Parameter estimation in a reservoir engineering application

A. M. Hanea
Institute of Applied Mathematics
Delft University of Technology, The Netherlands

M. Gheorghe
iBMG / iMTA
Erasmus University Rotterdam, The Netherlands

ABSTRACT: Reservoir simulation models are used both in the development of new fields, and in developed fields where production forecasts are needed for investment decisions. When simulating a reservoir one must account for the physical and chemical processes taking place in the subsurface. Rock and fluid properties are crucial when describing the flow in porous media. In this paper the authors are concerned with estimating the permeability field of a reservoir. The problem of estimating model parameters such as permeability is often referred to as a history matching problem in reservoir engineering. Currently one of the most widely used methodologies which address the history matching problem is the Ensemble Kalman filter (EnKF) (Evensen et al. 2007, Aanonsen et al. 2009). EnKF is a Monte-Carlo implementation of the Bayesian update problem. Nevertheless, the EnKF methodology has certain limitations. For this reason a new approach based on graphical models is proposed and studied. In particular, the graphical model chosen for this purpose is a dynamic non-parametric Bayesian network (NPBN) (Hanea 2009, Gheorghe 2010). The NPBN based approach is compared with the EnKF method. A two phase, 2D flow model was implemented for a synthetic reservoir simulation exercise and the results of both methods for the history matching process of estimating the permeability field are illustrated and compared.

1 INTRODUCTION

The objective of reservoir engineering is to optimise hydrocarbon recovery. Oil and gas are generally found in sandstones or limestones. There are several stages of the oil recovery process. In a primary recovery stage, reservoir drive comes from a number of natural mechanisms. Primary oil recovery is the process of pumping out the oil that flows naturally to the bottom of the well due to gravity and the pressure of the reservoir. Primary recovery ends when the pressure becomes too low. After that, one of the most common and efficient secondary recovery processes is the injection of water into an oil well, in order to force out some of the remaining thicker crude oil. As the water is forced into the reservoir, it spreads out from the injection well and pushes some of the remaining oil towards the producing wells [1]. The properties of the rock, e.g. porosity and permeability, are therefore important for oil extraction since they influence the ability of fluids to flow through the reservoir.

In this paper the authors are concerned with estimating the permeability field of a reservoir. The problem of estimating model parameters such as permeability is often referred to as a history matching problem in reservoir engineering.

To characterise the fluid flow into the reservoir we use a two phase (oil-water), 2D flow model which can be represented as a system of coupled nonlinear partial differential equations that cannot be solved analytically. Consequently, we build a state-space model for the reservoir.

2 DYNAMIC NON - PARAMETRIC BAYESIAN NETWORKS & KALMAN FILTER METHODS – DESCRIPTION & CONNECTION

In a state-space model, an underlying (hidden) state of the world that generates observations is assumed. This hidden state is represented by a vector of variables that we cannot measure, but whose state we would like to estimate. This hidden state vector evolves in

[1] In the secondary recovery process the water can be replaced by gas. Steam, carbon dioxide, and other substances can be injected into an oil-producing unit in order to maintain reservoir pressure. This is known as tertiary recovery.

time. The goal of many applications is to infer the hidden state given the observations up to the current time.

Let X_t represent the hidden state at time t, and $y_1, .., y_t$ the observations up to time t. The goal is to compute $P(X_t|y_1, ..y_t)$, called the belief state. We can update the belief state recursively using Bayes' rule, and obtain a probability distribution over the hidden state.

