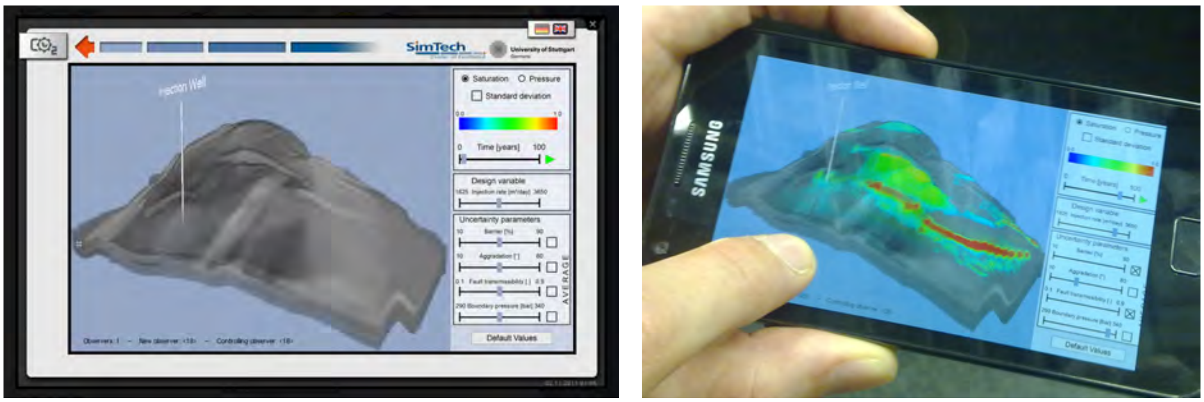
Employing Model Reduction for Uncertainty Visualization in the Context of CO2 Storage Simulation

Marcel Hlawatsch, Sergey Oladyshkin, and Daniel Weiskopf, *Member, IEEE Computer Society*



(a) (b)

Fig. 1. The developed software allows the user to interactively change uncertain parameters of the simulation. The resulting CO_2 concentration in the storage site is visualized in combination with the geological context of the storage site. The tool supports remote access. It is possible to interact with it via (a) web browser or (b) mobile device, which makes it easy to provide the public with access to the simulation. This may help communicate risks of the simulated technology.

Abstract—This work deals with the visualization of data resulting from a simulation of underground CO_2 storage. Several parameters of the simulation are unknown. A model reduction approach called polynomial chaos expansion was used in the simulation phase. The resulting polynomial data allows one to change the input parameters inside a certain range without the need of rerunning the simulation. Based on this, a visualization tool was developed that allows the user to interactively change these parameters. In this way, the parameter space can be explored and different scenarios can be played through. The tool can be accessed via web browser or mobile devices. Since underground CO_2 storage is a controversial technique, such a visualization tool can help communicate potential risks to the public. Furthermore, polynomial chaos expansion has high potential to trigger the development of future uncertainty visualization techniques.

Index Terms—Uncertainty visualization, model reduction, polynomial chaos expansion, computational steering

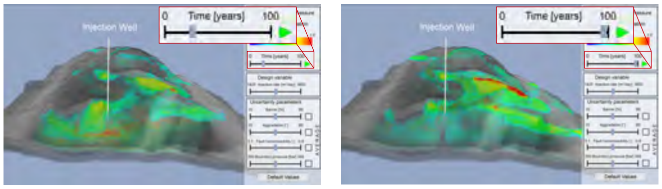
1 Introduction

Underground CO_2 storage is a controversial technology that may have a global impact in the future but may also exhibit certain risks. Respective experiments are expensive and difficult to conduct. Therefore, simulations of respective scenarios are crucial in the development of this technology. However, modeling the storage sites is difficult because they are typically not well explored and only global parameters are roughly known from measurements at potential storage sites. Therefore, the simulation model exhibits uncertain input parameters. Furthermore, the success of the technology also depends on the proper communication of potential risks to the public. We have developed a visualization tool that allows the interactive exploration of the parameter space of the simulation. By using a model reduction approach, a repeated execution of the simulation on parameter changes is avoided. Our tool therefore provides direct visual feedback to the user input. Basically, our method can be seen as a computational steering approach [6]. In a computational steering scenario, parameters of the simulation can be changed interactively during the runtime of the simulation, e.g., to guide the simulation into a certain direction. This is often realized by a strong coupling of simulation and visualization, with the latter providing the interface for changing the simulation parameters. An overview and discussion of computational steering can be found, e.g., in the work by Wright et al. [10]. As one example, Waser et al. [7] demonstrate the steering of a simulation for a flooding city. They later extended their approach for a semi-automatic exploration of the parameter space [8]. Ament et al. [1] integrate real-time FTLE computation in their steering environment. The exploration of parameter spaces is supported by the uncertainty-aware approach of Berger et al. [3]. However, all these approaches do not rely on model reduction and require a repeated execution of the simulation.

