

Copyright
by
Baosheng Liang
2007

**The Dissertation Committee for Baosheng Liang Certifies that this is the approved
version of the following dissertation:**

**AN ENSEMBLE KALMAN FILTER MODULE FOR AUTOMATIC
HISTORY MATCHING**

Committee:

Kamy Sepehrnoori, Supervisor

Mojdeh Delshad

Gary A. Pope

Steven L. Bryant

Thomas F. Edgar

Faruk O. Alpak

**AN ENSEMBLE KALMAN FILTER MODULE FOR AUTOMATIC
HISTORY MATCHING**

by

Baosheng Liang, B.S., M.S.

Dissertation

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

The University of Texas at Austin

December 2007

This dissertation is dedicated to my father Dunfa Liang, my mother Meilian Xie, my brothers Baowei Liang and Baoguo Liang, my sisters-in-law Liping Zhao and Xiangying Meng, my niece Fan Liang and my nephew Yu Liang, for their love.

Acknowledgements

Mark Twain said that “Good judgement comes from experience; experience comes from bad judgement.” Throughout this work, I have learned a lot and would like to take this opportunity to thank all the people who have helped and guided me. First and foremost, it is my great fortune to have Dr. Kamy Sepehrnoori as my supervising professor. Henri Poincaré said that “The scientist does not study nature because it is useful; he studies it because he delights in it, and he delights in it because it is beautiful. If nature were not beautiful, it would not be worth knowing, and if nature were not worth knowing, life would not be worth living.” I greatly appreciate that Dr. Sepehrnoori helped me select the automatic history matching topic and led me to explore the beauty of reservoir prosperities. Dr. Sepehrnoori has taught me not only specific technical knowledge, but also how to define problems and deliver solutions. Dr. Sepehrnoori’s inspiration, encouragement, patience, and humor were essential for my research and daily life at The University of Texas at Austin.

Dr. Mojdeh Delshad was always available: answering my many questions on UTCHEM questions, helping set up an application case, and editing my presentations. Her patience, encouragement and kindness built up my confidence under many challenging conditions. I would also like to thank other members of the supervising committee, Drs. Gary Pope, Steven Bryant, Thomas Edgar, and Faruk Alpak.

Special thanks to Dr. Jiang Zhang. His help on Linux and C++, comments on my research and care of my daily life contributed a lot. I am also grateful to Dr. Xian-Huan Wen, from Chevron Energy Technology Company, for his suggestions for the case studies used in this dissertation. Many thanks to my supervisor Suzanne Elliott, and friends Kefei Wang and Yujie Hu at Landmark Graphics during my summer internship in 2006.

I would like to thank Zhimin Du and Yingchuan Li, Professors at Southwest Petroleum University, for the encouragement, effort, and value they have given throughout my study in U.S.

I am truly indebted to my professors, fellow students and administrative personnel in the Department of Petroleum and Geosystems Engineering. In particular, Anne Reecer helped edit my draft and helped with paperwork for registration and travels. Tim Guinn and Roger Terzian helped me to solve Linux and Windows facility problems. Joanna Castillo helped my presentations in the annual meetings. Cheryl Kruzic was always smiling to process the registration for each semester. Mary Pettengill helped find the desired references for my research. I would also like to express my gratitude to my colleagues and friends Luan Wang, Ying He, Feng Pan, Ajay Suri, Bo Lu, Zhen Li, Zongyu Zhai, Weiping Bai, Xingru Wu, Chaoshun Hu, Jianxin Shi, Davood Ghorbani, Morteza Sayapour, Kiomars Eskandarivalvand, Yousef Ghomian, Yinghui Li, Zhaohui Lu, Yanping Sun, Reza Naimi-Tajdar, Omar Alan Vicencio, Choongyong Han, Yajun Liu, Chengwu Yuan, Guozhong Gao, Long Jin, Songcheng Li, Linlin Wang, Abdoljalil Varavei, Chun Shen, Qichong Li, Tao Xie, Liying Xu, Liying Zhang, Ping Zhao, Changli Yuan, Farhad Tahrahhom, Nariman Fathi, and Dr. Francisco Marcondes.

The financial support of the Reservoir Simulation Joint Industry Project in the Center for Petroleum and Geosystems Engineering at The University of Texas at Austin is gratefully acknowledged.

My mother, my father, my two grandmothers, my two elder brothers and sisters-in-law, my lovely niece and nephew have been unflagging in their affection, support, and encouragement, without which this work would not have been accomplished.

AN ENSEMBLE KALMAN FILTER MODULE FOR AUTOMATIC HISTORY MATCHING

Publication No. _____

Baosheng Liang, Ph.D.

The University of Texas at Austin, 2007

Supervisor: Kamy Sepehrnoori

The data assimilation process of adjusting variables in a reservoir simulation model to honor observations of field data is known as history matching and has been extensively studied for few decades. However, limited success has been achieved due to the high complexity of the problem and the large computational effort required by the practical applications. An automatic history matching module based on the ensemble Kalman filter is developed and validated in this dissertation.

The ensemble Kalman filter has three steps: initial sampling, forecasting through a reservoir simulator, and assimilation. The initial random sampling is improved by the singular value decomposition, which properly selects the ensemble members with less dependence. In this way, the same level of accuracy is achieved through a smaller ensemble size. Four different schemes for the assimilation step are investigated and direct inverse and square root approaches are recommended. A modified ensemble Kalman filter algorithm, which addresses the preference to the ensemble members through a non-equally weighting factor, is proposed. This weighted ensemble Kalman filter generates

better production matches and recovery forecasting than those from the conventional ensemble Kalman filter. The proposed method also has faster convergence at the early time period of history matching. Another variant, the singular evolutive interpolated Kalman filter, is also applied. The resampling step in this method appears to improve the filter stability and help the filter to deliver rapid convergence both in model and data domains. This method and the ensemble Kalman filter are effective for history matching and forecasting uncertainty quantification.

The independence of the ensemble members during the forecasting step allows the benefit of high-performance computing for the ensemble Kalman filter implementation during automatic history matching. Two-level computation is adopted; distributing ensemble members simultaneously while simulating each member in a parallel style. Such computation yields a significant speedup.

The developed module is integrated with reservoir simulators UTCHEM, GEM and ECLIPSE, and has been implemented in the framework Integrated Reservoir Simulation Platform (IRSP). The successful applications to two and three-dimensional cases using blackoil and compositional reservoir cases demonstrate the efficiency of the developed automatic history matching module.

Table of Contents

List of Tables	xv
List of Figures	xvi
Chapter 1: Introduction	1
1.1 Introduction to History Matching and Development Optimization	1
1.2 Research Objectives	3
1.3 Outline of Chapters	4
Chapter 2: Literature Review	6
2.1 History Matching	6
2.2 Automatic History Matching	8
2.3 Deterministic Algorithms	13
2.3.1 Adjoint Method	14
2.3.2 Other Approaches for the Calculation of Sensitivity Coefficients	15
2.3.2.1 Gauss-Newton and Modified Levenberg-Marquardt Algorithms	15
2.3.2.2 Conjugate Gradient Method	16
2.3.2.3 LBFGS Algorithm	17
2.3.3 Gradual Deformation Method	18
2.4 Stochastic Algorithms	20
2.4.1 Genetic Algorithms	20
2.4.2 Simulated Annealing	21
2.4.3 Scatter and Tabu Searches	22
2.4.4 Neighborhood Algorithm	23
2.4.5 Kalman Filter Methods	24
2.5 Summary	27
Chapter 3: Theory of the Ensemble Kalman Filter	29
3.1 Outline of the EnKF Algorithm	30
3.1.1 Measurement Perturbation	32

3.1.2 Forecasting Step.....	35
3.1.3 Assimilation Step.....	35
3.2 EnKF Assimilation Schemes	36
3.2.1 Direct Inverse Calculation	37
3.2.2 Standard EnKF Assimilation Calculation.....	37
3.2.3 Square Root Algorithm with Measurement Perturbations.....	39
3.2.4 Square Root Algorithm without Measurement Perturbations.....	42
3.3 Measurements of The EnKF Performance.....	42
3.4 Summary	45
Chapter 4: Investigation of Assimilation Schemes in the Ensemble Kalman Filter	50
4.1 Description of a Two-Dimensional Waterflooding Case.....	50
4.2 Discussion on Assimilation Schemes	53
4.3 Discussion on Assimilation Interval	57
4.4 Dependence of Measurement Errors.....	59
4.5 Summary	60
Chapter 5: Improved Initial Sampling Strategy in the Ensemble Kalman Filter ..	91
5.1 Ensemble Size.....	91
5.2 Singular Value Decomposition.....	93
5.3 Improved Initial Sampling Strategy.....	95
5.4 Case Study and Discussion	99
5.4.1 Property Adjustments of the Initial Sampling	99
5.4.2 Initial Sampling with the SVD.....	100
5.5 Summary	101
Chapter 6: Improved Assimilation Algorithm in the Ensemble Kalman Filter ..	109
6.1 Introduction.....	110
6.2 Weighted EnKF	112
6.3 First Case Study And Discussion.....	115
6.3.1 Creation of the Initial Geostatistical Model.....	116
6.3.2 Type of Measurement Data.....	119

6.3.3 Sensitivity Study of Measurement Interval.....	120
6.3.4 Comparisons between Conventional EnKF and Weighted EnKF	122
6.3.4.1 Evolution of a Matched Permeability Field.....	122
6.3.4.2 Evolution of a Matched Pressure Field.....	124
6.3.4.3 Evolution of Matched Saturation Field.....	125
6.3.4.4 Production History Matching and Recovery Forecasting.....	126
6.4 Second Case Study and Discussion	128
6.4.1 Exact or Approximated Reservoir Geologic Information.....	129
6.4.2 Case Description and Results Discussion.....	130
6.5 Summary	133
Chapter 7: A Singular Evolutive Interpolated Kalman Filter for Uncertainty Quantification	229
7.1 Introduction.....	230
7.2 Singular Evolutive Interpolated Kalman Filter.....	233
7.2.1 Initialization	236
7.2.2 Resampling	237
7.2.3 Forecasting.....	238
7.2.4 Assimilation	239
7.3 A Three-Dimensional Numerical Example	240
7.3.1 Model Description	240
7.3.2 Accuracy and Convergence of Model Domain Reconstruction	241
7.3.3 Comparison of SEIKF and EnKF Results	242
7.3.4 Quantification of Forecast Uncertainty.....	243
7.4 Summary	246
Chapter 8: Automatic History Matching with Distributed and Parallel Computing.....	259
8.1 Distributed, Parallel Computing and Portable Batch System	259
8.2 Structure of Distributed and Parallel Computing.....	262
8.3 Case Study and Discussion	263

8.3.1 Case Description	263
8.3.2 History Matching Results	264
8.3.3 High-Performance Computing	265
8.4 Summary	268
Chapter 9: Automatic History Matching Module	276
9.1 Introduction to UT_IRSP	276
9.2 Requirements of the Automatic History Matching Module	278
9.3 Structure of the Automatic History Matching Module	280
9.4 Compilation Procedure for the Automatic History Matching Module	282
9.5 Description of the Input File	284
9.6 Storage Hierarchy	288
9.7 MDM and SGSIM Input Files	289
9.7.1 MDM	290
9.7.2 SGSIM	292
9.8 Case Descriptions for Different Reservoir Simulators	294
9.8.1 Procedure for UTCHEM	294
9.8.2 Procedure for ECLIPSE	296
9.8.3 Procedure for CMG	297
9.9 Summary	297
Chapter 10: Summary, Conclusions, and Recommendations for Future Work	301
10.1 Conclusions	301
10.2 Recommendations for Future Work	304
Appendix A: Linux Programming	306
A.1 Introduction of Linux	306
A.2 Frequently used Linux commands	307
Appendix B: Portable Batch System	310
B.1 Introduction of Portable Batch System	310
B.2 Term Definitions of Portable Batch System	311
B.3 Environment Variables of Portable Batch System	312
B.4 Script Sample for Portable Batch System	314

B.5 Useful Commands for Portable Batch System.....	317
Appendix C: Example of Observation Data File	318
Bibliography	319
Vita	335

List of Tables

Table 6-1	Producer locations and corresponding permeabilities.....	136
Table 6-2	Injector locations and corresponding permeabilities.....	137
Table 6-3	Observation data during the simulation time.....	138
Table 6-4	Different sampling intervals in two scenarios.....	138
Table 7-1	Reservoir rock, fluid, and geometrical properties	248
Table 7-2	RMS of permeability fields from SEIKF and EnKF.....	248
Table 8-1	Observation data during the simulation time.....	269

List of Figures

Figure 2-1: Flowchart of gradual deformation method.....	28
Figure 3-1: Traditional history matching using all the available data simultaneously and generating one “best matched” model.	46
Figure 3-2: Kalman filter based model continuously updated through sequential data assimilation.	47
Figure 3-3: Basic workflow chart of the EnKF. Reservoir model parameters are updated via the EnKF as new measurements are available.	48
Figure 3-4: Illustration of the EnKF from the point view of Bayesian concept.	48
Figure 3-5: Schematic curves of cumulative density function (CDF) and probability density function (PDF) with P10, P50 and P90 for standard Gaussian distribution. The horizontal axis is the value of random event and vertical axis is either CDF or PDF.	49
Figure 4-1: Reference permeability field after the lognormal transformation for a two-dimensional waterflooding synthetic case. The legend unit is milidarcy with the range from 0 (dark blue) to 9 (dark red). The orientation of the field is 45 degree with clear high and low permeability channel zones.	62
Figure 4-2: Bottomhole pressure for one injector and three producers in the reference field.	62
Figure 4-3: Oil production rate for four producers in the reference field.	63
Figure 4-4: Water production rate for four producers in the reference field.	63

Figure 4-5: The evolution of mean (left hand side) and variance (right hand side) fields updated by the EnKF with direct inverse calculation method at selected times.	64
Figure 4-6: Well bottomhole pressures simulated from the mean permeability model updated by the direct inverse method at different times.	65
Figure 4-7: Well oil production rates simulated from the mean permeability model updated by the direct inverse method at different times.	66
Figure 4-8: Well water production rates simulated from the mean permeability model updated by the direct inverse method at different times.	67
Figure 4-9: The evolution of mean (left hand side) and variance (right hand side) fields updated by the EnKF with standard assimilation method at selected times.	68
Figure 4-10: Well bottomhole pressures simulated from the mean permeability model updated by standard EnKF assimilation method at different times.	69
Figure 4-11: Well oil production rates simulated from the mean permeability model updated by standard EnKF assimilation method at different times.	70
Figure 4-12: Well water production rates simulated from the mean permeability model updated by standard EnKF assimilation method at different times.	71
Figure 4-13: The evolution of mean (left hand side) and variance (right hand side) fields updated by using square root algorithm with measurement perturbations at selected times.	72

Figure 4-14: Well bottomhole pressures simulated from the mean permeability model updated by using square root algorithm with measurement perturbations at selected times.	73
Figure 4-15: Well oil production rates simulated from the mean permeability model updated by using square root algorithm with measurement perturbations at selected times.	74
Figure 4-16: Well water production rates simulated from the mean permeability model updated by using square root algorithm with measurement perturbations at selected times.	75
Figure 4-17: The evolution of mean (left hand side) and variance (right hand side) fields updated by using square root algorithm without measurement perturbations at selected times.	76
Figure 4-18: Well bottomhole pressures simulated from the mean permeability model updated by using square root algorithm without measurement perturbations at selected times.	77
Figure 4-19: Well oil production rates simulated from the mean permeability model updated by using square root algorithm without measurement perturbations at selected times.	78
Figure 4-20: Well water production rates simulated from the mean permeability model updated by using square root algorithm without measurement perturbations method at selected times.	79
Figure 4-21: RMS profiles of BHP, OPR, and WPR for direct inverse, square root with measurement perturbations, and square root without measurement perturbations.	82

Figure 4-22: The evolution of mean (left hand side) and variance (right hand side) fields updated by using square root algorithm with measurement perturbation method at selected times. The assimilation interval is taken as 60 days.83

Figure 4-23: Well bottomhole pressures simulated from the mean permeability model updated by using square root algorithm with measurement perturbation method at selected times. The assimilation interval is taken as 60 days.....84

Figure 4-24: Well oil production rates simulated from the mean permeability model updated by using square root algorithm with measurement perturbation method at selected times. The assimilation interval is taken as 60 days.....85

Figure 4-25: Well water production rates simulated from the mean permeability model updated by using square root algorithm with measurement perturbation method at selected times. The assimilation interval is taken as 60 days.86

Figure 4-26: Mean permeability fields and corresponding variances obtained at the end of 300 days from square root algorithm without measurement perturbation, with the assumption of the dependence and independence of measurement errors, respectively.87

Figure 4-27: RMS profiles of BHP, OPR, and WPR for dependent and independent measurement errors using square root without measurement perturbation method.....90

Figure 5-1: Approximating pictures with the SVD (Kahaner *et al.*, 1989).103

Figure 5-2: Illustration of the improved sampling strategy through the SVD.....	103
Figure 5-3: Mean permeability fields and the associated variance maps at the end of 300 days with and without initial sampling fix.....	104
Figure 5-4: Well oil and water production rates simulated from the mean permeability model updated with and without the initial sampling fix, respectively, along with the reference curves.....	105
Figure 5-5: Logarithm permeability fields of reference, and three updated results at the end of 300 days using 100 realizations, 200 realizations, and 100 realizations with the SVD, respectively.....	106
Figure 5-6: Well performances from different updated permeability fields with the comparisons with the references.....	108
Figure 6-1: Three-dimensional view of reservoir porosity.....	139
Figure 6-2: Three-dimensional view of reservoir permeability in X direction with the mD unit.....	139
Figure 6-3: Areal view of well configurations, where ● denotes producers and ▼ represents injectors.....	140
Figure 6-4: Experimental data (points) and the matched exponential semivariogram model (curve). The vertical scale is normalized and the horizontal unit is the number of grid sizes.....	140
Figure 6-5: At the time of 620 days, logarithm permeability fields of averages of ensemble members from two measurement scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....	142

Figure 6-6: At the time of 640 days, logarithm permeability fields of averages of ensemble members from two measurement scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....144

Figure 6-7: At the time of 660 days, logarithm permeability fields of averages of ensemble members from two measurement scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....146

Figure 6-8: At the time of 620 days, logarithm permeability fields of averages of ensemble members from two sampling scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....148

Figure 6-9: At the time of 700 days, logarithm permeability fields of averages of ensemble members from two sampling scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....150

Figure 6-10: At the time of 800 days, logarithm permeability fields of averages of ensemble members from two sampling scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....152

Figure 6-11: First-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....153

Figure 6-12: Second-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....154

Figure 6-13: Third-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....155

Figure 6-14: Fourth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....156

Figure 6-15: Fifth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....157

Figure 6-16: Sixth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....158

Figure 6-17: Seventh-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....159

Figure 6-18: Eighth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....160

Figure 6-19: Ninth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....161

Figure 6-20: Tenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....162

Figure 6-21: Eleventh-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....163

Figure 6-22: Twelfth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....164

Figure 6-23: Thirteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....165

Figure 6-24: Fourteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....166

Figure 6-25: Fifteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....167

Figure 6-26: Sixteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....168

Figure 6-27: Seventeenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....169

Figure 6-28: At the time of 150 days, from left to right in each layer, pressure maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively. Legend scale is the same for every profile, from 214.5368 psi (blue) through 248.34 psi (red).175

Figure 6-29: At the time of 150 days, from top to bottom, three-dimensional pressure views of the reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively. Legend scale is the same for every profile, from 214.5368 psi (blue) through 248.34 psi (red).176

Figure 6-30: At the time of 240 days, from left to right in each layer, pressure maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively. Legend scale is the same for every profile, from 17.886 psi (blue) through 52.013 psi (red).....182

Figure 6-31: At the time of 240 days, from top to bottom, three-dimensional pressure views of the reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively. Legend scale is the same for every profile, from 17.886 psi (blue) through 52.013 psi (red).183

Figure 6-32: At the time of 600 days, water saturation maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively: (a) cross-sectional profiles for P6-I1; (b) cross-sectional profiles for P1-I1; (c) cross-sectional profiles for P2-I1. Legend scale is the same for every profile, from 0.0 (blue) through 1.0 (red).....184

Figure 6-33: At the time of 700 days, water saturation maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively: (a) cross-sectional profiles for P6-I1; (b) cross-sectional profiles for P1-P5; (c) three-dimensional slab views. Legend scale is the same for every profile, from 0.0 (blue) through 1.0 (red).185

Figure 6-34: At the time of 800 days, from left to right in each layer, water saturation maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively. Legend scale is the same for every profile, from 0.0 (blue) through 1.0 (red).....191

Figure 6-35: At the time of 800 days, water saturation maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively: (a) cross-sectional profiles for P6-I1; (b) cross-sectional profiles for P1-P5; (c) three-dimensional slab views. Legend scale is the same for every profile, from 0.0 (blue) through 1.0 (red).192

Figure 6-36: At the time of 1500 days, water saturation maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively: (a) cross-sectional profiles for P6-I1; (b) cross-sectional profiles for P1-P5; (c) three-dimensional slab views. Legend scale is the same for every profile, from 0.0 (blue) through 1.0 (red).193

Figure 6-37: Oil rate in well P1 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (denoted as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).....195

Figure 6-38: Oil rate in well P2 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (denoted as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).....197

Figure 6-39: Oil rate in well P4 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (denoted as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).....199

Figure 6-40: Water rate in well P1 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (denoted as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).....201

Figure 6-41: Water rate in well P2 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (detonated as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).....203

Figure 6-42: Water rate in well P3 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (detonated as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).....205

Figure 6-43: Cumulative oil recovery in terms of original oil in place from initial, 150, 240, and 600 days in the conventional EnKF method (detonated as “oldenkf”) and the weighted EnKF method (denoted as “weighted”), and the reference (denoted as “true”).....206

Figure 6-44: Cumulative oil recovery in terms of original oil in place from 800 days in the conventional EnKF method (detonated as “oldenkf”) and the weighted EnKF method (denoted as “weighted”), and the reference (denoted as “true”).....207

Figure 6-45: Cumulative oil recovery in terms of original oil in place from 1500 days in the conventional EnKF method (detonated as “oldenkf”) and the weighted EnKF method (denoted as “weighted”), and the reference (denoted as “true”).....208

Figure 6-46: Root mean square in the eleventh layer versus time in two methods, the conventional EnKF method (detonated as “Conventional EnKF”) and the weighted EnKF method (denoted as “Weighted EnKF”).....209

Figure 6-47: First-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....210

Figure 6-48: Second-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....211

Figure 6-49: Third-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....212

Figure 6-50: Fourth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....213

Figure 6-51: Fifth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....214

Figure 6-52: Sixth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....215

Figure 6-53: Seventh-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....216

Figure 6-54: Eighth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....217

Figure 6-55: Ninth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....218

Figure 6-56: Tenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....219

Figure 6-57: Eleventh-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....220

Figure 6-58: Twelfth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....221

Figure 6-59: Thirteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....222

Figure 6-60: Fourteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....223

Figure 6-61: Fifteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....224

Figure 6-62: Sixteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....225

Figure 6-63: Seventeenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.....226

Figure 6-64: Twelfth-layer logarithm permeability field of the averages of ensemble members after several selected assimilation times by using the weighted EnKF, respectively. (a) Reference; (b)-(h) Left hand side is results from conditional realizations while right hand side is results from unconditional realizations. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.227

Figure 6-65: Root mean square in the twelfth layer versus time in two methods, the conventional EnKF method (denoted as “Conventional EnKF”) and the weighted EnKF method (denoted as “Weighted EnKF”).	228
Figure 7-1: Uncertainty quantification workflow.	249
Figure 7-2: Workflow of real-time data integration using SEIKF.	250
Figure 7-3: Layer-by-layer permeability maps for the reference reservoir. Permeabilities are transformed into the logarithmic domain. Areal locations of injector (dot + through-going line) and producer (dot) wells are shown on the permeability map Layer #1. Permeabilities range from 1 mD (blue) to 4313 mD (red).	251
Figure 7-4: Evolution of the mean permeability field in Layer #1. Results obtained by use of SEIKF.	252
Figure 7-5: Evolution of the mean permeability field in Layer #2. Results obtained by use of SEIKF.	253
Figure 7-6: Evolution of the mean permeability field in Layer #3. Results obtained by use of SEIKF.	254
Figure 7-7: Evolution of the mean permeability field in Layer #3. Results obtained by use of EnKF.	255
Figure 7-8: Layer-by-layer comparison of mean permeability fields at the end of the history-match.	256
Figure 7-9: Comparison of cumulative oil production forecasts conducted using 100 history-matched models obtained via SEIKF and EnKF. The red curve signifies the reference model response.	257
Figure 7-10: Quantification of forecast uncertainty.	258

Figure 8-1: Schematic structure of sequential, parallel, and distributed and parallel computing in the EnKF process.....	270
Figure 8-2: Three-dimensional view of reservoir permeability.....	271
Figure 8-3: Evolution of the mean permeability field in Layer #12 after logarithm transformation. Permeabilities range from 1 mD (blue) to 1100 mD (red).....	272
Figure 8-4: Evolution of the mean permeability field in Layer #20 after logarithm transformation. Permeabilities range from 1 mD (blue) to 1100 mD (red).....	273
Figure 8-5: History matching and forecasting of surface oil rate and surface gas oil ratio in well P1501 by using the reservoir field obtained at the different time.....	274
Figure 8-6: Elapsed time in five different execution modes.....	275
Figure 9-1: Environment of UT_IRSP with new “Hist Match” Module.....	299
Figure 9-2: Automatic history matching module flow chart	300

Chapter 1: Introduction

1.1 INTRODUCTION TO HISTORY MATCHING AND DEVELOPMENT OPTIMIZATION

Reservoir simulation is an effective method for helping engineers estimate the underground oil and gas resources; nearly all major reservoir development decisions are in some way based on simulation results. In practice, reservoir simulation is comprised of reservoir model building, history matching, and forecasting.

During a reservoir production life, data of different natures can be classified as static or dynamic depending on their association with the flow of fluids in the reservoir. Data that have originated from geology, electrical logs, core analysis, fluid properties, seismic and geostatistics can be generally classified as static, whereas the information originating from well testing, pressure shut-in surveys, production history, bottomhole pressure from permanent gauges, water cut, and gas-oil ratio can be classified as dynamic. Data assimilation stems from the need to improve the output of our model. In particular, we want to correct our reservoir model, reduce the parameter uncertainty, and increase prediction creditability by assimilating field production data.

The data assimilation process of adjusting variables in a reservoir simulation model to honor observations of field data is known in the petroleum industry as history matching and has been studied extensively for a long time. History matching is utilized to improve reservoir characterization and to provide a better understanding of general flow mechanisms. It is not only mathematically and computationally challenging, but also non-unique. Classical history matching procedures whereby reservoir parameters are adjusted manually by trial-and-error can be tedious and inconsistent with the geological models. Automatic history matching was subsequently proposed with the intention of lessening manual work. Automatic history matching attempts to maintain the geological

model plausibility and estimate some uncertain reservoir variables, which may represent gridblock permeability, porosity and saturation, transmissibility or fluids relative permeability, or geometric variables that describe the shape, size and location of wells, while honoring observed field data. The basic process is to start from an initial parameter guess and to improve it by integrating field data in an automatic loop. The quality of the fit between measured and computed data is generally evaluated by using an objective function which includes both model mismatch and data mismatch parts.

A great effort has been made to automate the history matching process, but with limited success due to the high complexity of the problem and the large computational effort required by the practical applications, either in objective function evaluation (non-gradient based minimization method), or in gradient computation (gradient-based minimization methods). On the other hand, the increase in deployment of permanent sensors for monitoring pressure, temperature, or flow rate has added impetus to the related problem of continuous model updating. Since data output frequency in this case can be very high, integrating all the available data to generate a reservoir flow model is impractical. Instead, it has become important to incorporate the data as soon as they are obtained so that the reservoir model is always being updated. Therefore, it is no wonder that few processes could offer feasible solutions in practice. Automatic history matching remains a challenging research topic.

The ensemble Kalman filter initially proposed by Evensen (1994) is a Monte Carlo approach, in which an ensemble of models is used to update the parameters sequentially according to the chronological order in which the data are acquired and assimilated. As a result, this method is free from tedious derivation and implementation of the complex adjoint equations required by efficient gradient-based history matching methods. Its implementation thus turns out to be significantly simpler and independent of

any reservoir simulator. The final multiple matched models are suitable for the non-unique nature of ill-posed history matching and can be applied to uncertainty quantification for future predictions. In addition, the ensemble Kalman filter works sequentially and is suitable for real-time reservoir history matching.

1.2 RESEARCH OBJECTIVES

The ultimate goal of this proposed research is to develop and apply an efficient module for performing automatic history matching. The tasks that will be addressed in this research are as follows:

1. An ensemble Kalman filter methodology will be comprehensively studied by integrating production measurements and geological model information for continuous reservoir model updating.

2. Improve the initial sampling through the singular value decomposition.

3. Investigate various assimilation schemes for the assimilation step in the ensemble Kalman filter methodology.

4. Propose and apply a modified ensemble Kalman filter algorithm, which addresses the preference to the ensemble members through a non-equally weighting factor.

5. Explore a variation of the ensemble Kalman filter, singular evolutive interpolated Kalman filter, for history matching and quantify the forecasting uncertainty.

6. Apply distributed and parallel high-performance computing to improve work efficiency.

7. Develop an automatic history matching module to fulfill the above objectives, validate it through various cases, and add the module to reservoir management framework IRSP, Integrated Reservoir Simulation Platform (Jiang Zhang, 2005).

1.3 OUTLINE OF CHAPTERS

In this work, automatic history matching using the ensemble Kalman filter will be studied. Chapter 2 presents a summary of previous work on automatic history matching. The performance evaluation of each method is also commented on in this chapter.

Chapter 3 contains the theory of the ensemble Kalman filter, a sequential method formed by a time loop including two steps: forecasting and assimilation. The latter step involves the numerical solutions for the assimilation equation.

Chapter 4 investigates several different approaches to calculating assimilation equation and compares their matched performances through the applications for a two-dimensional two-phase waterflooding case. The assimilation interval and the dependence of measurement errors are also discussed.

Chapter 5 extends the discussion of the ensemble Kalman filter by performing different strategies for the initial sampling. Rather than randomly picking up realizations, the singular value decomposition properly selects the important cases from a large sampling pool. The efficiency of this sampling strategy is demonstrated through implementing the two-dimensional case used in Chapter 4. The ensemble size and initial sampling fix issues are also discussed.

Chapter 6 changes the algorithm of the ensemble Kalman filter and presents an alternative assimilation equation. Currently, the ensemble mean is averaged in an equal weight for each member. We propose a different approach, which weighs each member by the degree of its closeness to the observed data. We consequently change the covariance and assimilation equation. The comparison between the proposed method and the traditional ensemble Kalman filter is performed through a seventeen-layer reservoir case from primary to waterflooding with various well schedules. In addition, we also

investigate the issue of the uncertain geological information in the initial ensemble generating.

Chapter 7 introduces a singular evolutive interpolated Kalman filter, a variation of the ensemble Kalman filter. The workflow using this method is then illustrated in detail. The ultimate goal of history matching is to guide our prediction. The uncertainty of the multiple history-matched models needs to be quantified. In this chapter, the singular evolutive interpolated Kalman filter and uncertainty quantification are studied in a three-dimensional two-phase waterflooding case. We also run the same case by using the ensemble Kalman filter and make comparisons with the singular evolutive Kalman filter.

Chapter 8 covers the distributed/parallel computing using our Linux cluster for a three-dimensional three-phase compositional reservoir case. The simulation results and distributed/parallel high performance computing are discussed.

Chapter 9 presents the automatic history matching module developed in this dissertation. This chapter discusses the module structure, compilation, applications, and outlines procedures for using different reservoir simulators in conjunction with the module.

Chapter 10 presents the summary, conclusions and recommendations for future work.

Chapter 2: Literature Review

The objective of this chapter is to provide some background and a big picture on history matching. We will review the concept of history matching and the basic formula and development process of automatic history matching. The applications of gradient- and stochastic-based methods for history matching will then be summarized.

2.1 HISTORY MATCHING

With the increase in computational capability, numerical reservoir simulation has become an essential tool for reservoir engineering. In field applications, numerical reservoir models are constructed from available static and dynamic data. The goal of building reservoir models at different development stages is different. History matching is a process of tuning a reservoir model, or multiple reservoir models, such as estimations of unknown geological structure, rock and fluid properties, to honor the geologic and engineering data by reproducing the reservoir production history. The ultimate purpose of history matching is to enhance the prediction accuracy of future reservoir performance by maximally utilizing current sources and increasing the understanding of geological structure and fluid flow behavior.

History matching involves the process of inversing the measurements into a large number of unknown parameters. Three straightforward challenges impede history matching performance.

First, unknown parameters. For real field cases, the number of gridblocks ranges from a few thousand to several million. The rock properties, such as porosity and permeability in each gridblock, the fluid properties, such as saturation and pressure of each phase in each gridblock, and the rock-fluid interactive properties, such as capillary pressure curve and relative permeability curves, are generally unknown and need tuning

through history matching. In addition, reservoir heterogeneity intensifies the physical complexity and contributes to the uncertainty.

Measurements are the second challenge. When building reservoir models, many sources of data are available and can be grouped as follows (Türeyen, 2005):

1. **Geological data:** Any data related to the style of geological deposition:
 - Core data: porosity, permeability and relative permeability measurements
 - Well-log data: any suite of logs that indicates lithology, petrophysics and fluid types near the wellbore
 - Sedimentological and stratigraphic interpretation
 - Outcrop analog data
2. **Geophysical data:** Any data originating from seismic surveys:
 - Surfaces and faults interpreted on three-dimensional seismic
 - Seismic attributes
 - Rock physics data
 - Time lapse four-dimensional seismic data
3. **Reservoir engineering data:** Any data related to testing and production of the reservoir:
 - PVT data
 - Well test data
 - Production data

Each piece of information has its own characteristics and no single source of information alone determines the reservoir absolutely. All the information combined only provides a part of the real reservoir. History matching in general is an ill-defined problem with non-unique solutions. Given production data from an actual field, it is possible to construct many reservoir models, which can differ significantly from each other. Even

though a history match can be achieved, the underlying geological continuity should always be taken into account. Accurate representation of the underlying geology is of great importance for reservoir development, particularly in determining the optimal well placements (Guyanguler and Horne, 2001; Yeten *et al.*, 2002a) and optimizing well rate controls (Brouwer *et al.*, 2001; Brouwer *et al.*, 2002; Brouwer *et al.*, 2004; Yeten *et al.*, 2002a; Yeten *et al.*, 2002b; Sarma *et al.*, 2005; Sarma *et al.*, 2006).

On the other hand, with the process of reservoir exploration and development, more and more information is provided with high frequency, thanks to the implementation of the advanced measurement technologies. It is then suitable to perform the work of history matching and reservoir model updating sequentially. The challenge is how to imitate the real sequential process, while at each stage the data with different scale and the level of accuracy and redundancy could be brought into a single model or multiple models effectively and consistently.

The final problem is uncertainty quantification. It is always challenging to reduce and quantify reservoir prediction uncertainty. The non-uniqueness of history matching makes the forecasting more difficult and the incompleteness of information forces reservoir engineers to interpret beyond data. Such interpretations are subject to personal experience and intuition and are associated with a great deal of uncertainty. The challenge is how to quantify the uncertainty for the matched model(s) and obtain the confidence interval for reservoir recovery predictions.

2.2 AUTOMATIC HISTORY MATCHING

Traditional history matching is done manually. The three challenges mentioned above make such an approach time consuming and subjective and make geological continuity difficult. Automatic history matching is proposed with the intention of alleviating manual work while honoring the information consistency.

The purpose of history matching is to minimize the discrepancies between observed data and simulated results. Typically, such discrepancy minimization is expressed by an objective function. In the early studies of history matching (Jacquard and Jain, 1965; Jahns, 1966; Carter *et al.*, 1974; Chen *et al.*, 1974; Chavent *et al.*, 1975), only dynamic data were incorporated, and the objective function was simply defined by the weighted norm or distance between the observed production data $d_{obs} \in R^{N_d}$, and predicted data, $g(m) \in R^{N_d}$:

$$O(m) = [g(m) - d_{obs}]^T W_d [g(m) - d_{obs}], \quad (2.1)$$

where W_d is an $N_d \times N_d$ matrix, called the data weighting matrix, and N_d is the number of observed data. If W_d is chosen as the inverse covariance matrix of the measurement errors of the data integrated, C_D^{-1} , then minimizing the objective function given by Eq. (2.1) generates the maximum likelihood estimate of the reservoir model. In petroleum history matching, this is an underdetermined problem since the number of data is less than the number of model parameters adjusted. Therefore, a regularization term was introduced to obtain a unique solution (Jacquard and Jain, 1965). In general, the objective function with a regularization term is

$$O(m) = [g(m) - d_{obs}]^T W_d [g(m) - d_{obs}] + [m - m_o]^T W_m [m - m_o], \quad (2.2)$$

where W_m is an $N_m \times N_m$ matrix, called the model weighting matrix, m_o is a fixed N_m dimensional vector, and N_m is the number of model parameters.

One approach to choosing W_d , W_m and m_o is based on probability theory. Assuming the prior model is a multivariate Gaussian random variable with mean and

covariance matrix C_M , the conditional probability density function (pdf) or posteriori pbf for model m given observation d_{obs} can be derived with an application of Bayes' theorem (Gavalas *et al.*, 1976; Jackson, 1979; Tarantola and Valette, 1982). This gives

$$p(m | d_{obs}) = \frac{p(d_{obs} | m)p(m)}{p(d_{obs})} = c \exp[-O(m)], \quad (2.3)$$

where c is the normalizing constant, and $O(m)$ is the objective function given by

$$O(m) = \frac{1}{2}[g(m) - d_{obs}]^T C_D^{-1}[g(m) - d_{obs}] + \frac{1}{2}[m - m_{prior}]^T C_M^{-1}[m - m_{prior}]. \quad (2.4)$$

Eq. (2.4) has the physical meaning: under the framework of Bayesian inference, the solution of the history matching inverse problem is a posteriori pdf on the space of the reservoir model. This posteriori pdf includes two parts. The first part is a likelihood function, which involves the difference between the predicted data from a given model and the observed data. The second part is the priori distribution, which comes from the static data, such as geologic, core, well logs, and seismic data. Minimizing the objective function given by Eq. (2.4) yields the maximum a posterior (MAP) estimate, m_{∞} . In addition, sampling the posteriori pdf expressed in Eq. (2.4) can generate a set of realizations of reservoir parameters. In this way, the uncertainties in the observed data and model parameters can be integrated in the inverse procedure.

It is difficult, however, to sample a posteriori pdf for non-linear problems. Markov chain Monte Carlo (MCMC) is a rigorous sampling method. Unfortunately, it appears to be computationally inefficient for practical applications even with modifications (Oliver *et al.*, 1997). Oliver *et al.* (1996) and Kitanidis (1995)

independently proposed the randomized maximum likelihood (RML) method to generate an approximate sampling of a posteriori pdf. A conditional realization is generated by minimizing the objective function given by

$$O(m) = \frac{1}{2} [g(m) - d_{uc}]^T C_D^{-1} [g(m) - d_{uc}] + \frac{1}{2} [m - m_{uc}]^T C_M^{-1} [m - m_{uc}], \quad (2.5)$$

where m_{uc} is an unconditional realization defined by

$$m_{uc} = m_{prior} + C_M^{1/2} z_M, \quad (2.6)$$

and m_{uc} is obtained by adding noise to the observed data d_{uc} , which is given by

$$d_{uc} = d_{obs} + C_D^{1/2} z_D. \quad (2.7)$$

Here, z_M and z_D , respectively, are N_m and N_d dimensional column vectors of independent standard random normal deviates; $C_M^{1/2}$ and $C_D^{1/2}$ denote the square root of C_M and C_D . If C_D is diagonal, generating the square root simply takes the square root of the diagonal elements. A series of conditional realizations can then be generated by minimizing the objective function in Eq. (2.5) with different realizations of m_{uc} and d_{uc} . It was found that the RML method produced distributions of reservoir properties compatible with those of MCMC in a single phase test case (Liu *et al.*, 2001).

In summary, we have discussed the theoretical formula for automatic history matching. Practically, minimizing the objective function by tuning a large amount of model parameters, and honoring the physical meaning within the capacity of computational time, is big challenge.

Since the computational time is very sensitive to the number of unknown parameters, reparameterization was explored to reduce the number of model parameters being adjusted. One commonly used method is zonation. This method divides the whole reservoir into a small number of zones. Inside each zone, the unknown parameters are treated as uniform (Jacquard and Jain, 1965; Jahns, 1966; Carter *et al.*, 1974). This simplicity makes the zonation method easy to apply. However, the boundary divisions of the zones are questionable. Modeling errors may be introduced through these boundaries, and the discontinuity of rock properties at zonation boundaries after adjustment is another issue.

If the eigenvalues of the covariance matrix of the prior model decay rapidly, such eigenvalues and eigenvectors can be truncated effectively reducing the number of model parameters (Gavalas *et al.*, 1976; Oliver, 1996; Reynolds *et al.*, 1996). Unfortunately, the decline of eigenvalues is very slow in the most commonly used variograms, such as spherical and exponential variograms. A nugget, if included in the variogram, also causes the problems (Reynolds *et al.*, 1996). The value of the eigenvalue approach is thus trivial.

Abacioglu *et al.* (2001) introduced another method of reparameterization called subspace method. This method requires the calculations of subspace vectors and the gradient of the objective function with respect to the subspace vectors. A small number of subspace vectors are used in the early iterations and then the number is increased gradually afterwards.

The pilot point method was proposed by Marsily *et al.* (1984) to reduce the number of unknowns. The concept is that only rock properties at a small number of selected pilot point locations are perturbed to match production data. The rock properties at other grid blocks are then distributed with Kriging interpolation. This method has been applied in ground water hydrology (Kitanidis, 1995; RamaRao *et al.*, 1995; Gómez-

Hernández, 1997) and in matching production data in petroleum industry (Bissell *et al.*, 1989; Roggero and Guerillot, 1996; Wen *et al.*, 1997; Xue and Datta-Gupta, 1997). However, the number and location of pilot points is somewhat subjective and case dependent. Another drawback of the pilot point method is its convergence behavior. The tendency to generate realizations with extreme values (overshooting or over perturbation), and the consequent oscillation of the objective function as the iteration proceeds have been the primary limitations of using the pilot point method in reservoir characterization (Xue and Datta-Gupta, 1997; Liu *et al.*, 2001). In addition, the number of pilot points is user-determined and can be a field-scale problem (Roggero and Hu, 1998).

We have discussed the various approaches to reducing the number of parameters during history matching above. Automatic history matching has been investigated for at least a few decades, and there are abundant methods published in the petroleum literature. Generally speaking, such methods can be classified under two main categories: deterministic and stochastic algorithms, which will be explained in more detail in the following two sections.

2.3 DETERMINISTIC ALGORITHMS

The purpose of history matching is to minimize the discrepancies between observed data and simulated results (i.e. objective function). Deterministic algorithms use traditional optimization approaches and obtain one local-optimum reservoir model within the number of simulation iteration constraints. In implementation, the gradient of objective function is calculated and the direction of the optimization search is then determined.

The methods for gradient calculation have been widely used. Such methods include adjoint method (Chen *et al.*, 1974; Chavent *et al.*, 1975; Makhoulouf *et al.*, 1993; Li

et al., 2001; He *et al.*, 2002; Li *et al.*, 2003) and other approaches for sensitivity coefficients (Alpak, 2005; Bissell *et al.*, 1994; He *et al.*, 1997; Vasco *et al.*, 1999).

2.3.1 Adjoint Method

All deterministic methods need the calculation of sensitivity coefficients. Sensitivity coefficients are defined as the partial derivatives of the simulator output with respect to the parameters being adjusted. The adjoint method requires derivations and solutions of adjoint equations, which generate the sensitivity of each production datum with respect to all the reservoir model parameters. Such a system of adjoint equations is similar to but different from the system of finite-difference equations in the reservoir simulator. First, reservoir simulation runs forward in time, while the adjoint problem is solved backward and requires information from the simulation results. Secondly, the system of finite-difference equations for the forward problem is strongly nonlinear while the adjoint system is linear.

The adjoint method was applied to a water-oil two-phase problem by Wu *et al.* (1999) and to three-dimensional three-phase problems by Makhoulf *et al.* (1993), Li *et al.* (2001) and Li *et al.* (2003). Rodrigues (2005) used the truncated singular value decomposition and adjoint method for sensitivity matrix calculation.

Usually, the computational time for adjoint equations in each time step is less than the corresponding time required for the forward simulation equations. However, if the number of observed data is large, this method is computationally expensive and is impractical for real problems. Zhang *et al.* (2003) pointed out that if the number of production data to be history matched exceeds one hundred, calculation of all sensitivity coefficients by the adjoint method is computationally prohibitive. In general, the adjoint method is unfeasible for multiphase flow due to computational time and the need for sufficient length of production history.