A state-space model starts with a prior, $P(X_1)$, a state-transition function, $P(X_t|X_{t-1})$, and an observation function, $P(Y_t|X_t)$[2]. We assume that the model is first-order Markov, i.e., $P(X_t|X_1, .., X_{t-1}) = P(X_t|X_{t-1})$. Similarly, we can assume that the observations are conditionally independent given the model, i.e. $P(Y_t|Y_{t-1}, X_t) = P(Y_t|X_t)$. There are many ways of representing state-space models, one of the most common being the Kalman Filter (KF) model. KF assumes that X_t is a vector of continuous random variables, and that $X_1, .., X_T$ and $Y_1, .., Y_T$ are joint normally distributed. The KF model was introduced by R. E. Kalman in 1960 (Kalman 1960). The author proposes a recursive procedure for inference about X_t:

$$X_t = G_t X_{t-1} + w_t;$$

$$Y_t = F_t X_t + v_t. \quad (1)$$

The random variables w_t and v_t represent the process and measurement noise, respectively. They are assumed to be independent, white, and normally distributed. In practice, the process noise covariance and measurement noise covariance matrices can change with each time step or measurement, however here they are assumed constant. G_t is a matrix that relates the state at the previous time step to the current step; F_t in the second equation of (1) relates the state to the measurements. The KF model assumes that the system is joint normal. This means the belief state must be unimodal, which is inappropriate for many problems, especially those involving discrete variables. The KF will recursively calculate the state vector X_t along with its covariance matrix, conditioned on the available measurements up to time t, under the criterion that the estimated error covariance is minimum. Conditioning on the measurements is referred to as the assimilation step of the procedure. The KF method becomes computationally expensive for large scale systems and it is not suitable for non linear systems. There are several algorithms developed in order to overcome these limitations. An example of such algorithm is the ensemble Kalman filter (EnKF) (Evensen 1994). EnKF represents the distribution of the system state using a collection of state vectors, called an *ensemble*, and replaces the covariance matrix by the sample covariance computed from the ensemble. Advancing the probability distribution function in time is achieved by simply advancing each member of the ensemble. The main advantage of the EnKF is that it approximates the covariance matrix from a finite number of ensemble members, thus becoming suitable for large non linear problems. Nevertheless, very often the number of variables to be estimated is much larger than the number of ensemble members. There are typically millions of state variables and less than a hundred ensemble members (e.g. Li et al. (2003)). In these situations the ensemble covariance is rank deficient, hence it contains large terms for pairs of points that are spatially distant. These are called spurious correlations, and since they are not physically accurate, there are algorithms that try to correct them (e.g. Anderson (2007),Hamill et al. (2001)). Unfortunately these algorithms introduce other inconsistencies in the system. Moreover, EnKF relays on the normality assumption although it is often used in practice for nonlinear problems, where this assumption may not be satisfied.

Because of these limitations we introduce a more general model, namely a dynamic Bayesian network (Dean and Kanazawa 1989, Dean and Wellman 1991). Dynamic Bayesian networks provide a much more expressive language for representing state-space models. They can be interpreted as instances of a static Bayesian networks (BNs) (Pearl 1988) connected in discrete slices of time[3].

At this point, a brief description of static BNs is appropriate. A static BN is a directed acyclic graph (DAG) whose nodes represent univariate random variables, which can be discrete or continuous, and the arcs represent direct influences. The BN stipulates that each variable is conditionally independent of all predecessors in an (non-unique) ordering of the variables, given its direct predecessors. The direct predecessors of a node i, corresponding to variable X_i are called parents and the set of all i's parents is denoted $Pa(i)$, or $Pa(X_i)$. Since uncertainty distributions need not conform to any parametric form, algorithms for specifying, sampling and analysing them should be non-parametric. Therefore we shall use non parametric Bayesian networks (NPBNs) (Hanea 2008). NPBNs associate nodes with random variables for which no marginal distribution assumption is made, and arcs with conditional copulae (Joe 1997, Nelsen 1999). These conditional copulae, together with the one-dimensional marginal distributions and the conditional independence statements implied by the graph uniquely determine the joint distribution, and every such specification is consistent (Hanea et al. 2006). The marginal distributions can be obtained from data or experts (Cooke 1991). Even though the empirical marginal distributions are used in most cases, parametric forms can be also fitted. The (condi-

[2]We can also consider input variables U_t. Then, the conditional probabilities become $P(X_t|X_{t-1}, U_t)$ and $P(Y_t|X_t, U_t)$. In this paper U_t will not be considered.

[3]We only consider discrete-time stochastic processes.

