous leakage flux rate as the 8th variable. We still apply logarithmic transformation on both input parameters and leakage flux rate to pre-process the input data. We then apply exponential transformation to post-process the output data for fair comparison of ROM predicted leakage rate to actual leakage rate. As shown in Fig. 6c it is found that the ROM performs much better compared to Figure 6b.

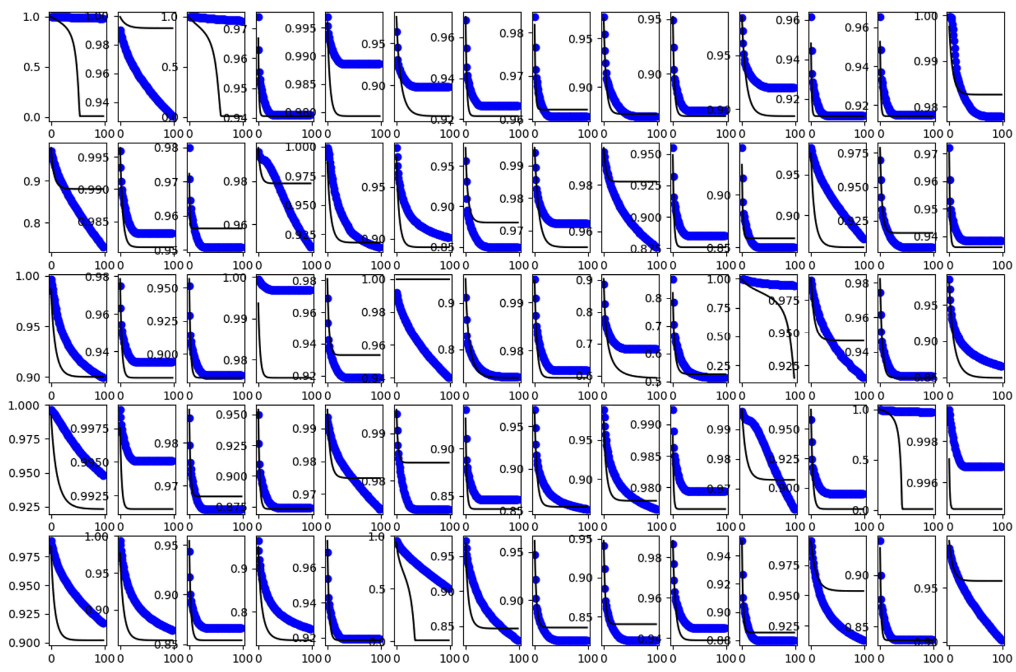


Figure 7: Dynamic ROM prediction to d 10 using polynomial of degree 3 by 376 samples evolved using previous 1 leakage flux where flow rate at d 0 is scaled to 1.0 (dynamic process is instable)

3.3 STABILITY AND NORMALIZATION

In this section, we will discuss the stability issues of the prediction of leakage rate based on previous leakage rate. We will also discuss how to scale the input parameters so that they are on the same scale.

We note that due to the dynamic nature of the ROM the prediction errors accumulate with time. The errors will be accumulated when the predicted current leakage rate is used for the prediction for the next leakage rate. Moreover, since we are using explicit ROM prediction, as shown in Fig. 7, the evolution of the dynamic ROM may not approximate the actual data well after day 10. To fix this problem, we can either develop an implicit ROM evolution method or include more leakage rates from previous time steps to predict the current leakage rate. In this work, we use leakage rates from two previous time steps with the results integrated to day 100 shown in Fig. 4.

We first take logarithmic transformation on both input parameters and leakage rate. Then, as for the dataset of each logarithmically transformed input parameter, we compute the range and normalize each of them based on each lower bound and higher bound so that lower bound for the scaled new input is 0 and higher bound is 1. As far as the logarithmically transformed leakage rate at each time step, we divide it by the logarithmically transformed initial leakage rate so that the scaled new logarithmically transformed initial leakage rate becomes 1 while all the logarithmically transformed leakage rates at subsequent time steps is less than 1. To sum up, we now have normalized logarithmically transformed input parameters and logarithmically transformed leakage rate varying between 0 and 1.