2.3.2 Other Approaches for the Calculation of Sensitivity Coefficients

Even if the adjoint method is efficient for large scale problems, it is still limited by the embedment into the source code of the reservoir simulator. Since the simulator source code is not easily reachable, there are other options for the calculation of sensitivity coefficients. Among them, the direct calculation method is a main approach. In this method, classical optimization algorithms such as Gauss-Newton, Levenberg-Marquardt, steepest descent, conjugate gradient, and quasi-Newton typically converge fast and have been successfully applied.

2.3.2.1 Gauss-Newton and Modified Levenberg-Marquardt Algorithms

Suppose m_k denotes the current most probable model parameters and δm_{k+1} is the search direction for the next step. The Gauss-Newton method gives the following iterative procedure:

$$\delta m_{k+1} = -[C_M^{-1} + G_k^T C_D^{-1} G_k]^{-1} [C_M^{-1} (m_k - m_{prior}) + G_k^T C_D^{-1} (g(m_k) - d_{obs})], \quad (2.8)$$

where G_k denotes the matrix of sensitivity coefficients evaluated at m_k , that is, the derivatives of predicted data with respect to reservoir model parameters. The reservoir model parameters are then updated by

$$m_{k+1} = m_k + \mu_k \delta m_{k+1}, \quad (2.9)$$

where μ_k is the step size, usually obtained by the restricted step method (Fletcher, 1987).

If the number of model parameters is much larger than the number of observed data, Eq. (2.8) can be modified as

$$\delta m_{k+1} = (m_k - m_{prior}) + C_M G_k^T [C_D + G_k^T C_M G_k^T]^{-1} [G_k (m_k - m_{prior}) - (g(m_k) - d_{obs})]. \quad (2.10)$$

Compared with Eq. (2.8), Eq. (2.10) changes the problem from the dimension of the number of model parameters into the dimension of the number of observed data. Thus, it will be more favorable than Eq. (2.8) during the real case application.

Gauss-Newton approach has been reported for slow convergence or unacceptable matches of pressure data in cases of bad initial estimates (Wu *et al.*, 1999; Li *et al.*, 2001). To solve such problem, Bi *et al.* (2000) introduced a modified Levenberg-Marquardt algorithm

$$\delta m_{k+1} = \frac{m_k - m_{prior}}{1 + \lambda_k} + C_M G_k^T [(1 + \lambda_k) C_D + G_k^T C_M G_k^T]^{-1} \left[\frac{G_k (m_k - m_{prior})}{1 + \lambda_k} - (g(m_k) - d_{obs}) \right], \quad (2.11)$$

where λ_k is the Levenberg-Marquardt parameter evaluated at the k^{th} iteration step.

As computing all sensitivity coefficients is impractical if the amount of observation data and the number of model parameters are large, other alternatives which require only the gradient of the objective function could be considered, such as conjugate gradient algorithms and quasi-Newton algorithms.

2.3.2.2 Conjugate Gradient Method

The conjugate gradient method was originally proposed for solving linear systems, and then extended to nonlinear optimization. Gavalas *et al.* (1976) and

Makhlouf *et al.* (1993) applied this method to estimate gridblock permeabilities by history matching production data. However, it appears that the authors did not use a preconditioning process. Since the efficiency of the conjugate gradient method relies heavily on the selected preconditioner, Zhang and Reynolds (2002) tried two preconditioners. Although both improved the performance, the algorithms are still significantly less robust than the quasi-Newton method. The quasi-Newton method will be introduced briefly in the next section.

2.3.2.3 LBFGS Algorithm

The first and second derivatives of the objective function are known as the gradient and Hessian matrix of the objective function, respectively. Quasi-Newton methods, which are based on generating an approximation of the inverse of the Hessian matrix, require only the gradient of the objective function and thus avoid the computation of individual sensitivity coefficients needed to directly form the Hessian matrix. Among various quasi-Newton methods, the limited memory BFGS (Broyden-Fletcher-Goldfarb-Shanno) is possibly the most promising and widely used.

H_k and g_k denote the Hessian matrix and gradient of the objective function, respectively, and k is the iteration index. y_k and s_k , respectively, are the difference in the gradient $y_k = g_{k+1} - g_k$ and the model parameter difference in the iteration $s_k = m_{k+1} - m_k$. In quasi-Newton methods, the inverse of the Hessian matrix H_k^{-1} is approximated by a symmetric positive definite matrix \tilde{H}_k^{-1} , which is updated through y_k and s_k from each iteration. The standard updating BFGS formula is constructed by

$$\tilde{H}_{k+1}^{-1} = \gamma_k \left(\tilde{H}_k^{-1} - \frac{\tilde{H}_k^{-1} y_k y_k^T \tilde{H}_k^{-1}}{y_k^T \tilde{H}_k^{-1} y_k} + \nu_k \nu_k^T \right) + \frac{s_k s_k^T}{s_k^T y_k}, \quad (2.12)$$

where γ_k is the scaling factor and

$$\nu_k = \left(y_k^T \tilde{H}_k^{-1} y_k \right)^{1/2} \left(\frac{s_k}{s_k^T y_k} - \frac{\tilde{H}_k^{-1} y_k}{y_k^T \tilde{H}_k^{-1} y_k} \right). \quad (2.13)$$

The limited memory BFGS (LBFGS) uses a limited number of previous vectors to construct the approximation of the Hessian inverse at each iteration. LBFGS was applied by Zhang and Reynolds (2002), Zhang *et al.* (2003), and Gao and Reynolds (2006).

2.3.3 Gradual Deformation Method

Gradual deformation method was proposed by Roggero and Hu (1998), and Hu (2000). Unlike traditional deterministic methods, gradual deformation method does not require the calculation of the complicated gradient of objective function. It reduces the unknown parameter space of the reservoir model to a few combination coefficients. Calibrating the reservoir model by fitting these combination coefficients preserves reservoir model structure. This method can be applied to modify the whole reservoir model or only certain subdomains. Moreover, this method is not limited to Gaussian-related models. Non-Gaussian models can be transformed to Gaussian models to meet the method requirement.

The principal idea is that new realizations of random field can be written as the linear combination of a set of independent random Gaussian fields with expected mean μ and covariance C_m

$$y(k) = \sum_{i=1}^n k_i (y_i - \mu) + \mu. \quad (2.14)$$

The coefficients k_i are required to satisfy

$$\sum_{i=1}^n k_i^2 = 1. \quad (2.15)$$

The algorithm searches the optimum combination of different realizations to generate the reservoir model parameters in Gaussian space. There are many different variations of gradual deformation algorithm. The most basic form uses two realizations in this way

$$y(t) = y_1 \cos(t) + y_2 \sin(t), \quad (2.16)$$

where t is the deformation parameter.

The advantage of this algorithm is the transformation of the history matching problem into a one-dimensional optimization problem. The basic procedure starts to generate and combine two geostatistical realizations in Gaussian space. The combined realization is then transformed into real space for reservoir model input. The mismatch between the production response of the generated realization and the observation data is calculated. By using the optimization algorithm, a new value of deformation parameter t is derived and used for the combination of current realization with another new generated realization. The mismatch of production data is calculated again. If the mismatch is reduced, the recently combined realization replaces the old one. Otherwise, the previous one is retained for the next iteration. This is a loop and continues to incorporate new generated realization until the stopping criterion is reached. Figure 2-1 shows the flowchart of the gradual deformation method.

Gradual deformation method has received wide attention. Caers (2003) combined gradual deformation method, multiple-point geostatistics, and streamline simulation for

history matching under a variety of geological scenarios. Liu and Oliver (2004) assessed gradual deformation method by comparing the distribution of conditional realizations for a small problem with the standard distribution from a MCMC method, and the results showed that gradual deformation method produced acceptable distribution. Hu and Jenni (2005) extended the application from pixel-based models (e.g., Gaussian-related stochastic models) to object-based models (e.g., Boolean models).

2.4 STOCHASTIC ALGORITHMS

Although automatic history matching has been investigated for a couple of decades, the studies are dominated by deterministic methods. The advantages of applying stochastic algorithms are impeded mainly by the huge computational time. Thanks to the rapid development of computer memory and computing speed, stochastic algorithms are receiving more and more attention. Stochastic algorithms have three main direct advantages. First, stochastic approach generates a number of equal probable reservoir models and therefore is more suitable to non-unique history matching problems. Second, it is straight forward to quantify the uncertainty of performance forecasting by using these equal probable models. Uncertainty quantification through stochastic history matching has become a hot topic at present. Last, unlike local solutions from all the deterministic algorithms, stochastic algorithms theoretically reach the global optimum.

Several algorithms have been discussed widely and even commercialized. In this section, we select and briefly review the most representative methods, including genetic algorithms, simulated annealing, scatter and tabu searches, and Kalman filter algorithms.

2.4.1 Genetic Algorithms

Genetic Algorithm (GA) procedures were developed by John Holland in the early 1970s, at the University of Michigan (Holland, 1975). Genetic algorithm is an imitation

of biological principals of evolution, or “survival of the fittest”. This means that the algorithm will continue the search around the best combinations of parameters to further improve the match and reject the bad ones. The method involves a population of chromosomes and possible solutions are called individuals. Each chromosome is typically encoded as a bit string and processed by “natural selection” from one generation to the next generation, associated with inheritance, mutation, selection, and crossover.

Genetic algorithms find applications in computer science, engineering, economics, chemistry, physics, mathematics and other fields. We have seen a large number of papers about these methods on history matching in the petroleum industry (Romero *et al.*, 2000; Lach *et al.*, 2005; Jutila and Goodwin, 2006). In particular, genetic algorithms have been embedded in commercial software MEPO® (a registered trade mark of Scandpower Petroleum Technology) and EnABLE® (a registered trade mark of Energy Scitech, Roxar). BP’s “Top-Down Reservoir Modelling” approach was proposed by Williams *et al.* (2004) and implemented by Kromah *et al.* (2005). The approach uses a genetic algorithm as a global optimizer in conjunction with the reservoir simulator to achieve flexible and scaleable history matching and uncertainty quantification. Such a concept seems promising for practical application. Unfortunately, due to the computation cost arising from the slow convergence, genetic algorithm is still very limited in real problems.

2.4.2 Simulated Annealing

Simulated annealing is a probabilistic algorithm for global optimization problems, specifically locating a good approximation to the global optimum of a given function in a large search space. The name comes from annealing in metallurgy, a technique involving heating and controlled cooling of a material to increase the size of its crystals and reduce their defects. The heat causes the atoms to become unstuck from their initial positions

and wander randomly through states of higher energy. The slow cooling gives those more chances of finding configurations with lower internal energy than the initial one. By analogy with this physical process, each step of simulated annealing replaces the current solution by a random neighbor, chosen with a probability that depends on the difference between the corresponding function values.

There have been many papers on simulated annealing in the petroleum industry. In particular, Panda and Lake (1993), Ouenes *et al.* (1993), and Portella and Prais (1999) have applied the simulated annealing technique into reservoir history matching.

2.4.3 Scatter and Tabu Searches

Parallel to the development of genetic algorithms, Fred Glover, at the University of Colorado, established the principles and operational rules for tabu search and a related methodology known as scatter search (Glover, 1977). Scatter search (Glover, 1994) is designed to operate on a set of points, called reference points, which constitute good solutions obtained from previous solution efforts. The approach systematically generates linear combinations of the reference points to create new points, each of which is mapped into an associated feasible point. Tabu search (Glover and Laguna, 1997) is an intelligent guidance for the search process in order to screen certain solutions from being chosen on the basis of information that suggests these solutions may duplicate or significantly resemble previous solutions. Such screening is often done by defining suitable attributes of moves or solutions, and by imposing restrictions on a set of the attributes according to the search history. Two prominent techniques for exploiting search history in tabu search are recency and frequency memories. Recency memory is typically (though not invariably) a short-term memory that is managed by structures or arrays called “tabu lists”, while frequency memory more often fulfills a long term search functionality. The

heart of tabu search lies in its use of adaptive memory, which makes it possible to use the search history to guide the solution process.

A commercial implementation of scatter search and tabu search has been released under the name of OptQuest® engine, a software system developed by OptTek Systems, Inc. OptQuest® is bundled with Crystal Ball® (product of Decisioneering, Inc) and DMS™ (product of Landmark Graphics Corporation).

In the petroleum industry, April *et al.* (2003a) introduced an optimizer containing scatter search, tabu search and neural networks into some simple examples in petroleum exploration and production. Cullick *et al.* (2003) used such an optimizer in multiple field scheduling and production strategy. April *et al.* (2003b) applied it to portfolio management. For history matching, Sousa *et al.* (2006) used scatter search for simple history matching cases. Cullick *et al.* (2006) combined scatter search with nonlinear neural network proxy for history matching problems with a small number of unknown parameters.

2.4.4 Neighborhood Algorithm

The neighborhood algorithm is a stochastic optimization algorithm initially aimed for seismic inversion problems (Sambridge, 1999a, 1999b). It is now applied for history matching problems (Christie *et al.*, 2005; Suzuki and Caers, 2006). Similar to simulated annealing and genetic algorithms, it tries to find models of acceptable data in a multidimensional parameter space. The sampling of parameter space in this method is guided directly by the spatial properties of Voronoi cells.

In particular, Christie and co-workers have applied to history matching and uncertainty quantification in some hydrocarbon production forecasts (Christie *et al.*, 2002; Subbey *et al.*, 2003; Subbey *et al.*, 2004; Litvak *et al.*, 2005; Nicotra *et al.*, 2005; Christie *et al.*, 2006; Rotondi *et al.*, 2006; Erbas and Christie, 2007).

2.4.5 Kalman Filter Methods

The Kalman filter is the best known sequential data assimilation scheme. It was developed by Kalman (1960) for solving the linear problem. In the petroleum industry, Corser *et al.* (2000) applied Kalman filters for a real-time drilling monitor processing. Unfortunately, most cases have a nonlinear relationship between the measurements and the model parameters. In addition, the number of parameters is large: often two or more variables per grid block in the simulation study.

The extended Kalman filter tries to solve the non-linear problem through linearization. Eisenmann *et al.* (1994) implemented the extended Kalman filter for measuring the flushed zone resistivity. Bloemen *et al.* (2006) reported the application of the extended Kalman filter for gas lift wells, in particular, the parameter estimations in the drift-flux model. However, this extended Kalman filter still has difficulties when used with highly non linear problems and may lead to a linear instability in the error covariance evolution (Evensen, 1994).

In order to handle these difficulties, many variations of Kalman filters have been proposed. Among them, the ensemble Kalman filter (EnKF), a Monte Carlo approach, is promising. The EnKF, initially proposed by Evensen (1994), sequentially updates multiple models to capture the probability density function in the parameter map, such as the mean and variance of statistical information. The correlation between reservoir response (for example, production rates, gas-oil ratio, water cut, and bottom hole pressure) and reservoir model parameters (static parameters such as porosity and permeability, and dynamic parameters such as reservoir pressure and oil saturation in each gridblock) is approximated directly from the ensemble, which is different from the explicit evolution of the covariance matrix in the standard and extended Kalman filters.

Because of the EnKF's simple formulation and easy implementation, it has gained popularity for weather forecasting, oceanography, hydrology, and petroleum engineering.

Many applications successfully using the EnKF have been published in the petroleum industry. Nævdal *et al.* (2002) used the EnKF to update static parameters in near-well reservoir models by tuning the permeability field. Later, this approach was further developed to update the two-dimensional three-phase reservoir model by continuously adjusting both the static permeability field and dynamic saturation and pressure fields at each assimilation step (Nævdal *et al.*, 2005). Gu and Oliver (2005) used the EnKF to update porosity and permeability fields, as well as the saturation and pressure fields, and then applied it to match three-phase production data for the three-dimensional PUNQ-S3 reservoir model. Furthermore, Brouwer *et al.* (2004) and Nævdal *et al.* (2006) used the combination of EnKF for continuous model updating with an automated adjoint-based water flood optimization to optimize water flooding strategy.

Results from previous studies have shown that the EnKF is very efficient and robust. Liu and Oliver (2005) evaluated the performance of the EnKF by comparing it with a gradient-based minimization method on history matching of geologic facies. They found that the EnKF method outperformed the gradient-based minimization method in both computation efficiency and applicability. Gao *et al.* (2005) used the randomized maximum likelihood method and the EnKF to quantify uncertainty for the PUNQ-S3 problem. Wen and Chen (2005) presented a modified version of the EnKF. They added a “confirming” step to run reservoir simulation using the most recent updated static model parameters so that the updated static parameters and dynamic parameters are always consistent. Lorentzen *et al.* (2005) studied the robustness of the EnKF by running ten ensemble cases using the different initial conditions. They demonstrated that the EnKF is well-suited for forecasting with uncertainty. Zafari and Reynolds (2005) proposed a

theoretical relation between randomized maximum likelihood and the EnKF. They then showed two examples in which the EnKF does not properly perform. Interestingly, they used a linear case to show that the “confirming” step suggested by Wen and Chen (2005) is inappropriate. Reinlie (2006) used the traditional EnKF and further conditioned local permeability information around wellbore. Skjervheim *et al.* (2005) used the EnKF to incorporate four-dimensional seismic data. It showed that the EnKF could handle large seismic data and had a positive impact on matching the permeability field, even in the case with highly noisy measurement data. Dong *et al.* (2006) reported their study by using the EnKF for reservoir description to history match both production data and time-lapse four-dimensional seismic data and had a conclusion similar to that identified by Skjervheim *et al.* (2005). Lorentzen *et al.* (2006) applied the EnKF as an optimization routine for controlling downhole chokes in smart wells with the aim of optimizing waterflooding. Their simple synthetic reservoir case demonstrated that the EnKF works robustly and the results are in good agreement with their reference. With a streamline simulator, Park and Choe (2006) studied two issues: the low value of the estimate error covariance after some history matching periods; and the number versus quality of the measurement data. They suggested a regeneration step when the estimate error covariance reaches one fifth level of the initial estimate error covariance. As for measurements, they found that water saturation measurements near the irreducible water saturation or residual oil saturation are not sensitive to reservoir static parameters and can be ignored. This ignorance and measurement selection helps to avoid ensemble deviation and improves the history matching of reservoir porosity and permeability. Arroyo-Negrete (2006) used the EnKF, streamline assisted tool and proposed streamline covariance localization. This approach was reported to be without problems such as overshooting and non-Gaussian distribution. Haugen *et al.* (2006) presented a successful

application of the EnKF to a North Sea field case, using the real production data. Similarly, Evensen *et al.* (2007) studied a North Sea case and investigated more parameters, including initial fluid contacts, vertical transmissivity multipliers, and fault transmissivity multipliers. They discussed the non-Gaussian parameter distribution and pointed out that the EnKF is theoretically unrealistic if used directly on a multimodal *prior*, such as a reservoir consisting of channels.

2.5 SUMMARY

A comprehensive literature review on history matching theory, representative methods on deterministic and stochastic categories, and wide applications has been presented. The idea of using multiple models to capture the statistic map of reservoir properties is intuitively superior to the traditional method of selecting a single “best” matched model. A stochastic method to integrate both static and dynamic parameters with measurements is also desirable.

The EnKF is a Monte Carlo approach, in which an ensemble of models, instead of one model as in traditional history matching methods and in other Kalman filter related methods, is promising in various areas and has been widely reported in the petroleum industry. By estimating the state error covariance function directly from the ensemble, the EnKF avoids computing the adjoint equations or derivatives of sensitivity coefficients. Thus, its implementation turns out to be significantly simpler and independent of any reservoir simulator. The final multiple matched models are suitable for the non-unique nature of ill-posed history matching and can be applied to uncertainty quantification for future predictions. Another benefit of the EnKF method is that it works sequentially and nicely mimics the reservoir development in a real-time fashion, making it applicable to high performance computing.

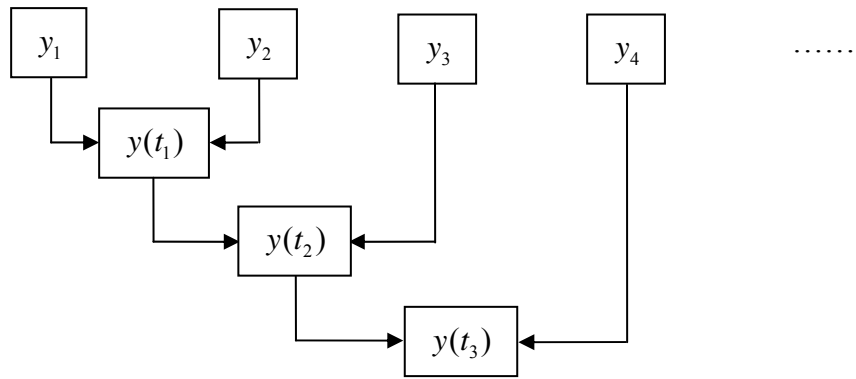


Figure 2-1: Flowchart of gradual deformation method.

Chapter 3: Theory of the Ensemble Kalman Filter

Data assimilation stems from the need to improve the model output. In particular, by assimilating field production data, we want to correct our reservoir model, reduce the parameter uncertainty, and increase prediction creditability. Such data assimilation is typically called history matching in the petroleum industry. In the previous chapter, we have reviewed the current status of history matching and shown that the ensemble Kalman filter (EnKF) is very promising. We summarize the basic characteristics as follows:

Traditional history matching updates only static quantities (such as porosity and permeability). It reruns the model iteratively until the match is reached:

- Repeated flow simulations of the entire production history
- Sensitivity coefficient calculations
- Not fully automated
- History matching repeated with all data when new data are available
- Not suitable for real-time reservoir model updating
- Difficult for uncertainty assessment

The EnKF updates reservoir model sequentially for both static and dynamic quantities (such as pressure and saturations):

- Suitable for updating non-linear reservoir simulation models on large scale
- One flow simulation for each ensemble member, easy for distributed computing
- No need of sensitivity coefficients
- Fully automated
- Production data assimilated sequentially in time

- Ensemble members updated sequentially in time and reflecting latest assimilated data
- Uncertainty of prediction always up-to-date and straightforward from the ensemble members

The objective of this chapter is to give the theory of the EnKF. Section 3.1 explains the outline of the EnKF algorithm in reservoir history matching. In Section 3.2, implementation procedures, particularly the assimilation step, are investigated and different matrix inversion schemes are given. Section 3.3 shows the approaches to measuring the result quantification, followed by a summary in Section 3.4.

3.1 OUTLINE OF THE ENKF ALGORITHM

As shown in Figure 3-1, reservoir engineers traditionally build one simulation model from a “most representative” geological interpolation. Through incorporating the whole production history and modifying the reservoir model, one “best match” is obtained. This model is then applied for recovery predictions and parameter sensitivity studies.

Kalman filter based methods perform sequentially and only update the model with the latest available data. An assimilation step is implemented to modify the model parameters, based on the difference between reservoir simulation responses and the data from the field. The updated model is then used to run forward until reaching the next measurement time, as shown in Figure 3-2.

Different from the general Kalman filter, the EnKF runs multiple simulation models independently, assimilates only the new measurements, and updates the multiple models simultaneously. After each updating, the EnKF describes model parameters through two statistical properties: mean and variance, the first representing the most probable model and the second depicting the change range, i.e., uncertainty. Aside from

the initial sampling, the EnKF consists of two steps for each time-recursive process: a forecasting step based on current state variables (which solves the flow equations with current static and dynamic parameters) and an assimilation step (which updates the state variables). Figure 3-3 illustrates the basic workflow chart of the EnKF. The evolution of dynamic variables is outputted from the reservoir simulator, dictated by the flow equations.

State variables include three types of parameters: (1) static parameters (e.g., permeability and porosity fields that are traditionally called static because they do not vary with time. However, in the EnKF, static parameters are updated with time. We use this traditional concept for convenience), (2) dynamic parameters (e.g., pressure and phase saturations of the entire model that are usually solutions of the flow equations), and (3) production data (e.g., well production rate, bottom-hole pressure, water cut, etc., which are usually measured at wells). State variables for each simulation model form a state vector and the ensemble of state variables forms an ensemble matrix. Thus, we have

$$y_{k,j} = \begin{bmatrix} m_s \\ m_d \\ d \end{bmatrix}_{k,j}, \quad (3.1)$$

where $y_{k,j}$ is the j -th ensemble member of state vector at time t_k . m_s and m_d are static and dynamic vectors, and d is the production data vector. In this dissertation, m_s is the permeability at each cell of the reservoir model with dimension of N being the total number of active cells; m_d has the dimension of $2N$ and includes pressure and water saturation at each cell; and d includes measurements, such as bottom-hole pressure, oil production rate and water production rate at wells with dimension of $N_{d,k}$. The dimension of state vector is $N_{y,k}$, which is equal to $3N + N_{d,k}$ and can change with

time t_k to account for the different amount of production data at the different time. Suppose there are a total of N_e number of models, we define the ensemble matrix holding the ensemble members $y_i \in \mathfrak{R}^{N_y}$ as

$$Y_k^f = (y_{k,1}^f, y_{k,2}^f, \dots, y_{k,N_e}^f), \quad (3.2)$$

with the dimension of $N_{y,k} \times N_e$.

In the remaining parts, we drop out the superscript f and subscript k for convenience. The ensemble mean is stored in each column of \bar{Y} , which is defined as

$$\bar{Y} = Y 1_{N_e}, \quad (3.3)$$

where $1_{N_e} \in \mathfrak{R}^{N_e \times N_e}$ is the matrix where each element is equal to $\frac{1}{N_e}$. We then define

the ensemble perturbation matrix as

$$Y' = Y - \bar{Y} = Y(I - 1_{N_e}). \quad (3.4)$$

The ensemble covariance matrix $C_{y,e} \in \mathfrak{R}^{N_y \times N_y}$ can be defined as

$$C_{y,e} = \frac{Y'(Y')^T}{N_e - 1}. \quad (3.5)$$

3.1.1 Measurement Perturbation

Given a vector of measurements $d \in \mathfrak{R}^{N_d}$ with N_d being the number of measurements, we can define the N_e vectors of perturbed observations as

$$d_j = d + \varepsilon_j, \quad j = 1, 2, \dots, N_e \quad (3.6)$$

which can be stored in the columns of a matrix

$$D = (d_1, d_2, \dots, d_{N_e}) \in \mathfrak{R}^{N_d \times N_e}, \quad (3.7)$$

while the ensemble of perturbations, assumed to be unbiased and Gaussian with ensemble mean equal to zero, can be stored in the matrix

$$E = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_{N_e}) \in \mathfrak{R}^{N_d \times N_e}, \quad (3.8)$$

from which we can construct the ensemble representation of the measurement error covariance matrix

$$C_{d,e} = \frac{EE^T}{N_e - 1}. \quad (3.9)$$

Note that $C_{d,e}$ is a diagonal matrix if the measurement errors are uncorrelated.

Considering the construction of the state vector, the relationship between the observed data and the true state vector can be written as

$$d_{obs} = Hy^{true} + \varepsilon, \quad (3.10)$$

where H is a matrix operator, depending on the number of observations, which relates the state vector to reservoir response data. Because the data are part of the state vector as

shown in Eq. (3.1), H is a trivial matrix whose elements are only ones or zeros. Following the structure of Eq. (3.1), we can always arrange it as

$$H = [\mathbf{0} | \mathbf{I}], \quad (3.11)$$

where \mathbf{I} is an identity matrix.

In implementation, the construction of H is not essential. Pre-multiplied by H simply selects the corresponding rows of a matrix. Similarly, post-multiplied by H^T chooses the corresponding columns of a matrix.

Burgers *et al.* (1998) showed that observations must be treated as random variables to avoid a too-low variance after data assimilation. Random perturbations are added into the measured data, creating a suite of observation sets for the ensemble models. The relationship between the perturbed observation and the true state vector is written as

$$d_{obs,j,k} = H_k y^{true} + \varepsilon_k + \nu_{j,k} = d_{obs,k} + \nu_{j,k} \quad (3.12)$$

where ε_k is the unknown measurement error; $\nu_{j,k}$ is the perturbation added to the noisy measured data, $d_{obs,k}$, to form the observations for the j -th ensemble members. Both ε_k and $\nu_{j,k}$ are Gaussian distributions with zero mean and $C_{d,e,k}$ error covariance, i.e., $E[\varepsilon_k \cdot \varepsilon_k^T] = E[\nu_{j,k} \cdot \nu_{j,k}^T] = C_{d,e,k}$; the noise and perturbation are uncorrelated, i.e., $E[\varepsilon_k \cdot \nu_{j,k}^T] = E[\nu_{j,k} \cdot \varepsilon_k^T] = 0$ for all j and k . Here, k is the index for time t_k .

3.1.2 Forecasting Step

The EnKF is a sequential method, which means that the model is integrated forward in time, and whenever new measurements are available, these measurements are used to enrich the model before the further integration progresses. Within the framework of model integration, a reservoir simulator is employed for reservoir modeling. The state variables $y_{k,j}$ are advanced in time as

$$y_{k,j} = f(y_{k-1,j}) \quad (j = 1, 2, \dots, N_e), \quad (3.13)$$

where f represents reservoir flow equations, coded in the reservoir simulator.

Note that in this forward step only the dynamic variables m_d (such as reservoir pressure and saturation in each gridblock) and corresponding response data are updated. The static variables m_s (such as permeability in each gridblock) are unchanged. Also note that the independence of the ensemble models at the forecasting step allows the benefit of distributed processing.

3.1.3 Assimilation Step

The assimilation equation, expressed in terms of the ensemble covariance matrices, is

$$Y^u = Y + C_{y,e} H^T (H C_{y,e} H^T + C_{d,e})^{-1} (D - HY) \quad (3.14)$$

$$= Y + Y' Y'^T H^T (H Y' Y'^T H^T + E E^T)^{-1} (D - HY). \quad (3.15)$$

In Eq. (3.14), the term $C_{y,e} H^T (H C_{y,e} H^T + C_{d,e})^{-1}$ is called Kalman gain. When the ensemble size, N_e , is increased by adding random samples, the analysis computed

from this equation will converge towards the exact solution of Eq. (3.14) with $C_{y,e}$ and $C_{d,e}$ replaced by the exact covariance matrices C_y and C_d .

A big assumption for the EnKF is that the model parameters are Gaussian distribution. Here, we explain this assumption within Bayesian statistical framework. Suppose at the end of time t_1 , the probability density function (PDF) of the state vector is known. At the end of time of t_2 , new measurements are available and the PDF up to t_2 is regarded as the *prior* PDF. By using the assimilation step, the EnKF members are conditioned with these new measurements and the PDF turns into the *posterior* PDF. From this point of view, the solution provided by the EnKF is based on maximizing the *posterior* PDF of the state vector within the context of Bayesian inversion. It is equivalent to minimizing the variances of the *posterior* covariance matrix with the assumption that the variables including model errors, measurement errors and variables in the state vector are Gaussian distributions. Such distribution assumption makes the *posterior* PDF still Gaussian distribution. The sequential EnKF process within Bayesian frame is illustrated in Figure 3-4.

Note that reservoir permeability is usually lognormally distributed and needs a transformation to meet the EnKF requirements. Also note that it is insufficient to describe a non-Gaussian distribution only by mean and covariance. The study of the EnKF on non-Gaussian model parameters is another important area which we will not discuss. In this dissertation, we assume all the parameters meet the EnKF assumption requirement.

3.2 ENKF ASSIMILATION SCHEMES

For the EnKF assimilation step, Evensen (2004) recommended the square root schemes. However, we have not seen clear and extensive discussion on the EnKF implementation. In this section, we provide four approaches to solve the assimilation step

described by Eq. (3.10): direct inverse calculation, standard EnKF assimilation calculation, and square root algorithms with and without measurement perturbations.

3.2.1 Direct Inverse Calculation

The intuitive way to solve Eq. (3.10) is to directly calculate and invert covariance matrices. We assume the independence among the measurement errors and $(HC_{y,e}H^T + C_{d,e})$ is full rank and invertible. Hence, the measurement error covariance matrix $C_{d,e}$ is diagonal. The basic procedure for solving Eq. (3.10) is as follows:

1. Construct $C_{y,e}H^T$ and $(D - HY)$;
2. Construct $HC_{y,e}H^T + C_{d,e}$;
3. Compute $(HC_{y,e}H^T + C_{d,e})^{-1}(D - HY)$ with solving linear equations;
4. Compute $C_{y,e}H^T(HC_{y,e}H^T + C_{d,e})^{-1}(D - HY)$;
5. Finally update Eq. (3.10): $Y^u = Y + C_{y,e}H^T(HC_{y,e}H^T + C_{d,e})^{-1}(D - HY)$.

3.2.2 Standard EnKF Assimilation Calculation

For Eq. (3.15), the potential singularity of the inverse computation requires the use of a pseudo inverse. The traditional way is to compute the eigenvalue decomposition directly,

$$HY'Y'^T H^T + EE^T = Z\Lambda Z^T, \quad (3.16)$$

which has the inverse (or pseudo inverse if the matrix is singular)

$$(HY'Y'^T H^T + EE^T)^{-1} = Z\Lambda^{-1}Z^T. \quad (3.17)$$

The cost of the eigenvalue decomposition is proportional to N_d^2 and becomes unaffordable for large N_d , where N_d is the number of measurements.

An alternative solution for large N_d is to compute and store only the first N_e columns of N_d :

With the assumption of uncorrelated reservoir responses and measurement errors, we have

$$HY' E^T \equiv 0$$

$$HY' Y'^T H^T + EE^T = (HY' + E)(HY' + E)^T. \quad (3.18)$$

Compute the singular value decomposition:

$$HY' + E = U\Sigma V^T;$$

Then Eq. (3.18) becomes

$$HY' Y'^T H^T + EE^T = U\Sigma V^T V \Sigma^T U^T = U\Sigma \Sigma^T U^T. \quad (3.19)$$

With comparison of Eq. (3.19) and Eq. (3.16), $\Sigma \Sigma^T$ and the singular vectors contained in U are identical to Λ and the first eigenvectors in Z , respectively. The benefit of this procedure is the efficient inversion computation in most practical situations.

Eq. (3.15) then turns into

$$Y^u = Y + Y'(HY')^T U \Lambda^{-1} U^T (D - HY). \quad (3.20)$$

This can be implemented with the following steps:

1. Compute the matrix HY' and the singular value decomposition $(HY' + E)$ to get U and Σ ;
2. Compute $X_1 = \Lambda^{-1} U^T$;
3. Compute $X_2 = X_1 (D - HY)$;
4. Compute $X_3 = U X_2$;
5. Compute $X_4 = (HY')^T X_3$;
6. Finally update: $Y^u = Y + Y' X_4$.

3.2.3 Square Root Algorithm with Measurement Perturbations

Introducing the matrix holding the measurements of the ensemble perturbations, $S = HY' \in \mathfrak{R}^{N_d \times N_e}$, we now define the matrix $B \in \mathfrak{R}^{N_d \times N_d}$ as

$$B = SS^T + (N_e - 1)C_d \quad (3.21)$$

and its ensemble approximation B_e as

$$B_e = SS^T + (N_e - 1)C_{d,e} = SS^T + EE^T. \quad (3.22)$$

Assuming that B is of full rank such that B^{-1} exists, we compute the eigenvalue decomposition $Z \Lambda Z^T = B$ and obtain

$$B^{-1} = Z \Lambda^{-1} Z^T, \quad (3.23)$$

where all matrices are of dimension $N_d \times N_d$.

A solution for the ensemble perturbations then is

$$Y^{u'} = \left(Y' V_2 \sqrt{I - \Sigma_2^T \Sigma_2} \right) \Theta^T, \quad (3.24)$$

where $X_2 \in \mathfrak{R}^{N_d \times N_e}$ as $X_2 = \Lambda^{-\frac{1}{2}} Z^T S$, and the singular value decomposition of X_2 , $U_2 \Sigma_2 V_2^T = X_2$, with $U_2 \in \mathfrak{R}^{N_d \times N_d}$, $\Sigma_2 \in \mathfrak{R}^{N_d \times N_e}$ and $V_2 \in \mathfrak{R}^{N_e \times N_e}$.

Note that the additional multiplication with a random orthogonal matrix Θ^T can be easily constructed, by using the right singular vectors from a singular value decomposition of a random $N_e \times N_e$ matrix.

When B is singular it is possible to compute the pseudo-inverse B^+ of B . It is convenient to formulate the analysis schemes in terms of the pseudo-inverse, since the pseudo-inverse $B^+ \equiv B^{-1}$ when B is of full rank. The algorithm then is valid in the general case.

Here we use an algorithm where the inverse is computed in the N_e – dimensional ensemble space rather than the N_d – dimensional measurement space. The key to this algorithm is a new approach for computing the inverse of B in the case when $N_d > N_e - 1$. The case when $N_d \leq N_e - 1$ is trivial since B will have full rank.

We assume that B has rank equal to $N_e - 1$, which will be the case if the ensemble is chosen properly and the measurement operator has full rank. The singular value decomposition of S is:

$$U_0 \Sigma_0 V_0^T = S, \quad (3.25)$$

with $U_0 \in \mathfrak{R}^{N_d \times N_d}$, $\Sigma_0 \in \mathfrak{R}^{N_d \times N_e}$ and $V_0 \in \mathfrak{R}^{N_e \times N_e}$.

The pseudo-inverse of S is defined as $S^+ = V_0 \Sigma_0^+ U_0^T$, where $\Sigma_0^+ \in \mathfrak{R}^{N_e \times N_d}$ is diagonal and defined as $diag(\Sigma_0^+) = (\sigma_1^{-1}, \sigma_2^{-1}, \dots, \sigma_{N_e-1}^{-1}, 0)$. The matrix product is $\Sigma_0 \Sigma_0^+ = \tilde{I}_{N_e-1} \in \mathfrak{R}^{N_d \times N_d}$, where \tilde{I}_{N_e-1} has the first $N_e - 1$ diagonal elements equal to one and the rest of the elements in the matrix are zero.

Defining $X_0 = \Sigma_0^+ U_0^T E$ which is an $N_e \times N_e$ matrix with rank equal to $N_e - 1$, we proceed with a singular value decomposition $U_1 \Sigma_1 V_1^T = X_0$, where all matrices are $N_e \times N_e$. Through designating X_2 as

$$X_2 = \left(I + \Sigma_1^2 \right)^{-\frac{1}{2}} U_1^T \tilde{I}_{N_e-1} V_0^T, \quad (3.26)$$

we then end up with the same final update Eq. (3.24).

For implementation, we give the following steps:

1. Compute the singular value decomposition: $U_0 \Sigma_0 V_0^T = S$;
2. Form the matrix product: $X_0 = \Sigma_0^+ U_0^T E$;
3. Compute the singular value decomposition of X_0 : $U_1 \Sigma_1 V_1^T = X_0$;
4. Form the matrix product: $X_1 = U_0 \Sigma_0^{+T} U_1$;
5. Update the ensemble mean from the equation:
$$\bar{y}^u = \bar{y}^f + Y^T S^T X_1 \left(I + \Sigma_1^2 \right)^{-1} X_1^T \left(d - H \bar{y}^f \right);$$
6. Form the matrix product: $X_2 = \left(I + \Sigma_1^2 \right)^{-\frac{1}{2}} X_1^T S = \left(I + \Sigma_1^2 \right)^{-\frac{1}{2}} U_1^T \tilde{I}_{N_e-1} V_0^T$;
7. Compute the singular value decomposition of X_2 : $X_2 = U_2 \Sigma_2 V_2^T$;
8. Evaluate the analyzed ensemble perturbations from
$$Y^{u'} = \left(Y^T V_2 \sqrt{I - \Sigma_2^T \Sigma_2} \right) \Theta^T$$
and add the mean to arrive at the final analyzed ensemble.

3.2.4 Square Root Algorithm without Measurement Perturbations

Another approach is to use square root algorithm with the ensemble measurement error covariance matrix $C_{d,e}$ approximating C_d . This method needs to specify $C_{d,e}$, which can be generated by first sampling the matrix of ensemble perturbations E and then computing $C_{d,e}$ with Eq. (3.8).

To implement this procedure, Eq. (3.25) and its pseudo-inverse are used: $U_0 \Sigma_0 V_0^T = S$ and $S^+ = V_0 \Sigma_0^+ U_0^T$. $X_0 = (N_e - 1) \Sigma_0^+ U_0^T C_d U_0 \Sigma_0^{+T}$ is then computed.

We proceed with an eigenvalue decomposition of X_0 , $Z \Lambda Z^T = X_0$. After that, with the definition of $X_1 = U_0 \Sigma_0^{+T} Z$, the solution for the analysis ensemble perturbations can be expressed as

$$Y^{u'} = \left(Y' \sqrt{I - X_2^T X_2} \right) \Theta^T, \quad (3.27)$$

where X_2 is defined as

$$X_2 = (I + \Lambda)^{-\frac{1}{2}} X_1^T S = (I + \Lambda)^{-\frac{1}{2}} Z^T \tilde{I}_{N_e-1} V_0^T. \quad (3.28)$$

This approach is similar to the previous square root algorithm except that we first specify the measurement error covariance matrix.

3.3 MEASUREMENTS OF THE ENKF PERFORMANCE

It is highly desirable to quantify the matching results from the EnKF. Basically, the statistical information such as mean and variance describing the most likely trend and uncertainty, respectively, can be directly used for result quantification. Here, we discuss

the quantifications for the matching parameters and prediction results in two scenarios: one is for a synthetic case and the other is for the real field application.

For a synthetic case, the reference is known and can be compared with the mean of ensemble by the root mean square (RMS)

$$\begin{aligned} \text{RMS} &= \sqrt{\frac{1}{N} \sum_{i=1}^N (y_{\text{mean},i} - y_{\text{ref},i})^2} \\ &= \sqrt{\frac{1}{N} \frac{1}{N_e} \sum_{i=1}^N \sum_{j=1}^{N_e} (y_{i,j} - y_{\text{ref},i})^2}, \end{aligned} \quad (3.29)$$

where, i is the gridblock index; j is the index for ensemble members; N is the total number of reservoir active gridblocks; N_e is the total number of ensemble members; $y_{\text{mean},i}$ is the mean value in the i -th gridblock of the matched ensemble; $y_{\text{ref},i}$ is the “true” value in the i -th gridblock of the reference.

In addition, it is easy to plot and compare the predictions of production rate for each well and overall cumulative oil recovery from the matching field with the corresponding results from the reference model.

However, the reality is more difficult since we do not know the true answer. Fortunately, at the end of the final matching time, we can rerun the updated mean model from the beginning. By using the measurements from each well, similar to Eq. (3.29), the root mean square can be defined as

$$\text{RMS} = \sqrt{\frac{1}{N_t} \sum_{i=1}^{N_t} \frac{1}{N_{w,i}} \sum_{j=1}^{N_{w,i}} \frac{1}{N_{w,i,j}} \sum_{k=1}^{N_{w,i,j}} (y_{i,j,k}^{\text{mean}} - y_{i,j,k}^{\text{measure}})^2}, \quad (3.30)$$

where, i is time index; j is well index; N_t is the total number of measurement times; $N_{w,i}$, the total number of wells at the i -th measurement time, varies over the different time periods; $N_{w,i,j}$, the total number of measurement data in the j -th well at the i -th measurement time, varies both in the different wells and over the different time periods; $y_{i,j,k}^{mean}$ is the k -th simulation data in the j -th well at the i -th measurement time; $y_{i,j,k}^{measure}$ is the k -th measurement datum in the j -th well at the i -th measurement time. Note that the RMS is a scalar and we expect it is gradually smaller and smaller.

For prediction quantification, we run each matched model and obtain the statistical information, such as P10, P50, and P90. In other words, if we finally get N_e number of matched models, we run each case forward to the final time. During the prediction period, for any specified time, we have N_e number of prediction values and therefore we can calculate P10, P50, and P90 from the cumulative density function. Plotting all the P10 data with the prediction time gives P10 curve. Similarly, P50 and P90 curves are available. Figure 3-5 shows the probability density curve and cumulative density curve, P10, P50, and P90 in the standard Gaussian distribution.

Here, the cumulative density function is the cumulation of the probability of all the outcomes up to a given value. Similarly, the probability density function is the probability density of a continuous random variable.

Note that P10, a pessimistic estimation, means the value of the real reservoir has 10 percent possibility below this P10 value. In a similar way, P90, an optimistic estimation, means the value of the real reservoir has 90 percent possibility below this P90 value.

Also note that P50 shows the most likely trend while the interval of P10 and P90 can be taken as the uncertainty band.

3.4 SUMMARY

Chapter 3 has concentrated on the theory of the EnKF. The implementation flowchart, basic assumptions and relationship with Bayesian concept are illustrated. The EnKF is a Monte Carlo approach using multiple models sequentially. The initial realizations are usually generated randomly from the geological *prior* knowledge. A closed loop consisting of forecasting and assimilation steps follows. The forecasting step is implemented by running all the independent reservoir simulation models directly. Four different schemes for the assimilation step are studied in this chapter. The coding steps for each method are given in detail. The approaches for qualifying the matching results and predictions are also discussed.