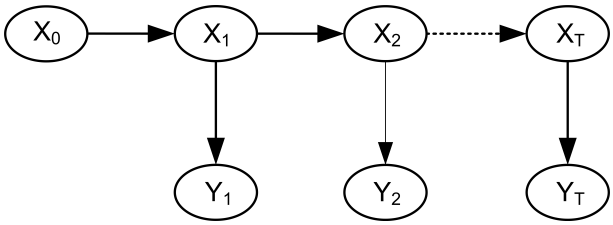

Figure 1: The KF model as a dynamic BN

tional) copulae used in this method are parametrised by (conditional) rank correlations that can be calculated from data or elicited from experts (Morales, Kurowicka, & Roelen 2007). The name NPBN is somewhat inappropriate but it is used to stress the fact that the joint distribution is specified via marginal distributions, upon which no restrictions are placed, and the dependence structure given in terms of a non parametric measure of dependence.

A dynamic NPBN is a way to extend a static NPBN to model probability distributions of collections of random variables, $Z_1, Z_2, .., Z_T$. The variables can be partitioned in $Z_t = (X_t, Y_t)$ to represent the hidden and output variables of a state-space model. A dynamic NPBN is defined to be a pair, $(B_1, B_{\rightarrow})$, where B_1 is a NPBN which defines the prior $P(Z_1)$, and $B_{\rightarrow}$ is a two-slice temporal NPBN which defines $P(Z_t|Z_{t-1})$ as follows:

$$P(Z_t|Z_{t-1}) = \prod_i P(Z_t^i|Pa(Z_t^i)), \qquad (2)$$

where Z_t^i is the i^{th} node at time t, which could be a component of X_t, or of Y_t.

The parents $Pa(Z_t^i)$ can be either in the same time slice or in the previous time slice[4].The arcs between slices are from left to right, reflecting the flow of time.

The difference between a dynamic NPBN and a KF model is that the latter requires joint normality, whereas a dynamic NPBN allows arbitrary marginal distributions. In addition, a dynamic NPBN allows for a much more general graph structure. Figure 1 presents a general KF model as a dynamic BN.

3 CASE STUDY

We construct a synthetic example by simulating a five-spot injection-production strategy. In other words, the reservoir has an injector in the middle of the field (where water is injected) and 4 producers, one in each corner (where oil is pumped out from). The *true* permeability field is randomly chosen from an ensemble of possible models (see Figure 2) and the synthetic production data is generated using this *true* model. Synthetic measurements are obtained by adding normally distributed errors to the production data.

[4]We assume the model is first-order Markov, for a fair comparison with the ENKF method.

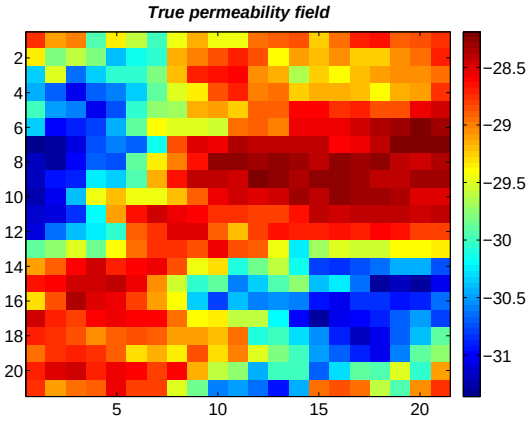

Figure 2: The *true* permeability field.

3.1 *Experimental setup*

The reservoir model considered here is a 2D square with a uniform Cartesian grid consisting of 21 grid cells in each direction. The reservoir is considered a closed space where liquid gets in and out only through the drilled wells. Therefore, the drilled wells become the reservoir's boundaries. A well model is available for the injection and extraction of fluids through the drilled wells. The flow is specified at the wells by bottom hole pressure (bhp) and fluid flow rates (q). The well model imposes that either the bottom hole pressure or the fluid flow rates must be prescribed. We consider the case where the injection well is constrained by prescribed flow rates and production wells are constrained by bottom hole pressure. The two phase flow model is combined with the well model and implemented in a simple in-house simulator.