4. APPLICABILITY AND LIMITATIONS OF THE RESULTS

In this section, we will discuss the applicability and limitations of the results.

4.1 THE INTEGRATION OF THE DEVELOPED ROM TO IAM

We will incorporate the developed ROM in the integrated assessment model (IAM) and/or the OpenIAM. For the pressure change, IAM typically runs with one-year time steps in phase I IAM, while the time step becomes variable in the phase II IAM. So, over a year time step the pressure is generally considered constant then is adjusted for the new value. For the Kimberlina case, taking one-year time steps, we see the differential pressure increase by as much as 6.66 MPa in a single time step, a maximum differential pressure increases of 8.3 MPa over 20 years, and the pressure slowly declining to initial conditions over about 200 years. These numbers are very close to the injection point, farther away we would still see several MPa increase in pressure during injection. The time scales for reaction and change in permeability in the high-fidelity model is significantly shorter than 1 year. Thus, for the purpose of integrating with the IAM, we will internally run the ROM with smaller time steps and provide the IAM with the output every 0.25 year. In the current study, the parameters are assumed to be time independent. In general, however, they can be time varying. It will be more natural to use dynamic ROM to account for the time-varying parameters in our future study. The reason is because one doesn't need to parameterize the time-varying parameters as required by the static ROM, if implemented. Instead, the time-varying information of the parameters will be learned automatically during the marching process.

4.2 THE EXTENSION TO LONGER TIME WINDOW

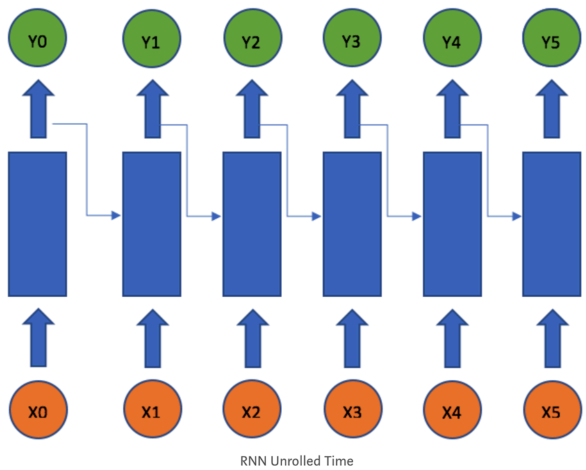
A total of 376 training examples were used to generate this ROM. For 320 cases the permeability of the leakage pathway changes and reaches its final value within 100 days. Thus, under our assumption of constant pressure and saturation driving force, the leakage rate through these fractures won't change even if the simulation is run longer than 100 days. For the remaining 56 cases, the coupled model will be run for a longer time window to quantify the long-term evolution of leakage rate. This new data will be generated, and the ROM will be expanded for cases in which a longer time window is necessary to predict the long-term leakage.

4.3 THE JUSTIFICATION TO LARGER SAMPLES FOR CROSS VALIDATION

We have implemented cross validation for static ROM by randomly choosing the training samples and validation samples. We are working on expanding the number of samples and use relative root mean square error to implement k-fold cross validation of various input parameters for dynamic ROM. We will use sklearn module from Python for k-fold cross validation. K-Folds cross-validator provides train/test indices to split data in train/test sets. We will split dataset into k consecutive folds (without shuffling by default). Each fold will then be used once as a validation set while the k - 1 remaining folds form the training set.

5. CONCLUSIONS AND FUTURE WORK

In this work, we have developed a polynomial based ROM for the prediction of leakage flux rate from the user-provided input parameters. The evolution from the previous states to the current state is predicted by a polynomial based dynamic ROM. The initial conditions required by the dynamic ROM are calculated from a polynomial based static ROM from the user-provided input parameters. We have demonstrated that 89% of the results predicted by the dynamic ROM of polynomial degree 3 have less than 30% prediction errors by using relative root mean square error as the metric.