This chapter is a fundamental background for the following several chapters, where we will discuss the impact of different assimilation schemes, improve the initial random sampling strategy, and modify the EnKF method by using a weighted mean in the assimilation equation.

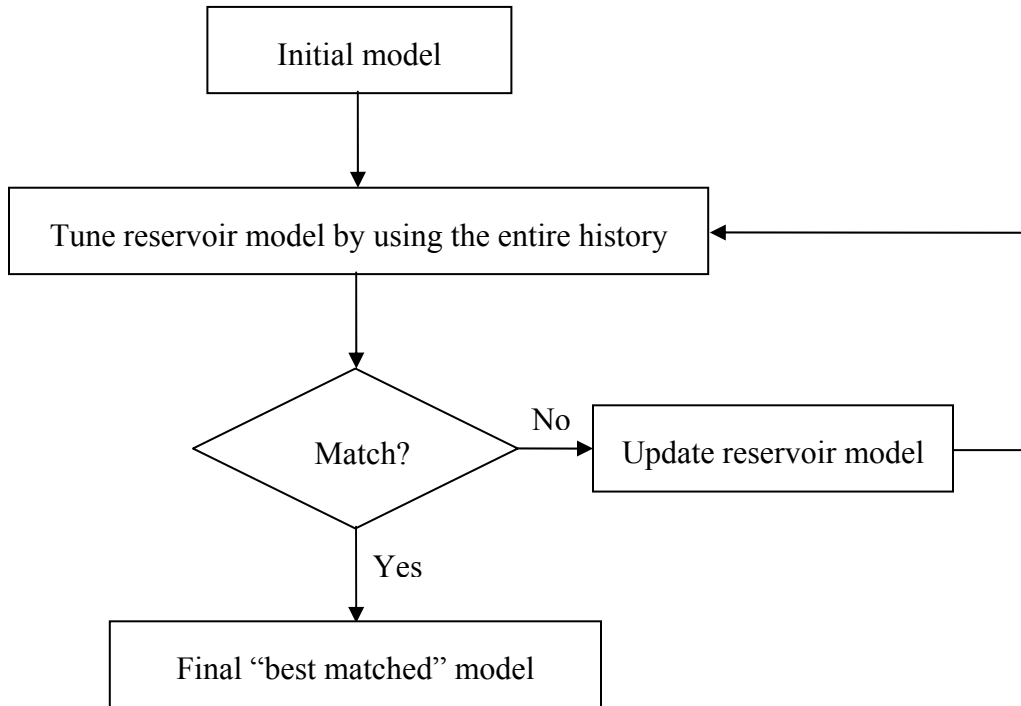


Figure 3-1: Traditional history matching using all the available data simultaneously and generating one “best matched” model.

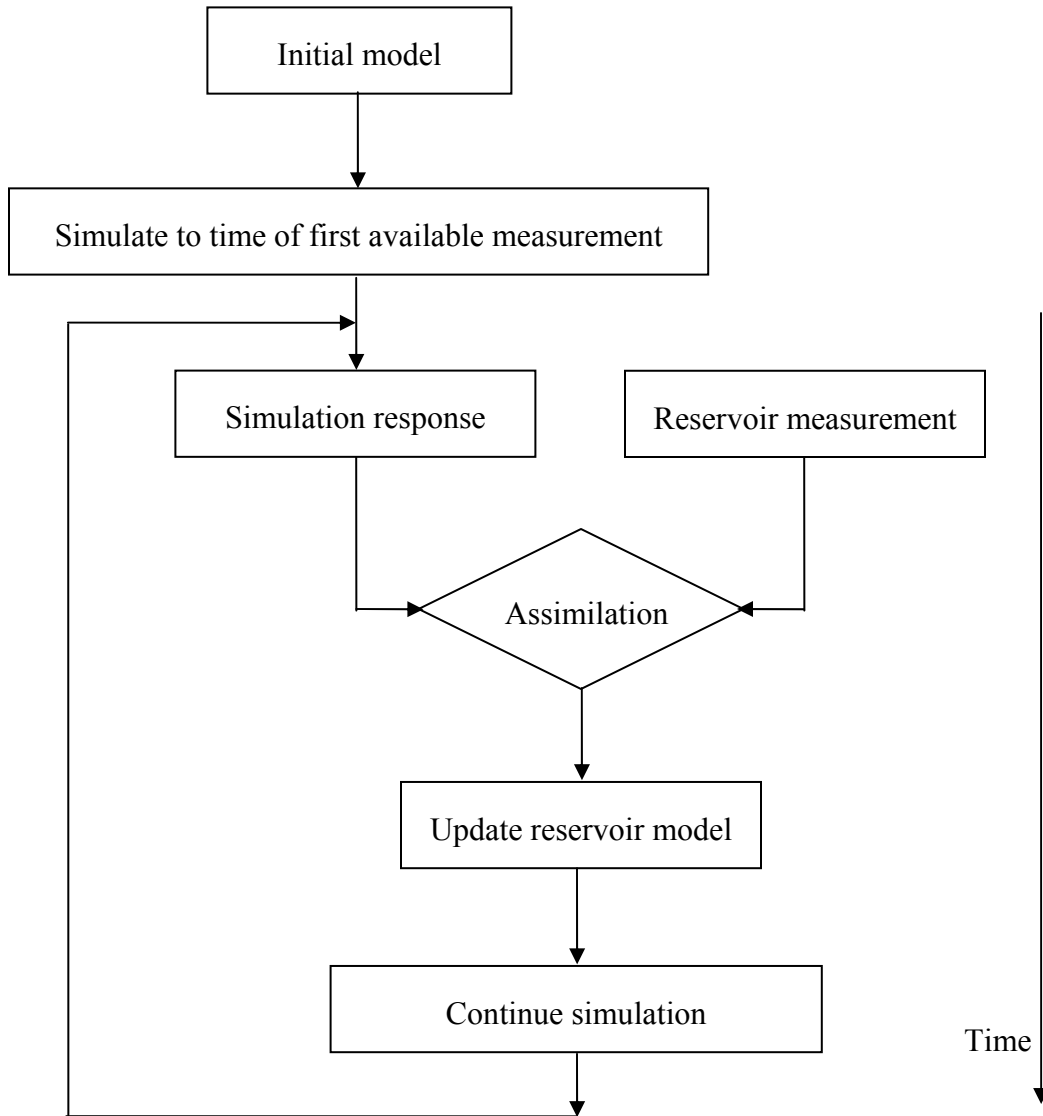


Figure 3-2: Kalman filter based model continuously updated through sequential data assimilation.

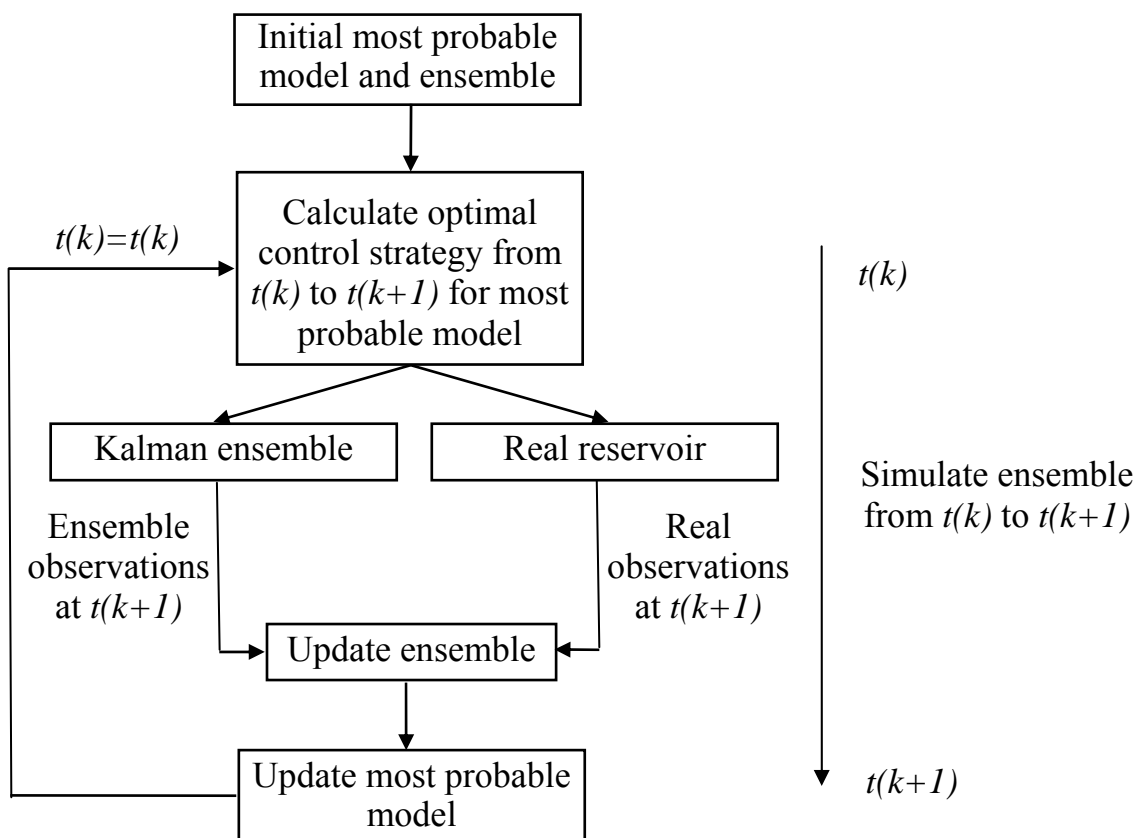


Figure 3-3: Basic workflow chart of the EnKF. Reservoir model parameters are updated via the EnKF as new measurements are available.

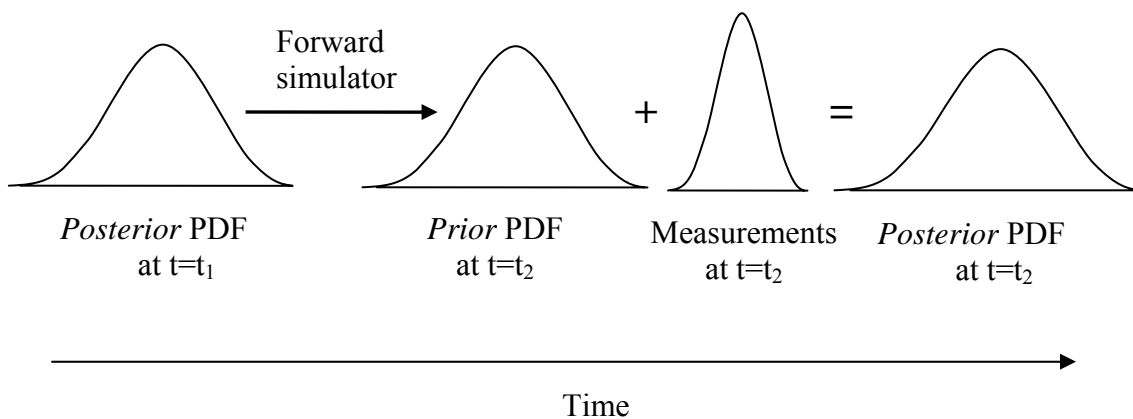


Figure 3-4: Illustration of the EnKF from the point view of Bayesian concept.

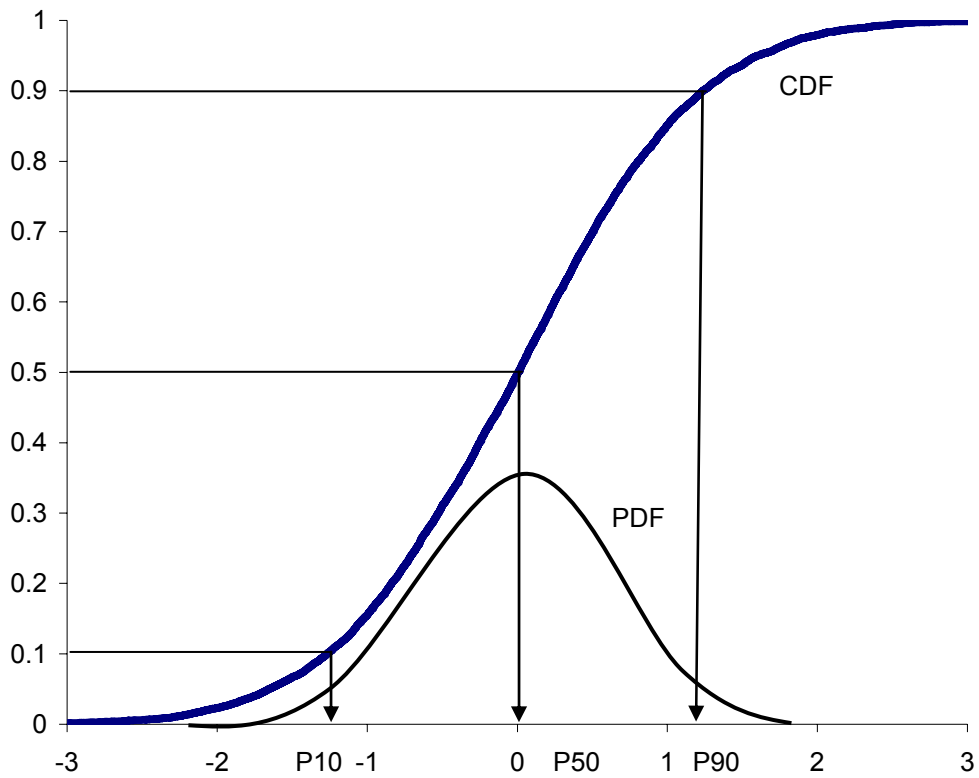


Figure 3-5: Schematic curves of cumulative density function (CDF) and probability density function (PDF) with P10, P50 and P90 for standard Gaussian distribution. The horizontal axis is the value of random event and vertical axis is either CDF or PDF.

Chapter 4: Investigation of Assimilation Schemes in the Ensemble Kalman Filter

In the previous chapters, we have reviewed the current status of history matching and the basic theory of the ensemble Kalman filter (EnKF). In this chapter, we will apply the EnKF to a two-dimensional waterflooding case. The advantage of the EnKF is demonstrated and the different assimilation methods are investigated. In addition, the impacts of assimilation interval and the assumption of independent measurement errors in the history matching results are discussed.

4.1 DESCRIPTION OF A TWO-DIMENSIONAL WATERFLOODING CASE

Figure 4-1 shows a geostatistical permeability reference map of a two-dimensional field ($50 \times 50 \times 1$ grid with cell size 20 feet \times 20 feet \times 2 feet), generated by the Sequential Gaussian Simulation method `sgsim.exe` from GSLIB package (Deutsch and Journel, 1998). The unit of permeability is millidarcy. After logarithm transformation, the permeability field, $\ln k$, has a Gaussian histogram with mean 6.0 and variance 3.0. The variogram of $\ln k$ is spherical with a range of 200 feet and 40 feet in the direction of 45 degree and 135 degree, respectively. All wells are vertical. In an aerial map, one injector (I) is in the center and four producers (P1 to P4 in anti-clockwise) are located in the four corners.

Initially, the reservoir is oil saturated and the pressure is 6,000 psi at the top. The injector has a primary constraint with a constant injection rate of 700 STB/day and a secondary constraint with a maximum bottomhole pressure (BHP) of 10,000 psi. All producers are producing constantly with a total volume of 200 STB/day and will switch to BHP control if the BHP drops down to 4,000 psi. The mobility ratio of water and oil is 10. Corey-type model with exponential 2 is used for relative permeability curves. For convenience, both oil and water have zero residual saturation. Compressibility and

capillary pressure are ignored. The commercial reservoir simulator ECLIPSE 100TM is used for reservoir simulation, running from 0 day up to 700 days. The results of BHP in each well, oil production rates (OPR) and water production rates (WPR) in producing wells are shown in Figures 4-2 through 4-4.

The reference permeability field is considered as truth and its corresponding dynamic data (BHP, OPR and WPR) are measured directly from the simulation output every 30 days up to 300 days. The measurement errors obey Gaussian distributions with standard deviations of 3.0 psi, 1.0 STB, and 2.0 STB for BHP, OPR and WPR, respectively.

Initial ensemble of 200 permeability models is generated unconditionally by the Sequential Gaussian Simulation method `sgsim.exe` with the same histogram and variogram as the reference field. Other parameters (porosity = 0.2, relative permeability curves, initial pressure = 6,000 psi, and initial water saturation = 0.0) are assumed known without uncertainty. Each ensemble member is updated at every 30 days by assimilating the observed production data (BHP, OPR, and WPR), followed by a confirming step proposed by Wen and Chen (2005). Also note that the reservoir reference and operation schedules are identical to the case used by Wen and Chen (2005).

There are totally 2,500 permeability values, unknown in each gridblock. We also have the same amount of unknown water saturation and pressure in each gridblock. Permeability is deemed as a static parameter while water saturation and pressure are dynamic parameters. Therefore, the state vector for the j -th ensemble member is formed in the following order

$$y_j = \begin{bmatrix} \ln k_1 \\ \ln k_2 \\ \vdots \\ \ln k_{2500} \\ S_{w,1} \\ S_{w,2} \\ \vdots \\ S_{w,2500} \\ P_1 \\ P_2 \\ \vdots \\ P_{2500} \\ OPR_1 \\ \vdots \\ OPR_4 \\ WPR_1 \\ \vdots \\ WPR_4 \\ BHP_1 \\ \vdots \\ BHP_4 \\ BHP_5 \end{bmatrix}_j, \quad (4.1)$$

where, $\ln k_1$ to $\ln k_{2500}$, $S_{w,1}$ to $S_{w,2500}$ and P_1 to P_{2500} denote lognormal permeability values, water saturations and reservoir pressures for the whole field, respectively. OPR_1 to OPR_4 and WPR_1 to WPR_4 are oil production rates and water production rates in four producers at the current observation time, respectively. BHP_1 to BHP_4 and BHP_5 symbolize bottomhole pressures in four producers and one injector at the current observation time.

4.2 DISCUSSION ON ASSIMILATION SCHEMES

With the same observation data and same initial realizations, we investigated the impact of different assimilation schemes on the matched results and their corresponding variances.

First, we used direct inverse calculation. Figure 4-5 shows the evolution of mean (i.e., estimation) and variance fields (i.e., uncertainty) computed from the ensemble at the end of 0, 30, 60, and 300 days, respectively. At the beginning, no measurement information is available. The geological *prior* knowledge in this case is uniform. At the end of the first month, the first 13 observations are obtained and assimilated into 200 ensemble models. The mean of these 200 updated models captures the main features of the reference permeability field, such as high and low permeability zones. At the same time, the uncertainty (shown as variance of the models) reduces, especially in the areas near well locations. At the end of 120 days, the permeability field has been well recovered and the variance decreases significantly. Later on, the field does not change much, which means that the useful information from the field measurements is less than that at the beginning time. Compared with the reference field, the matched result at the end of 300 days has the similar structure. However, the width of the high permeability band is smaller and looks a little bit messy.

With the mean permeability fields updated from different times, we reran the cases from the beginning and performed the predictions up to 700 days. Figure 4-6 presents well bottomhole pressures using the mean models updated at the end of 30, 60, 120, and 300 days, respectively. By using the mean models assimilating only production data (e.g., 30, and 60 days) in the early time, the predicted well performances, although improved, still significantly deviate from the reference case. Hence, the assimilation production data up to 60 days is not sufficient to capture the real spatial structure.

However, the updated model at 120 days has results very close to the performance of the reference model. In addition, it is of interest to figure out how the EnKF works gradually on the basis of four plots in Figure 4-6. The model at 30 days has values too high for all the wells. The model at 60 days is then adjusted and turns out to be too small. The model is tuned again and achieves very good matches within the range of results from 30 days and 60 days. Therefore, the EnKF demonstrates its high capability for fast convergence with very limited observation numbers (13 values at every measurement time in this case).

Similarly, Figure 4-7 and Figure 4-8 illustrate well oil production rates and water production rates by using the matched models at different times. Except for well P4, there is no significant difference between the curves from the matched models at the different times and the reference curves. At the end of 120 days, the model achieves good matches on both oil and water production rates at well P4. In particular, the peak of oil production rate and the water breakthrough time are well recovered. From Figure 4-6 to Figure 4-8, we see that the final model not only reproduces the production history, but also gives satisfactory predictions.

Figure 4-9 presents the evolution of mean and variance fields updated by the EnKF with standard assimilation method at the end of 30, 60, 120, and 300 days. Comparing with Figure 4-6, we found that the value change in the mean permeability field at the different assimilation step is milder. In other words, there is no dramatically high permeability zone, nor extremely low permeability area. The whole map does not show clearly high and low permeability bands, either. These characteristics are further confirmed from the variance maps. In comparison with the picture in Figure 4-6 at the corresponding time, the variance changes more slowly and still has high values even at the end of 300 days. This indicates that the standard EnKF assimilation scheme has

slower convergence and higher uncertainty, compared with the performance from the direct inverse approach at the same time period. The standard EnKF has difficulty to acquire severely heterogeneous reservoir field. Consequently, the uncertainty, represented by the ensemble variance, still keeps a high level even at the late production period.

In the similar procedure described previously, Figure 4-10 to Figure 4-12 lay out BHP, OPR and WPR by running the models updated from different times. Generally, we obtained acceptable matches for each well. However, these results are not as good as those obtained from the direct inverse approach.

Figure 4-13 shows the evolution of mean and variance fields updated by using square root algorithm with measurement perturbations. Figure 4-14 to Figure 4-16 are the corresponding well production curves associated with the references. This method converges fast and has clearly high and low permeability zones at the end of 300 days. All the well bottomhole pressures and production rates have good agreement with the references.

Figures 4-17 through 4-20 are results from the square root algorithm without measurement perturbations. This method also achieves satisfactory matches. Simply from the matched permeability fields and well data, we realized that the four methods all performed well. Roughly, direct inverse method, square root algorithms with and without measurement perturbation methods are better than standard EnKF assimilation method.

The CPU times for EnKF assimilations are similar: all within one second. Since the number of observation data at each assimilation step (thirteen values) is quite small and less than the ensemble size (two hundred), we do not worry about the rank loss and easily inverse the covariance matrix. More generally speaking, as long as the number of observation data is smaller than the ensemble size, we can use direct calculation.

However, in the case of large data at each assimilation step, particularly with the installation of permanent sensors and four-dimensional time lapse seismic data, the direct inverse is time consuming and no longer practical. The square root algorithms with and without measurement perturbations look promising since they change the inverse calculation from the traditional N_d – measurement space to the N_e – dimensional ensemble space. Here, N_d denotes the number of unknown model parameters and N_e represents the number of ensemble members. Typically, $N_d \gg N_e$. In addition, the square root algorithms are applicable for low-rank conditions. Hence, we recommend the square root algorithms for the assimilation step for the large scale observations.

With mean models from different methods at the different assimilation times, we ran simulation from the beginning up to 700 days. The purpose is to illustrate and compare history matching (from 0 day to 300 days) and forecasting (from 300 days to 700 days) behaviors with the reference results. We calculated the root mean square (RMS) values of the differences between well production data with the references. Figure 4-21 shows the RMS profiles of BHP, OPR, and WPR for direct inverse, square root with measurement perturbations, and square root without measurement perturbations. In Figure 4-21, these three methods have similar performances on well BHP, but square root algorithm with measurement perturbations has the best overall results for well OPR and WPR. From this point of view, we recommend this square root algorithm with measurement perturbations for the assimilation step in the EnKF implementation.

However, Lawson and Hansen (2004) expressed a doubt on the performance of the square root algorithm in the severely nonlinear problem because this algorithm more easily diverges from the non-Gaussian distribution and causes the ensemble collapse. Similarly, Leeuwenburgh *et al.* (2005) evaluated the standard EnKF assimilation and the square root algorithm using an ocean general circulation model. They found that the

mean states produced with the standard EnKF are generally slightly better than those from the square root algorithm. They also showed that the ensembles after assimilation step are highly non-Gaussian introduced by the square root algorithm. The authors projected that this problem is possibly a consequence of using the singular value decomposition, and such decomposition makes the abnormal realization contribute greatly to the updated ensemble members. In this paper, the authors' projection was confirmed by the skewness measure and χ^2 tests. Evensen (2007) reevaluated the square root schemes by a simple linear advection model and conjectured that the random rotation in the square root schemes had an undesirable impact on the results. Even for the linear system, such random rotation is necessary to ensure that a randomized ensemble which properly represents the error statistics will be obtained. Therefore, the real reason is still unclear.

In our application, we did not face the problem addressed by Leeuwenburgh *et al.* (2005). Though further investigation is needed, since the square root algorithm strictly follows the idea of the Kalman filter, we believe this algorithm is a good approach.

4.3 DISCUSSION ON ASSIMILATION INTERVAL

The measurement data, utilized in the discrete style, is always associated with errors and leads to uncertainty. Meanwhile, the determination of measurement frequency always makes the engineer more fidgety. Ideally, we want to extract the important information only at certain critical times and update reservoir models while monitoring the remaining data for some possible analyses. A current trend with the advanced instruments is to obtain as many measurements as possible. For example, increasing deployment of permanent sensors leads to a high frequency of data output. If we update the reservoir model through history matching as soon as the new data is available, it will be very time consuming and implementation unfriendly because of the relative large

ensemble size in the EnKF method. On the other hand, production data may be redundant and correlated. No new information will be absorbed if the same type of data is observed in a too-short period. The smart interval could liberalize engineers from the heavy work of data measurement and assimilation into the reservoir model. In addition, the redundant data might result in unexpected trouble. Park and Choe (2006) stated that some unwanted measurements could cause a worse adjustment. Therefore, it is valuable to assess the impact of the assimilation interval and properly sample the data from the field leaving the measurements insensitive to the model updating. In this section, different assimilation intervals and corresponding effects on the updated results are discussed.

For the same waterflooding case, we used the square root algorithm with measurement perturbation, but increased the assimilation interval from 30 days to 60 days. The simulation output frequency was the same as previously, i.e., 30 days. Figure 4-22 shows the evolution of mean and variance fields at the end of 60 days, 120 days, and 300 days. We found that they are very similar to the results using an assimilation interval of 30 days, as shown in Figure 4-13. Figures 4-23 through 4-25 show the well performances using the permeability field at the end of 300 days. They are identical with the plots at the end of 300 days in Figures 4-14 through 4-16. This implies we can increase our assimilation interval save the time for data communication.

Note that the increase of assimilation interval is based on our specific case. Theoretically, such an interval should be treated case by case. In practice, information such as water and gas breakthrough, increase or drop of water cut and gas-oil ratio, new operation schedule or pattern, and adding new well, is very important and should be measured and incorporated into model updating. On the other hand, the data in the plateau period of the curves contains less information and contributes less to history matching. Such data can be screened out.

4.4 DEPENDENCE OF MEASUREMENT ERRORS

In the previous literature, measurement errors were assumed to be independent. In reality, however, the measurement errors from the same instrument at the different time should have certain correlations. Here, we try to analyze the impact of this assumption on the EnKF results.

We ran two scenarios with the square root algorithm without measurement perturbation: one applies dependent measurement errors and the other assumes independent measurement errors. Figure 4-26 shows the obtained mean permeability fields at the end of 300 days. Intuitively, we found no significant difference in either mean permeability fields or associated variances.

To further compare these two methods, we calculated the RMS between well production data and the references. Figure 4-27 shows the RMS plots of the performances. For the BHP plots in four wells, the RMS in the dependence of measurement errors drops faster and outperforms independence of measurement errors. Results from the independent measurement errors for the OPR in well P1 converge faster at the beginning and turn worse later. In the end, both methods have similar RMS values. The method of dependent measurement errors has a better performance on the OPR in wells P2 and P4, while the method of independent measurement errors is superior to the dependent case on the OPR in well P3. Observations similar to the OPR are illustrated in RMS plots for the WPR values in wells P1, P2, P3 and P4. Overall, except the OPR and WPR results in well P3, the performances of the dependent measurement errors are better than those from the independent measurement errors.

Hence, the assumption of independent measurement errors has a negative impact on the final matching results while the dependent measurements improve the

performance. Note that we use the dependence of measurement errors from Figure 4-17 to Figure 4-20.

Simulations were performed using 10% noise in the measurements. The results of the simulations were in good agreement with the reference permeability map. However, when we performed the simulations with 20% noise, the results did not agree well with the reference data.

4.5 SUMMARY

It has been shown that after several sequential data assimilations, the updated models from the EnKF, though quite different from the early models are still consistent with the early data and that the material balance errors are small in comparison to the uncertainty in the actual values (Gu and Oliver, 2006). In this sense, the EnKF in the sequential manner, which only utilizes the latest data while honoring all the previous information, is encouraging for reservoir history matching.

Chapter 4 has concentrated on the demonstration and studies of a waterflooding synthetic case by using the EnKF. With limited measurements, the EnKF is promising in its fast convergence and reasonable matching results. The results from four assimilation methods are discussed. The influences of the assimilation interval and the assumption of independent measurement errors are also addressed.

With the same initial realizations and same measurement data, we studied four methods extensively: direct inverse, standard EnKF assimilation, and square root algorithms with and without measurement perturbations. The results show that direct inverse method, square root algorithms with and without measurement perturbation methods are better than the standard EnKF assimilation method. For a small number of observations, direct inverse is convenient. For the consideration of algorithm generality, we recommend the square root approaches.

It is desirable to investigate when we should assimilate data and implement history matching. In order to explore the impact of assimilation interval on the results, we doubled the assimilation interval time in the study and still obtained the same level of accuracy for the final results. Therefore, the data in the stable period can be sampled in a large time interval. On the other hand, it is always helpful to analyze and validate the measurements to capture the useful data such as water breakthrough, dramatic change of water cut and gas-oil ratio, and new well pattern replacement.

Measurement errors in the literature were assumed to be independent. In this chapter, we found that the consideration of correlation among measurement data improved the matching performance. In practice, it is reasonable to take the measurements to be dependent, especially for the same type of data in the same well or even from different wells.

This chapter focused on the issues on the assimilation step in the EnKF implementation. The initial realization sampling and improved strategy will be discussed in the next chapter.

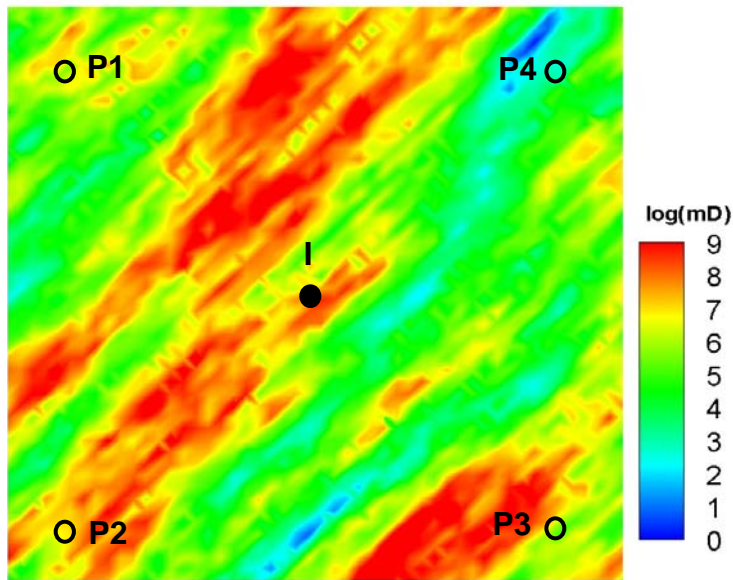


Figure 4-1: Reference permeability field after the lognormal transformation for a two-dimensional waterflooding synthetic case. The legend unit is milidarcy with the range from 0 (dark blue) to 9 (dark red). The orientation of the field is 45 degree with clear high and low permeability channel zones.

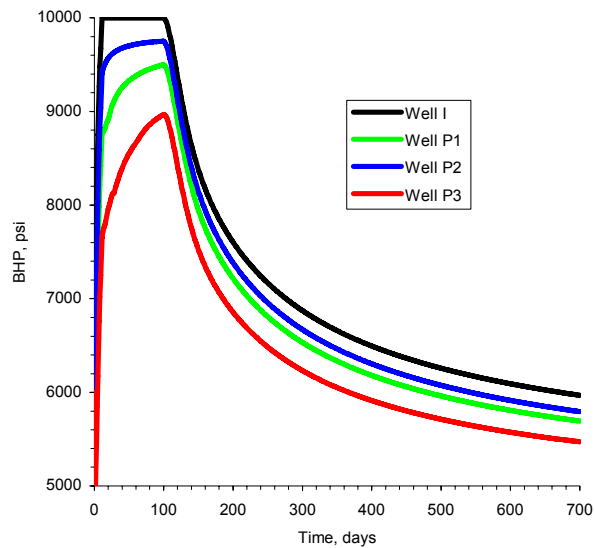


Figure 4-2: Bottomhole pressure for one injector and three producers in the reference field.

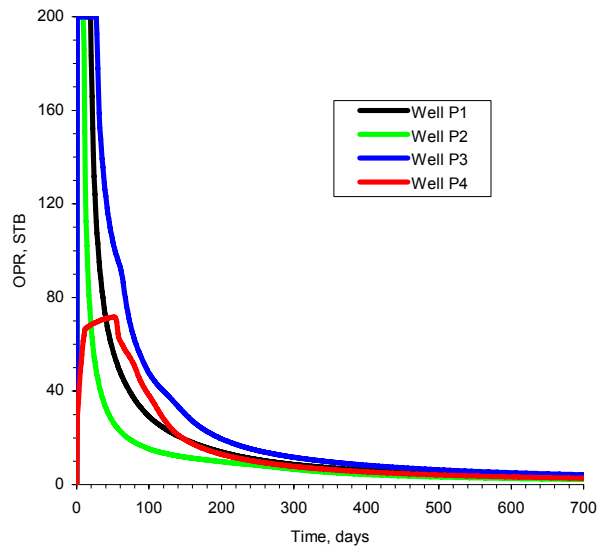


Figure 4-3: Oil production rate for four producers in the reference field.

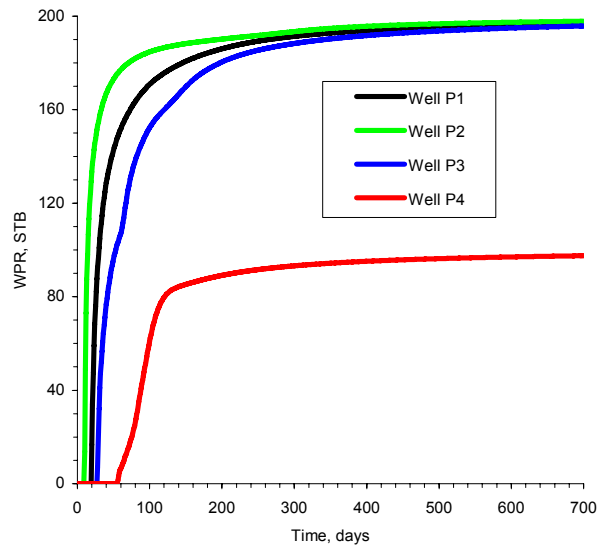


Figure 4-4: Water production rate for four producers in the reference field.

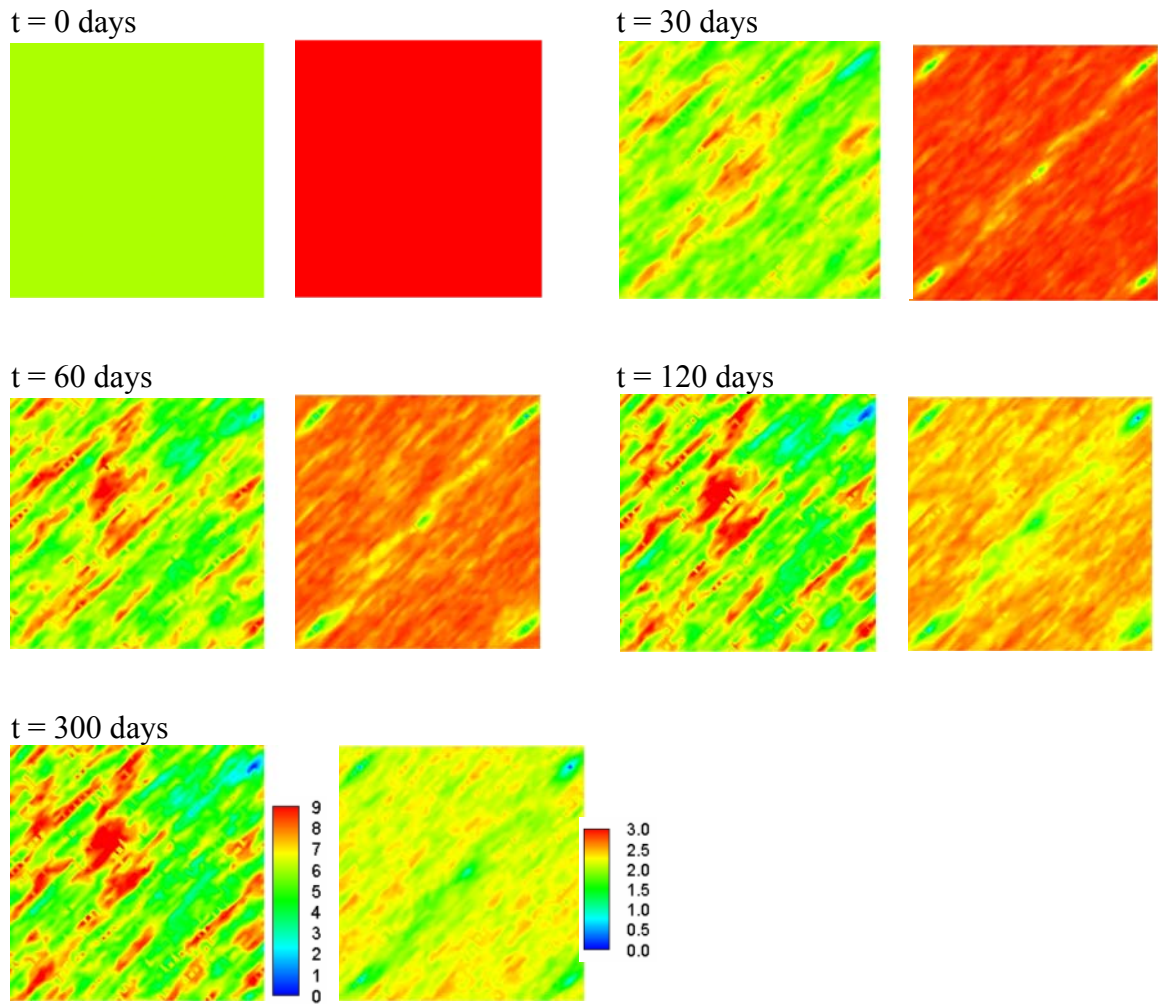
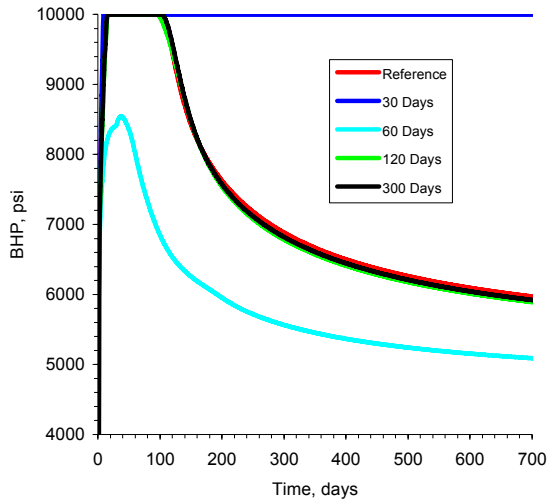
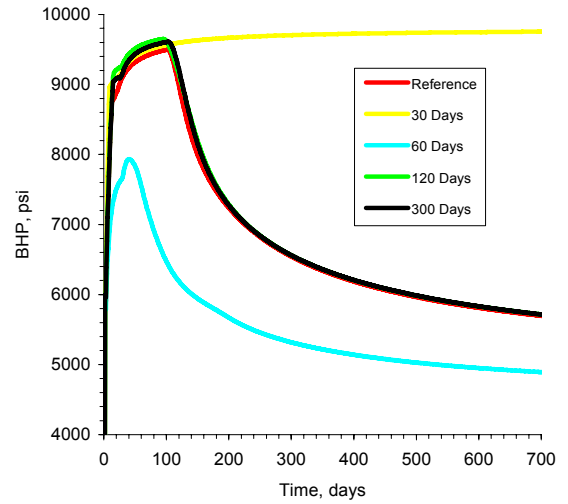


Figure 4-5: The evolution of mean (left hand side) and variance (right hand side) fields updated by the EnKF with direct inverse calculation method at selected times.

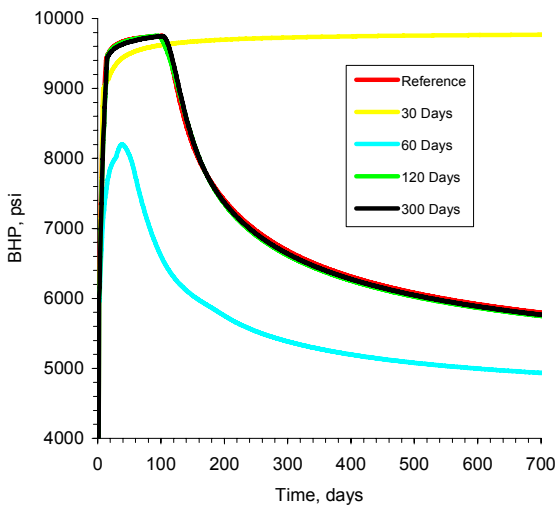
(a) BHP at Injector



(b) BHP at Well P1



(c) BHP at Well P2



(d) BHP at Well P3

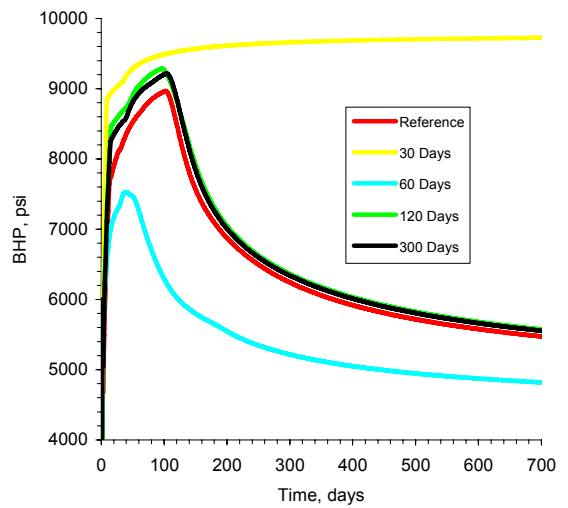
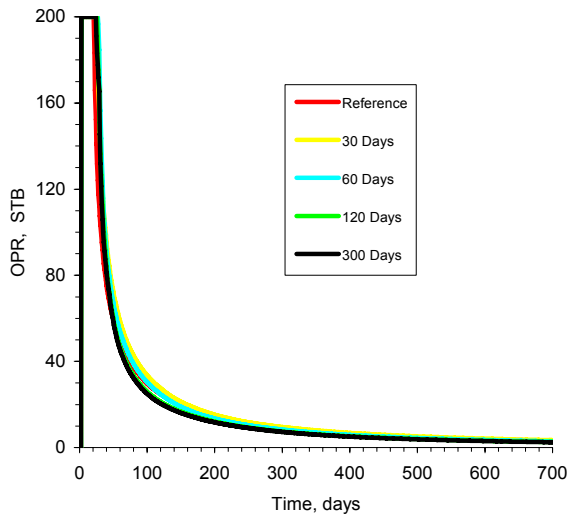
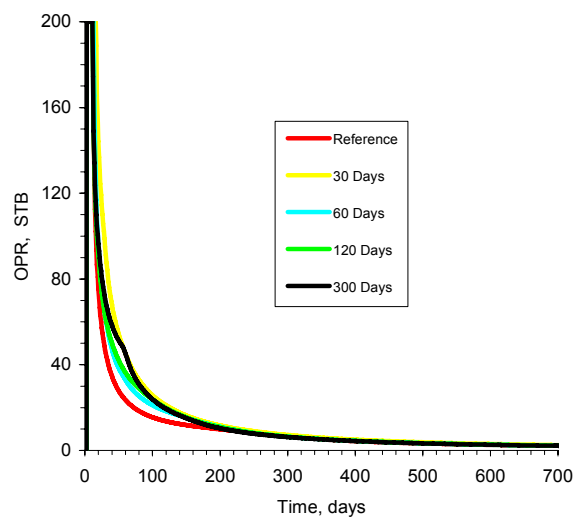


Figure 4-6: Well bottomhole pressures simulated from the mean permeability model updated by the direct inverse method at different times.