The state vector contains pressures (p) corresponding to each grid cell. Since we want to perform a parameter estimation, the state vector is augmented with the parameter of interest, i.e. the natural logarithm of the permeability[5] (log(k)). Given the well model constraints, we measure bottom hole pressure at the injector and total flow rates at the producers. The final form of the vector Z_t is:

$$Z_t = \begin{pmatrix} \log k(t) \\ p(t) \\ bhp(t) \\ q(t) \end{pmatrix}$$

The reservoir is initialized with pressure equal to $3 \cdot 10^7[Pa]$ in every grid cell. We perform simulations for 420 days, considering measurements every 60 days.

For the NPBN based approach we build a DAG on the variables defined in the state vector, hence the DAG should contain 1328 nodes. Given the incipient stage of modelling a petroleum engineering problem with a NPBN, a simplification of the model is in order. Based on expert's opinions we decided to exclude the variables representing saturations and how to set the arc directionality amongst remaining variables. A

[5]We consider the log(k) instead of k because the values of the permeability are of order $10^{-13}[m^2]$.

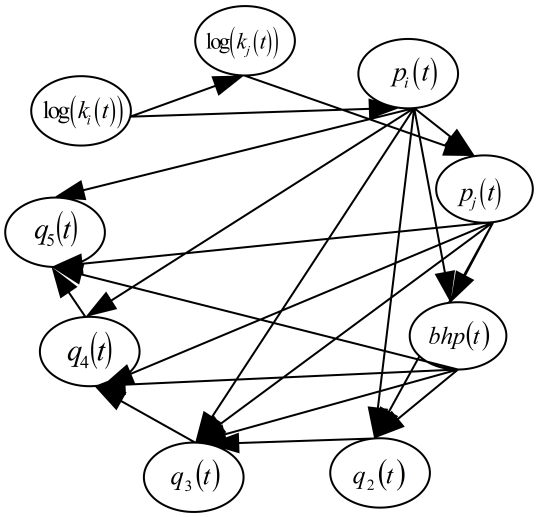

Figure 3: The DAG of the NPBN.

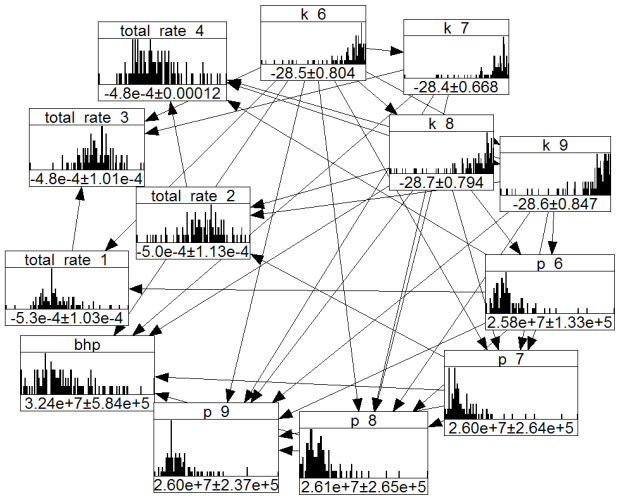

Figure 4: The learned NPBN after 60 days.

schematic representation of a potential BN is shown in Figure 3.

3.2 *Results & comparisons*

We will first estimate the permeabilities in a reduced number of grid cells. The initial goal of this study was to estimate the entire permeability field with both methods and compare results. Unfortunately, at this stage of the research, the NPBN approach cannot handle more than 500 variables, so we shall restrict our analysis to parts of the grid, rather than the entire reservoir.