$$Y_t = \tanh(wY_{t-1} + u\,x_t)$$

Figure 8: Undergoing work: recurrent neural network

We observed that the dynamic ROM via MARS regression provided better accuracy compared to dynamic ROM here using polynomial. We will consider dynamic ROM via MARs in the future. Moreover, since the high-fidelity model is a non-linear coupling between different physics, our future research involves replacing MARS with recurrent neural networks (RNN) as shown in Fig. 8. The RNN (Hecht-Nielsen 1992, Faucett 1994) model has been found to be useful in applications that make use of sequential data such as natural language processing (NLP). Unlike traditional neural network, RNN treats the current state as dependent on the previous states. Figure 8 shows a typical RNN where the current state Y_t is expressed as a non-linear mapping of the previous states. The RNN are *recurrent* since they use the same network to predict the future states as a function of the previous states.

This work was prepared by LLNL under contract DE-AC52-07NA27344. The work was funded by NRAP, Office of Fossil Energy, and U.S. DOE.

6. BIBLIOGRAPHY

Berkooz, Gal, Philip Holmes, and John L Lumley. 1993. "The proper orthogonal decomposition in the analysis of turbulent flows." *Annual review of fluid mechanics* 539--575.

Bliznyuk, Nikolay, David Ruppert, and Christine A Shoemaker. 2012. "Local derivative-free approximation of computationally expensive posterior densities." *Journal of Computational and Graphical Statistics* 476--495.

Faucett, L. 1994. "Fundamentals of neural networks." *Architecture, Algorithms.*

Funahashi, Ken-Ichi. 1989. "On the approximate realization of continuous mappings by neural networks." *Neural networks* (Elsevier) 183--192.

Ghanem, Roger G, and Pol D Spanos. 1991. *Stochastic Finite Element Method: Response Statistics.* Springer.

Hecht-Nielsen, Robert. 1992. "Theory of the backpropagation neural network." *Neural networks for perception* 65--93.

Hornik, Kurt, Maxwell Stinchcombe, and Halbert White. 1989. "Multilayer feedforward networks are universal approximators." *Neural networks* 359--366.

Iyer, Jaisree, Stuart D C Walsh, Hao, Yue, and Susan A Carroll. 2018. "Assessment of two-phase flow on the chemical alteration and sealing of leakage pathways in cemented wellbores." *International Jounral of Greenhouse Gas Control* 69: 72-80.

Iyer, Jaisree, Stuart D C Walsh, Yue Hao, and Susan A Carroll. 2017. "Incorporating reaction-rate dependence in reaction-front models of wellbore-cement/carbonated-brine systems." *International Journal of Greenhouse Gas Control* 160-171.

Joseph, V Roshan. 2012. "Bayesian computation using Design of experiments-based Interpolation technique." *Technometrics* 209--225.

Marzouk, Youssef M, Habib N Najm, and Larry A Rahn. 2007. "Stochastic spectral methods for efficient Bayesian solution of inverse problems." *Journal of Computational Physics* (Journal of Computational Physics) 560--586.

Marzouk, Youssef, and Dongbin Xiu. 2009. "A stochastic collocation approach to Bayesian inference in inverse problems."

Rasmussen, Carl Edward. 2004. *Gaussian processes in machine learning.* Springer.

Van Der Maaten, Laurens, Eric Postma, and Jaap Van den Herik. 2009. "Dimensionality reduction: a comparative." *J Mach Learn Res* 66--71.

Walsh, Stuart D C, Harris E Mason, Wyatt L Du frane, and Susan A carroll. 2014. "Experimental calibration of a numerical model describing the alteration of cement/caprock interfaces by carbonated brine." *International Journal of Greenhouse Gas Control* 22: 176-188.

Walsh, Stuart D C, Harris E Mason, Wyatt L Du Frane, and Susan A Carroll. 2014. "Mechanical and hydraulic coupling in cement–caprock interfaces exposed to carbonated brine." *International Journal of Greenhouse Gas Control* 109–120.

Walsh, Stuart D C, Wyatt l Du Frane, Harris E Mason, and Susan A Carroll. 2013. "Permeability of wellbore-cement fractures following degradation by carbonated brine." *Rock Mechanics and Rock Engineering* 46: 455-464.