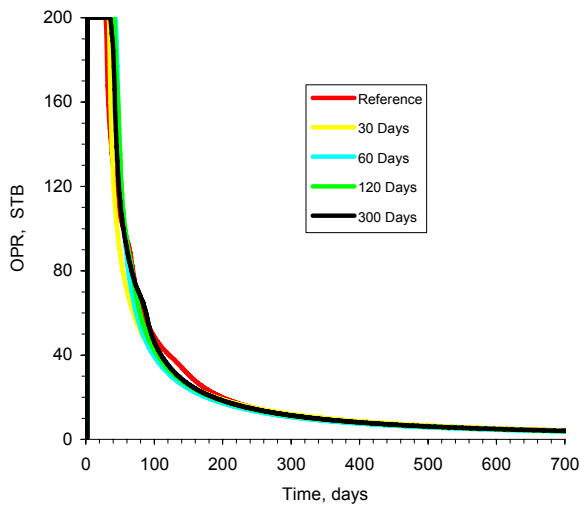
(a) OPR at Well P1



(b) OPR at Well P2



(c) OPR at Well P3



(d) OPR at Well P4

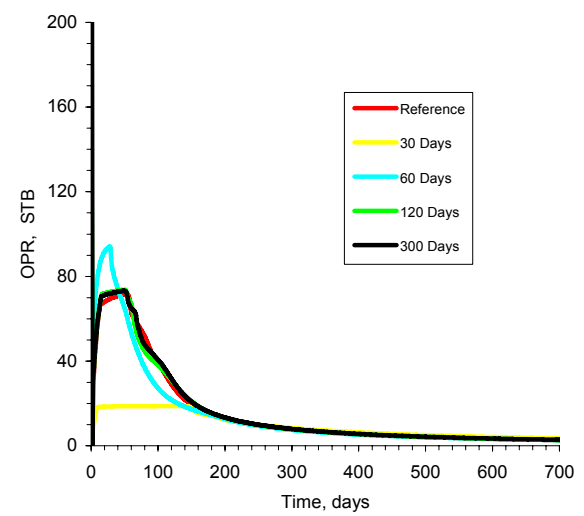
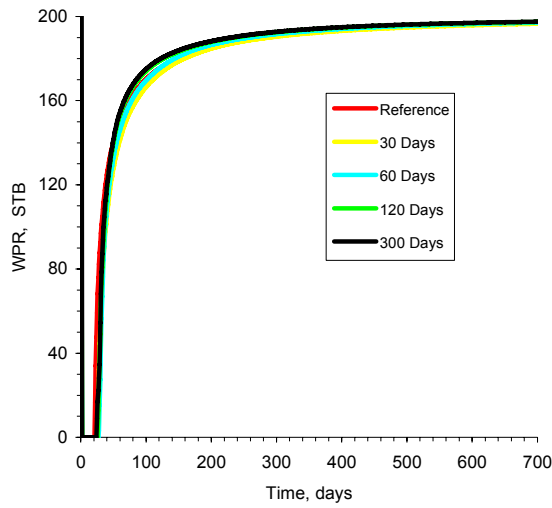
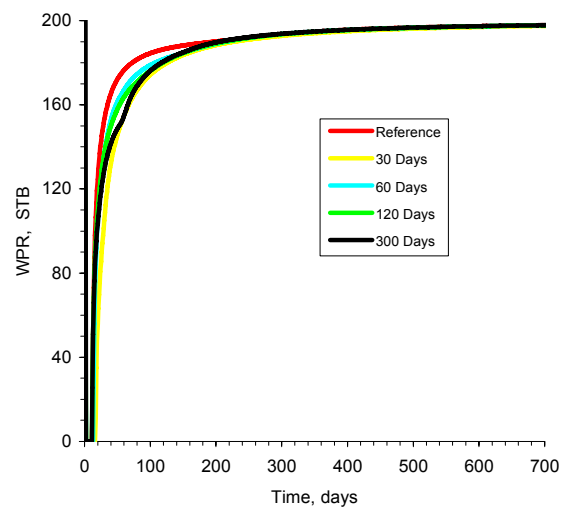


Figure 4-7: Well oil production rates simulated from the mean permeability model updated by the direct inverse method at different times.

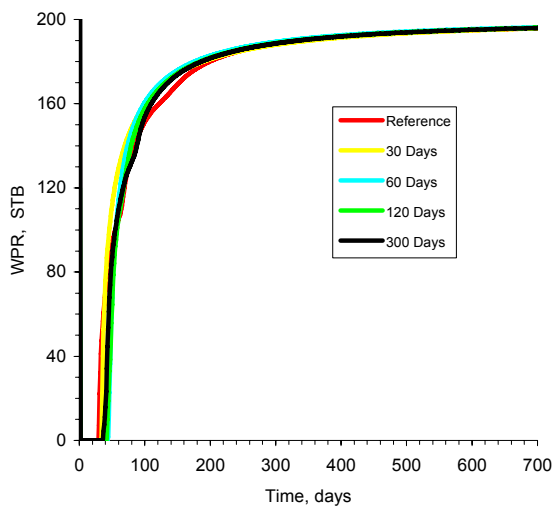
(a) WPR at Well P1



(b) WPR at Well P2



(c) WPR at Well P3



(d) WPR at Well P4

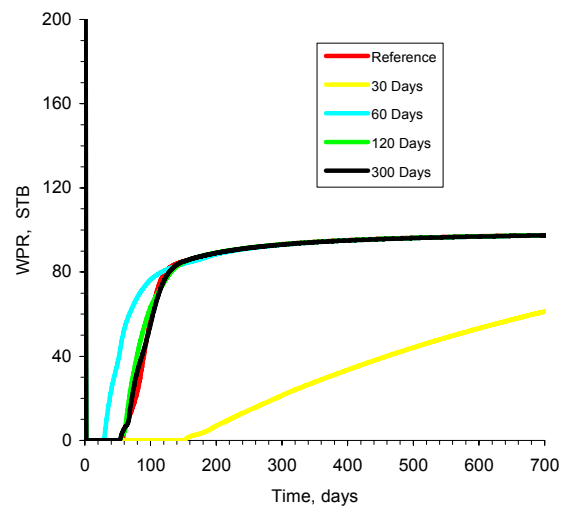


Figure 4-8: Well water production rates simulated from the mean permeability model updated by the direct inverse method at different times.

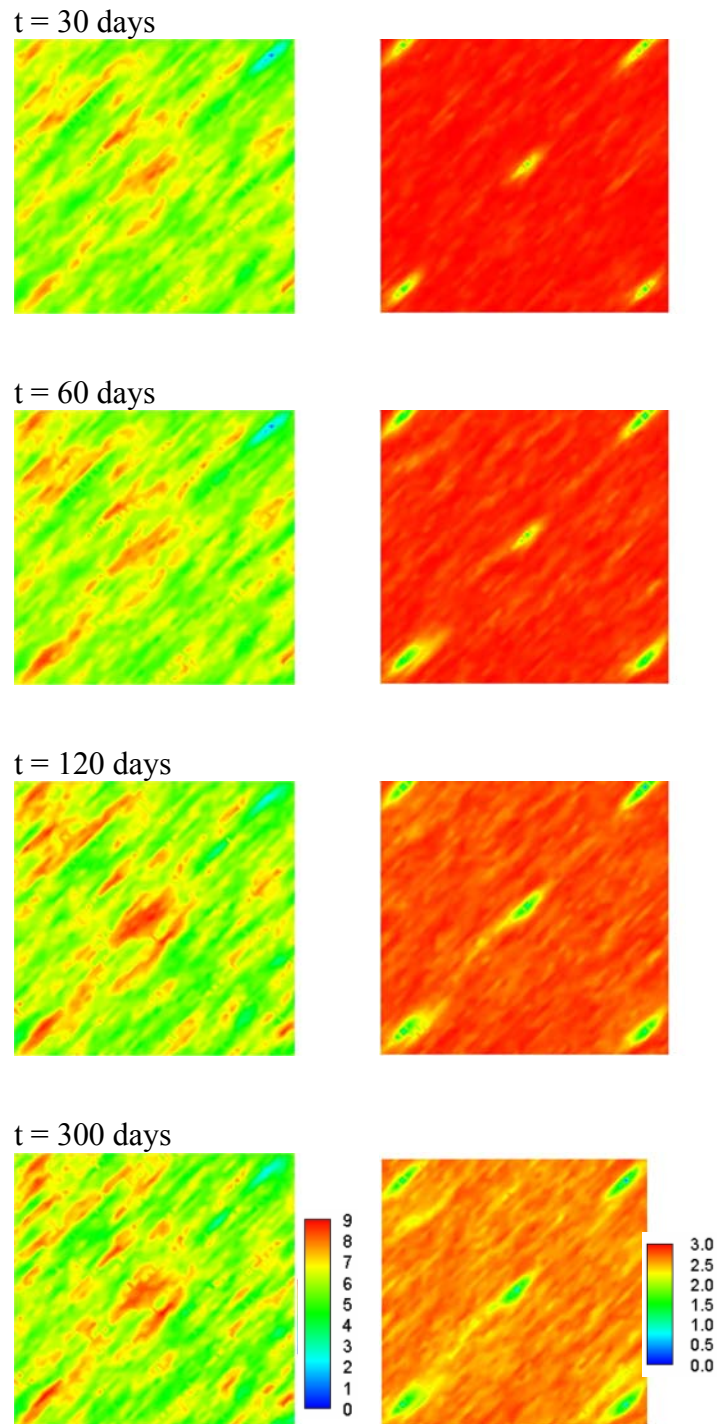
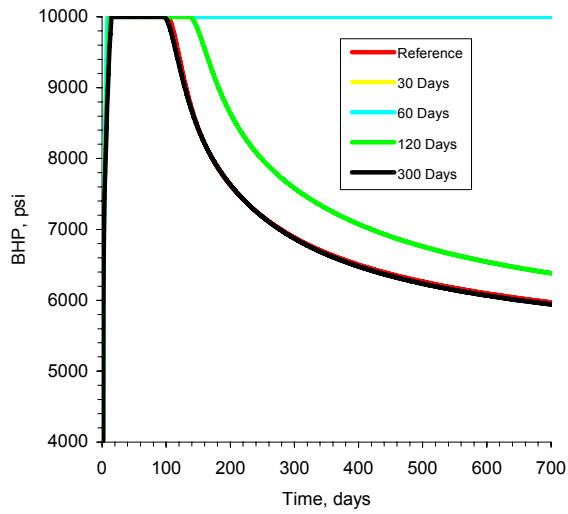
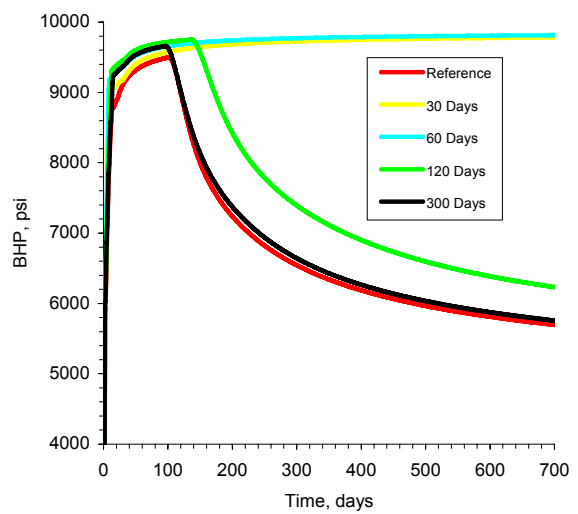


Figure 4-9: The evolution of mean (left hand side) and variance (right hand side) fields updated by the EnKF with standard assimilation method at selected times.

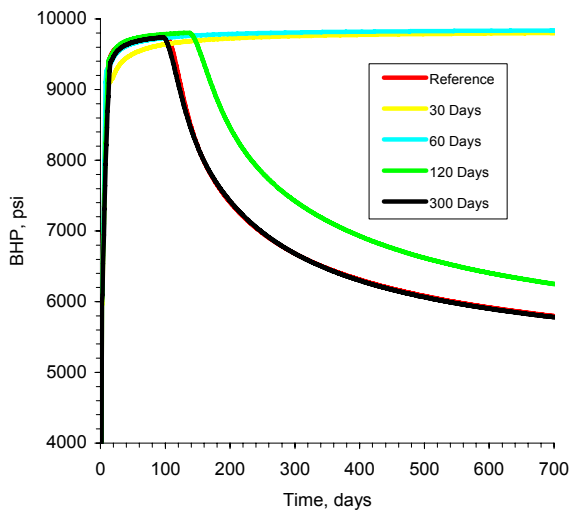
(a) BHP at Injector



(b) BHP at Well P1



(c) BHP at Well P2



(d) BHP at Well P3

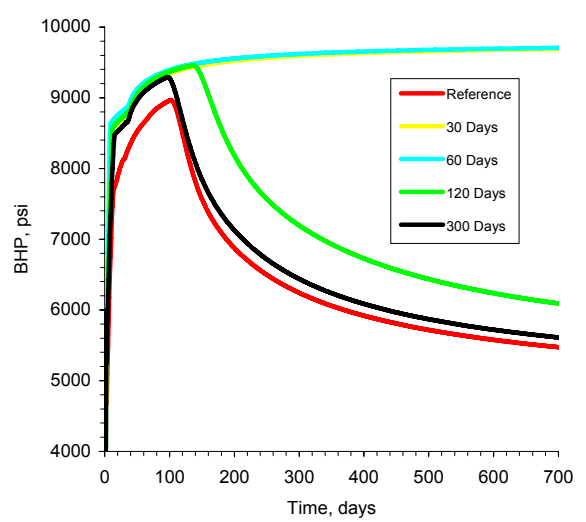
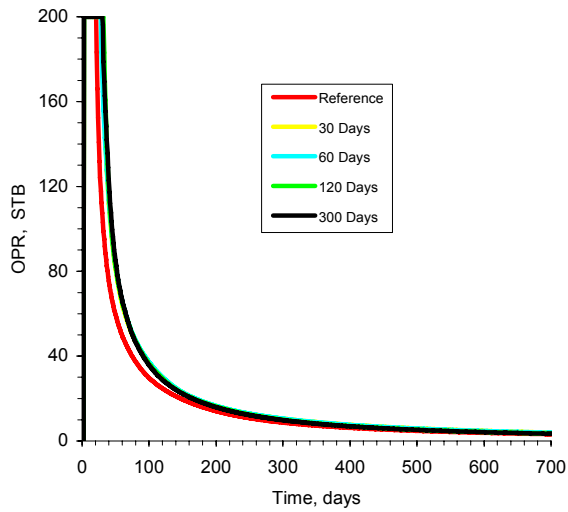
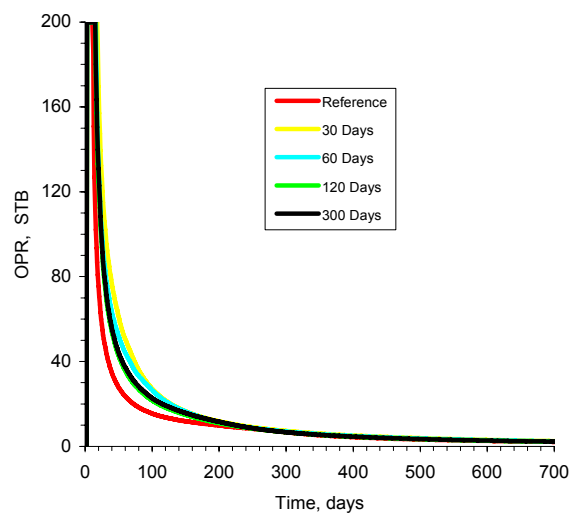


Figure 4-10: Well bottomhole pressures simulated from the mean permeability model updated by standard EnKF assimilation method at different times.

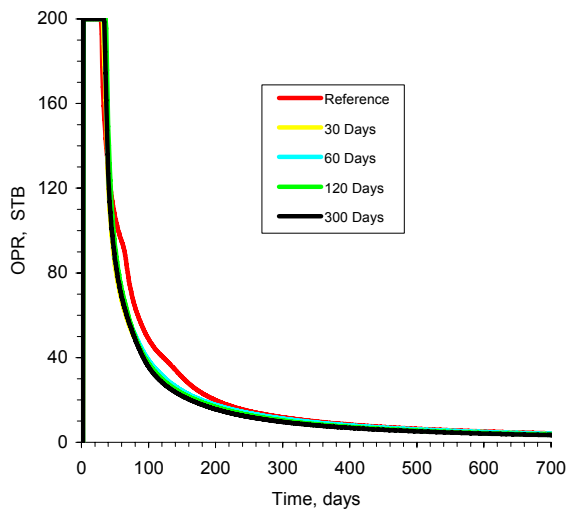
(a) OPR at Well P1



(b) OPR at Well P2



(c) OPR at Well P3



(d) OPR at Well P4

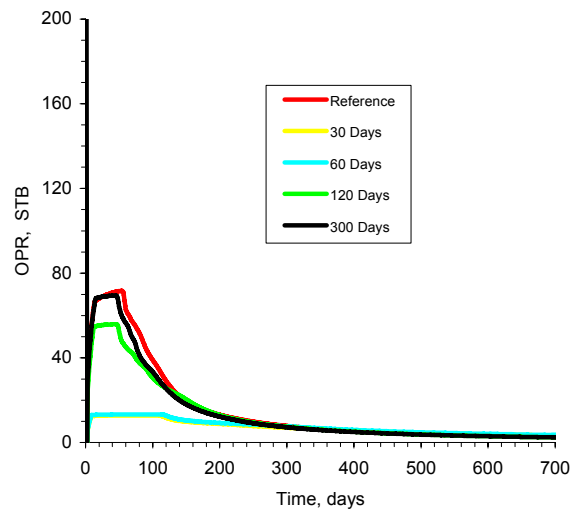
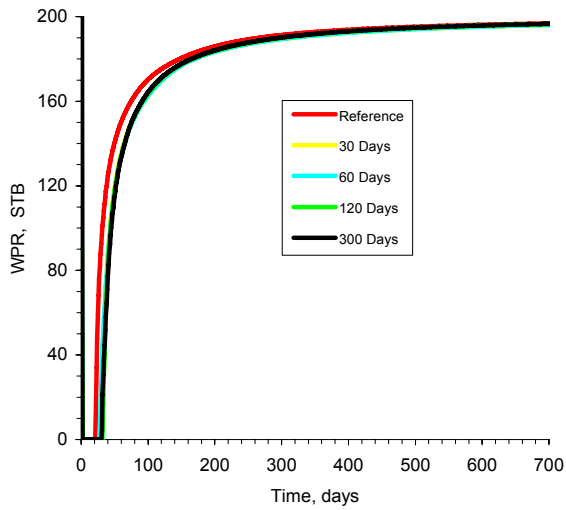
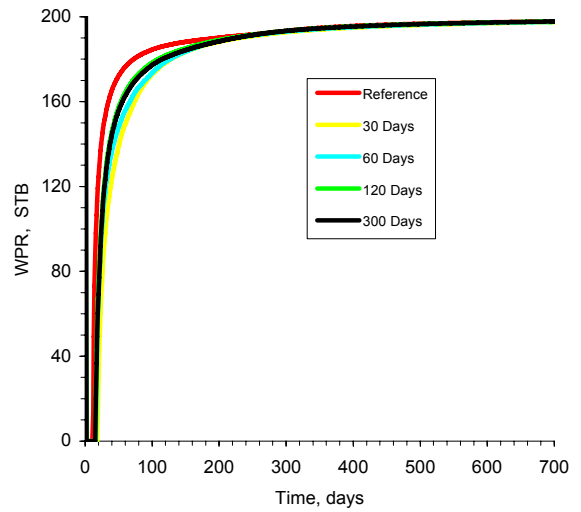


Figure 4-11: Well oil production rates simulated from the mean permeability model updated by standard EnKF assimilation method at different times.

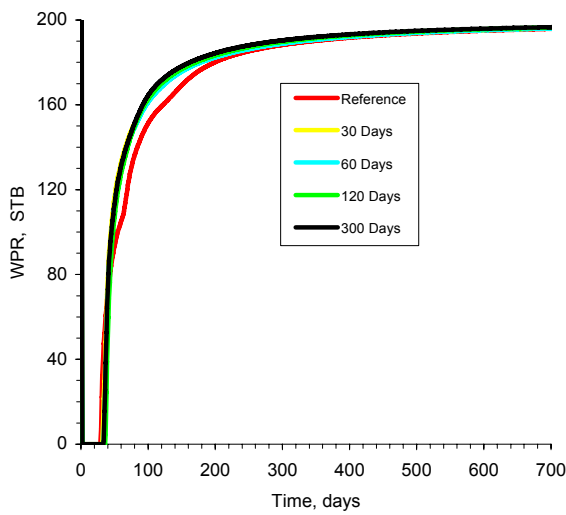
(a) WPR at Well P1



(b) WPR at Well P2



(c) WPR at Well P3



(d) WPR at Well P4

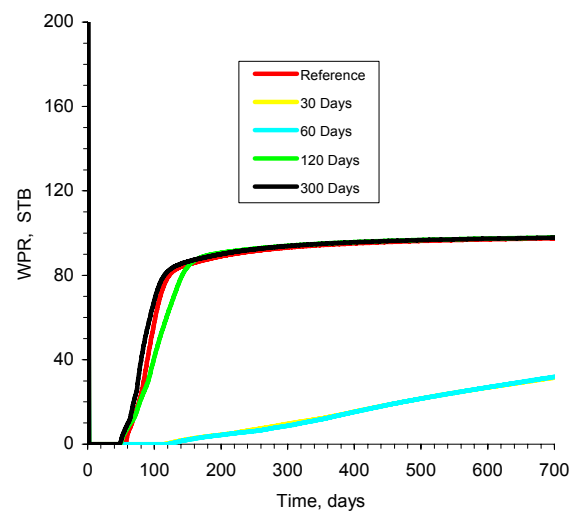


Figure 4-12: Well water production rates simulated from the mean permeability model updated by standard EnKF assimilation method at different times.

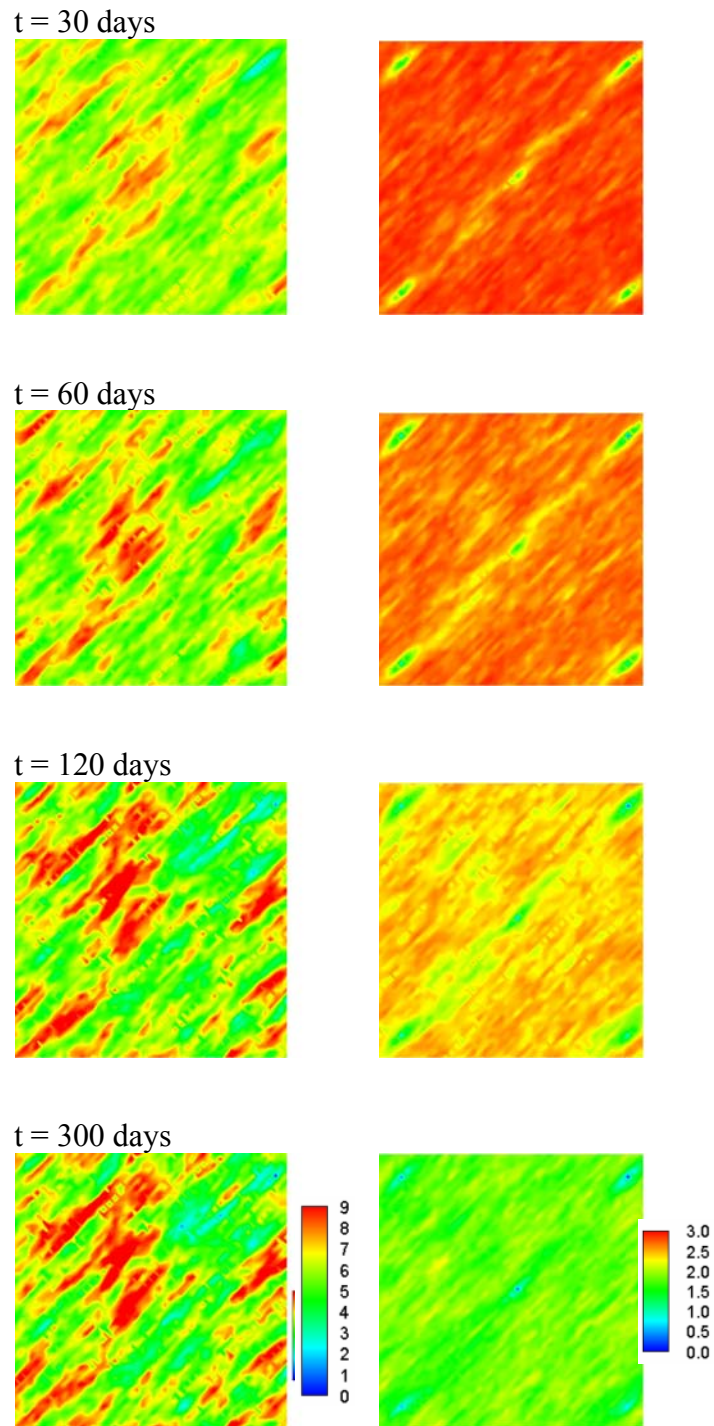


Figure 4-13: The evolution of mean (left hand side) and variance (right hand side) fields updated by using square root algorithm with measurement perturbations at selected times.

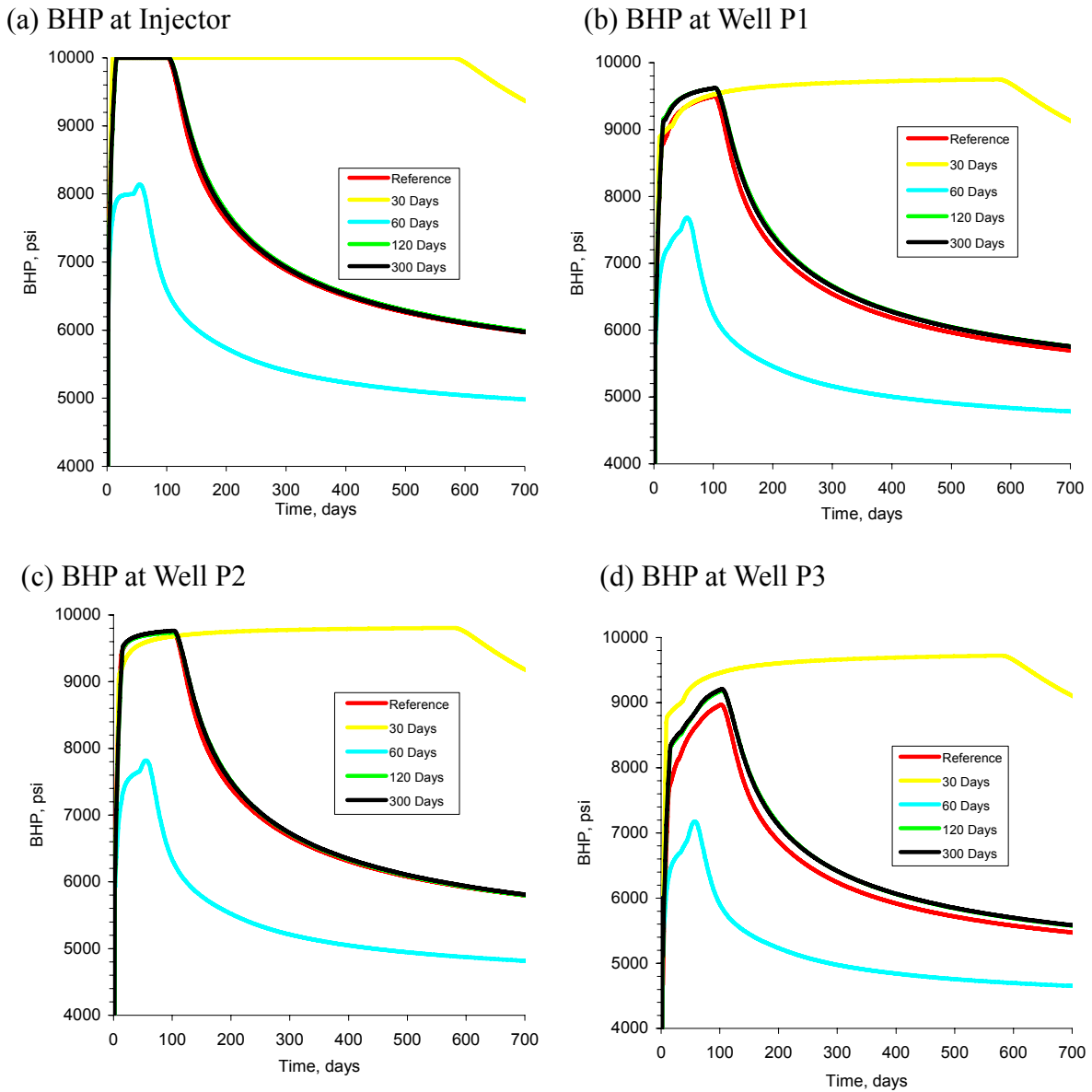
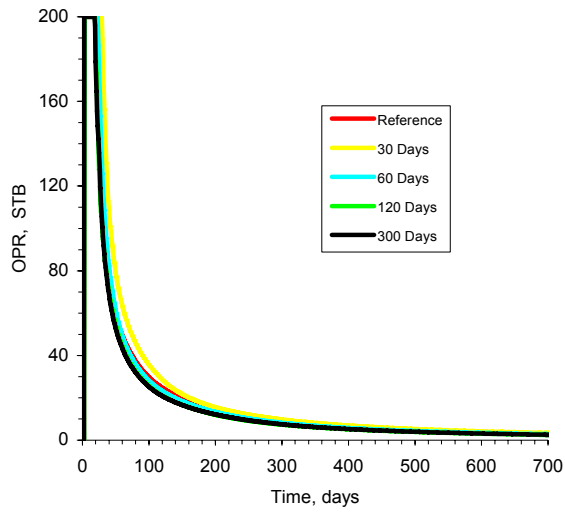
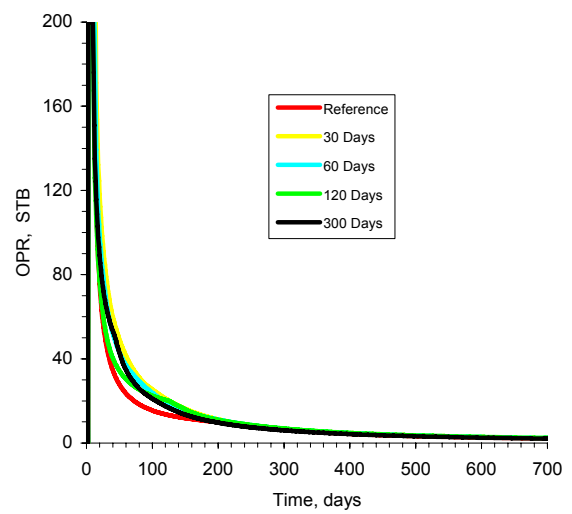


Figure 4-14: Well bottomhole pressures simulated from the mean permeability model updated by using square root algorithm with measurement perturbations at selected times.

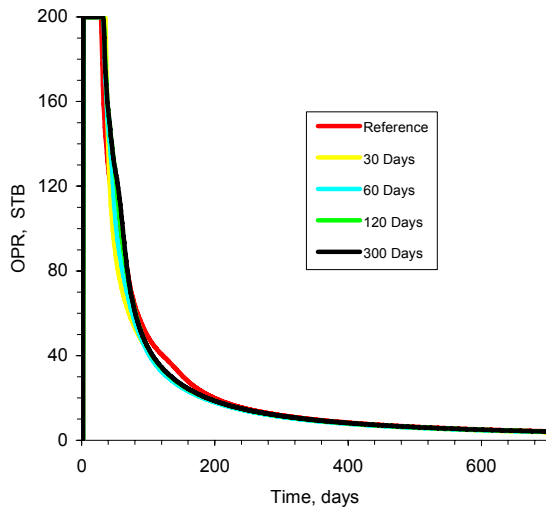
(a) OPR at Well P1



(b) OPR at Well P2



(c) OPR at Well P3



(d) OPR at Well P4

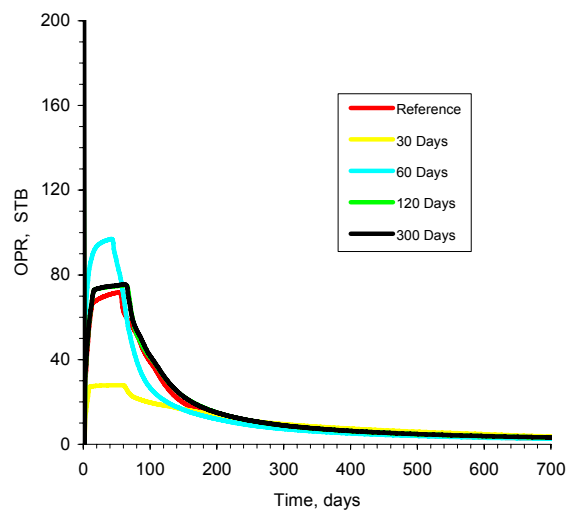
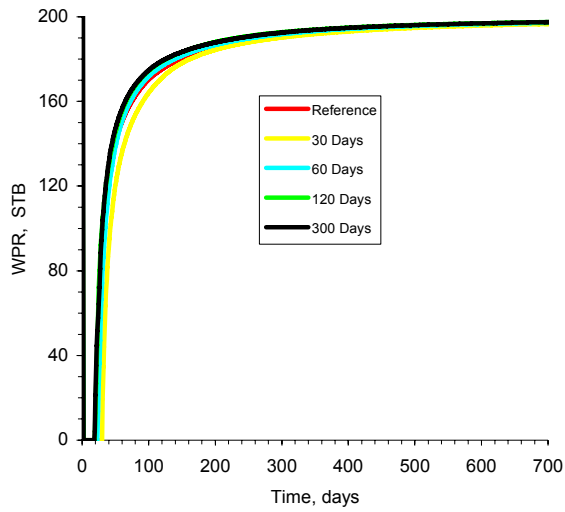
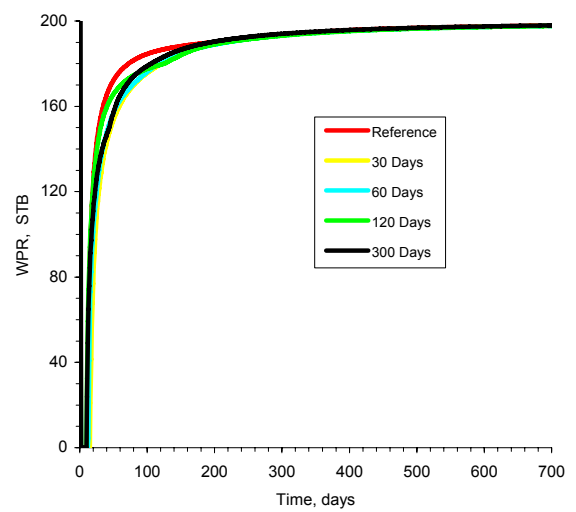


Figure 4-15: Well oil production rates simulated from the mean permeability model updated by using square root algorithm with measurement perturbations at selected times.

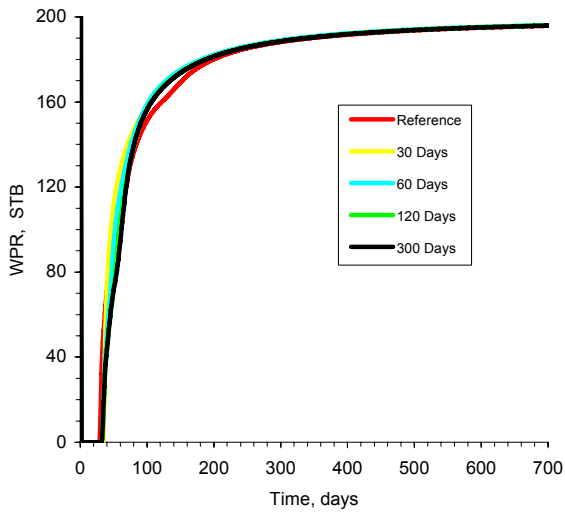
(a) WPR at Well P1



(b) WPR at Well P2



(c) WPR at Well P3



(d) WPR at Well P4

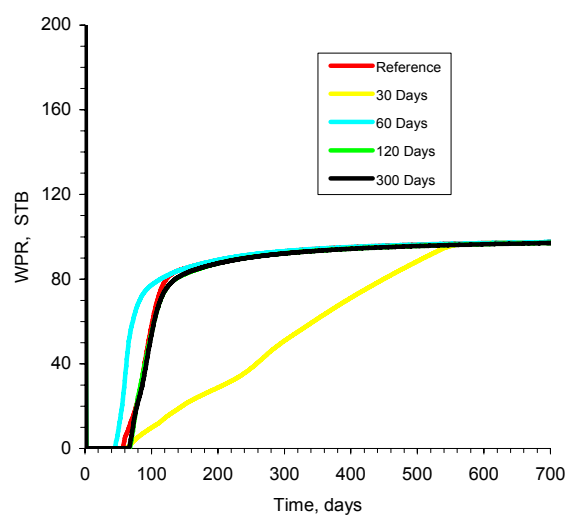


Figure 4-16: Well water production rates simulated from the mean permeability model updated by using square root algorithm with measurement perturbations at selected times.

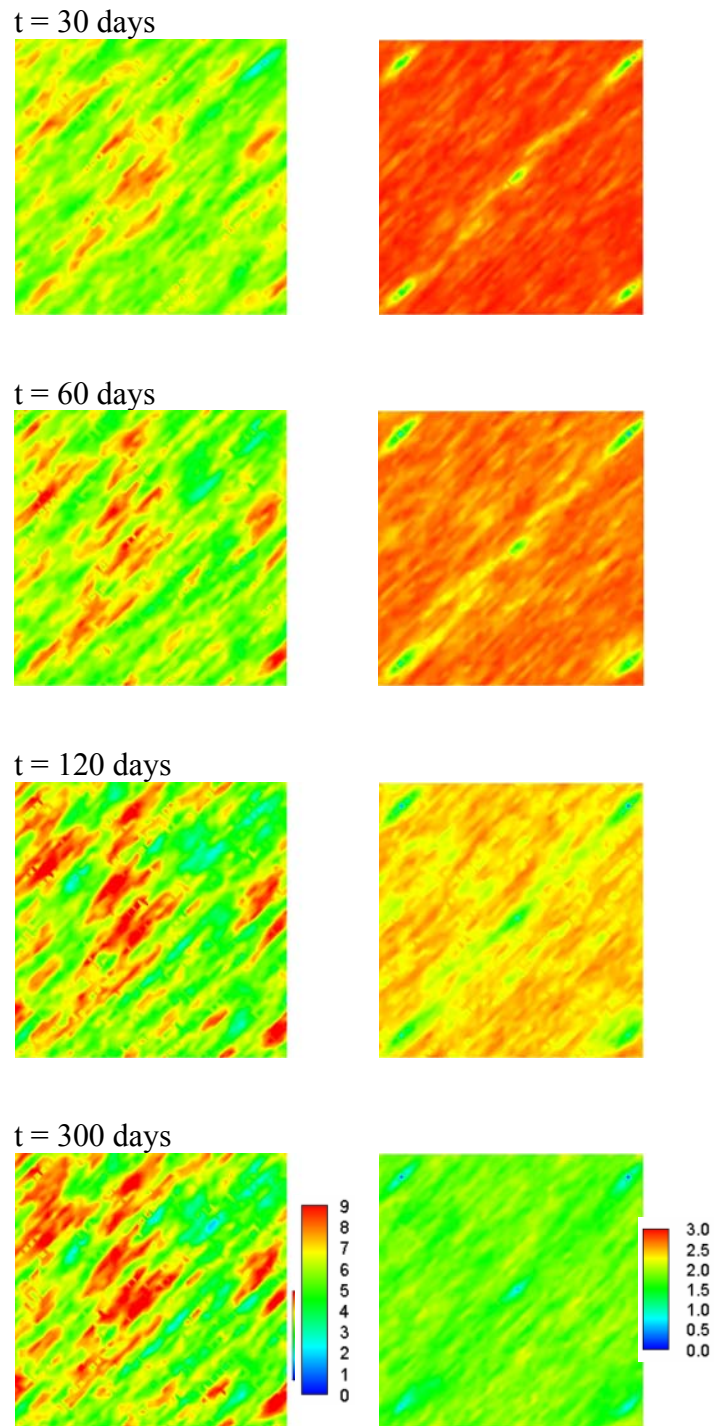
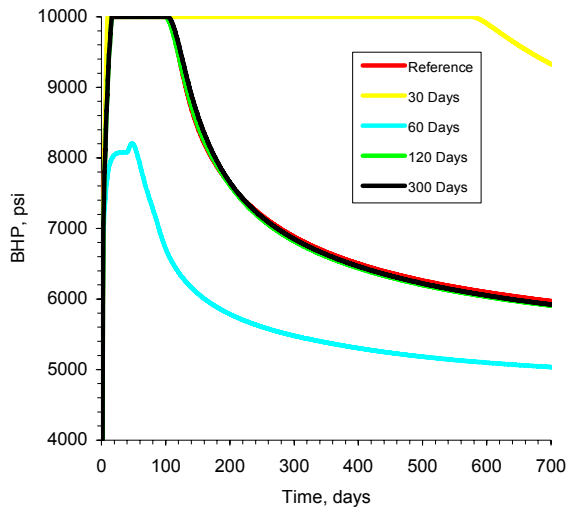
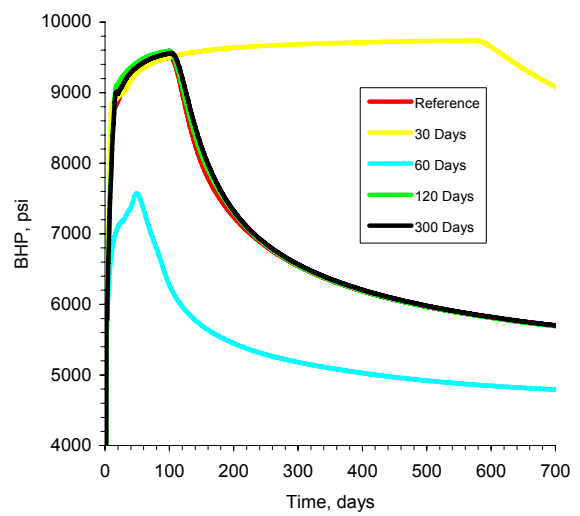


Figure 4-17: The evolution of mean (left hand side) and variance (right hand side) fields updated by using square root algorithm without measurement perturbations at selected times.

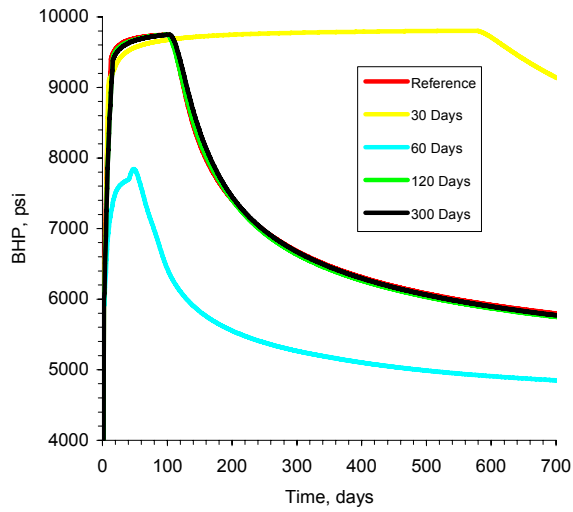
(a) BHP at Injector



(b) BHP at Well P1



(c) BHP at Well P2



(d) BHP at Well P3

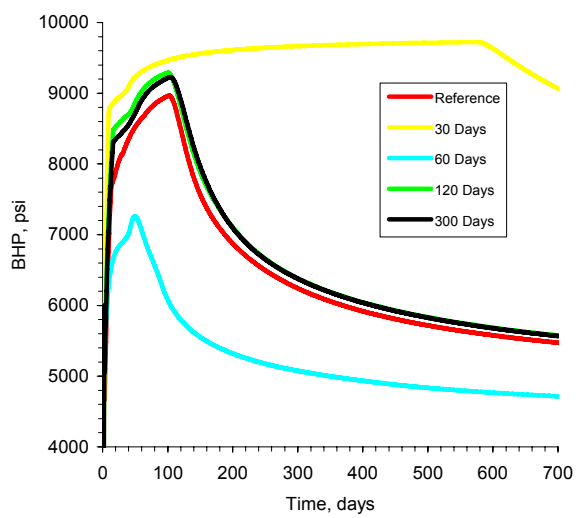
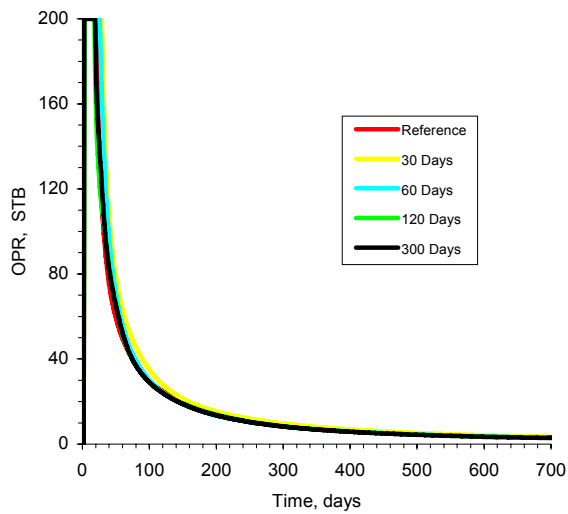
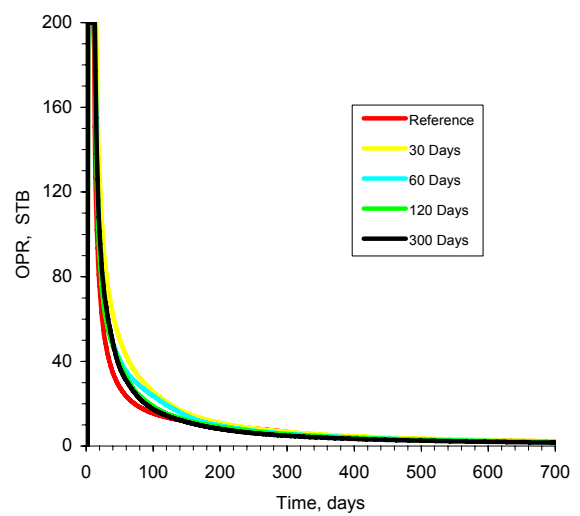


Figure 4-18: Well bottomhole pressures simulated from the mean permeability model updated by using square root algorithm without measurement perturbations at selected times.