We arbitrarily choose 4 different locations. We measure bottom hole pressure (*bhp*) at the injector well and the total flow rates (q) denoted now by *total_rate_i*, $i = 1, .., 4$, at each producer. Any location that is not a well has its corresponding pressure and permeability. We denote them by *p_j*, and *k_j*, $j = 6, .., 9$, respectively. Therefore, we are interested in the joint distribution of 13 variables. We run the simulator for the first 60 days, and obtain their joint distribution (in form of a data set). We can now represent the joint distribution using a static NPBN. Using a saturated NPBN[6] translates into representing all possible dependencies present in the data set, including the noisy ones. Moreover, the visual advantage of the graphical model vanishes since a saturated graph is dense and un-intuitive. Another choice is to learn a validated NPBN from data. A learning algorithm is introduced in Hanea et al. (2010). The only assumption of the algorithm is that of a joint normal copula. This means that we model the data as if it were transformed from a joint normal distribution. The marginal distributions are taken directly from data and the empirical rank correlation matrix is calculated. The algorithm assigns arcs between strongly correlated variables. Missing arcs will correspond to (conditional) independent statements. A joint distribution that approximates the distribution given by the simulator is therefore obtained. For details we refer to Hanea et al. (2010). The learning procedure involves validation. First, we validate that the joint normal copula adequately represents the multivariate data. If this is the case we then learn a model and validate that it is an adequate model of the saturated graph.

After 60 days, the normal copula assumption is validated, hence we learn the NPBN presented in Figure 4. The NPBN model is build using the software Uninet (Morales-Napoles et al. 2007). Nodes of an NPBN can be visualised as ellipses or histograms. The mean and standard deviation of each variable are shown on the graph.

The static NPBN can now be used to perform the conditionalization/assimilation step. Given the observed values of measurements at the wells, we can calculate the joint conditional distribution of the other variables[7]. It is worth noting that the observable variables are not normally distributed. Nevertheless, normally distributed noise is added when generating measurements for a fair comparison with the EnKF method. After conditioning, we stipulate the conditional distribution by sampling it. Further, we introduce the updated distribution in the simulator, and we run it for another 60 days. In this way we obtain the distribution of the variables after 120 days (with 1 assimilation step after 60 days). The new joint distribution will be modelled with another static NPBN. The 2 NPBNs connected through the simulator are basically a dynamic NPBN with changed structure and parameters over time, and with functional temporal relationships. We repeat the above steps for a period of 420 days. Every time step we validate the normal copula and the model. We thus build a dynamic NPBN for 7 discrete times.

The results of estimating the permeabilities for the chosen locations using a saturated NPBN, a learned NPBN, and the EnKF method are further presented. To measure the quality of the estimation we compare it with the *truth*. A measure of discrepancy is the root

[6] In a saturated NPBN all nodes are connected.

[7] The normal copula assumption facilitates analytical conditioning (Hanea et al. 2006).

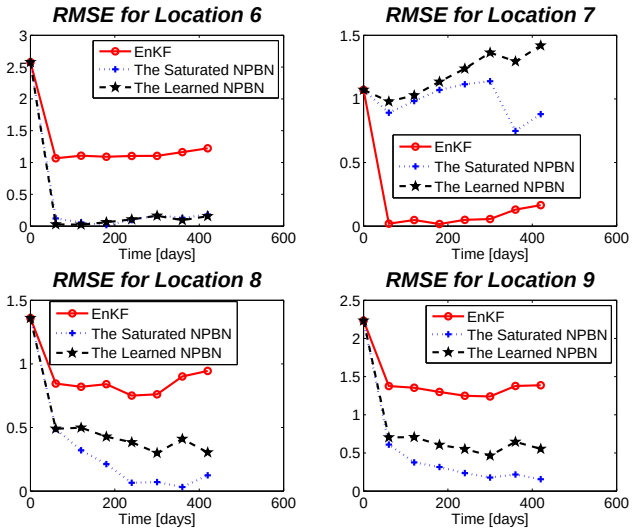

Figure 5: RMSE for each location using EnKF vs. NPBN.