- *Marcel Hlawatsch and Daniel Weiskopf are with the Visualization Research Center, University of Stuttgart (VISUS). E-mail: {marcel.hlawatsch|daniel.weiskopf}@visus.uni-stuttgart.de.*
- *Sergey Oladyshkin is with the Department of Hydromechanics and Modelling of Hydrosystems, University of Stuttgart. E-mail: Sergey.Oladyshkin@iws.uni-stuttgart.de.*

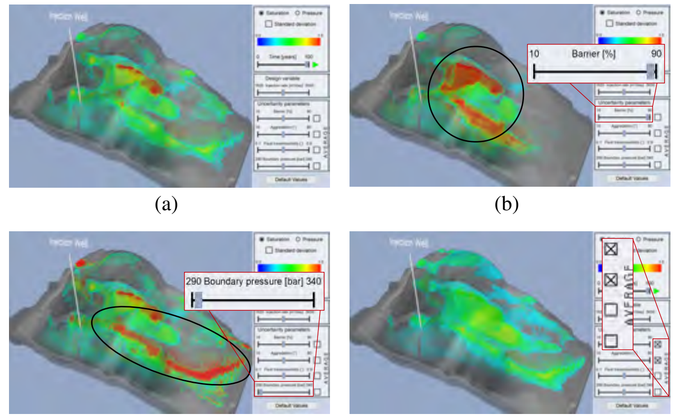
2 Model Reduction

Directly applying Monte Carlo methods to handle uncertain input parameters for a simulation model tends to be computationally intensive because the simulation model must be evaluated many times with different parameter sets. In most cases, the simulation runtime is too long for interactive applications. The computational effort can be reduced by approximating the model dependence on the input parameters. In our case, polynomial chaos expansion (PCE) was used to construct



(a) (b)

Fig. 2. Visualization of different time steps of the simulation result. Time steps can be manually selected with a slider (close-up) or automatically displayed as an animation sequence. CO_2 saturation is shown.



(a) (b) (c) (d)

Fig. 3. Exploring the parameter space. (a) The CO_2 saturation at the end of the time range with default parameters. Parameters of the simulation like (b) the percentage of barriers or (c) the boundary pressure can be interactively changed and their influence observed, e.g., areas with increased CO_2 saturation (marked black). (d) It is also possible to average parameters instead of setting them to specific values.

the response surface of the simulation model. While the original PCE approach requires a Gaussian distribution of input parameters [9], a generalization of PCE called arbitrary polynomial chaos (aPC [4, 5]) works also with other probability distribution shapes.

According to Ashraf et al. [2], aPC approximates the model response Γ for the input parameters Θ with polynomial basis functions:

$$\Gamma \approx \sum_{i=1}^{n_c} c_i \Pi_i(\Theta),$$

with n_c being the number of expansion terms and c_i being the expansion coefficients. Π_i are the polynomials for the n input parameters $\Theta = [\theta_1, ..., \theta_n]$. To obtain the coefficients c_i, a non-intrusive technique was used in our application. In this case, the simulation is treated as a black-box, i.e., modifications of the simulation code are not required.

The dataset generated with aPC contains the respective coefficients c_i for all points in the domain. To obtain the simulation result Γ for a specific point of the domain, the polynomials are evaluated with the current input parameter set and the coefficients for this point.

More details of the theoretical background of aPC and our simulation scenario can be found in the paper by Ashraf et al. [2].