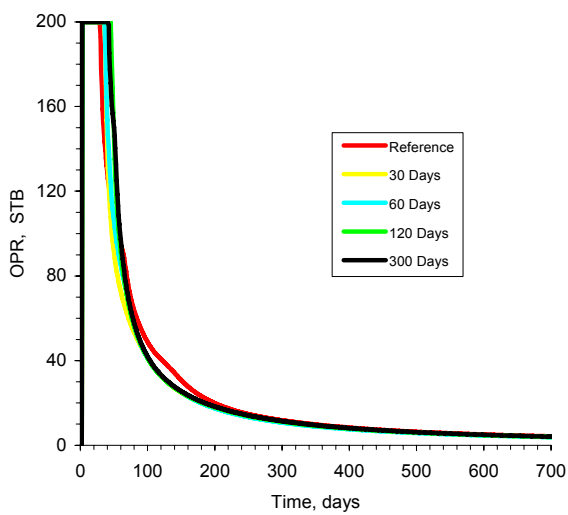
(a) OPR at Well P1



(b) OPR at Well P2



(c) OPR at Well P3



(d) OPR at Well P4

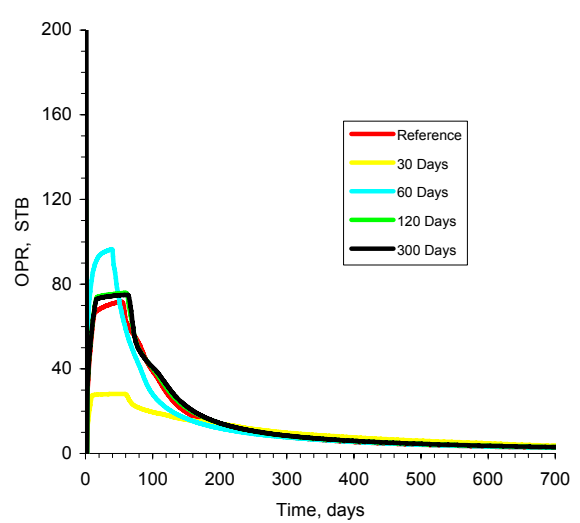
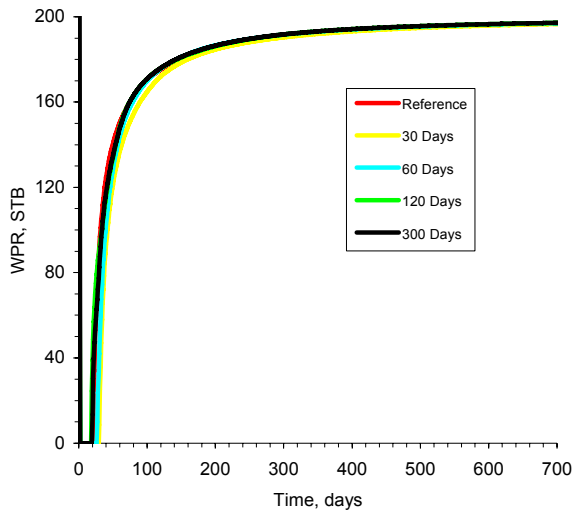
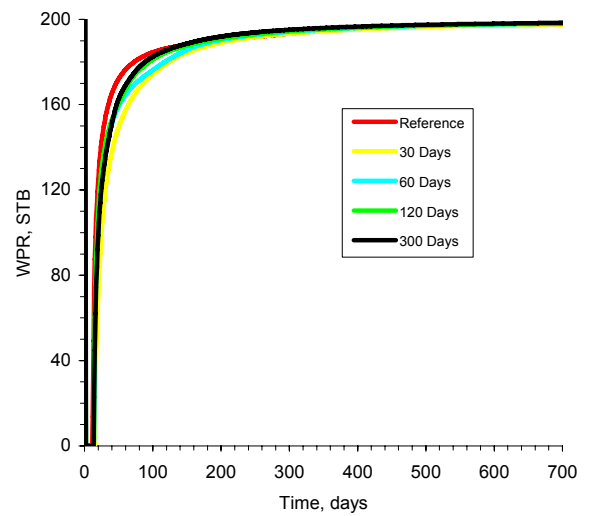


Figure 4-19: Well oil production rates simulated from the mean permeability model updated by using square root algorithm without measurement perturbations at selected times.

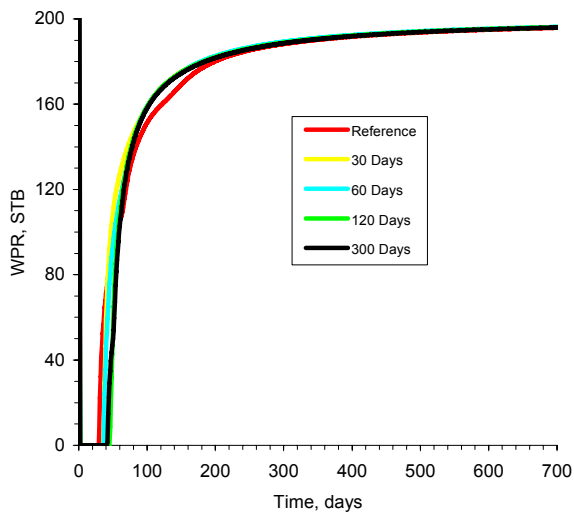
(a) WPR at Well P1



(b) WPR at Well P2



(c) WPR at Well P3



(d) WPR at Well P4

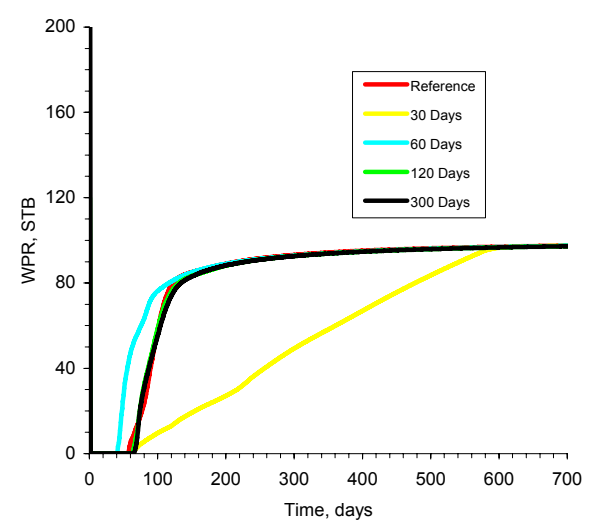
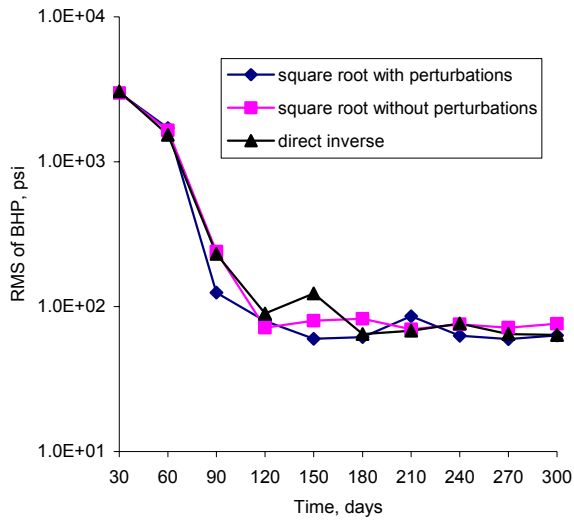
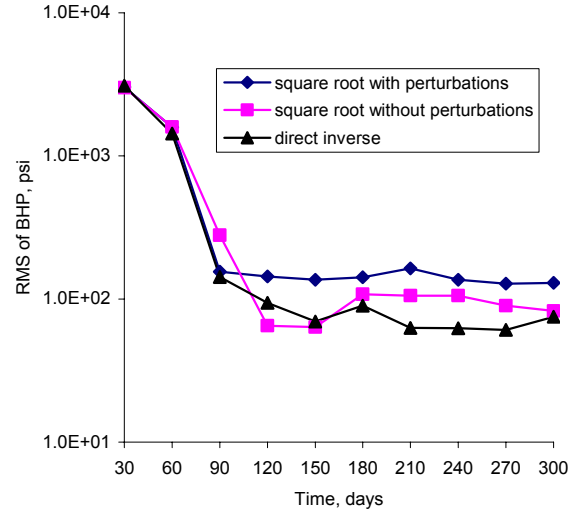


Figure 4-20: Well water production rates simulated from the mean permeability model updated by using square root algorithm without measurement perturbations method at selected times.

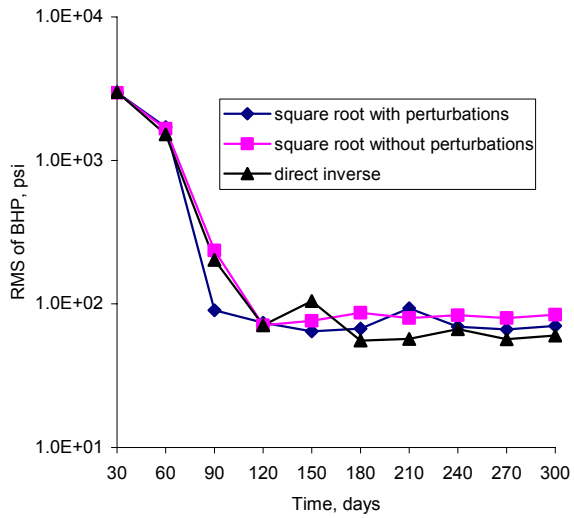
(a) BHP at Well I



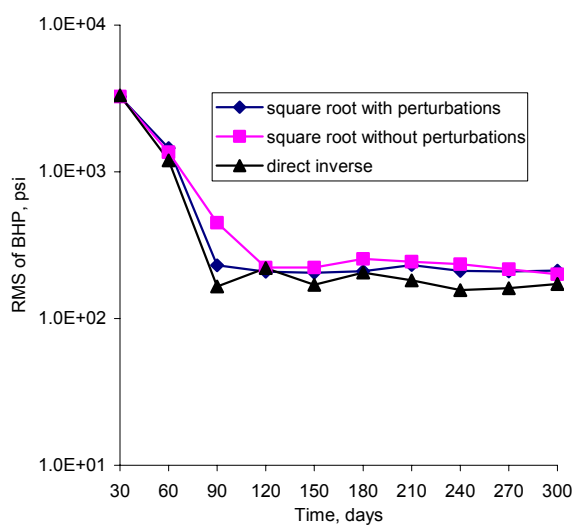
(b) BHP at Well P1



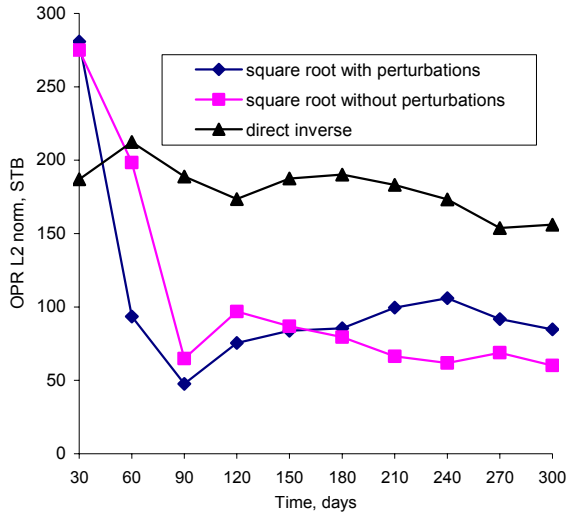
(c) BHP at Well P2



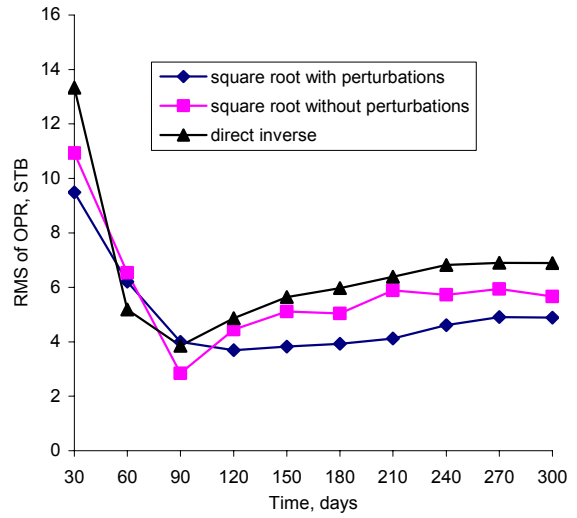
(d) BHP at Well P3



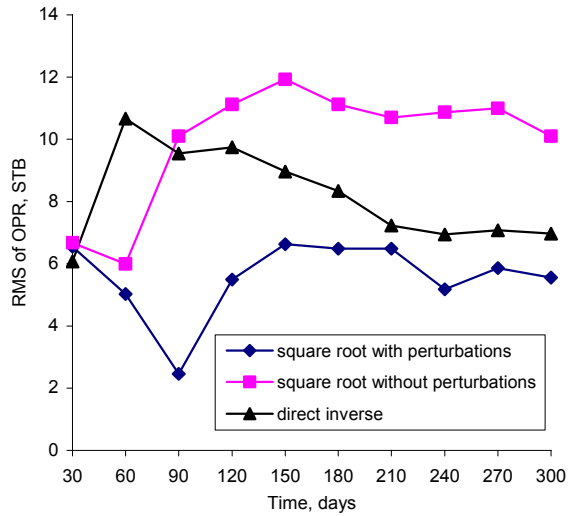
(e) OPR at Well P1



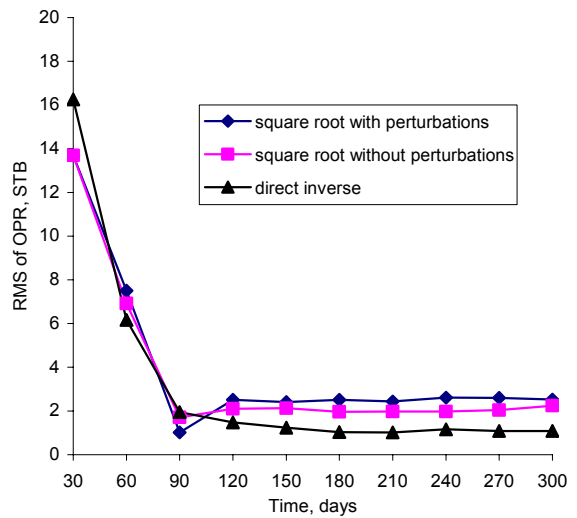
(f) OPR at Well P2



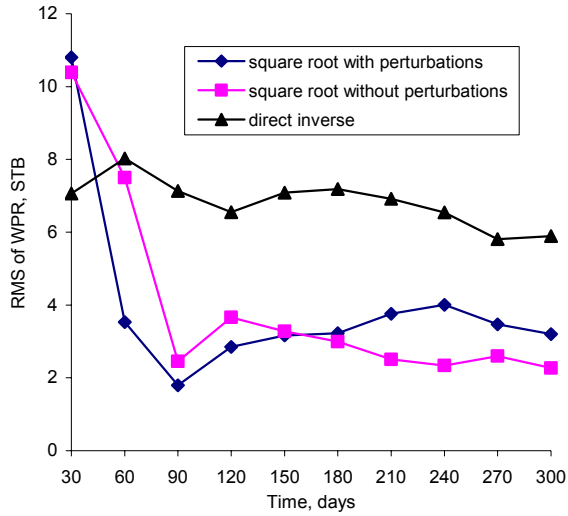
(g) OPR at Well P3



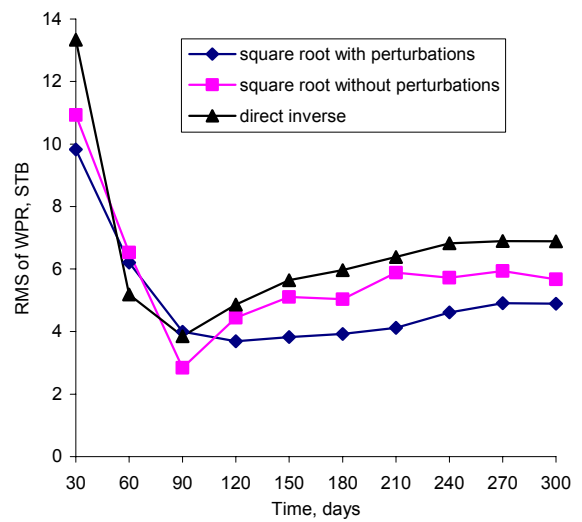
(h) OPR at Well P4



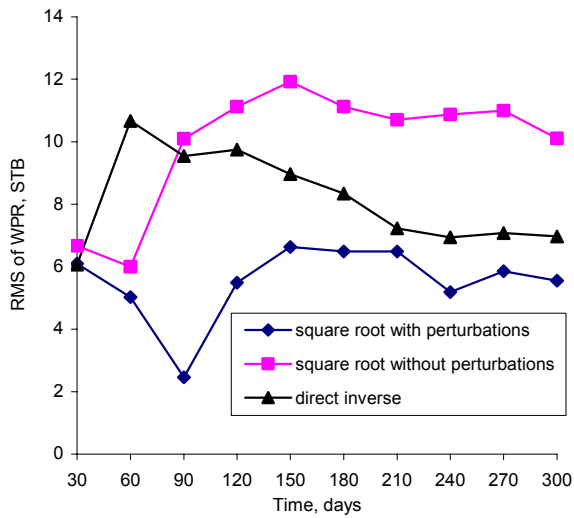
(i) WPR at Well P1



(j) WPR at Well P2



(k) WPR at Well P3



(l) WPR at Well P4

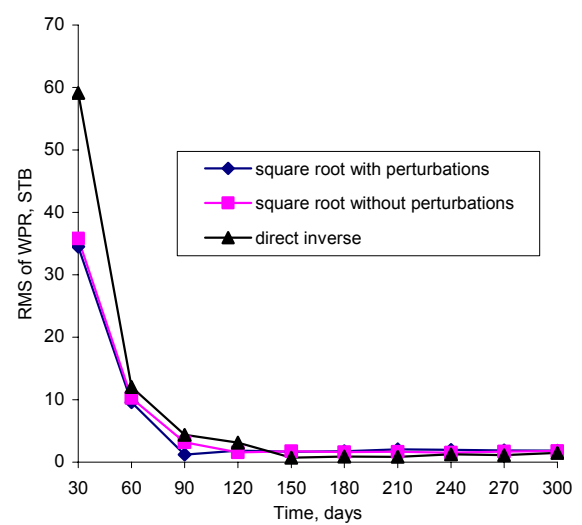


Figure 4-21: RMS profiles of BHP, OPR, and WPR for direct inverse, square root with measurement perturbations, and square root without measurement perturbations.

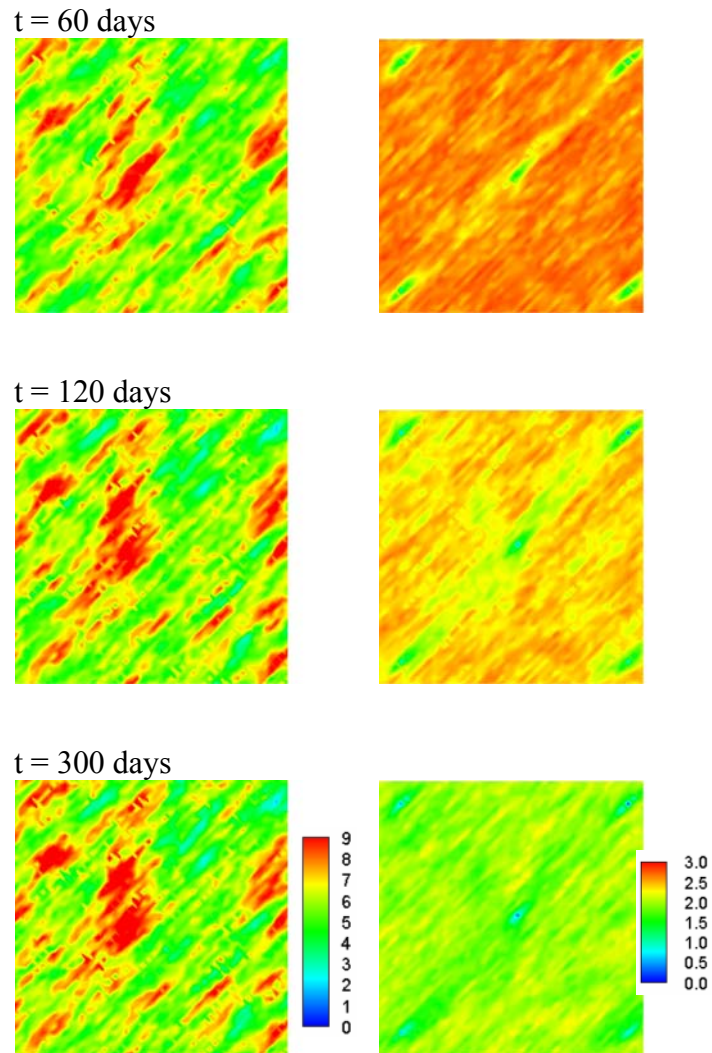
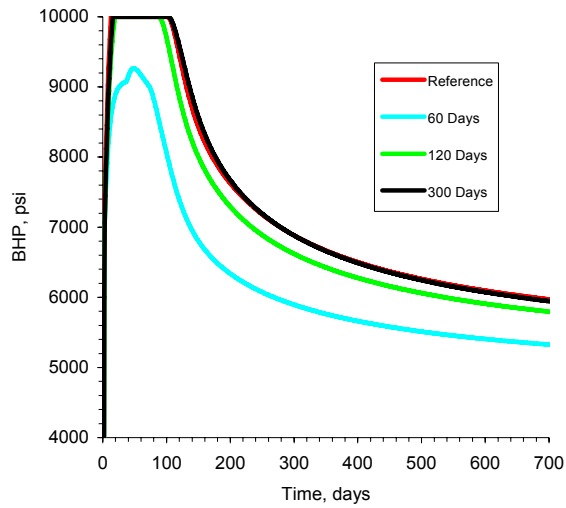
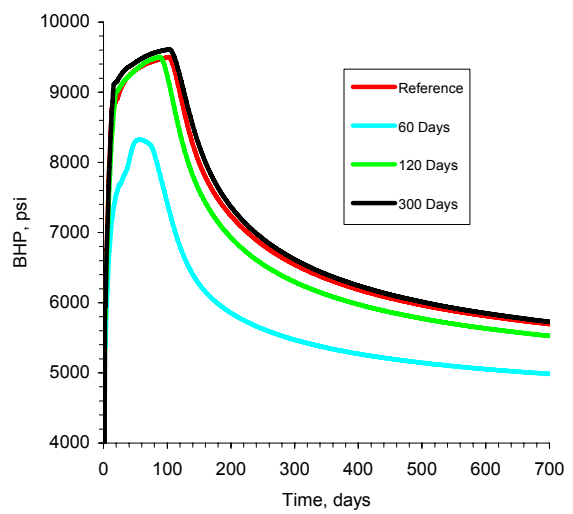


Figure 4-22: The evolution of mean (left hand side) and variance (right hand side) fields updated by using square root algorithm with measurement perturbation method at selected times. The assimilation interval is taken as 60 days.

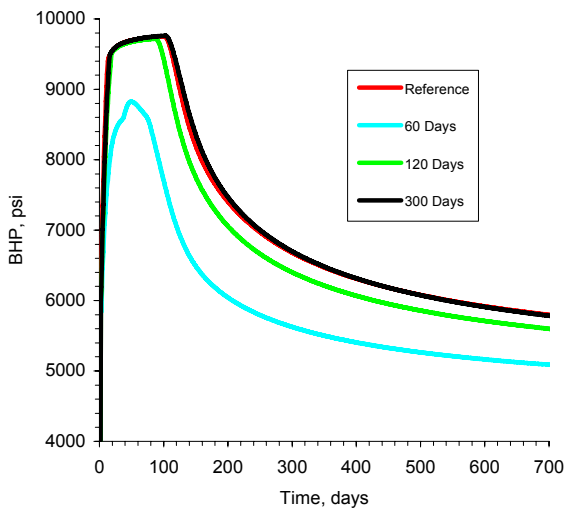
(a) BHP at Injector



(b) BHP at Well P1



(c) BHP at Well P2



(d) BHP at Well P3

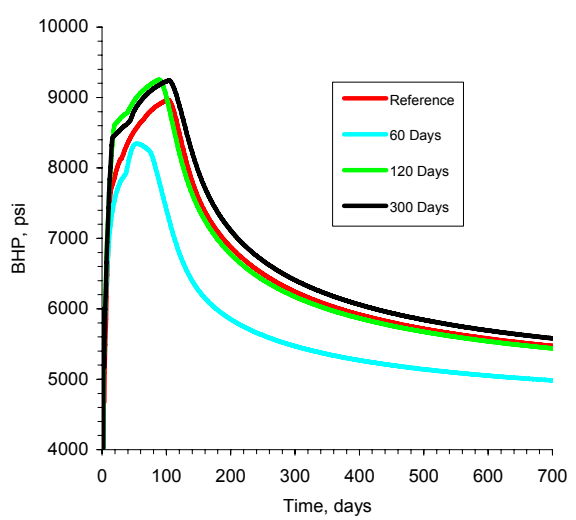
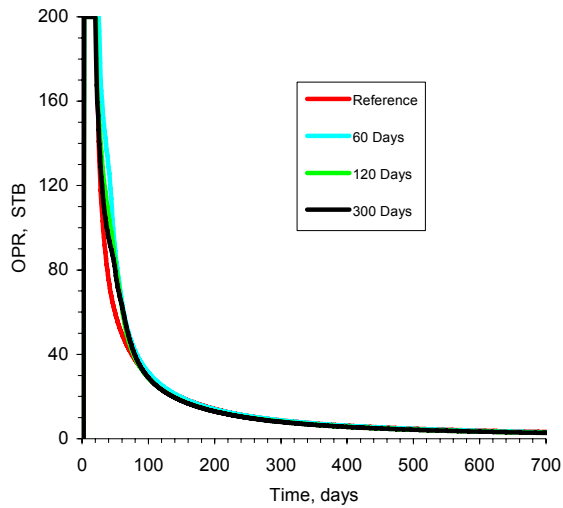
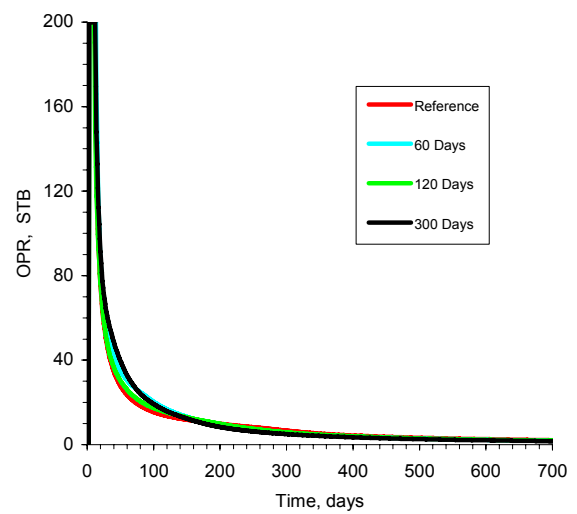


Figure 4-23: Well bottomhole pressures simulated from the mean permeability model updated by using square root algorithm with measurement perturbation method at selected times. The assimilation interval is taken as 60 days.

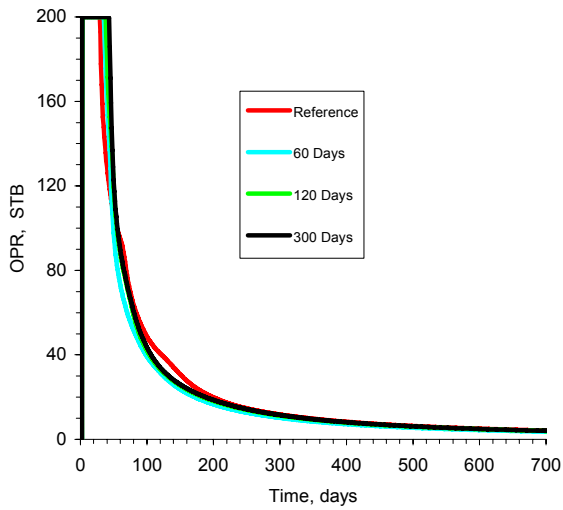
(a) OPR at Well P1



(b) OPR at Well P2



(c) OPR at Well P3



(d) OPR at Well P4

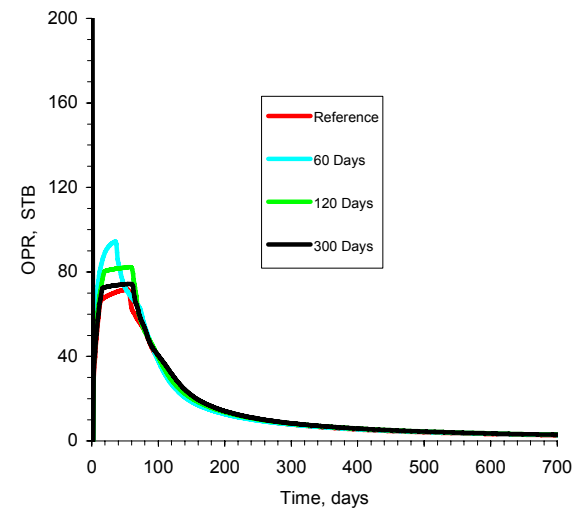
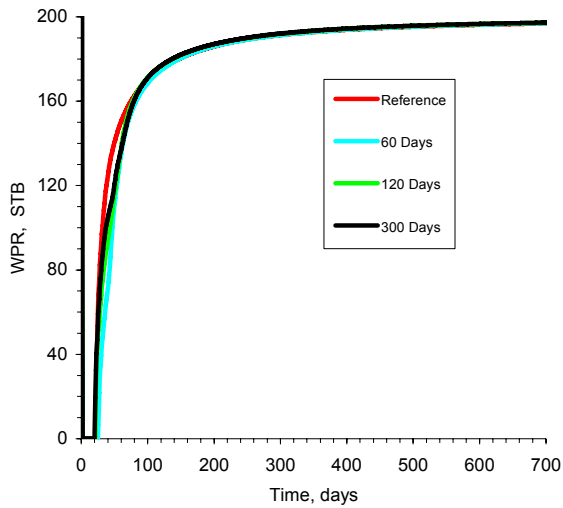
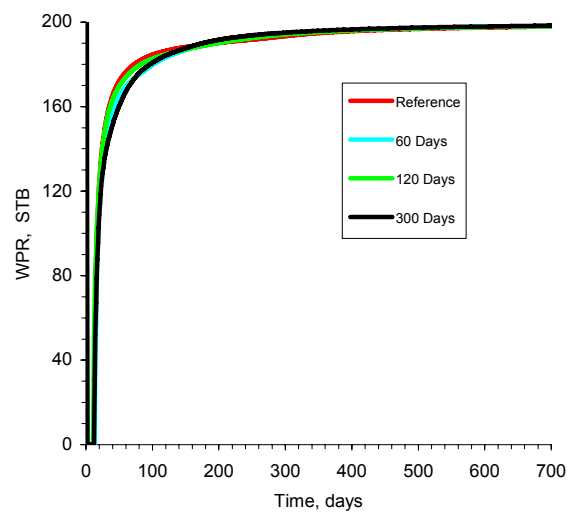


Figure 4-24: Well oil production rates simulated from the mean permeability model updated by using square root algorithm with measurement perturbation method at selected times. The assimilation interval is taken as 60 days.

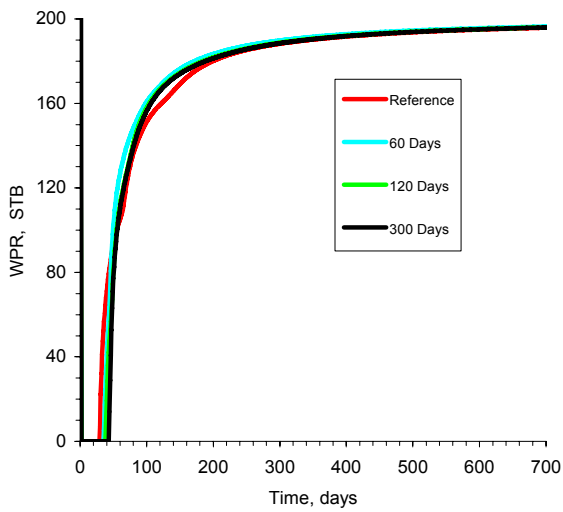
(a) WPR at Well P1



(b) WPR at Well P2



(c) WPR at Well P3



(d) WPR at Well P4

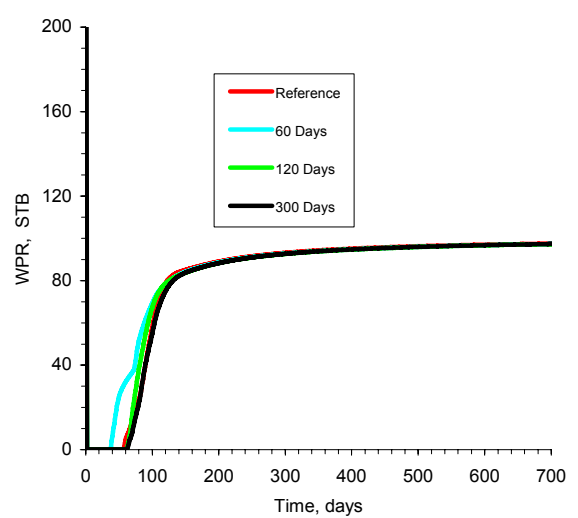
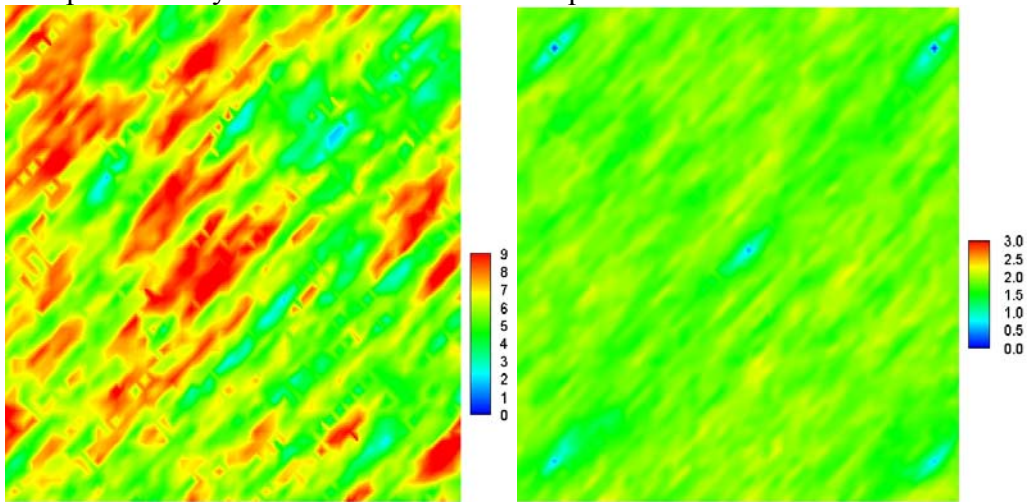


Figure 4-25: Well water production rates simulated from the mean permeability model updated by using square root algorithm with measurement perturbation method at selected times. The assimilation interval is taken as 60 days.

(a) Mean permeability field and variance with dependence measurement errors



(b) Mean permeability field and variance with independence measurement errors

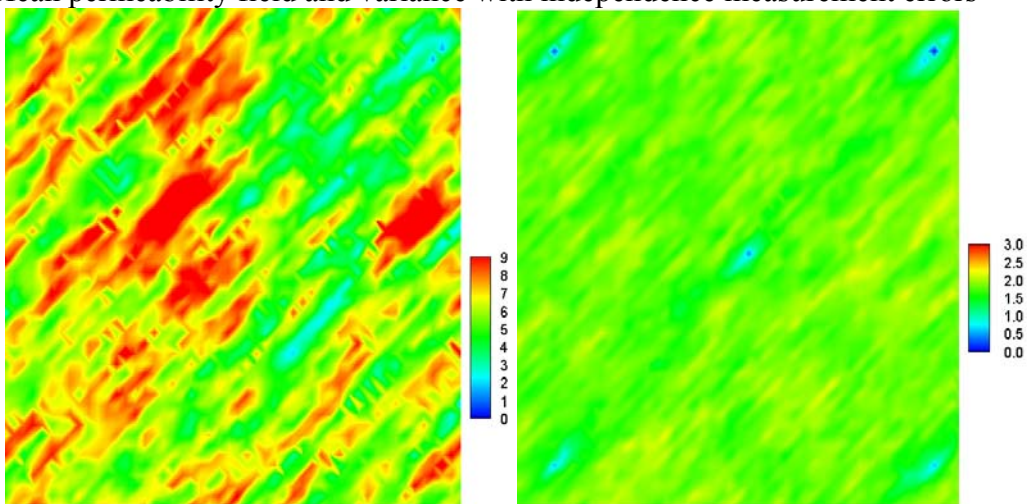
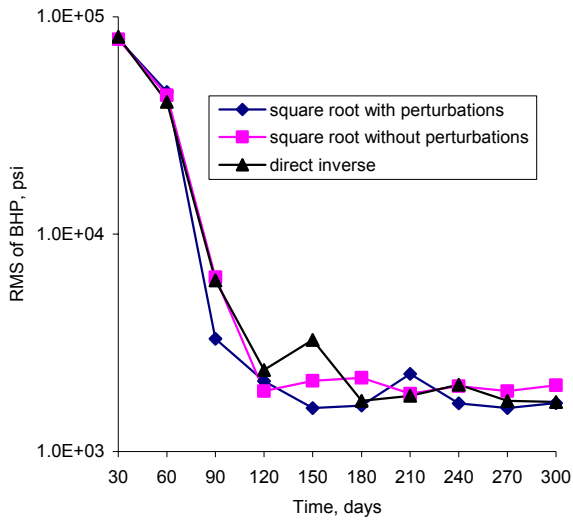
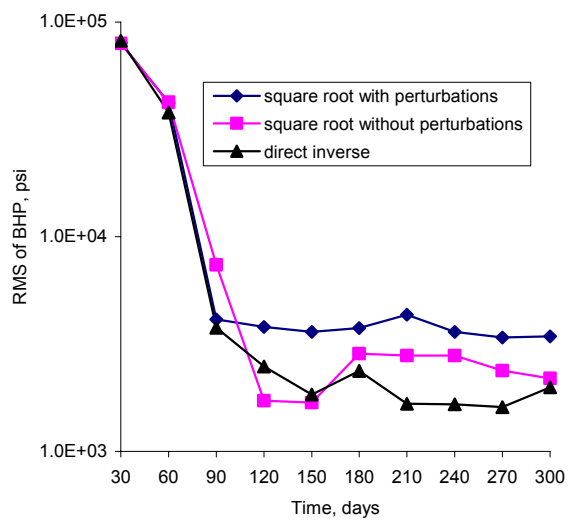


Figure 4-26: Mean permeability fields and corresponding variances obtained at the end of 300 days from square root algorithm without measurement perturbation, with the assumption of the dependence and independence of measurement errors, respectively.