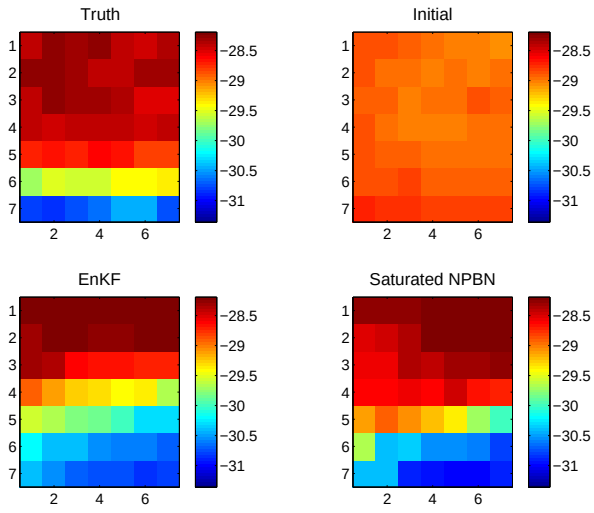

Figure 6: The true (top-left), the initial (top-right), the EnKF estimated (bottom-left), and the NPBN estimated (bottom-right) permeability field of a 7×7 grid.

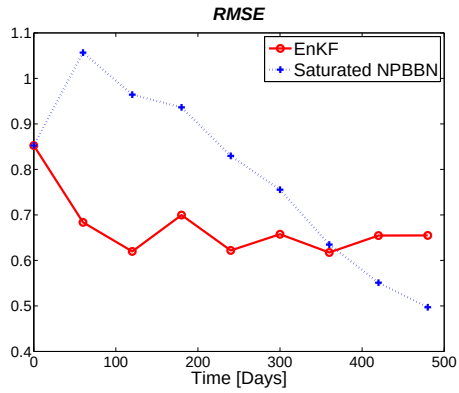

Figure 7: RMSE for the estimated permeability field of a 7×7 grid using EnKF vs. NPBN.

mean square error (RMSE), that is computed by taking the sum of the squares of the errors (difference between the predicted and actual values), computing the average and then taking the square root.

Figure 5 shows the RMSE for each location, at each time step. For locations 6, 8 and 9 both the saturated and the learned NPBN give better estimations than the EnKF method. However, EnKF performs better than the NPBN for location 7. A possible explanation is that the EnKF algorithm incorporates the information from all the grid points in the calculations, whereas the NPBN works only with the information given by the variables represented in the DAG. The saturated NPBN performs better then the learned one in most of the cases. That would suggest that the correlations present in the data are significant even if small.

Let us now consider a 7×7 and a 13×13 grid. For estimating the 49 permeabilities we use a NPBN on 103 variables. When estimating the permeabilities for the 13×13 grid we have 423 variables. The validation steps involved in the learning procedure for NPBNs become inconclusive for such a large number of variables (Gheorghe 2010). Hence, we perform experiments using only the saturated graph.

Figure 6 shows the true permeabilities, the initial ensemble, and the estimated permeabilities after 480 days using the EnKF and using the saturated NPBN. A visual comparison would suggest that the saturated NPBN gives a better estimate than the EnKF for this grid. This conclusion is supported by comparing the RMSE values as well (see Figure 7).

The behavior of the RMSE is quite different for the 2 methods. For the EnKF, the RMSE decreases at the 1^{st} time step and then has an oscillating behavior with a tendency to stabilize around the 4^{th} time step to a value of 0.7. The RMSE for the NPBN has an increase after the 1^{st} time step and afterwards is decreasing for every time step reaching a value of 0.5 by the 8^{th} time step. The RMSE should, theoretically, decrease for every time step. However, in practice this is not always the case. The NPBN method uses copulae parametrised by rank correlations. The estimation of the rank correlation matrix is very sensitive to noise, i.e, a wrong input can be amplified and result in wrong estimates. This could be a possible explanation for the RMSE behaviour after the 1^{st} time step when using the NPBN. As expected, incorporating more measurements improves the performance of the NPBN based approach. On the other hand it seems that after the 4^{th} time step the EnKF method does not benefit from more information.