3 Implementation and Results

The implementation uses CUDA to evaluate the field of polynomials in parallel on the GPU and compute the respective scalar field. To display the resulting scalar field, a standard GPU-based volume ray casting approach is used. To provide the geological context, the resulting CO_2 saturation is combined with data describing rock porosity during visualization. The overall frame rate including the evaluation of the polynomials and rendering the data was typically around 40 fps for the view shown in Figure 1 with a viewport of 818×466 (measured on a Intel Core 2 Quad Q6600 with 2.40 GHz clock rate, 4 GB RAM, and nVidia GeForce GTX 560 GPU). Therefore, the user gets direct visual feedback when changing parameters like the simulation time (Figure 2) or properties of the storage site (Figure 3).

One goal of the implementation was to make it publicly available as a demonstrator for the related research project. Therefore, the possibility was integrated to transfer the generated images via remote framebuffer protocol (RFB), which also transfers the user interaction. In this way, the visualization tool is accessible with web browsers and mobile devices (Figure 1). This is a server-side rendering approach, i.e., the client only displays the images generated on the server.

4 Conclusion

Although the visualization tool presented here is rather simple from the view of visualization research, it is already an improvement compared to the tools our collaboration partners use. They typically generate their visualizations with MATLAB; mainly plots and colored representation of the data grid. Furthermore, direct interaction is not provided. We had good experiences with our visualization tools at open house events at our university. Such tools could be used to communicate risks to the public, e.g., people could try out if there are scenarios in which their neighborhood may be affected by the stored CO_2.

Our approach provides so far only a direct visualization of the result for a specific parameter set. The uncertainty must be explored in a manual way by changing parameters. However, we think that PCE approaches are an interesting basis for novel uncertainty visualization approaches. They allow a fast evaluation of complex simulation models. This offers many possibilities for sensitivity analysis, computational steering, or visualization of uncertainty.

Acknowledgments

The authors thank the German Research Foundation (DFG) for financial support within the Cluster of Excellence in Simulation Technology (EXC 310) and within project A01 of SFB/Transregio 161.

References

[1] M. Ament, S. Frey, F. Sadlo, T. Ertl, and D. Weiskopf. GPU-based two-dimensional flow simulation steering using coherent structures. In *Proceedings of the Second International Conference on Parallel, Distributed, Grid and Cloud Computing for Engineering*, 2011. Paper 18.

[2] M. Ashraf, S. Oladyshkin, and W. Nowak. Geological storage of CO2: Application, feasibility and efficiency of global sensitivity analysis and risk assessment using the arbitrary polynomial chaos. *International Journal of Greenhouse Gas Control*, 19(0):704–719, 2013.

[3] W. Berger, H. Piringer, P. Filzmoser, and E. Gröller. Uncertainty-aware exploration of continuous parameter spaces using multivariate prediction. *Computer Graphics Forum*, 30(3):911–920, 2011.

[4] S. Oladyshkin, H. Class, R. Helmig, and W. Nowak. A concept for data-driven uncertainty quantification and its application to carbon dioxide storage in geological formations. *Advances in Water Resources*, 34(11):1508–1518, 2011.

[5] S. Oladyshkin and W. Nowak. Data-driven uncertainty quantification using the arbitrary polynomial chaos expansion. *Reliability Engineering & System Safety*, 106:179 – 190, 2012.

[6] R. van Liere, J. D. Mulder, and J. J. van Wijk. Computational steering. In H. M. Liddell, A. Colbrook, L. O. Hertzberger, and P. M. A. Sloot, editors, *HPCN Europe*, volume 1067 of *Lecture Notes in Computer Science*, pages 696–702. Springer, 1996.

[7] J. Waser, R. Fuchs, H. Ribičić, B. Schindler, G. Blöschl, and M. E. Gröller. World lines. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1458–1467, 2010.

[8] J. Waser, H. Ribicic, R. Fuchs, C. Hirsch, B. Schindler, G. Bloschl, and M. Gröller. Nodes on ropes: A comprehensive data and control flow for steering ensemble simulations. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1872–1881, 2011.

[9] N. Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60(4):pp. 897–936, 1938.

[10] H. Wright, R. Crompton, S. Kharche, and P. Wenisch. Steering and visualization: Enabling technologies for computational science. *Future Generation Computer Systems*, 26(3):506–513, 2010.