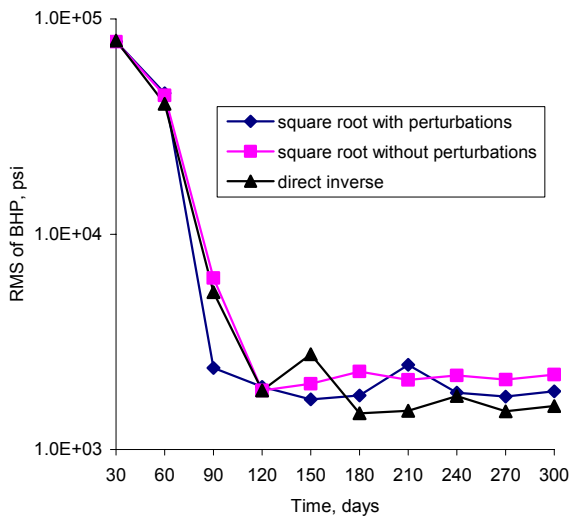
(a) BHP at Well I



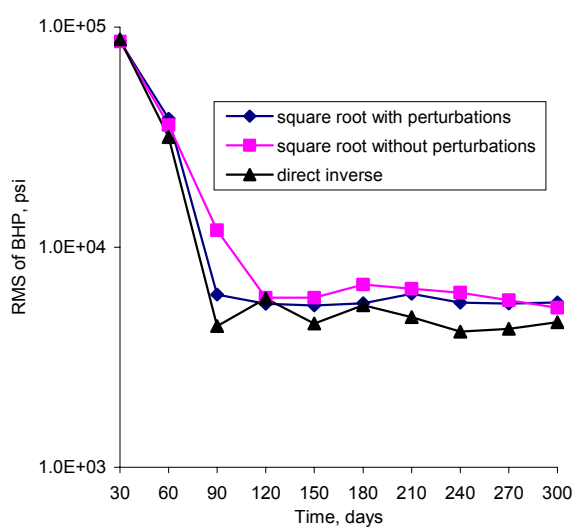
(b) BHP at Well P1



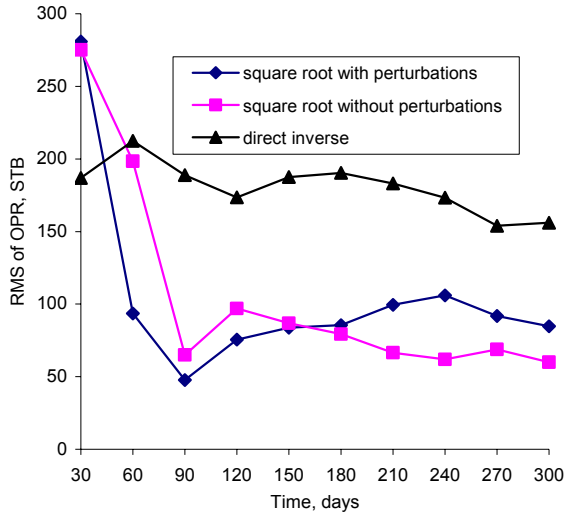
(c) BHP at Well P2



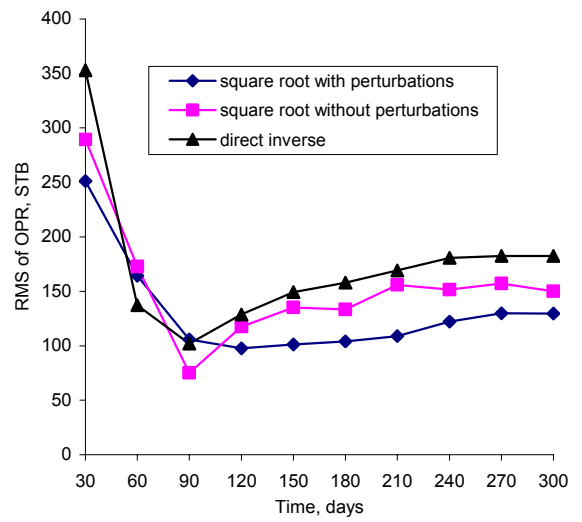
(d) BHP at Well P3



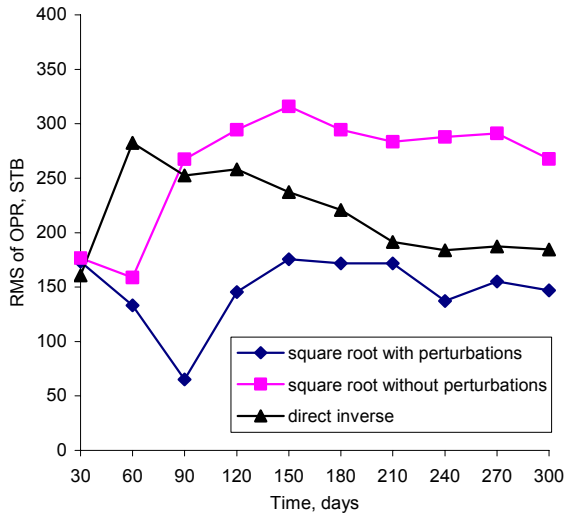
(e) OPR at Well P1



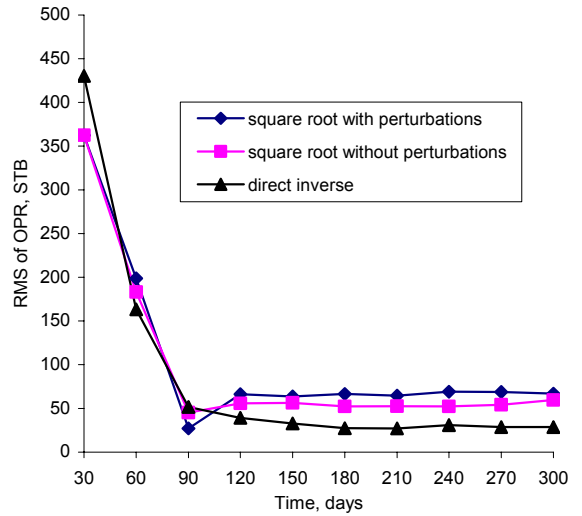
(f) OPR at Well P2



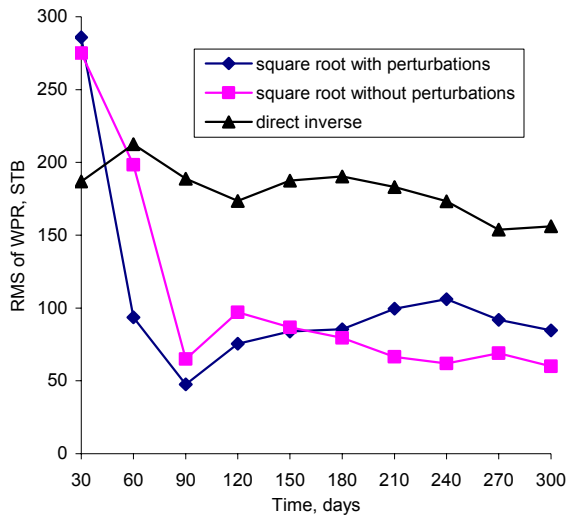
(g) OPR at Well P3



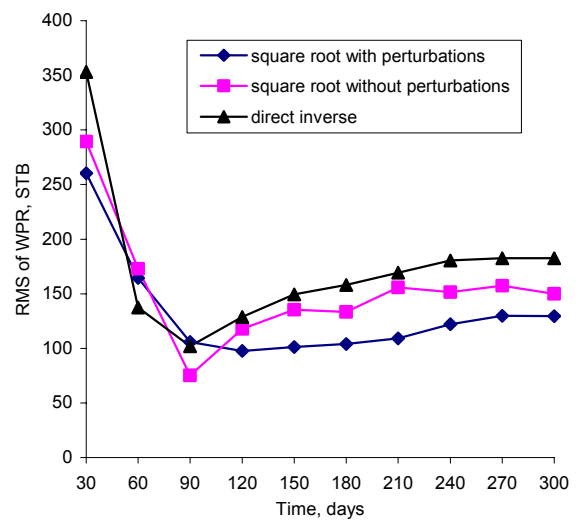
(h) OPR at Well P4



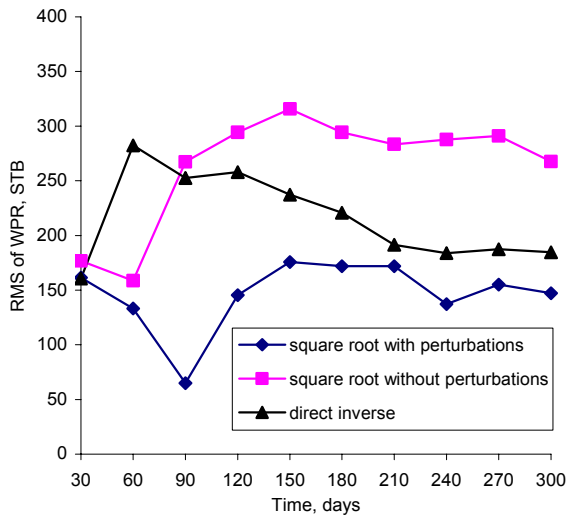
(i) WPR at Well P1



(j) WPR at Well P2



(k) WPR at Well P3



(l) WPR at Well P4

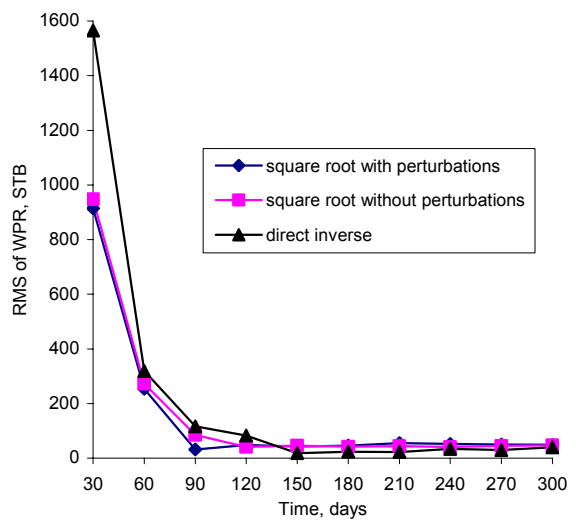


Figure 4-27: RMS profiles of BHP, OPR, and WPR for dependent and independent measurement errors using square root without measurement perturbation method.

Chapter 5: Improved Initial Sampling Strategy in the Ensemble Kalman Filter

Time cost is a major concern in reservoir simulation. The ensemble Kalman filter (EnKF) requires multiple members and consequently runs multiple simulations during history matching. Therefore, the number of ensemble size directly relates to the overall efficiency of the EnKF. In general, the ensemble members are randomly sampled and a large number of realizations are needed. Without selection, however, the sampled realizations possibly have high correlations and impair the method performance. It is desirable to investigate the possible ways to reduce the number of ensemble members while retaining the high level of accuracy.

The objective of this chapter is to improve the initial sampling strategy by using the concept of singular value decomposition (SVD). The idea is that we cleverly generate the initial sampling and abate the interdependences. We first address the issue of ensemble size, and then the SVD theory and its implementation in the initial sampling are introduced. Through a synthetic case, Section 5.4 provides a comparison of different initial sampling strategies and different numbers of ensemble size, followed by a summary in Section 5.5.

5.1 ENSEMBLE SIZE

Because the whole unknown space is represented by a limited number of realizations, random sampling is always an issue for the Monte Carlo approaches. In order to acquire the representative space, a large number of realizations are usually required. Meanwhile, the feature of randomization leads to the possibility of high dependences among these members, which greatly reduces the sampling efficiency. As a Monte Carlo approach, the EnKF has the above discussed problem and relies heavily on

the large number of ensemble size to maintain the required statistics. As a result, there are two closely related issues. First, the realizations are generated randomly. This limited number of random realizations narrows the unknown space and underestimates the uncertainty of the real problem. Second, in history matching approximately 200 realizations are needed to maintain stability because a too-small size does not represent the statistical information and results in filter divergence or even collapse. Since each simulation case runs from a few minutes to a couple of days, the number of simulation cases needed during history matching is sensitive.

Liu and Oliver (2005) and Gu and Oliver (2005, 2006) reported a relative small ensemble numbers (40 in their studies) if the uncertainty study was not a primary goal. However, Wen and Chen (2005) addressed this issue and found that 200 ensemble were needed for their case. We believe that the number of ensemble size is dependent on the complexity of the reservoir case. Since most reservoir models have a large number of unknown parameters with high heterogeneity, 200 realizations are typically required, particularly in the cases we investigated. Therefore, 200 simulation runs are definitely a computational burden.

Two directions can be considered to alleviate the computation stress. One proposal is to save simulation time, such as grid reduction technique, fast proxy instead of full reservoir model, and parallel/distributed high performance computing. However, this approach does not solve the core issue, i.e., the possible correlation among realizations. The second solution is to selectively sample the unknown space. Methods for sampling a given random variable for efficient uncertainty propagation are available, such as the polynomial chaos expansion, Karhunen-Loeve expansion, and principal component analysis (Reynolds *et al.*, 1996; Xiu and Karniadaskis, 2003; Zhang *et al.*, 2005; Yadav, 2006; Sarma *et al.*, 2007). Unfortunately, such methods are usually

computationally slow for reservoir cases because of the eigenvalue and eigenvector calculations. In this chapter we use the SVD to reduce the possible correlations aroused during the EnKF initialization stage. We first generate a very large sampling pool. Based on this pool, we properly generate a small number of realizations using the SVD. With this small number of realizations, we demonstrate that the same level of accuracy compared to the results from the conventional approach is achieved.

5.2 SINGULAR VALUE DECOMPOSITION

The SVD, related to the concept of orthogonality, is extremely useful in matrix computation and matrix rank problem. It is also widely applied in satellite data storage and digital image processing. Here, a brief mathematical explanation is given.

A set of vectors $\{x_1, \dots, x_p\}$ in \mathfrak{R}^m is orthogonal if $x_i^T x_j = 0$ whenever $i \neq j$ and orthonormal if $x_i^T x_j = \delta_{ij}$. Orthogonal vectors are maximally independent because they point in totally perpendicular directions. A matrix $Q \in \mathfrak{R}^{m \times m}$ is said to be orthogonal if $Q^T Q = I$, where I is an identity matrix.

If A is a real m -by- n matrix then there exist orthogonal matrices $U = [u_1, \dots, u_m] \in \mathfrak{R}^{m \times m}$ and $V = [v_1, \dots, v_n] \in \mathfrak{R}^{n \times n}$, such that

$$U^T A V = \text{diag}(\sigma_1, \dots, \sigma_p) \in \mathfrak{R}^{m \times n} \quad p = \min\{m, n\}, \quad (5.1)$$

where $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_p \geq 0$.

In other words, if A is a real m -by- n matrix with $m \geq n$ then the SVD can be expressed as

$$A = U \Sigma V^T, \quad (5.2)$$

where U and V^T are orthogonal and Σ is diagonal. That is, $U^T U = I_m$, $V V^T = I_n$, U is an m -by- m orthogonal matrix, V is an n -by- n orthogonal matrix, and

$$\Sigma = \begin{pmatrix} \sigma_1 & & & & & \\ & \sigma_2 & & & & \\ & & \ddots & & & \\ & & & \sigma_{n-1} & & \\ & & & & \sigma_n & \\ & \mathbf{0} & & & & \end{pmatrix}$$

is an m -by- n diagonal matrix. In addition, $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0$. The quantities σ_i are called the singular values of A , and the columns of U and V are called the left and right singular vectors. The number of non-zero singular values is equal to the rank of the matrix. Thus, if A is singular then at least $\sigma_n = 0$. In practice, singular values are rarely exactly zero, but if A is “nearly singular” some of the singular values will be small. The ratio of σ_n/σ_1 can be regarded as a condition number of the matrix A . If the ratio is close to 1, then the columns of A are very independent; if the ratio is large, then the columns of A are nearly dependent.

Note that the matrices AA^T and $A^T A$ have the same nonzero eigenvalues and that the singular values of A are the positive square roots of these eigenvalues. Moreover, the left and right singular vectors are particular choices of the eigenvectors of AA^T and $A^T A$, respectively.

More details about orthogonality and the SVD can be referred to Golub and Van Loan (1989), Kahaner *et al.* (1989) and Forsythe *et al.* (1977).

5.3 IMPROVED INITIAL SAMPLING STRATEGY

In aerospace engineering, the satellite digitizes the picture by subdividing it into picture elements called pixels, and each pixel is represented by a single number. If each photograph were divided into 500×500 pixels, the satellite would have to send 250,000 numbers for each picture, which would take a great deal of time for transmission and affect the satellite efficiency. Thus, data compression is needed.

The idea of data compression is that we can consider this 500×500 array of numbers as a matrix A and approximate this matrix with a “simpler” matrix which requires fewer numbers through the SVD. Suppose u_i and v_i are the i -th columns of U and V , respectively, Eq. (5.2) can be written as

$$A = U\Sigma V^T = \sum_{i=1}^n \sigma_i u_i v_i^T. \quad (5.3)$$

To compress the data, the smaller singular values are set to zero. If only 10 singular values are used, the approximation is

$$A = U\Sigma V^T = \sum_{i=1}^n \sigma_i u_i v_i^T \approx \sum_{i=1}^{10} \sigma_i u_i v_i^T. \quad (5.4)$$

Thus, the approximate picture depends only on the first 10 columns of U and V and the rest are zero. Instead of 250,000 numbers, the approximate picture relies on only 10,000 numbers.

Figure 5-1 shows the approximate pictures with the SVD for fingerprint (Kahaner *et al.*, 1989), which captures almost all the features of the original data by using only 10

singular values. Such capability of SVD motivates our modification on the EnKF initial sampling strategy in this chapter.

In practice, we first generate a large number of random realizations and form a matrix A , which represents the whole unknown reservoir property space. Then, using the SVD, we take a small number of singular values and singular vectors to form a new matrix while maintaining the main structure of the matrix A , especially the mean and variance of each column.

Recall that state vector $y_{k,j}$ is the j -th ensemble member at time t_k . The dimension of state vector is $N_{y,k}$, which can change with time t_k to account for different amounts of production data at different times. With ignorance of time step, the ensemble matrix holding N_e ensemble members is defined as $Y^f = (y_1^f, y_2^f, \dots, y_{N_e}^f)$, with dimension of $N_{y,k} \times N_e$. For simplification, we drop out time index k in the later discussion.

The ensemble mean is stored in each column of \bar{Y} and the ensemble perturbation matrix is defined as $Y' = Y - \bar{Y}$. Therefore, the ensemble covariance matrix $C_{y,e} \in \mathfrak{R}^{N_y \times N_y}$ is expressed as

$$C_{y,e} = \frac{Y'(Y')^T}{N_e - 1}. \quad (5.5)$$

Suppose the real error covariance matrix is C_y , we can compute the eigenvalue decomposition

$$C_y = Z\Lambda Z^T, \quad (5.6)$$

where the matrices Z and Λ contain the eigenvectors and eigenvalues of C_y . Now approximating the error covariance matrix with its ensemble representation, i.e., $C_{y,e} \approx C_y$, we can write

$$C_{y,e} = \frac{1}{N_e - 1} Y'(Y')^T = \frac{1}{N_e - 1} U \Sigma V^T V \Sigma U^T = \frac{1}{N_e - 1} U \Sigma^2 U^T, \quad (5.7)$$

where, U , Σ , and V^T result from a SVD and contain the singular vectors and singular values of Y' . In the limit when the ensemble size goes to infinity, the N_y singular vectors in U will converge towards the N_y eigenvectors in Z and the square of the singular values, Σ^2 , divided by $(N_e - 1)$, will converge towards the eigenvalues, Λ . This shows that there are two strategies for the approximation $C_{y,e} \approx C_y$:

We can increase the ensemble size, N_e , by sampling additional model states and adding these to the ensemble. As long as the addition of new ensemble members increases the space spanned by the overall ensemble, this will result in a more accurate representation of C_y by an ensemble covariance, $C_{y,e}$.

Alternatively, we can improve the condition number of the ensemble matrix by ensuring that the first N_e singular vectors in U are similar to the N_e first eigenvectors in Z . Thus, with a given moderate ensemble size, the absolute error in the representation $C_{y,e}$ of C_y will be smaller for ensembles generated by such an improved sampling than that for random ensembles.

The first approach is the standard Monte Carlo method used in the traditional EnKF where the convergence is slow. The second approach has a selection process through which ensemble members have less dependence and span a larger space. These two strategies are, of course, used in combination when the initial ensemble is created in

the EnKF: generate a large sample pool with the first approach and then select important ones using the second approach. In this way, an N_e member ensemble with better conditioning is generated while the computation of C_y eigenvectors, which is too large to allow the direct computation, is avoided. In more detail, such a concept is illustrated in Figure 5-2: first generate a large ensemble βN_e (β is a natural number and greater than 1), and then resample N_e members along the first N_e dominant singular vectors of this larger start ensemble. Note that, in application, an extra scaling step is needed to retain the correct variance in the new ensemble.

Evensen (2004) proposed a sampling strategy in the EnKF on an oceanography application. Based on his proposal, we apply the following sampling procedures for reservoir history matching process.

1. Sample a large ensemble of reservoir states with βN_e members and store the ensemble perturbations in $Y' \in \mathfrak{R}^{N_y \times \beta N_e}$;
2. Compute the SVD, $\hat{Y}' = \hat{U} \hat{\Sigma} \hat{V}^T$;
3. Retain only the first $N_e \times N_e$ quadrant of $\hat{\Sigma}$ which is stored in $\Sigma \in \mathfrak{R}^{N_e \times N_e}$;
4. Scale the non-zero singular values with $\sqrt{\beta}$;
5. Generate an N_e ensemble using only the first N_e singular vectors in \hat{U} and store in U ;
6. Generate a random orthogonal matrix $V_1^T \in \mathfrak{R}^{N_e \times N_e}$ by the SVD of a random $N_e \times N_e$ matrix, $M = U_1 \Sigma_1 V_1^T$;
7. Obtain the improved sampling members $Y' = U \frac{1}{\sqrt{\beta}} \Sigma V_1^T$;
8. Rescale the ensemble perturbation Y' to ensure the zero mean and specified variance.

5.4 CASE STUDY AND DISCUSSION

The two-dimensional waterflooding case in Chapter 4 is used continuously here. We will delve into two main concerns of the initial sampling in the EnKF application: one is whether the mean and variance of the initial random sampling members should be fixed to the specified mean and variance; the other is how to improve the initial sampling strategy with limited ensemble numbers.

5.4.1 Property Adjustments of the Initial Sampling

In the previous chapter, 200 permeability fields are initially sampled using the Sequential Gaussian Simulation method `sgsim.exe` (Deutsch and Journel, 1998). For the limited ensemble size, the mean and variance of each sample are not exactly same as those specified, and will fluctuate to some extent. To investigate the influence of this fluctuation on the updated results, we ran two cases by using the direct inverse calculation for the EnKF assimilation step. In the first case, we generated 200 initial ensembles from `sgsim.exe` and used them directly for permeability fields. In the second case, based on initial ensemble realizations from `sgsim.exe`, we corrected their means and variances. For each realization, this was done by subtracting the mean and then dividing the number of gridblocks by the square root of the ensemble variance. We then transformed the logarithm permeability with mean 6.0 and variance 3.0 to the real field. Figure 5-3 shows the mean permeability fields with and without sampling fix at the end of 300 days. The results from the two approaches are almost the same. With these matched fields, we reran the cases from the beginning. No significant differences were found in the production histories from the two cases and the reference, i.e., well bottomhole pressures, water rates and oil rates. Figure 5-4 presents oil and water rates at well P3 by the mean permeability fields shown in Figure 5-3. We have not seen any big differences in the results with and without initial sampling fix process.

The results also indicate that it is not critical whether the initial sampling members have exactly the same statistical information as the reference field, such as mean and variance. Therefore, the EnKF is pretty robust and does not have a strict requirement for the initial samplings.

5.4.2 Initial Sampling with the SVD

We now compare the results from different ensemble numbers: (1) 100 random realizations, (2) 200 random realizations, and (3) 100 realizations generated by the SVD from 400 random realizations.

Figure 5-5 shows logarithm permeability fields of the reference and three history matching results using the above different ensemble numbers at the end of 300 days. In this figure, 100 random realizations are insufficient to represent the unknown model space and the results deviate from the references to some extent. The performance is improved when the ensemble number is doubled. On the other hand, the permeability field generated by 100 realizations with the SVD is identical to the field generated by 200 realizations without the SVD. Both of them capture the main feature of the reference field and are better than the results from 100 random realizations. In particular, many thin and high permeability strips and dots exist in the field from 100 random realizations and have the potential for computational convergence problems. Compared with the reference map, the results from 200 realizations and 100 realizations with the SVD have clearer configuration for high and low zones. This suggests that a large number of realizations is needed if sampled randomly, and that the SVD sampling strategy has a positive impact on the matching result.

For each of these permeability fields, we ran one simulation from the beginning up to 700 days and plotted well production curves. Figure 5-6 illustrates the BHP of wells P1 and P2, OPR and WPR of well P4. For BHP in well P1 and P2, the results from 100

realizations drop earlier and faster than the reference curves and generally underestimate the bottomhole pressure. The performances from 200 random realizations and 100 realizations with the SVD are very close to the references, and 100 realizations with the SVD are even superior to 200 random realizations. Similar observations are obtained from OPR at well P4. The results from 200 random realizations and 100 random realizations overestimate and underestimate the oil production, respectively. The curve from 100 realizations with the SVD lies between the reference and the curve from 200 random realizations. As for water cut at well P4, the curve from 100 random realizations is higher than the reference while the breakthrough from 200 random realizations is postponed.

Therefore, with some overhead time at the beginning to generate 100 realizations from 400 random initializations, the results from 100 realizations with the SVD perform comparably with those from 200 random realizations. The half deduction of the ensemble size saves much time since the most time is spent in the simulation running in the real practice.

5.5 SUMMARY

The initial sampling is a big issue for all Monte Carlo methods. Traditionally, the number of realizations is too large to be practical. This Monte Carlo method needs a large number of realizations to get stable results and therefore requires a large amount of computational time. For history matching, the realizations are randomly generated with geostatistical constraints. These randomly selected realizations may have a high correlation and affect the representation of the sampling space. Also, the random feature creates stability problems. In this paper we have investigated the initial sampling strategy for the EnKF and introduced an improved sampling approach based on the SVD: we

sample a large number of realizations and then use the SVD to resample a small group of realizations with less linear dependence. In more detail, we have done three aspects.

First, we consider ensemble size. It is difficult to find the proper number of realizations for the EnKF during history matching. Through a two-dimensional case, we found that 100 random realizations are insufficient to represent the whole unknown space of reservoir model parameters. Two hundred realizations generate better results; however, the total simulation time is doubled. Of course, this conclusion is based only on our specific case. Considering the strong nonlinearity and heterogeneity in the reservoir, we suggest that if the initial random sampling is used, 200 realizations are required to ensure a good capture of the initial uncertainty.

Second, we consider the sampling fix. Supposing we know the mean and variance of the initial reservoir model from the geological knowledge, the sampled multiple realizations are not guaranteed to have the exact mean and variance. Our study demonstrates that there is no big difference in the matching performances from random realizations with and without sampling fix for the desired mean and variance. This means that the EnKF itself is robust. More importantly, since it is impossible to accurately know the mean and variance of the reservoir in reality, we do not need to care much the mean and variance of the initial realizations.

Third, we consider an improved sampling strategy with the SVD. We used the sampling strategy with the SVD to study the permeability field and found that the modified sampling strategy has a positive impact on the matched results. In our example, by using 100 realizations, we achieved the same level of accuracy compared to the results from the conventional approach using 200 realizations. Considering the cost of reservoir simulation for each realization, this reduction of realizations can save a significant amount of time and expense.

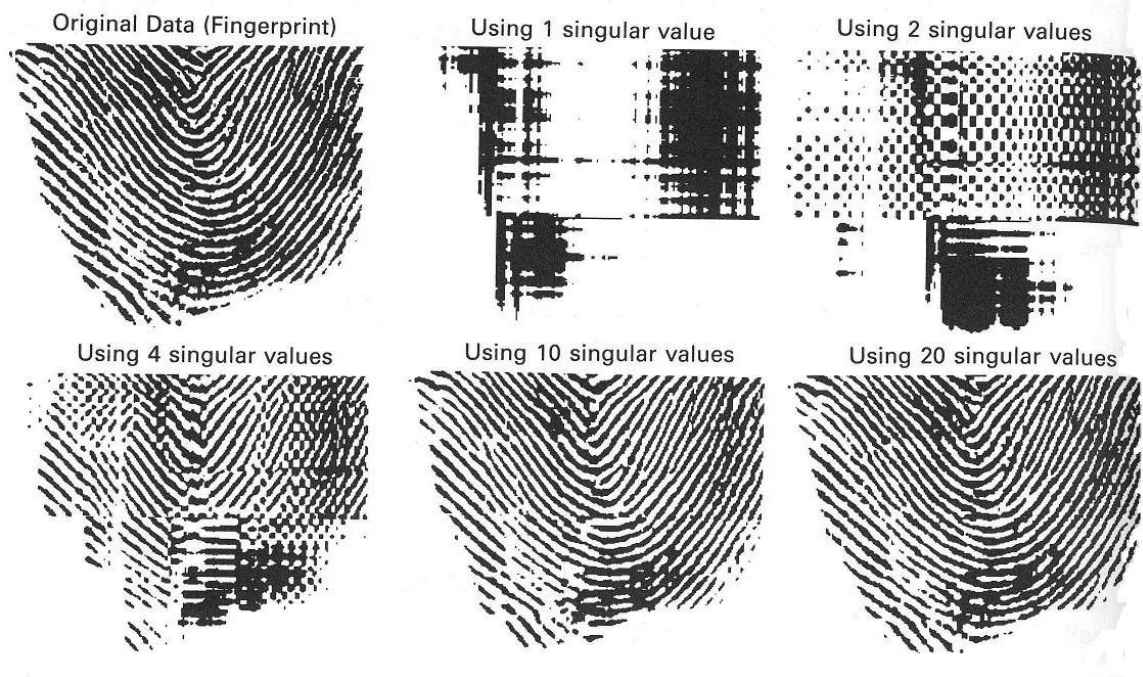


Figure 5-1: Approximating pictures with the SVD (Kahaner *et al.*, 1989).

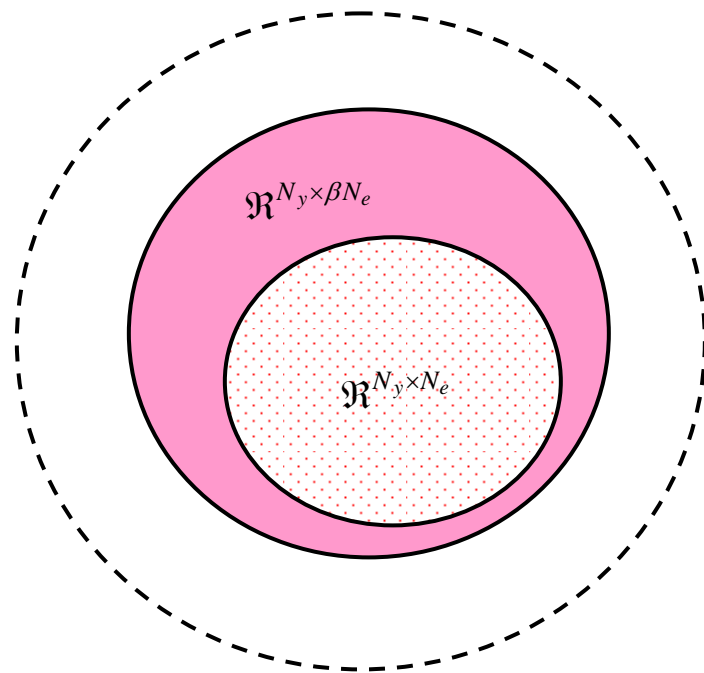
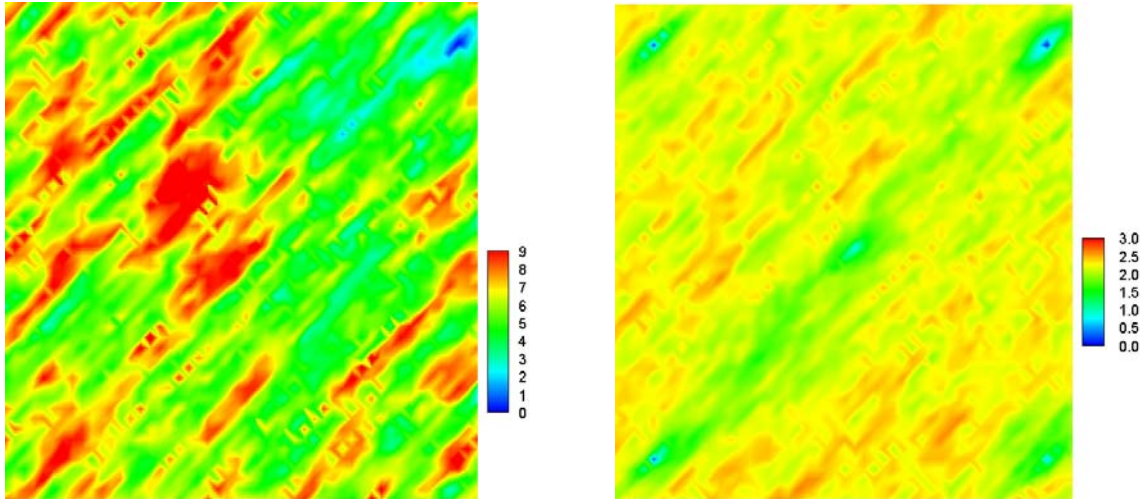


Figure 5-2: Illustration of the improved sampling strategy through the SVD.

(a) Mean $\ln(k)$ and variance with the sampling fix at then end of 300 days



(b) Mean $\ln(k)$ and variance with the sampling fix at then end of 300 days

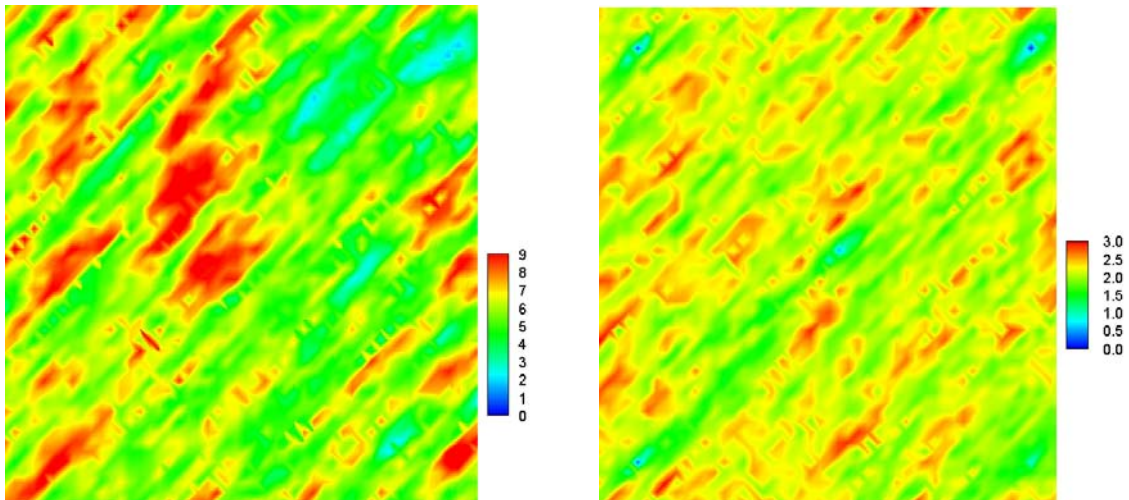
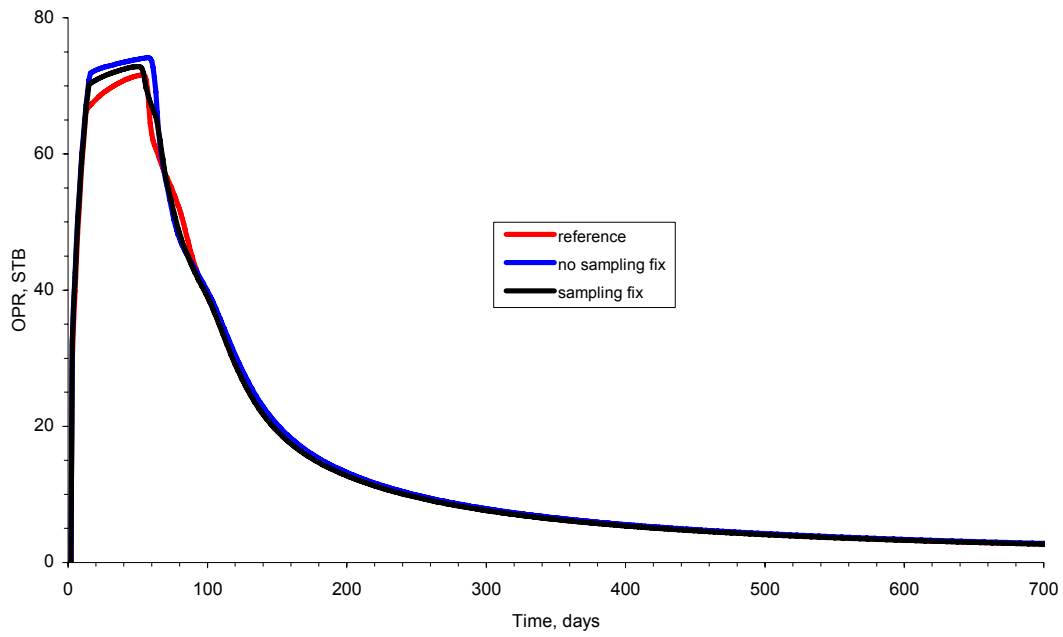


Figure 5-3: Mean permeability fields and the associated variance maps at the end of 300 days with and without initial sampling fix.

(a) OPR at well P3



(b) WPR at well P3

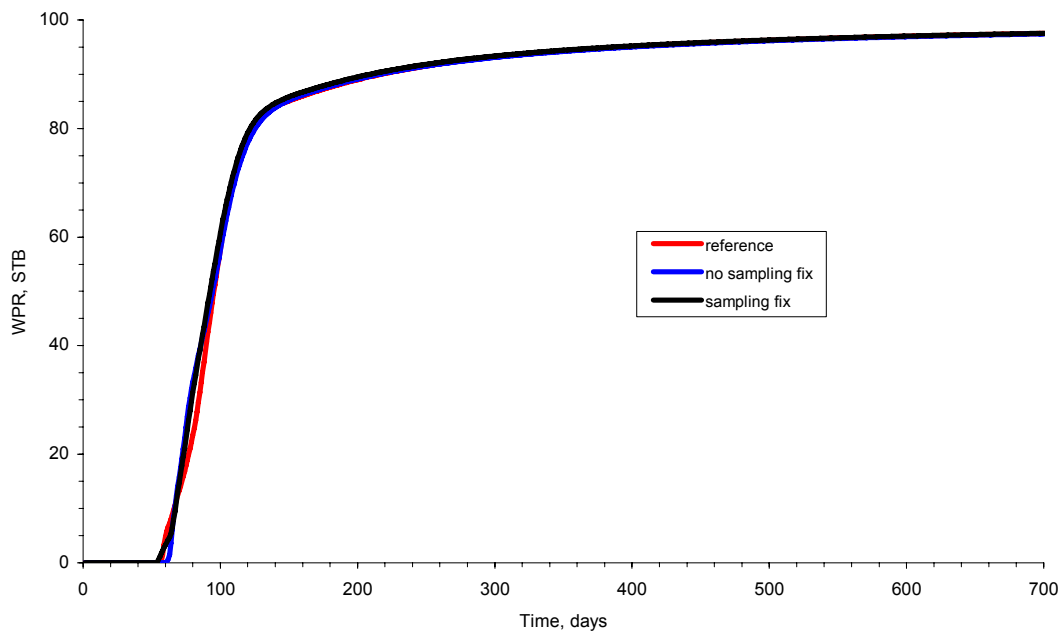
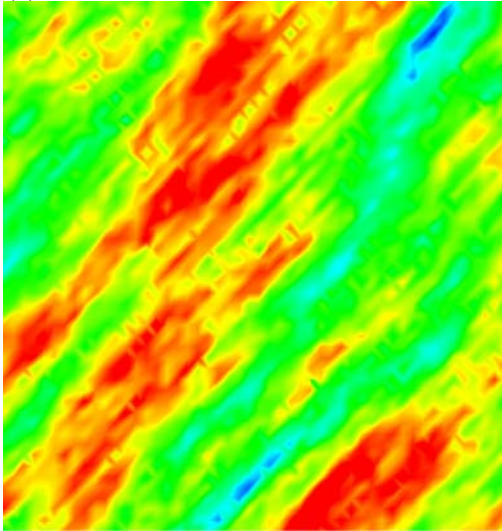
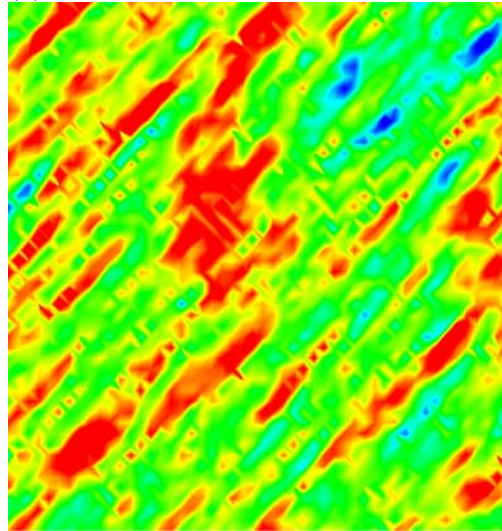


Figure 5-4: Well oil and water production rates simulated from the mean permeability model updated with and without the initial sampling fix, respectively, along with the reference curves.

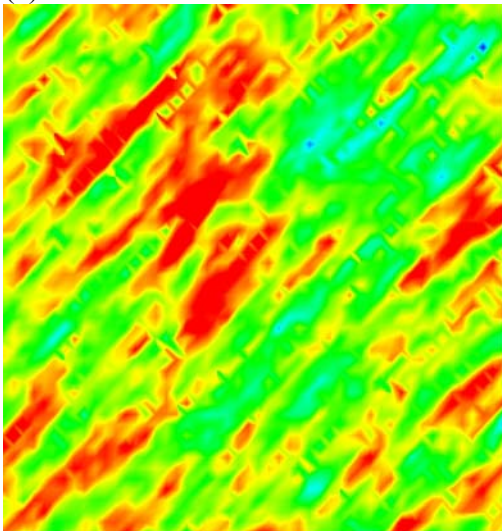
(a) Reference



(b) 100 Random Realizations



(c) 200 Random Realizations



(d) 100 Realizations with the SVD

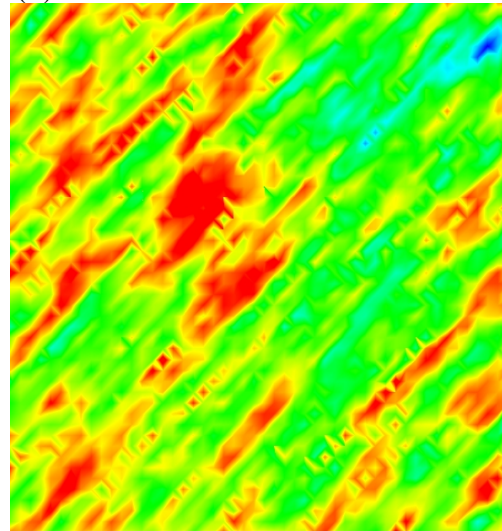
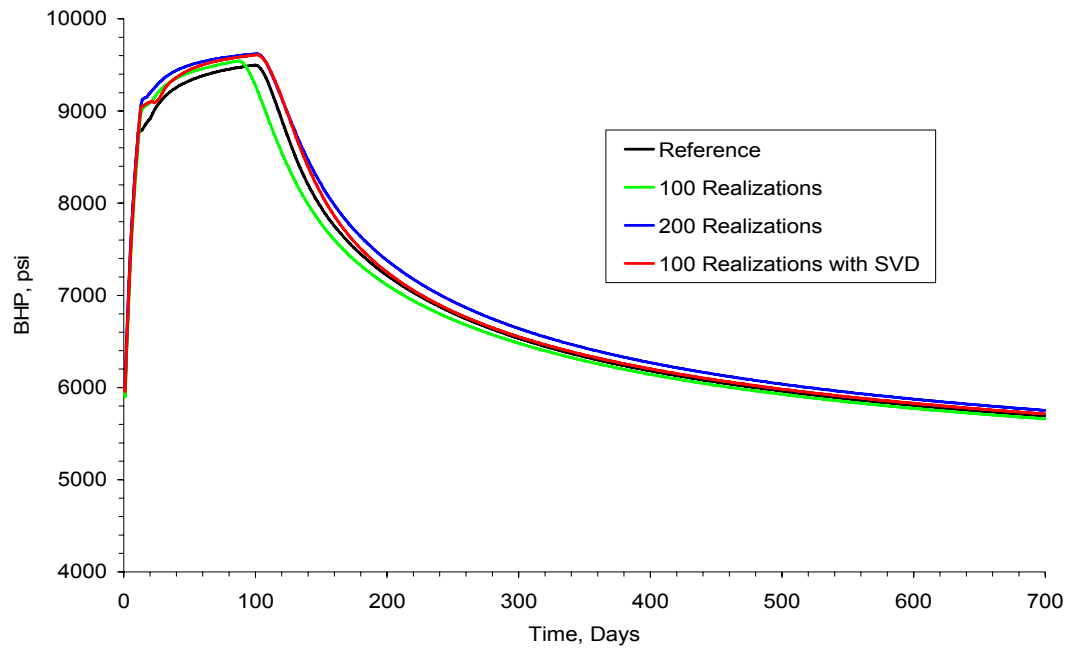
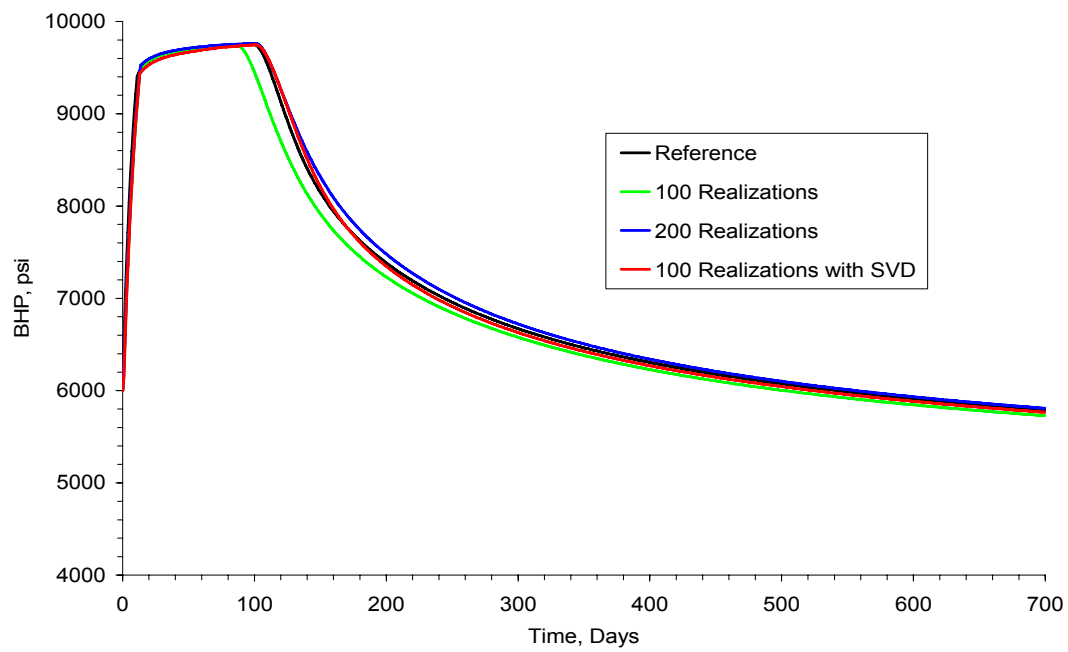


Figure 5-5: Logarithm permeability fields of reference, and three updated results at the end of 300 days using 100 realizations, 200 realizations, and 100 realizations with the SVD, respectively.