In the case of a 13×13 grid, Figure 8 only indicates that the 2 methods are comparable. One could say that the two fields look equally well or even that the estimate using the saturated NPBN looks slightly better than the field estimated using EnKF. However, the RMSE from Figure 9 contradicts the visual analysis. The RMSE for the entire field is clearly smaller and more stable for the EnKF than for the saturated NPBN. The RMSE for EnKF has a considerable decrease after the 1^{st} time step and then an almost constant behavior. It stabilizes around the value of 0.5. Note that the RMSE for EnKF shows that after the 1^{st} time step the EnKF does not really assimilate any new information. It is worth stressing that the RMSE

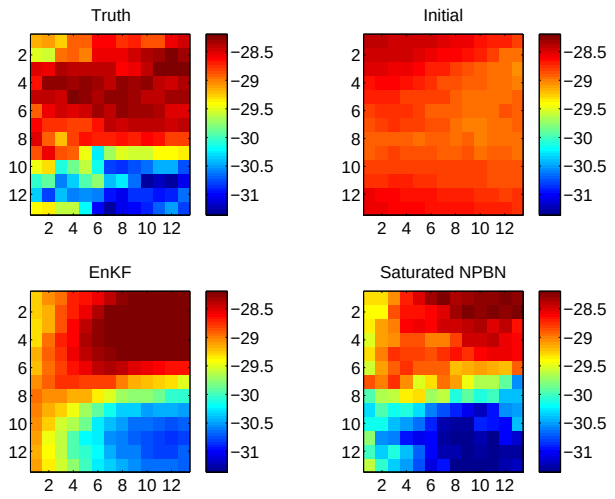

Figure 8: The true (top-left), the initial (top-right), the EnKF estimated (bottom-left), and the NPBN estimated (bottom-right) permeability field of a 13×13 grid.

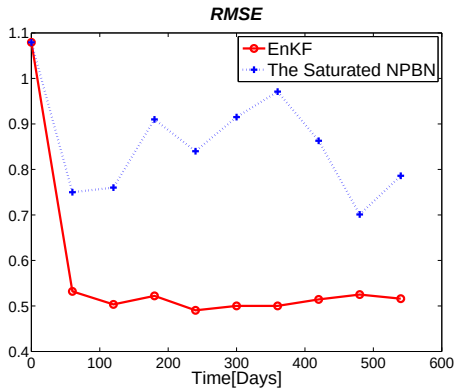

Figure 9: RMSE for the estimated permeability field of a 13×13 grid using EnKF vs. NPBN.

shows only a certain general/average behavior of an estimated filed, and it can often be misleading. In Figure 9 the simulation is performed for one extra time step in order to check if the RMSE for the NPBN continues to decrease. Unfortunately this is not the case. One possible explanation of this unstable behavior could be that estimating the joint distribution of 423 variables requires more ensemble members than used here (i.e., 900).

4 CONCLUSIONS

To our knowledge this is the first attempt to approach a history matching problem in reservoir simulation using a NPBN based approach. Comparisons have been made between the results of applying the new method and those obtained with the EnKF method. The results, just like the theory behind, indicate that the 2 methods are comparable. Unfortunately we cannot say yet that one method outperforms the other. However we would like to point out the differences between them.

The NPBN method uses parameters whose estimates are more sensitive to the number of ensemble members used. A larger ensemble is needed when working with an NPBN. However, this requirement does not interfere with the speed of the calculations.

The measurements are generated by adding normally distributed noise to the truth. Looking at the histograms of the variables we notice that neither bottom hole pressure, nor the total flow rate are normally distributed. Since the assumption of normality is used in the EnKF method, we used the same setting for the NPBN based approach. Nevertheless, the NPBN based approach affords adding errors with closer distributions to the true ones. An improvement of the estimates is then expected.

One of the most important assumptions made by the EnKF method is that the conditional joint distribution is joint normal. When the NPBN model was built for 4 locations we observed that the margins of the assumed Gaussian distribution were far from being normally distributed. Note that no validation of the assumption of joint normality is performed in the EnKF method. On the other hand, the NPBN based approach uses the assumption of the joint normal copula, and no assumption about the marginal distributions. In contrast with the EnKF, 2 validation steps are performed for the NPBN based method. However, as the number of variables in the graph increases, the validation steps become unpractical. Different, more meaningful statistical tests are being investigated at the moment of writing this paper.

We presented results of estimating the permeabilities for: 4 different, randomly chosen locations, a 7×7 grid block and a 13×13 grid. For 3 out of 4 locations, the estimates obtained with the NPBN approach were closer to the truth than those obtained with the EnKF. Better results were also obtained with the NPBN when estimating the permeabilities for a 7×7 grid. However, for a 13×13 grid, the 2 methods showed undistinguishable performance. One could argue that the RMSE for the NPBN method shows an worse estimate. It is worth mentioning that the RMSE is only an average measure of performance. The visual inspection of the estimated permeability fields can sometimes offer a better insight into the performance of the methods. In this particular case, the images of the fields show that the NPBN based approach gives similar results to those obtained with the EnKF method.

Our goal was to estimate the permeabilities for the entire field using both methods. However building a saturated NPBN for a larger grid becomes computationally infeasible. The maximum number of permeabilities that we could estimate was 169 out of 441. This constitutes a considerable limitation of the NPBN based approach. Nevertheless, interpolation methods could be employed for estimating the permeabilities in larger fields.

A definite conclusion about which approach performs better is premature. There are reasons to believe that further research is worthwhile.

REFERENCES

Aanonsen, S., G. Naedval, D. Oliver, A. Reynolds, & B. Valles (2009). The Ensemble Kalman Filter in Reservoir Engineering. *SPE Journal*.

Anderson, J. (2007). Exploring the need for localization in ensemble data assimilation using a hierarchical ensemble filter. *Physica D: Nonlinear Phenomena*, 99111.

Cooke, R. (1991). *Experts in Uncertainty : Opinion and Subjective Probability in Science*. Environmental Ethics and Science Policy Series. Oxford University Press.

Dean, T. & K. Kanazawa (1989). A model for reasoning about persistence and causation. *Artificial Intelligence 93(1-2)*, 127.

Dean, T. & M. Wellman (1991). Planning and control. *Morgan Kaufmann*.

Evensen, G. (1994). Sequential data assimilation with nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research 99*(C6), 1014310162.

Evensen, G., J. Hove, H. Meisingset, E. Reiso, K. Seim, & O. Espelid (2007). Using the EnKF for assisted history matching of a North Sea reservoir model. *SPE Reservoir Simulation Symposium Huston*(Texas), USA.

Gheorghe, M. (2010). Non parametric Bayesian belief nets versus ensemble Kalman Filter in reservoir simulation. *MSc Thesis, Delft University of Technology*.

Hamill, T., J. Whitaker, & C. Snyder (2001). Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter. *Monthly Weather Review 129*, 27762790.

Hanea, A. (2008). *Algorithms for Non-parametric Bayesian belief nets*. Ph. D. thesis, TU Delft, Delft, the Netherlands.

Hanea, A. (2009). *Tackling a Reservoir Engineering Problem with a NPBN Approach*. Lecture notes, Summer School on Data Assimilation: Section 3.

Hanea, A., D. Kurowicka, & R. Cooke (2006). Hybrid Method for Quantifying and Analyzing Bayesian Belief Nets. *Quality and Reliability Engineering International 22*(6), 613–729.

Hanea, A., D. Kurowicka, R. Cooke, & D. Ababei (2010). Mining and visualising ordinal data with non-parametric continuous BBNs. *Computational Statistics and Data Analysis 54*(3), 668–687.

Joe, H. (1997). *Multivariate Models and Dependence Concepts*. London: Chapman & Hall.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 3545.

Li, R., A. C. Reynolds, & D. Olivier (2003). History matching of three phase flow production data. *SPE Journal 8(4)*, 328–340.

Morales, O., D. Kurowicka, & A. Roelen (2007). Eliciting conditional and unconditional rank correlations from conditional probabilities. *Reliability Engineering and System Safety*. doi: 10.1016/j.ress.2007.03.020.

Morales-Napoles, O., D. Kurowicka, R. Cooke, & D. Ababei (2007). Continuous-discrete distribution free Bayesian belief nets in aviation safety with UNINET. *Technical Report TU Delft*.

Nelsen, R. (1999). *An Introduction to Copulas*. Lecture Notes in Statistics. New York: Springer - Verlag.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo: Morgan Kaufman Publishers.