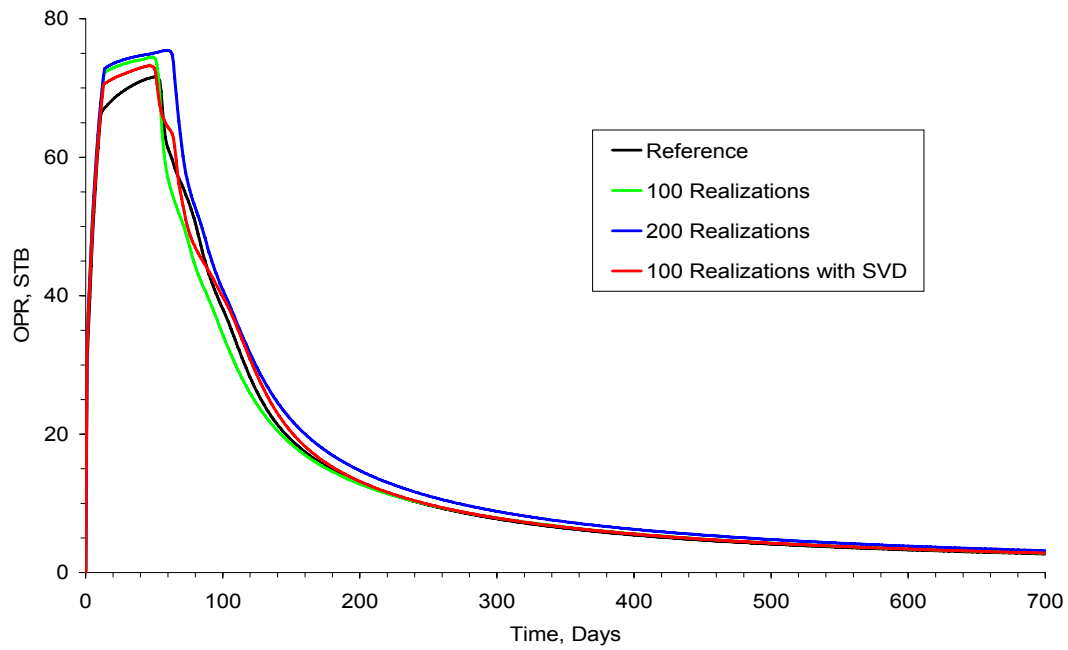
(a) BHP at well P1



(b) BHP at well P2



(c) OPR at well P4



(d) WPR at well P4

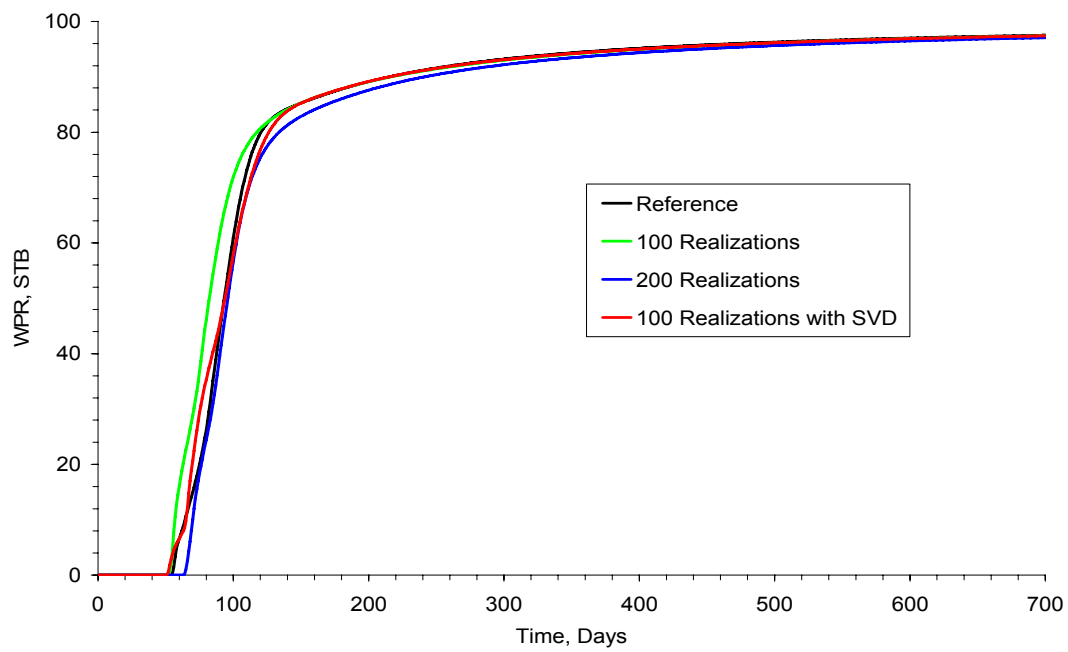


Figure 5-6: Well performances from different updated permeability fields with the comparisons with the references.

Chapter 6: Improved Assimilation Algorithm in the Ensemble Kalman Filter

The ensemble Kalman filter (EnKF) involves the initial sampling, forecasting and assimilation steps. It tunes multiple members sequentially and updates the statistical mean and variance of the model. Many applications have been reported in various literatures. Unfortunately, we have not seen much information on the modification of the EnKF algorithm. As we know, the forecasting step is implemented by running the reservoir model simulator. In the previous chapters, the improved initial sampling strategy and different approaches to solving the assimilation equation have been addressed. In the assimilation equation, the ensemble mean is calculated through equally weighing all the members. Therefore, the contribution factor to the mean from each member is the same. The purpose of this chapter is to propose a modified assimilation equation by introducing a weighting factor for each ensemble member. We also investigate the EnKF application for a modified field case of a complex seventeen-layer reservoir which has a strong heterogeneity. Throughout this case, the performances of the EnKF on production history match and forecasting, field permeability match, dynamic reservoir saturation and pressure are discussed. In addition, we investigate the impact of geological uncertainty in the initial ensemble generation on the final matching results. Two scenarios which have the same semivariogram as the reference field are implemented and their results are discussed.

We first give a detail discussion of the EnKF assimilation equation and other stochastic methods used in the engineering and science fields. Section 6.2 presents the proposed algorithm through modifying the mean and consequently reshaping the Kalman gain. A case study where the geologic information is estimated in the initial ensemble

members is demonstrated in Section 6.3, followed by another case study where the true geologic information is applied in the initial ensemble generation. Section 6.5 presents a summary of the chapter.

6.1 INTRODUCTION

From a statistical point of view, Bayes' theorem illustrates that the probability density of an unknown parameters ψ given a new set of observations d is expressed by

$$f_m(\psi | d) = \frac{f_d(d | \psi) f_m(\psi)}{\int f_d(d | \psi) f_m(\psi) d\psi}. \quad (6.1)$$

This means the probability density function (pdf) of the model given the observations is expressed by the product of the prior pdf of the model and that of the observations given to the model. The denominator is a constant for normalization. The equation also presents such a simple formulation for data assimilation that the posterior pdf can be obtained by simply multiplying the densities of model and observations.

However, the highly nonlinear nature of reservoir structure and fluid flow together with the relatively sparse observations make the history matching problem nonlinear. The EnKF allows the errors evolve with the nonlinear model equations by performing an ensemble of model runs. The ensemble members are a certain representation of the prior model pdf and the posterior pdf is represented by a weighting of these ensemble members. This weighting is dependent on the value of the observations given an ensemble member. Consequently, everything seems straightforward. Unfortunately, all the Kalman-based methods assume at the assimilation step that either the model is linear or that the pdf of the model parameters is a Gaussian distribution. For

observations, the assumption is that the pdf is a Gaussian distribution and that the measurement operator, which produces the model equivalent of an observation, is linear. Through reported applications, we have seen most cases obtained acceptable matching results even though the above-mentioned assumptions were not fulfilled. The problem is that the EnKF can produce unbalanced model parameters at assimilation steps because each updated ensemble member is just a linear combination of the prior ensemble. For example, negative water saturations or unbalanced dynamic parameters can occur. This is due to the Gaussian assumption on the pdf, while water saturation, for example, does not meet that assumption. The research on non-physical updated values and non-Gaussian pdf such as multimodal distribution of model parameters is not in the scope of this dissertation and we assume the unknown parameters satisfy the EnKF requirements. We are more interested in the information utilization of multiple members in the assimilation equation.

Two interesting aspects of the assimilation equation are its formulation and implementation, respectively. All the implementations of the Kalman-filter-based methods need the matrix inversion, which involves the sizes of ensemble members and the number of observations. This has been comprehensively discussed in the previous chapters. As for the assimilation equation, the key concept of the EnKF is that this method captures second-order moment, i.e., mean and variance, through simplifying the model parameters and observations as the Gaussian distributions. The mean is the most probable model and is taken by averaging all the ensemble members. The contribution of each member is equal. However, we know that the closeness of all the members to the true field is different. Some are close and some are far away. In other words, some ensemble members have little to do with the observations while some are close to the true model. Ideally, when calculating the mean, we want to give more weight to the close

members while giving less favor to others. This is the motivation of the formulation modification of the EnKF assimilation equation in this chapter.

In literature, the research on sequential importance sampling (Doucet *et al.*, 2001) provides us with an idea. In this method, the prior pdf represented by multiple particles is multiplied with the observation pdf to obtain the posterior pdf represented by the new multiple particles. This posterior pdf is then randomly sampled to give each particle equal weight. Such random sampling introduces an additional Monte Carlo variation which is unnecessary. According to the concept of the survival of the fittest, a weighted sampling directly based on the posterior pdf is needed to amplify and diminish some particles in the population. Sequential importance resampling (Doucet *et al.*, 2001; van Leeuwen, 2003) adds a partly deterministic scheme to condition the sampling so that the expected number selected from each particle is proportional to the multiplication of the particle weight and the population number. Therefore, particles with very low weight have a very low probability of being drawn, while particles with large weights can be drawn more than once. This is nothing more than abandoning those members that contain no information and stressing those that have.

Therefore, for the EnKF, we can modify the assimilation in a similar way: we first create an ensemble of realizations, run that ensemble forward until observations become available, weigh each ensemble member through the difference between its response and real observations, and continue the integration.

6.2 WEIGHTED ENKF

Recall that the assimilation equation was explained in Chapter 3. Each ensemble member can be updated using the previous one and a weighted difference between the observation data and model response. The weighting matrix is called the Kalman gain

and is denoted as $K_{e,k}$, where k is the index for the assimilation time. The assimilation equation for the j -th ensemble member is then expressed by

$$y_{k,j}^u = y_{k,j}^p + K_{e,k} (d_{obs,k,j} - H_k y_{k,j}^p), \quad (6.2)$$

$$K_{e,k} = C_{y,e,k}^p H_k^T (H_k C_{y,e,k}^p H_k^T + C_{d,e,k})^{-1}, \quad (6.3)$$

where, $C_{y,e,k}$ and $C_{d,e,k}$ are ensemble covariance matrix and measurement error covariance matrix, respectively. $d_{obs,k,j}$ is the measurement vector for the j -th member at the k -th assimilation time. p and u represent prior and updating, respectively. H is a matrix operator and $H = [\mathbf{0} | \mathbf{I}]$. The subscript e means values represented by the ensemble members. T is the matrix transpose operator.

Any element $c_{k,m,i}$ in the covariance matrix $C_{y,e,k}$ is computed as follows

$$c_{k,m,i} = \frac{1}{N_e - 1} \sum_{h=1}^{N_e} (y_{k,m,h} - \bar{y}_{k,m}) (y_{k,i,h} - \bar{y}_{k,i}) \quad (6.4)$$

$(m = 1, 2, \dots, N_y; i = 1, 2, \dots, N_y).$

Above, $c_{k,m,i}$ is the covariance between the parameters in the m -th and i -th rows in the ensemble matrix at the k -th time index. $y_{k,m,h}$ and $y_{k,i,h}$ are the m -th and i -th parameters in the stator vector for the h -th ensemble member at the k -th time index, respectively. h is the index for the number of ensemble members and $h = 1, 2, \dots, N_e$. N_y is the dimension of the state vector. At the k -th time index, $\bar{y}_{k,m}$ and $\bar{y}_{k,i}$ are the averages of the m -th and i -th rows in the ensemble matrix at the k -th time index,

respectively. $\bar{y}_{k,m}$ is deemed the most probable value for the m -th parameter in the reservoir model at the k -th time index and is calculated by

$$\bar{y}_{k,m} = \frac{1}{N_e} \sum_{h=1}^{N_e} y_{k,m,h}. \quad (6.5)$$

Rather than evaluating each member equally with $\frac{1}{N_e}$, we introduce a weighting factor w_h to adjust the contribution of the state vector for the h -th ensemble member according to the difference between its simulation responses and the observation data.

Suppose that at the k -th time index, we are operating N_{well} wells and each well has N_{pr} observation data, such as bottomhole pressure, oil and water rates. We then define a dimensionless factor for the h -th ensemble member as

$$1 - \frac{\sum_{q=1}^{N_{well}} \sum_{l=1}^{N_{pr}} (d_{obs,l,q,h} - d_{resp,l,q,h})^2}{\sum_{q=1}^{N_{well}} \sum_{l=1}^{N_{pr}} d_{obs,l,q,h}^2 + \sum_{q=1}^{N_{well}} \sum_{l=1}^{N_{pr}} d_{resp,l,q,h}^2},$$

which is between 0.0 and 1.0. $d_{obs,l,q}$ and $d_{resp,l,q}$ denote the observation and simulation response for the l -th datum in the q -th well. When the observation and simulation data are exactly the same, the factor is 1.0, while it turns into 0.0 if all the responses are 0.0.

Since we have N_e ensemble, we need the normalization of the weighting factor for each member. We then define the following weighting factor expression for the h -th ensemble member:

$$w_h = \frac{1 - \frac{\sum_{q=1}^{N_{well}} \sum_{l=1}^{N_{pr}} (d_{obs,l,q,h} - d_{resp,l,q,h})^2}{\sum_{q=1}^{N_{well}} \sum_{l=1}^{N_{pr}} d_{obs,l,q,h}^2 + \sum_{q=1}^{N_{well}} \sum_{l=1}^{N_{pr}} d_{resp,l,q,h}^2}}{N_e - \sum_{h=1}^{N_e} \left[\frac{\sum_{q=1}^{N_{well}} \sum_{l=1}^{N_{pr}} (d_{obs,l,q,h} - d_{resp,l,q,h})^2}{\sum_{q=1}^{N_{well}} \sum_{l=1}^{N_{pr}} d_{obs,l,q,h}^2 + \sum_{q=1}^{N_{well}} \sum_{l=1}^{N_{pr}} d_{resp,l,q,h}^2} \right]} \quad (h = 1, 2, \dots, N_e). \quad (6.6)$$

We therefore redefine Eq.(6.5) as

$$\bar{y}_{k,m} = \sum_{h=1}^{N_e} (w_h \cdot y_{k,m,h}). \quad (6.7)$$

Consequently, we modify the covariance matrix expressed by Eq. (6.4), Kalman gain in Eq. (6.3) and eventually the assimilation equation in Eq. (6.2).

6.3 FIRST CASE STUDY AND DISCUSSION

A three-dimensional reservoir model has sizes of 1660.14ft-by-1886.48ft-by-38.5ft, modified from a section of a real large reservoir. After discretization, it is modeled by 22-by-25-by-17 gridblocks with various grid sizes in each direction. The average porosity is 0.285. The average permeability in X direction is 1330mD with Dykstra Parsons coefficient 0.702 while the average permeabilities and Dykstra Parsons coefficients are 1336mD and 0.703 in Y direction and 669mD and 0.736 in Z direction. Ten to twelve layers are high permeability layers. The porosity and permeability in X direction are shown in Figure 6-1 and Figure 6-2, respectively. The average top depth is 2000 ft with reference pressure 550psi. Initial water saturation is 0.2.

During primary recovery, only two wells, producers P1 and P4 are operated with a constant production rate of $600\text{ft}^3/\text{day}$ for each well. At the end of 150 days, wells P1 and P4 are shut down. Producers P2 and P5 are operated with a constant production rate of $600\text{ft}^3/\text{day}$ for each well till 240 days.

After 240 days, a five-spot well pattern is used for waterflooding. Producers P1, P2, P4, P5 and the injector I1 are open. Each producer has a constant production rate of $1200\text{ft}^3/\text{day}$ and the injector has a constant injection rate of $600\text{ft}^3/\text{day}$.

After 600 days, we open 10 producers and 7 injectors, seventeen wells in total for an inverted-seven-spot well pattern. All producers are operated with the constant bottomhole pressure of 300psi. All injectors have constant injection rates. The injection rate of the well in the center is $30,000\text{ft}^3/\text{day}$ while all the other wells have $15,000\text{ft}^3/\text{day}$. The waterflooding is ended at 1500 days. All the wells are vertical and fully perforated. Well configurations are shown in Figure 6-3. Tables 6-1 and 6-2 give well locations in X and Y directions, and measured permeability values in each layer, respectively.

Basically, the total amount of observation time is 36 with various observation frequencies. The observation data include oil production rate, water production rate, bottomhole pressure in each layer in producers, injection rate, and injection bottomhole pressure in each layer. Table 6-3 gives the details. Two hundred ensemble members are implemented.

6.3.1 Creation of the Initial Geostatistical Model

The information of reservoir initialization from geologists is the precondition for reservoir engineering, especially for history matching. However, the geostatistical information in most cases is not directly obtainable. The purpose of our work in this section is to integrate reservoir characterization and reservoir engineering and illustrate how to combine them during history matching.

As we know, geological features and associated petrophysical properties are generally not distributed isotropically within a depositional environment. Geostatistics provides a method for identifying and quantifying anisotropic behavior in data with metrics that are used during interpolation to preserve directions and scales of continuity (Yarus and Chambers, 2006). This method is called variography, and the set of metrics it produces is identified from a graph called semivariogram. Variography is an interpolation method for an unsampled location, embracing the distance and directional-weighting issues. The concept of the semivariogram is to compare pairs of data at regular separation distances and to determine the degree of interdependency. Semivariogram consists of the sill, range and nugget three components. The inflection point at which the semivariogram flattens is called the sill and is theoretically equal to the true variance of the data. The distance at which the sill is reached is called the correlation range, or scale, and defines the distances within which there is a predictable relationship with variance. Beyond the inflection point, the data are not correlated, and no predictable relationship can be defined. The nugget effect occurs when the slope of the semivariogram intersects the Y-axis above the origin, suggesting the presence of random or uncorrelated noise at all distances.

To generate a semivariogram in this case study, values in the seventeen well locations and additional twenty sampled places are used. The reference map is analyzed and a permeability trend with a counterclockwise 135 degree is estimated. For these irregularly spaced data, a GSLIB (Deutsch and Journel, 1998) variogram program, GAMV, generates output file “gamv.out” with semivariograms in three directions. Further analysis of these semivariograms reveals that the reservoir has only one structure. The results in “gamv.out” are then taken as the experimental data to gauge the regression process of the semivariogram modeling. In the regression, a type of semivariogram model

with parameters including sill, correlation range and nugget effect are determined. VMODEL, a program in GSLIB, offers a tool for fast regression. Through trial and error, the exponential model is selected and correlation ranges 12, 5 and 1 (unit: grid size) in three directions are determined by data regression. The next step is to plot the results by using GSLIB program VARGPLT, as shown in Figure 6-4. VARGPLT takes the special output format used by the semivariogram programs GAMV and VMODEL and creates graphical displays. We can see they are well matched, especially within the correlation range, while the uncorrelated part (sill) does not need to be taken in consideration. In this respect, the parameters required for permeability simulation are generated.

Kriging is the approach to interpolating the permeability field for the reservoir characterization onto a grid by using the conditional data and the spatial model, i.e., semivariogram. A major advantage of kriging over other interpolation algorithms is the ability to use more than one variable simultaneously to predict the value at an unsampled location. Conditional simulation reflects the proper spatial relationships among the various geological elements and their petrophysical properties as well as the heterogeneous nature of those properties. The key point for conditional simulation is that it captures the heterogeneity. In our case application, the data in seventeen well locations are assumed measured and ordinary kriging is selected. The conditional SGSIM can directly generate the three-dimensional permeability map for the whole reservoir without any further mean or variance transformation.

Once the initial realizations are generated, a transformation from the coordination system in geostatistics to the reservoir system is needed. For a reservoir with dimensions $N_x \times N_y \times N_z$, suppose three directions in geostatistics and reservoir are I_g, J_g, K_g and I_r, J_r, K_r , respectively. If the difference of half block size due to the block-

centered and point-centered discretizations is ignored, it is easy to derive the transformation formula

$$\begin{cases} I_r = I_g \\ J_r = N_Y + 1 - J_g \\ K_r = N_Z + 1 - K_g \end{cases} \quad (6.8)$$

6.3.2 Type of Measurement Data

In intuition, more measurements bring about more information and hence lead to more accurate recognition of the research object. In reality, however, a large amount of data can be available continuously while not all the data are informative. Instead, the value of information needs evaluation. The aim of this section is to investigate the impact of different amounts of measurement data on the matching results.

In this case study, when the time is greater than 600 days, ten producers and seven injectors are operating with a constant bottomhole pressure and a constant injection rate, respectively. Two scenarios are considered. In the first one, only oil and water rates in each producer are measured. There are a total of 20 data points at each observation time. In the second scenario, bottomhole pressure in each layer in each injecting well is also accounted for. There will be 119 (17×7) more data and totally 139 ($20 + 119$) data points are collected at each observation time. All the other information, such as everything before 600 days and the direct inverse assimilation method for solving assimilation equation, is kept exactly the same in these two scenarios.

Figures 6-5 through 6-7 show the matching permeability results of each layer by using two types of measurement data at the time of 620 days, 640 days and 660 days, respectively. At the time of 620 days, layers 5 through 7 and 9 through 17 are identical

while there are observable differences in layers 1 through 4 and layer 8. However, the overall shape including high and low permeability locations is similar in the results from the two measurement scenarios. At the time of 640 days, new measurement data are further incorporated into the ensemble in the two scenarios. Figure 6-6 represents that the matching results from the two scenarios are approaching to the same direction and have not much difference, except the discrepant magnitudes of low permeability values in layers 6, 12 and 15. At the time of 660 days, though there are some mismatches of high and low permeability areas in some layers, each layer profile is very similar. Note that in the areal coordinate system the original point is located at the left-bottom corner, the X axis is towards right while the Y axis towards up.

Figures 6-5 through 6-7 indicate that the matching results using 20 production rates have the same level of accuracy with the results from 20 production rates plus 119 bottomhole pressures in seven injectors. Consequently, in this case the EnKF history matching is not sensitive to bottomhole pressures in injecting wells. In other words, adding more pressure data does not bring obviously better or worse matching results.

6.3.3 Sensitivity Study of Measurement Interval

In our case study, when the time is greater than 600 days, inverted-seven-spot well pattern with 10 producers and 7 injectors is implemented. Through running the reference case, we notice that from 600 days to 1500 days, oil rates soar drastically from the low rates before 600 days, and then drop quickly. Correspondingly, the trend of the water rate rises and reaches a plateau. Between 800 days and 1500 days, production rates are almost stable and the information should be less valuable. Since the sampling interval is important for a good history matching, we focus on the time period between 600 days and 800 days, and investigate two scenarios with different measurement intervals. Table 6-4 gives the sampling frequencies with time intervals in details. Therefore, from 0 day

through 1500 days, the first scenario has a total of 36 times of observation while the second has 64 times. As for the measurement type, the first scenario is the same as the first scenario in Section 6.3.2, using data types listed in Table 6-3; while the second scenario is the same as the second one in Section 6.3.2, using 139 data in the time interval between 600 days and 1500 days.

Figures 6-8 to 6-10 show the matching permeability results of each layer from these two scenarios at the time of 620 days, 700 days and 800 days, respectively. In Figure 6-8, plots in layers 1, 3, 5, 6 and 13 from two different scenarios have major differences. The figures for the remaining 14 layers have similar shapes while each plot from the second scenario (high sampling frequency) has a larger contrast: the high permeability is larger and the low permeability is smaller than the corresponding values in the plot from the first scenario (low sampling frequency). Consequently, the results in the high sampling frequency are more heterogeneous. Note that the high permeability layers 10 through 12, which are the main contributor for the production matching, are almost the same in both scenarios. Also note that the coordinate systems here are same as in Figure 6-5.

Similar observations are shown in the plots in Figure 6-9. However, there exist obvious differences in layers 4, 6, 8, 13 and 15. In particular, the structures in layers 13 and 15 from two sampling strategies are very different. In Figure 6-10, however, the differences are smaller. Only layers 4, 13 and 15 show the discrepancy while all the plots in other layers are much identical, both in structure shapes and value magnitudes. This implies that with time increasing, useful information can be obtained in the first scenario even though this information is missing at the beginning because of the sparse data sampling. In this sense, the importance of very frequent data sampling is weakened, especially considering the implementation cost to get the data.

Meanwhile, since the data in the second scenario not only has a dense sampling interval but also includes the 119 bottomhole pressure data points from seven injectors, the differences between the two scenarios are attributed to the combined effect of these two factors.

Therefore, history matching in this case is not very sensitive to high frequency sampling if the production history is pretty long. In our case, the 20-day or 30-day frequency leads to results similar to those from a high frequency such as one day.

6.3.4 Comparisons between Conventional EnKF and Weighted EnKF

Based on the understanding from Sections 6.3.1-6.3.3, we implement two history matching approaches using conventional EnKF and weighted EnKF. Here, the conventional EnKF refers to the direct inverse assimilation scheme. The initial ensemble, measurements, reservoir and fluid properties are exactly same as the first scenario in Section 6.3.3. The simulation period is from 0 day up to 1500 days. The observation data and sampling intervals during the simulation time are presented in Table 6-3 and Table 6-4 (Scenario 1), respectively.

6.3.4.1 Evolution of a Matched Permeability Field

Figures 6-11 through 6-27 show the matching permeability results of each layer from these two scenarios at the time of 0 day, 150 days, 240 days, 600 days, 800 days, and 1500 days, respectively. For easy comparison, the reference is given first, followed by the initial model averaged by 200 ensemble members. Note that the same initial model is used in the two approaches to demonstrate the impact of two different assimilation methods. The coordinate systems in these figures are same as in Figure 6-5.

Some similarities exist in these figures. At the early time, 150 days and 240 days, the model does not change much because only two producers operate and the available

information is confined within a small area around the reservoir center. From 240 days to 600 days, a five-spot well pattern which covers the whole reservoir is implemented, and more information contributes to more change in the matched model. After 600 days, all seventeen wells open and big changes appear in all layers: the low permeability area is shown in blue and the high permeability area shown in red. This phenomenon not only matches the idea of sequential data assimilation in the EnKF, but indicates that a sufficient production history is needed in order to collect enough observation data.

Excitingly, the high permeability layers such as 10-12 are recovered pretty well in each method. Meanwhile, we notice the mismatches between the reference maps and the results in several layers generated after 800 days. For instance, the reference values in low permeability areas are higher than the matches from 800 days and 1500 days in both methods. Similar phenomena are observed in layers 2, 3, 4, 7, 8, 13, 14, and 15. The matched layers are obviously more heterogeneous than the references. One possible reason is that we treat all the seventeen layers as one geological structure expressed by one semivariogram while the high permeability layers such as 10-12 seem to have more complex structures than other relatively homogeneous layers. When the EnKF methods blindly modify all the layers simultaneously, the low permeability layers are changed in more heterogeneous way with undesired high fluctuation.

When we simply compare the matching maps from the two different methods in each low-permeability layer, it is hard to tell which one is better, since the references are relatively homogeneous while both matching results, though different shapes, have obvious high- and low-permeability contrasts. In layer 5, the weighted EnKF generates a worsened low-permeability structure after 800 days, compared with corresponding conventional EnKF results. In layer 6, both methods recover some low-permeability areas but fail to capture the locations of these areas. Similarly, in layer 7, undesired low-

permeability zones appear in both methods. It also seems that the trend of the low permeability zone is totally different in the reference and in the results from the two methods. This might be explained by the simplification of the initial reservoir model. In layers 10 and 11, we see both low- and high-permeability structures and the locations are better captured in the weighted EnKF than in those from the conventional EnKF in the corresponding times. In layer 12, the basic shapes in both methods are good but both fail to capture the high-permeability structure at the right bottom corner.

In summary, both methods almost capture the main reservoir features in each layer. However, we could not evaluate them by eyes simply from the permeability figures. The investigation of other properties is needed.

6.3.4.2 Evolution of a Matched Pressure Field

Since reservoir pressure is included in the state vector of the EnKF and changed gradually at each assimilation step, we will plot the matched pressure maps. Before 240 days, the reservoir is operating in a natural depletion mode with constant well production rates. After that, an injector with constant bottomhole pressure is applied. Hence, reservoir pressure drops fast during the natural depletion period. Layer and three-dimensional pressure maps after the assimilation steps at the end of 150 days and 240 days are illustrated in Figures 6-28 through 6-31, associated with the pressure maps from the reference model at the corresponding times. Note that the coordinate system in these figures is different from previous figures: the gridding starts from the top left corner; the X axis increases from left to right and the Y axis increases from top to bottom, as shown in Figure 6-3.

From these plots, we see that after the assimilation step, the pressure map in each layer matches very well with the reference. This demonstrates the EnKF's capability of handling multiple parameters simultaneously. After 240 days, constant bottomhole

pressure is implemented in the injecting well. Pressure fields, gradually stable with time, are easy to match and the comparisons are ignored here.

6.3.4.3 Evolution of Matched Saturation Field

After the natural depletion period, water injection begins and reservoir water saturation increases. We select the maps after the assimilation steps at the end of 600 days, 700 days, 800 days, and 1500 days, respectively. To save space, each layer comparison is shown only at 800 days, while several cross-sectional profiles and three-dimensional maps are presented at 600 days, 700 days and 1500 days. Figure 6-32 illustrates that good matches are achieved at the end of 600 days: three sections crossing injector I1 are shown. We find that the high water saturation values surrounding injector I1 are well recovered in both conventional EnKF and weighted EnKF. On the other hand, the number of low water saturation blocks, shown in blue in the plots, is smaller in these two methods than in the reference. Such discrepancy is improved in Figure 6-33. With time increasing, the number of low water saturation blocks is reduced. Cross-sectional profiles illustrate good matches at low and high water saturation blocks. Such matches are also illustrated in the three-dimensional slab views.

Next, we plot the comparison of each layer at the end of 800 days, shown in Figure 6-34. Again, we see exciting matches in each layer. In layers 1 through 8, the high water saturation blocks are similar in two methods and the low water saturation areas in the weighted EnKF are better than in the conventional EnKF. Overall water saturation in layers 9 through 17 is higher than that in layers 1 through 8. The shapes of high water saturation areas in these layers 9 through 14 are visually better in the weighted EnKF than those from the conventional EnKF. In addition, Figures 6-35 and 6-36 give the cross-sectional profiles of P6-I1 and P1-P5, as well as three-dimensional slab views, at the end of 800 days and 1500 days, respectively. These plots also demonstrate good

matches with the references. For the high and low water saturation values, the results from the weighted EnKF are better than those from the conventional EnKF.

6.3.4.4 Production History Matching and Recovery Forecasting

In order to evaluate the matching results, we rerun the case from 0 day to 1800 days by using the permeability models obtained at the beginning, at 150 days, 240 days, 700 days, 800 days and 1500 days. Note that we have two sets of reservoir models at each time: one is from the conventional EnKF and the other is from the weighted EnKF. Water and oil rates versus time are plotted for each producing well. In addition, reservoir cumulative oil production in terms of the percent of original oil in place versus time and reservoir overall water cut versus time are plotted.

Figure 6-37 shows the oil rates in well P1 by using the permeability models generated at different times. The reference curve is also plotted. We find that the oil rate from the initial model has a large deviation from the reference, especially for the peak production. At the time of 150 days, the rate from the conventional EnKF is almost unchanged while the result from the weighted EnKF is much improved. This means that the weighted EnKF converges faster than the conventional EnKF at the early time period of history matching. At the end of 600 days, however, the peak from the conventional EnKF is just slightly higher than that from the weighted EnKF. Oil rates from both methods are similar and are very close to the reference curve due to more information gathered. The only large difference from the reference lies in the period from 600 days to 1000 days. Such a difference is alleviated in Figure 6-37 (d), where the plots are provided from models in 1500 days. This indicates that EnKF is continuously modifying and we have not seen the divergence at late time though the permeability maps in 800 days and 1500 days seem different from the reference maps. Similarly, the improvement of the peak value and the period between 600 days and 1000 days can be found in Figures 6-38

and 6-39. The better performance of the weighted EnKF is also demonstrated. For the remaining oil rate plots in other producing wells, we see that the weighted EnKF converges faster than the conventional EnKF before 240 days and both gradually adjust to the reference. During the late time (i.e., after 800 days), the weighted EnKF is still slightly superior to the conventional EnKF. Another other finding is that the improvement continues even at the late simulation time and no divergence is actually found.

Figures 6-40 through 6-42 present water rate curves in wells P1, P2 and P3 by reservoir models from difference assimilation times in the conventional EnKF and the weighted EnKF, respectively. The reference is also presented for easy comparison. The superior performance of the weighted EnKF is pretty obvious in these groups of figures. Before 240 days, the results from the weighted EnKF are closer to the reference. This means the weighted EnKF has faster convergence to the reference at the early period of history matching. At the late time, for example, after 800 days, the weighted EnKF reaches better matches than the conventional EnKF. Again, no divergence is found even at the end of 800 days and 1500 days. This confirms that even though the permeability fields generated by the two EnKF methods have visual differences from the reference field, shown in Figures 6-11 through 6-27, we have good matches in reservoir water saturation, shown in Figures 6-32 through 6-36, and production history, shown in Figures 6-37 through 6-42. Note that in Figures 6-37 through 6-42, we have 300-day recovery forecasting from 1500 days to 1800 days. Good forecast matching is illustrated in these figures.

Further, we plot cumulative oil recovery in terms of original oil in place with time by using both the conventional and weighted EnKF at the end of different assimilation times, shown in Figures 6-43 through 6-45. At the first several assimilation times,

including the initial, 150 days, 240 days and 600 days, the plots are too close to differentiate, as seen in Figure 6-43. The enlarged windows of cumulative oil recovery are shown in Figures 6-44 and 6-45. We see that the weighted EnKF is better than the conventional EnKF though both are close to the reference curve.

In addition, with time increasing, the matches from both production history and recovery forecasting improve. Therefore, the root mean square of each layer should drop continuously with time if the root mean square is an effective method of quantifying history match. Interestingly, the root mean square of each layer keeps rising with time increasing in the overall trend. One typical plot is shown in Figure 6-46, where the final values are higher than the initialization. We think the reason for this is that many high and low channels appearing in the matched results do not exist in the corresponding locations in the reference, even though the overall shape is similar. As a result, the calculation method in the root mean square makes the differences in each block larger. In this sense, the root mean square is not sufficient to calibrate the matching performance of reservoir permeability, at least when the high and low permeability channels appear. We believe that the production history, such as oil and water rates in wells and in the overall field, is the most effective quantification of the matching performance.

6.4 SECOND CASE STUDY AND DISCUSSION

The EnKF starts from the ensemble members, representing reservoir initial recognition and large uncertainty. Currently, all the ensemble members are generated from the geostatistic information, which needs the semivariogram. In reality, however, an accurate semivariogram is not easy to obtain especially at the early beginning. It is also not uncommon during the reservoir exploration that the geologic structure, such as fracture, initial oil-water contact and fault, is found only gradually. Therefore, the initial

ensemble members could not represent such features. In this section we will investigate the impact of geological recognition on the final matching results.

Following a brief introduction, we give our second case study and discussion. The results where the semivariogram is known are compared with the results where the semivariogram is estimated. The accurate semivariogram leads to a higher quality of the matching permeability field than that from the case with the approximated semivariogram. However, in both scenarios, production history is well matched.

6.4.1 Exact or Approximated Reservoir Geologic Information

As more data is accumulated, an understanding of the reservoir characterization becomes clear with time. During a relatively long period at the beginning, reservoir structures, such as high permeability zone, fault, aquifer size, and water-oil contact, are not well determined. Even basic geostatistic information, such as correlation lengths and angles for semivariogram, are unclear. On the other hand, we need these reservoir structures and geostatistic information to generate the initial set of ensemble members. The only way to do this is to make a guess about the required information and then modify it during later production history. As for the EnKF, the question is to what extent such a guess affects the convergence of the history matching toward the real reservoir field.

Specifically, for the case we discussed in Section 6.3, the exact semivariogram of the reference is not available. We guessed the basic trend of permeability field and plotted the semivariogram using one geological structure on the basis of limited sampled permeability data. The final results match well with the well production history. However, the permeability map in each layer does not match our reference well. Except for the nature of the non-uniqueness, we believe that the uncertainty of reservoir semivariogram is another reason. If we understand the reservoir structure including all

the reservoir characterization information correctly, will the matching permeability field improve?

A new case will be investigated in this section, in which all the reservoir information and well operation are same as the case used in Section 6.3, except that the reference permeability field is changed to have the same semivariogram as the initial ensemble member.

6.4.2 Case Description and Results Discussion

Except for the permeability field, the three-dimensional reservoir model used here is exactly same as that in Section 6.3. Well operations and observation frequency are also unchanged, as shown in Tables 6-1 through 6-3. Bottomhole pressure data in all injectors after 600 days are included in the observation data. The total simulation period is from 0 day to 1000 days.

In Section 6.3, the reference permeability field, generated from other sources, was used directly. Here, we adopt the semivariogram result from Section 6.3.1. Though we have replaced the reference, we let it have the same semivariogram as the initial ensemble members. The 200 initial ensemble members are the same as those used previously. The conventional and weighted EnKF are run and the evolution of permeability layers with time is plotted together with the reference layer.

Figures 6-47 through 6-63 show the evolution of the permeability profile in each layer with time, together with the reference map. In Figure 6-47, at the end of 800 days, both the conventional and the weighted EnKF have already captured the main features of the first layer. However, we clearly see that the results from the weighted EnKF, especially at the end of 1000 days, show the high permeability area at the left bottom corner. The structure of low permeability in the weighted EnKF is also better than that from the conventional EnKF. Similar observations can be found in layers 2-5. In

particular, the weighted EnKF has a better low permeability area at the right bottom corner in the fourth layer at the end of 1000 days, shown in Figure 6-50. Both methods have a similar performance in the fifth, sixth and seventh layers, shown in Figures 6-51 to 6-53. Figure 6-54 illustrates that the weighted EnKF performs better in the high permeability areas located in the middle of the left boundary and right bottom corner. However, both methods have difficulty recovering the low permeability spots at the top section in the reference map. The shape from the weighted EnKF at 1000 days, where the left part is well recovered, is better than the conventional EnKF in the ninth layer. The results in the eleventh layer are identical. The right bottom corner is also better in the weighted EnKF. In the high permeability layers 12 and 13, the weighted EnKF generates almost the same maps as the references. Too many low permeability areas are demonstrated in the conventional EnKF at the end of 1000 days in the fourteenth layer compared with both the reference and the weighted EnKF. The results in layers 15 through 17 are similar in both methods.

Figures 6-47 through 6-63 demonstrate that both the conventional EnKF and weighted EnKF generate comparable results, which capture all the geologic structures in the reference. The similarities of these matched results to the reference are closer to the results we got in Figures 6-11 through 6-27. This indicates that if the initial ensemble members embrace the correct geostatistic information such as semivariogram, more accurate permeability results can be expected, in turn improving the future development decision. However, it is a paradox since the accurate knowledge of reservoir geologic information is impossible at the beginning. The impact of the approximation of geologic information on history matching deserves further investigation.

Another finding is that we clearly see that the weighted EnKF in each layer has better convergence to the reference than the conventional EnKF at the end of each

assimilation time. It is very encouraging that the permeability from the weighted EnKF is comparable to the reference, not only the overall geological structures but also the locations of the high and low permeability areas. The group results in Figures 6-47 to 6-63 again confirm the advantage of the weighted EnKF over the conventional EnKF.

In the above studies, the 200 initial realizations were conditioned by permeability data in 17 well locations. Another scenario was implemented where the initial realizations were not conditioned. In this case we used the average permeability of 1330mD. The results for the twelfth layer are plotted in Figure 6-64. Both simulations used the weighted EnKF. Although a uniform average realization was used at initial time, the permeability map at the end of 800 days agrees well with the reference map and captures the main structure. However, the results for the simulation using conditioned well data display much better agreement with the reference map.

Investigating the twelfth-layer maps in Figure 6-58, we notice that the structures are well recovered in both the weighted EnKF and the conventional EnKF after 800 days, as shown in Figure 6-58 (e) and (f). Also notice that at the end of 600 days, the result in the weighed EnKF has started to display a relatively low permeability area in the middle of the left hand side, shown in Figure 6-58 (d). Such a low permeability area is later confined and regressed to the shape of the reference, as shown in Figure 6-58 (e) and (f). The low permeability zones located at the right corner are captured in both methods and are very close to the reference. We can see that the methods are tuning the permeability layer with time and gradually approaching the reference. The final maps present good shapes. Figure 6-65 shows the root mean square of the matched results of this layer with the reference. Interestingly, before 600 days both methods are stable and almost unchanged. The value from the weighted EnKF even rises. After 600 days, the value in the conventional EnKF first drops and then rises. The final value is higher than the initial

value. This curve indicates that the matched results in the conventional EnKF are getting worse as time increases and the initial model before history matching is the best one, which is contrary to the observation we get in Figure 6-58. On the other hand, after a plateau period, the result from the weighted EnKF starts to drop. At the end of matching, the value is already very low. This is consistent with our observation in Figure 6-58, which implies that the weighted EnKF achieves good matches at the end. Bringing the curves of the conventional EnKF and the weighted EnKF together, we conclude that it is controversial to use the root mean square of the permeability field as a matching criterion. In particular, like our case, when high and low permeability channels appear, the results may not match the corresponding areas in the reference. Therefore, the value from the root mean square is still high or even higher than the initialization, though the whole structure of the field can be captured well.

6.5 SUMMARY

This chapter is a detailed application of the EnKF in a synthetic complex reservoir. A weighted EnKF is proposed and demonstrates the better performance than the conventional EnKF.

Through geostatistic software package, we first build the suitable semivariogram systematically by using the hard information from the sampled locations. The type of measurement data is then studied. In the case study, we find that it is not critical to include the bottomhole pressure data in injectors. The production rates are more sensitive to the overall change of the permeability structure. Similarly, we also compare the results from different sampling frequency. As we know, the knowledge of the sampling interval is very important to us since we do not want to miss any valuable information, while we also do not want to over-sample too much data considering the time and money costs. In our study, we use very dense sampling, (i.e., one day interval). The compared results

illustrate that as long as we have a long matching period, the 20-day sampling interval still gives us reasonable matching results which are almost identical to those from the high frequency sampling case at the late time though some structure fails to recover at the early beginning.

The matched permeability field is continuously updating as more measurements are available, even at very late time when the water cut is above 95 percent. The field pressure and water saturation maps show that both the conventional EnKF and the weighted EnKF give an amazing match of the field pressure and water saturation at the end of each assimilation step.

Well production history matching curves demonstrate that despite some reported divergence, both EnKF methods keep improving even at late time. The well production is matched reasonably even though the permeability field maps are somehow different from the references. Their non-uniqueness may cause this phenomenon. Another reason is that the inaccurate reservoir geological structure such as the semivariogram leads to the visual drift away of the permeability field, which is confirmed in the other case study in this chapter.

The weighted EnKF generates better matches for oil and water production rates in the producer and give closer cumulative oil recovery in terms of original oil in place. The recovery forecasting from the weighted EnKF is also better than that from the conventional EnKF. Additionally, we notice that the weighted EnKF demonstrates faster convergence at the early assimilation time periods.

Further, two scenarios were studied where geological information such as semivariogram is known and used in the initial ensemble generation. Better matching structures are obtained in both the conventional EnKF and the weighted EnKF methods, if the semivariogram is known, than those in which the semivariogram is unknown.

Again, the weighted EnKF exhibits a performance superior to the conventional EnKF. This means that the initial geologic information is important to the EnKF performance. However, to what extent such impact on the final result is hard to measure.

Another interesting finding is that when the high and low permeability channels appear, the root mean square is not sufficient to calibrate the matching performance. Production history, such as oil and water rates, can be deemed the most effective approach for the matching quantification.

Table 6-1 Producer locations and corresponding permeabilities

Well	1	2	3	4	5	6	7	8	9	10
X	8	15	18	15	8	5	5	18	18	5
Y	7	7	13	19	19	13	1	1	25	25
Layer 1	602.51	488.79	1134	850.12	1063.6	619.37	441.41	1183.3	861.5	982.72
Layer 2	828.79	1208.9	2718	4086.4	1214.8	2634.4	1004.8	1406	1555.5	1829.5
Layer 3	2141.1	232.04	1978.1	2292.7	484.15	1328.5	1414.9	299.44	3350.3	1768.8
Layer 4	2103.5	354.92	2106.7	366.95	615.18	488.26	1982.9	407.42	905.51	1960.4
Layer 5	393.98	412.75	185.13	478.74	343.84	243.04	480.88	467.25	325.24	1913.5
Layer 6	309.35	555.69	1151	1267	225.19	395.26	379.55	379.62	822.95	817.87
Layer 7	766.81	554.56	831.76	536.65	4416.3	460.41	434.96	911.9	1719	852.16
Layer 8	2876.2	2050.8	1071.8	448.8	553.48	2396.3	574.67	1001.4	1960.9	1127
Layer 9	3562.3	1129.4	508.81	466.37	5267.8	1340.8	596.86	6013.3	127.75	317.41
Layer 10	20054.2	10677.8	21527.2	867.38	3476.5	921.22	8608.8	17005.4	25122.5	869.4
Layer 11	25796.5	8795.5	1516.7	213.72	531.63	17625	20684.5	994.36	785.71	17588.6
Layer 12	15214.2	225.31	236.77	1729.2	19797	17697.3	9894.4	20667.8	3929.2	9986.9
Layer 13	4562.3	8304.4	5086.2	628.69	2793	3257.3	15867.7	393.03	4118.5	19756.3
Layer 14	364.45	792.61	2220.2	3514.6	389.54	2712.9	726.32	391.12	3087.7	1476.8
Layer 15	115.41	2166.5	7298.3	3444.6	111.18	630.17	1469.7	365.93	449.61	4303.9
Layer 16	1303.2	2926.7	2766.7	1165.1	1130.6	676.68	3236.8	696.4	1370.5	575.7
Layer 17	305.79	1227.6	534.62	970.89	288.53	1412.2	1074.6	1014.7	119.72	1579.5

Table 6-2 Injector locations and corresponding permeabilities

Well	11	12	13	14	15	16	17
X	11	11	22	22	11	1	1
Y	13	1	7	19	25	19	7
Layer 1	478.9	569.4	484.4	1111.6	975	728.4	549.3
Layer 2	920.2	1153	523.1	1596.6	1883.9	2021	1758.4
Layer 3	1099.1	2228.1	1058.1	3318.6	1042.6	3951.3	294.9
Layer 4	295.7	1005.4	482.6	427.4	1013.2	1708.7	1778.3
Layer 5	266.3	316.4	353.9	296.7	485.8	1517	754
Layer 6	470.7	347.4	413.6	280.7	319.4	575.1	244.4
Layer 7	1176.5	200.7	1605.6	137.2	515.1	590.8	201.2
Layer 8	1357.6	2732.4	2184	324.9	529.8	1672.5	1122.3
Layer 9	800.8	823.6	14300.5	385.2	3161.6	4955.3	3081.3
Layer 10	17064.3	1256.2	2041.6	4475.4	3781.2	1031.6	6153.9
Layer 11	11516.3	10937.4	1140.9	219.8	19096.6	20420.4	19945.4
Layer 12	16054.1	315.3	139.7	2238.7	21172.9	17805.2	4756.2
Layer 13	1241.4	4809	232.7	875.5	1002.1	1329.9	4749.2
Layer 14	5555.7	740.6	1867.3	6635.7	443.9	726	618.7
Layer 15	3230.3	2501	1222	3569.3	639.3	3116.9	164.7
Layer 16	1172.8	1776	1690.4	492.4	448.9	1551.4	2413.3
Layer 17	1310.2	464.9	1142.2	259.2	148.5	818.2	1185.7

Table 6-3 Observation data during the simulation time

Time(days)	Oil Rate	Water Rate	Bottomhole Pressure	Injection Rate	Total
[0-150]	2×1	2×1	2×17	0	38
(150-240]	2×1	2×1	4×17	0	72
(240-600]	4×1	4×1	4×17	1	77
(600-1500]	10×1	10×1	0	0	20

Table 6-4 Different sampling intervals in two scenarios

Time (days)	Sampling Interval (days)	
	Scenario 1	Scenario 2
[0-150]	30	30
(150-240]	30	30
(240-600]	30	30
(600-610]	20	1
(610-700]		5
(700-800]		10
(800-1000]	50	50
(1000-1200]	200	200
(1200-1500]	300	300

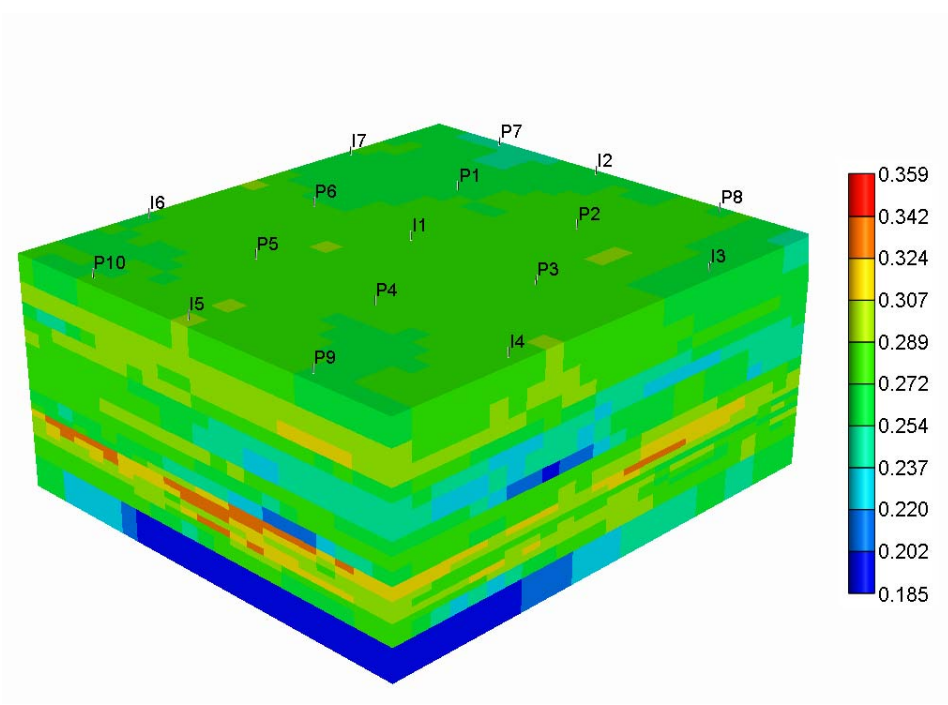


Figure 6-1: Three-dimensional view of reservoir porosity.

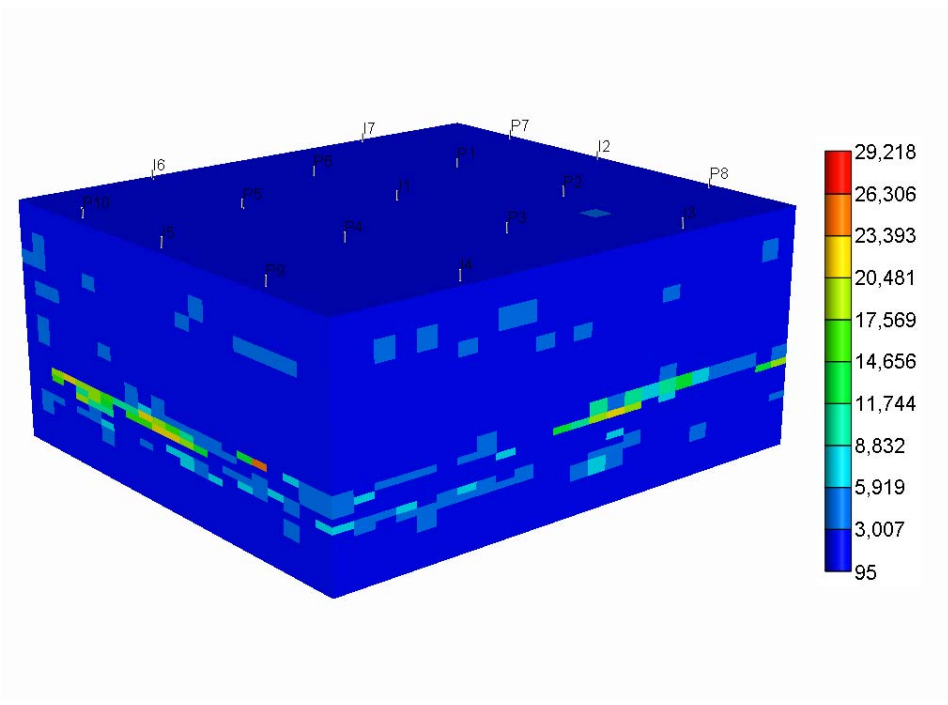


Figure 6-2: Three-dimensional view of reservoir permeability in X direction with the mD unit.

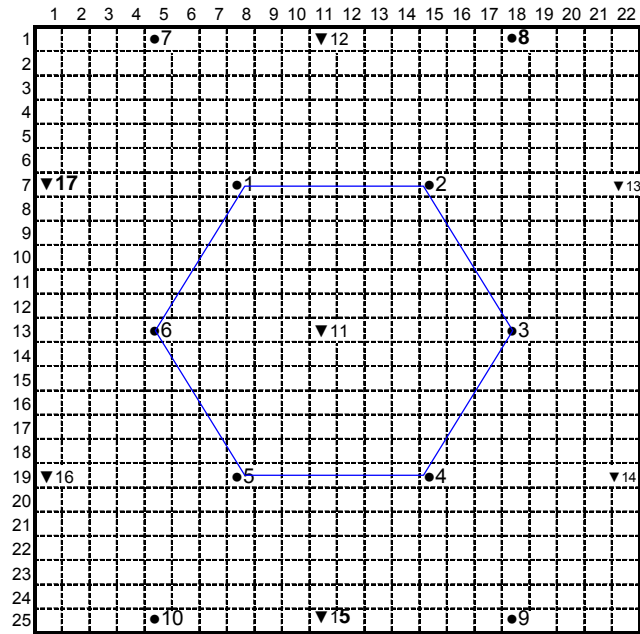


Figure 6-3: Areal view of well configurations, where ● denotes producers and ▼ represents injectors.

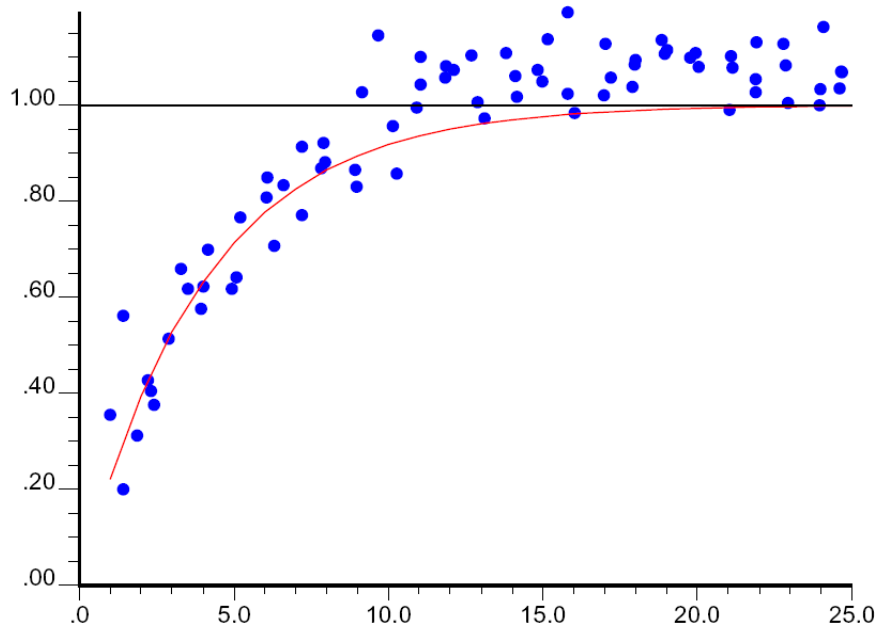
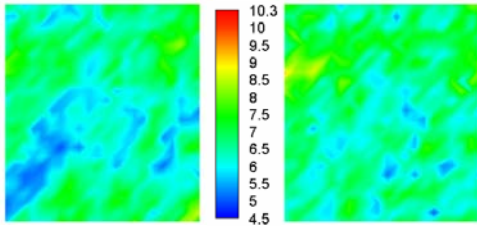
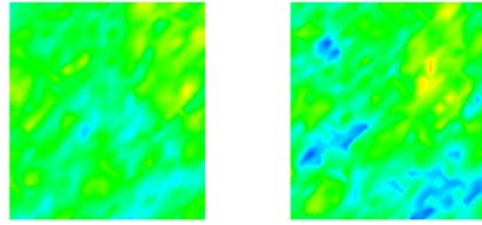


Figure 6-4: Experimental data (points) and the matched exponential semivariogram model (curve). The vertical scale is normalized and the horizontal unit is the number of grid sizes.

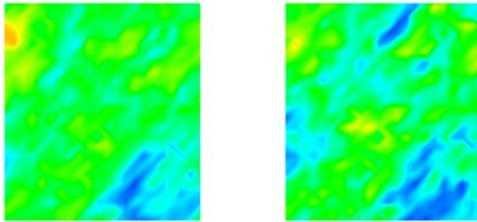
(a) First layer



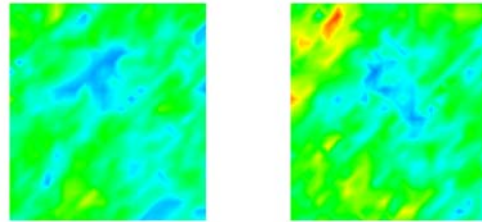
(b) Second layer



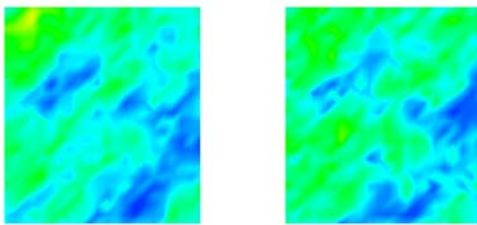
(c) Third layer



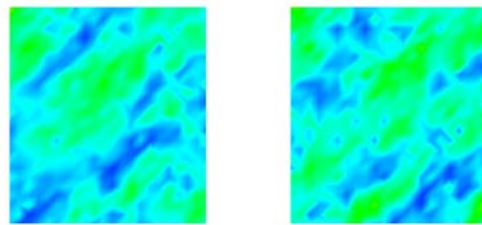
(d) Fourth layer



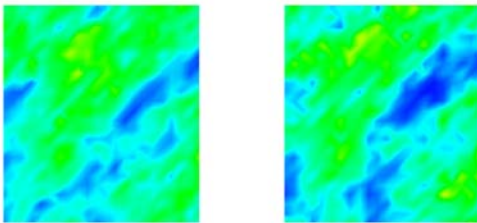
(e) Fifth layer



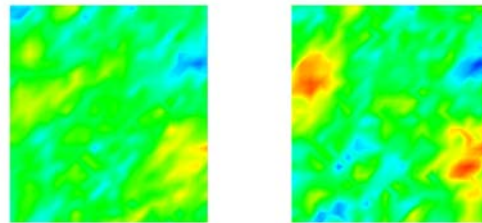
(f) Sixth layer



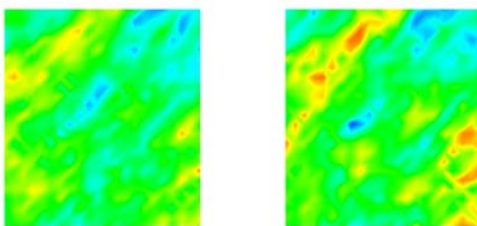
(g) Seventh layer



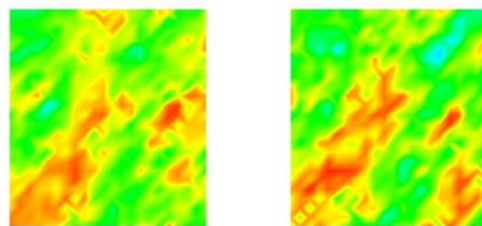
(h) Eighth layer



(i) Ninth layer



(j) Tenth layer



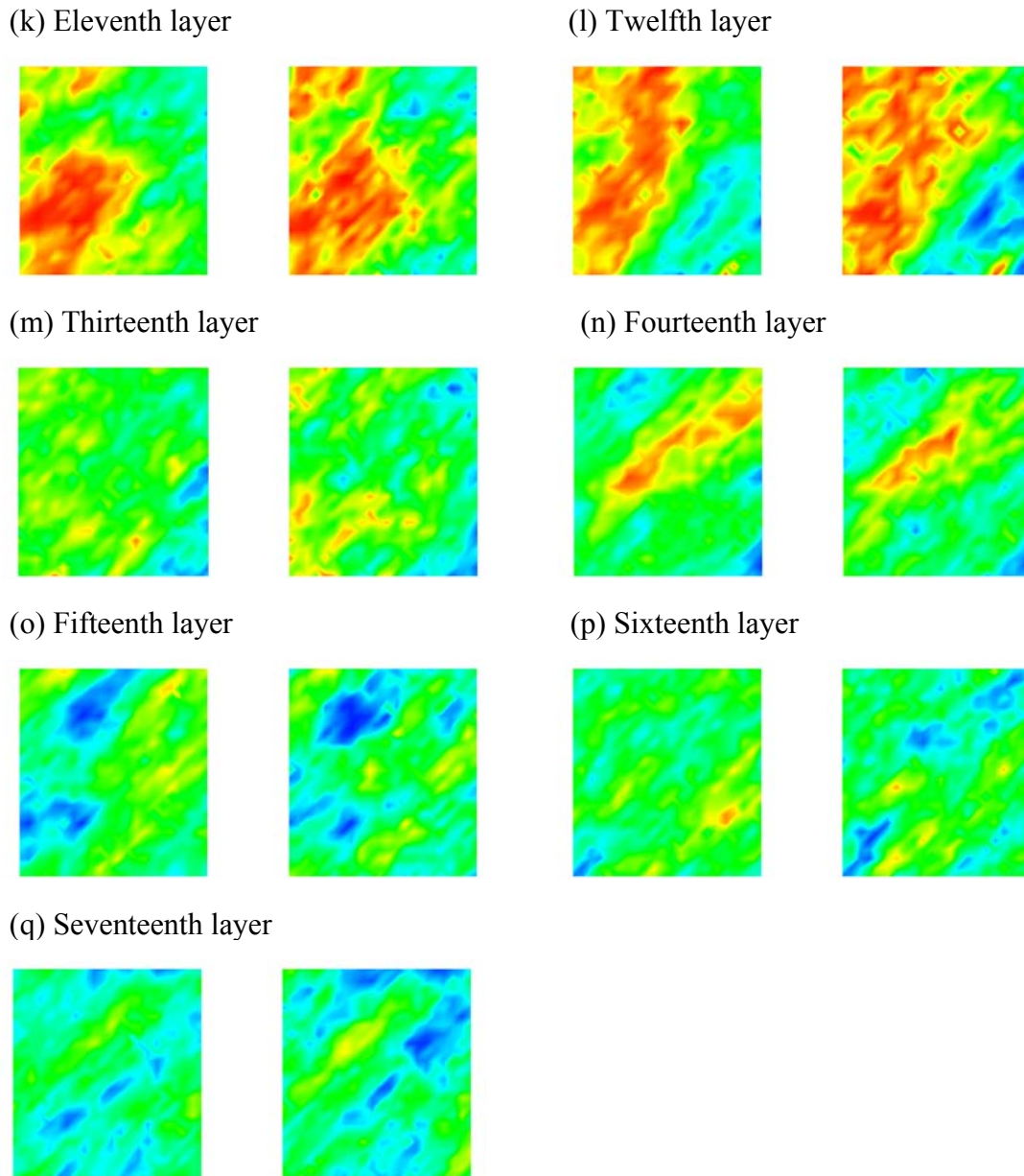
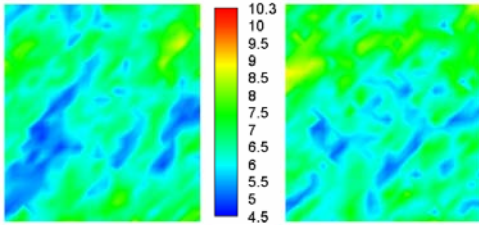
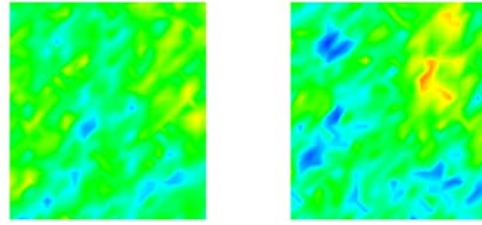


Figure 6-5: At the time of 620 days, logarithm permeability fields of averages of ensemble members from two measurement scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

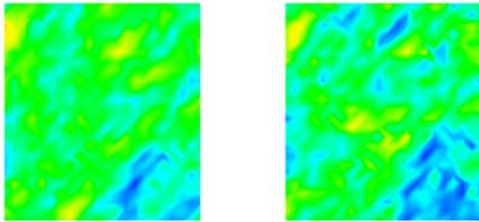
(a) First layer



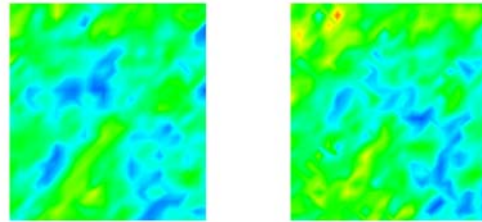
(b) Second layer



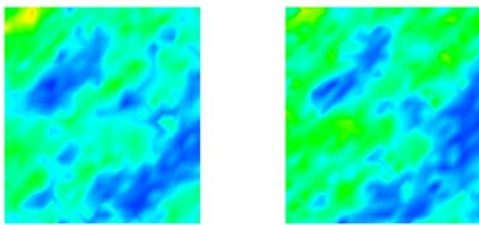
(c) Third layer



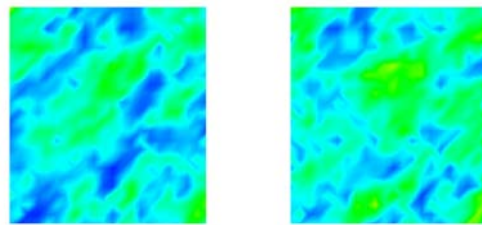
(d) Fourth layer



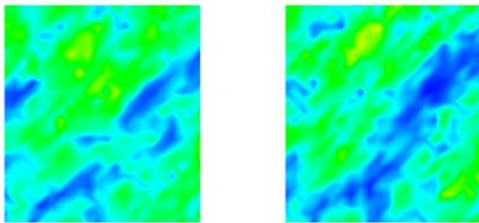
(e) Fifth layer



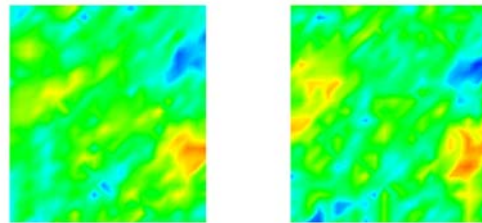
(f) Sixth layer



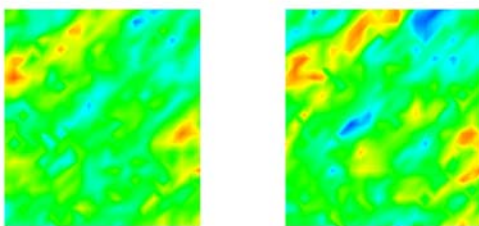
(g) Seventh layer



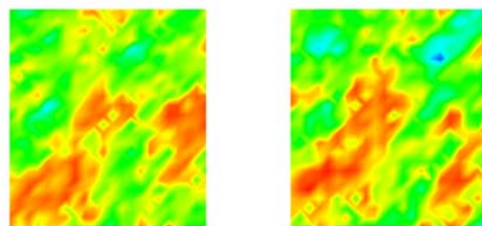
(h) Eighth layer



(i) Ninth layer



(j) Tenth layer



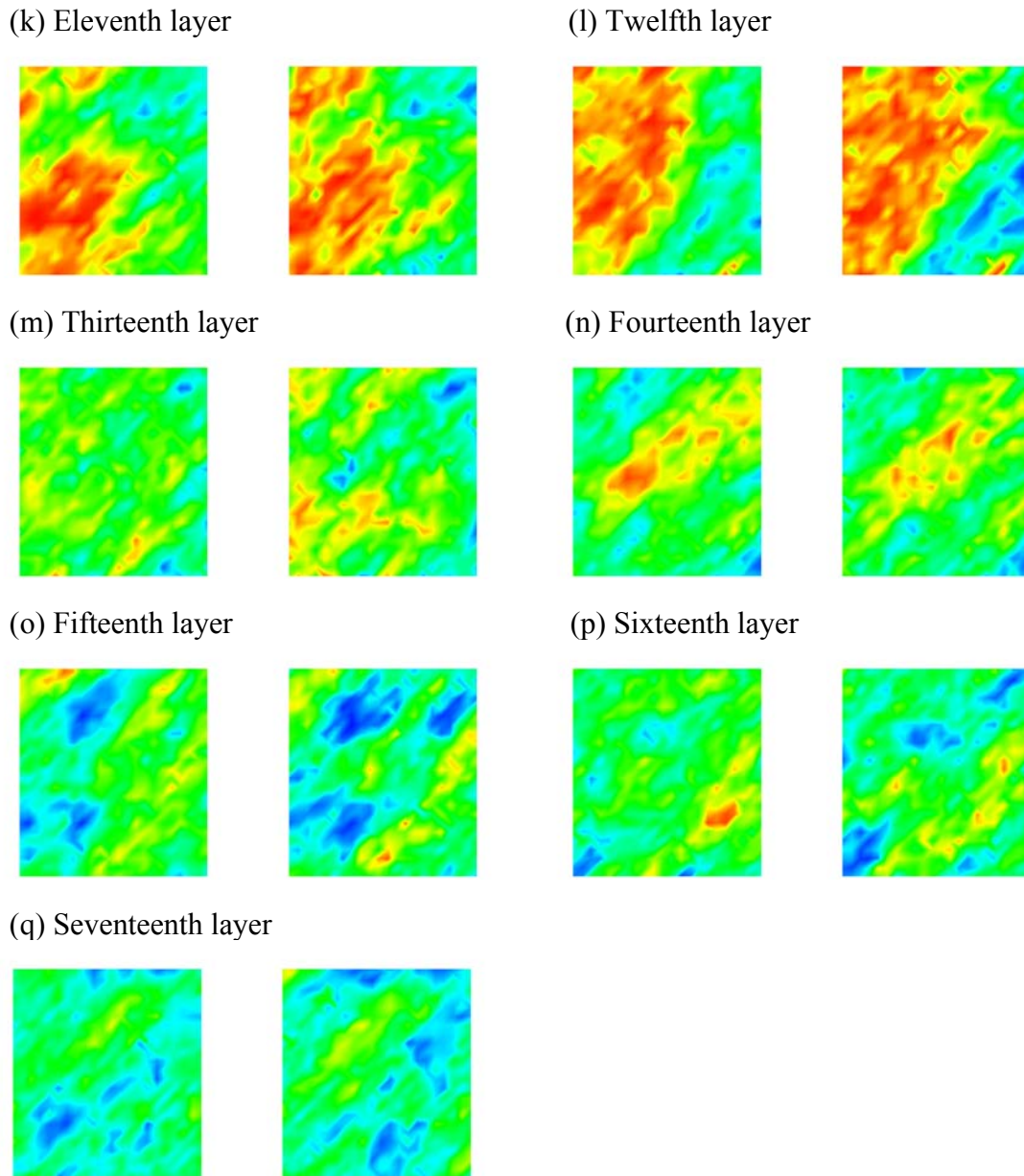
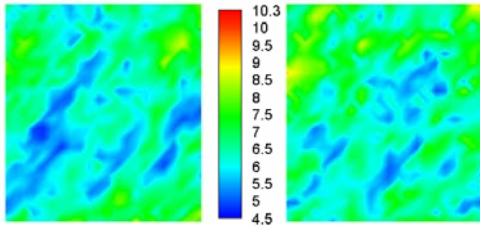
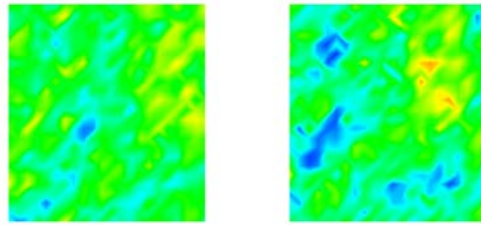


Figure 6-6: At the time of 640 days, logarithm permeability fields of averages of ensemble members from two measurement scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

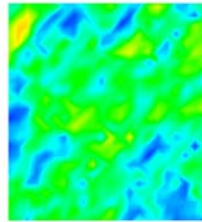
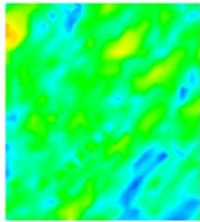
(a) First layer



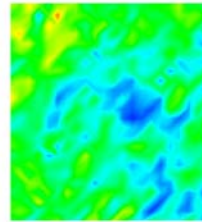
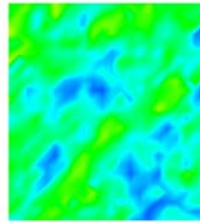
(b) Second layer



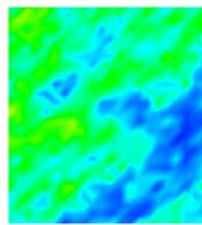
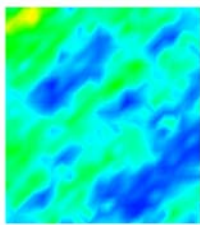
(c) Third layer



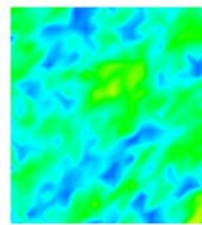
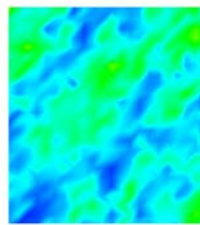
(d) Fourth layer



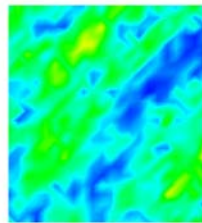
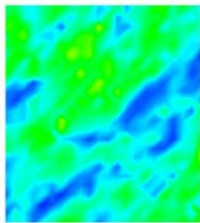
(e) Fifth layer



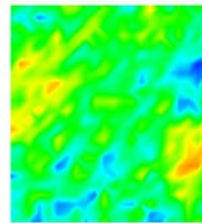
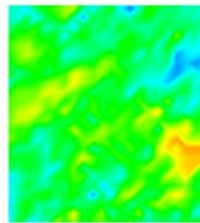
(f) Sixth layer



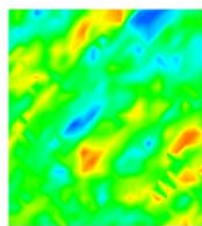
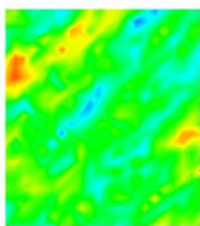
(g) Seventh layer



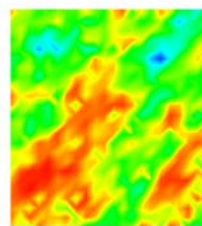
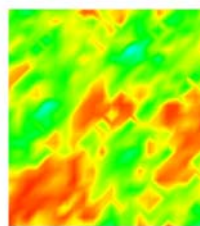
(h) Eighth layer



(i) Ninth layer



(j) Tenth layer



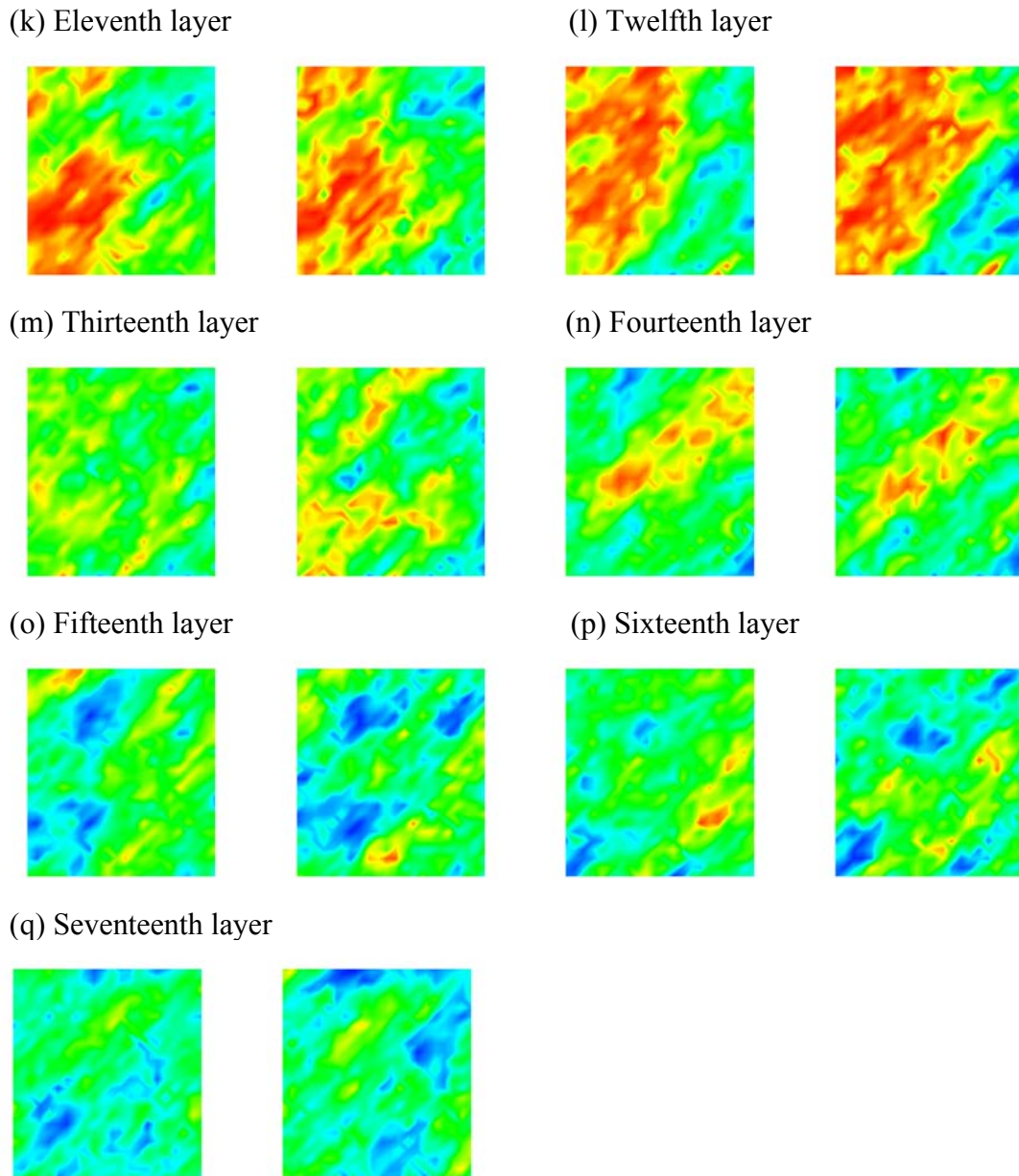
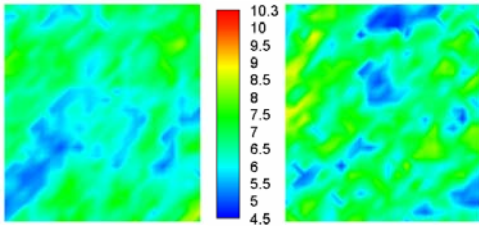
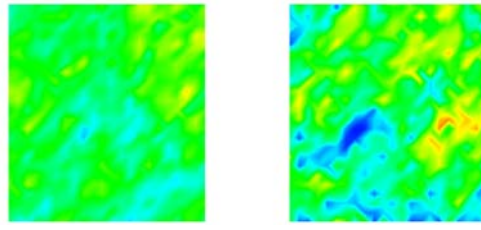


Figure 6-7: At the time of 660 days, logarithm permeability fields of averages of ensemble members from two measurement scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

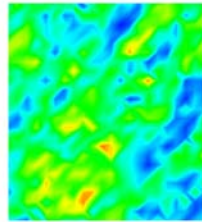
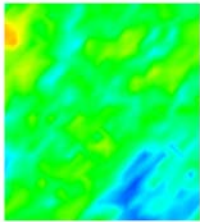
(a) First layer



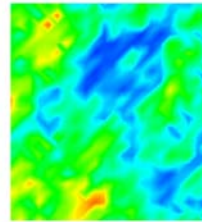
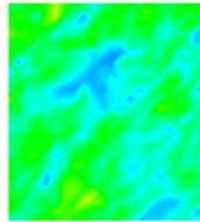
(b) Second layer



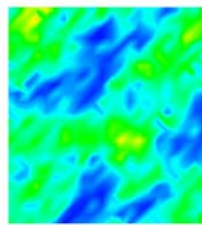
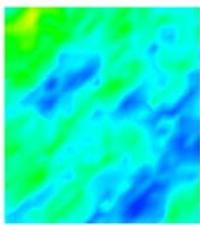
(c) Third layer



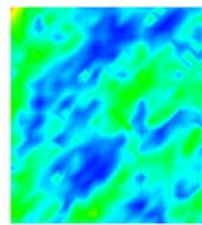
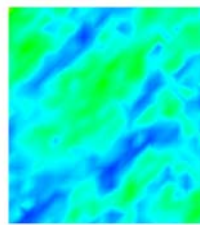
(d) Fourth layer



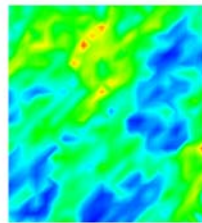
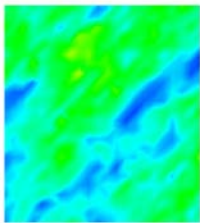
(e) Fifth layer



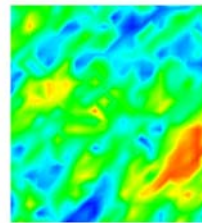
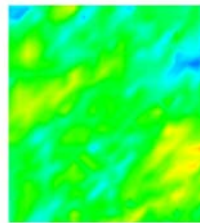
(f) Sixth layer



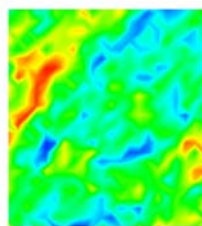
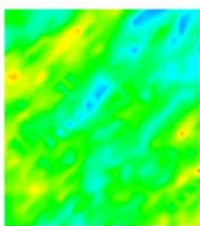
(g) Seventh layer



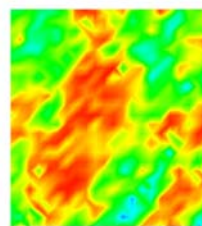
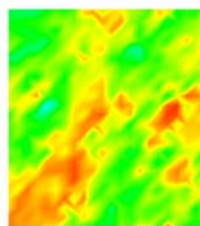
(h) Eighth layer



(i) Ninth layer



(j) Tenth layer



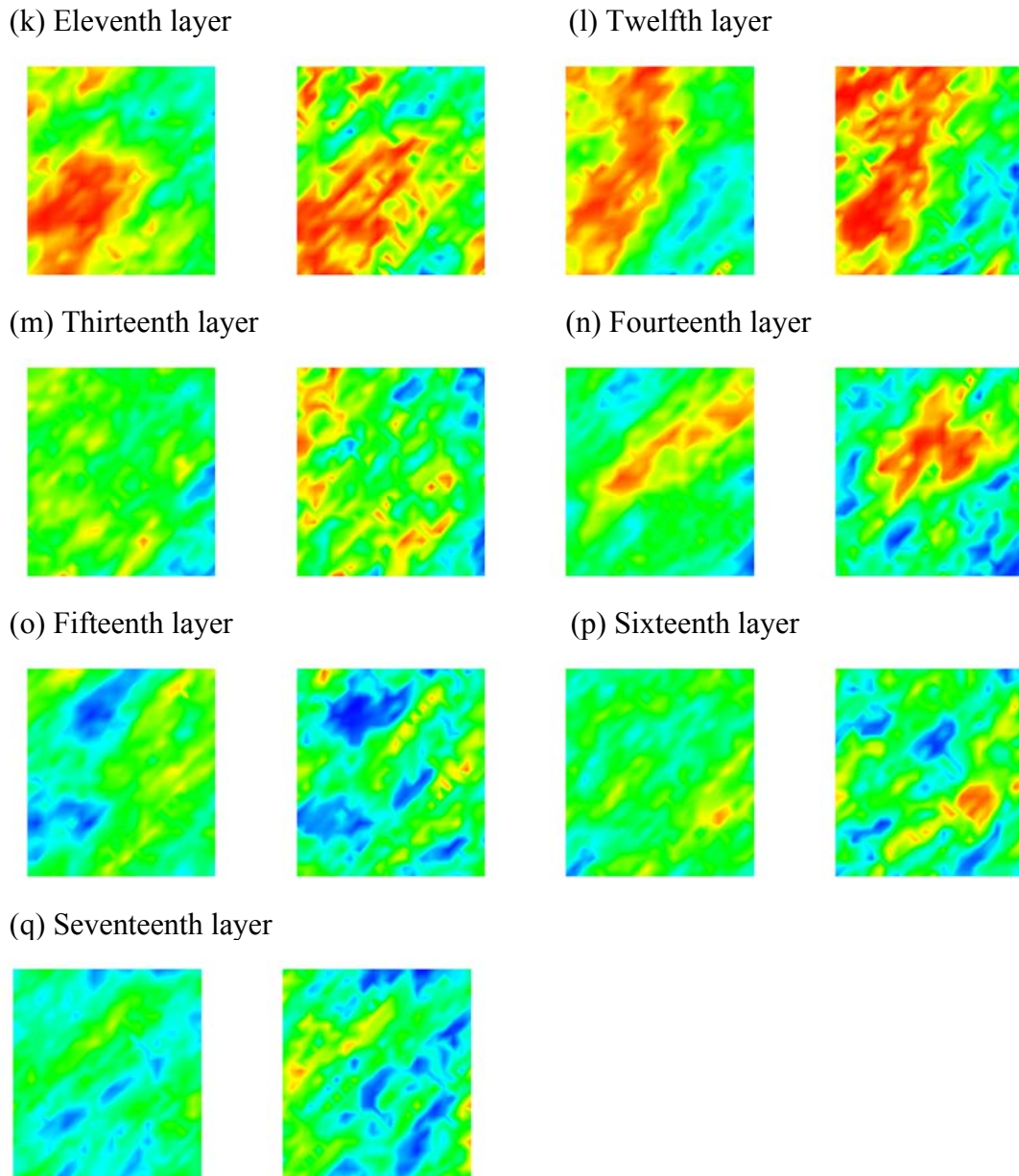
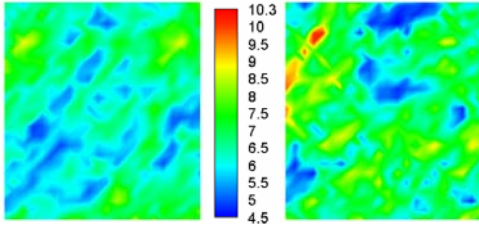
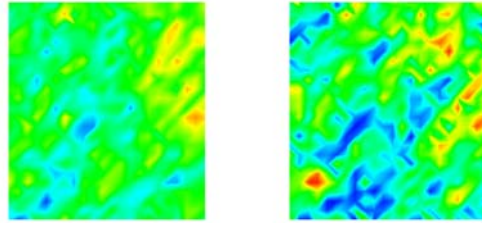


Figure 6-8: At the time of 620 days, logarithm permeability fields of averages of ensemble members from two sampling scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

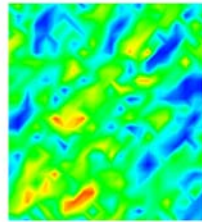
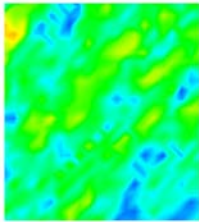
(a) First layer



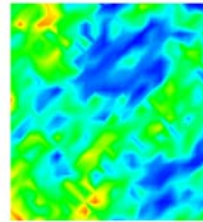
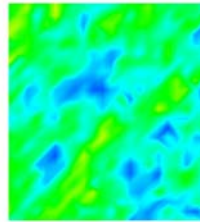
(b) Second layer



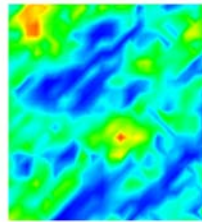
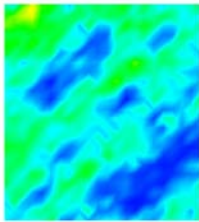
(c) Third layer



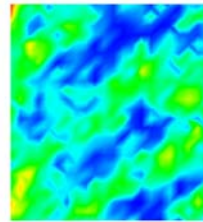
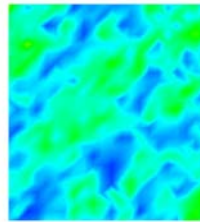
(d) Fourth layer



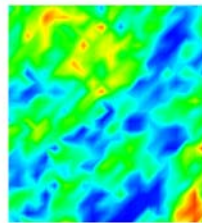
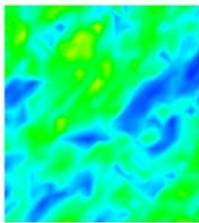
(e) Fifth layer



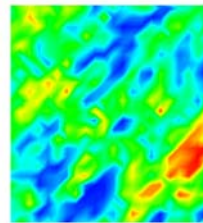
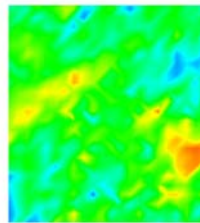
(f) Sixth layer



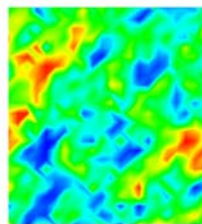
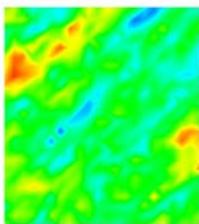
(g) Seventh layer



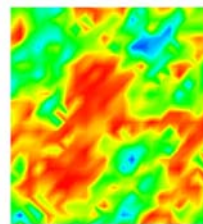
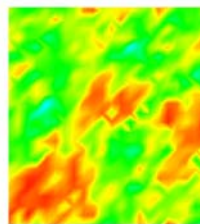
(h) Eighth layer



(i) Ninth layer



(j) Tenth layer



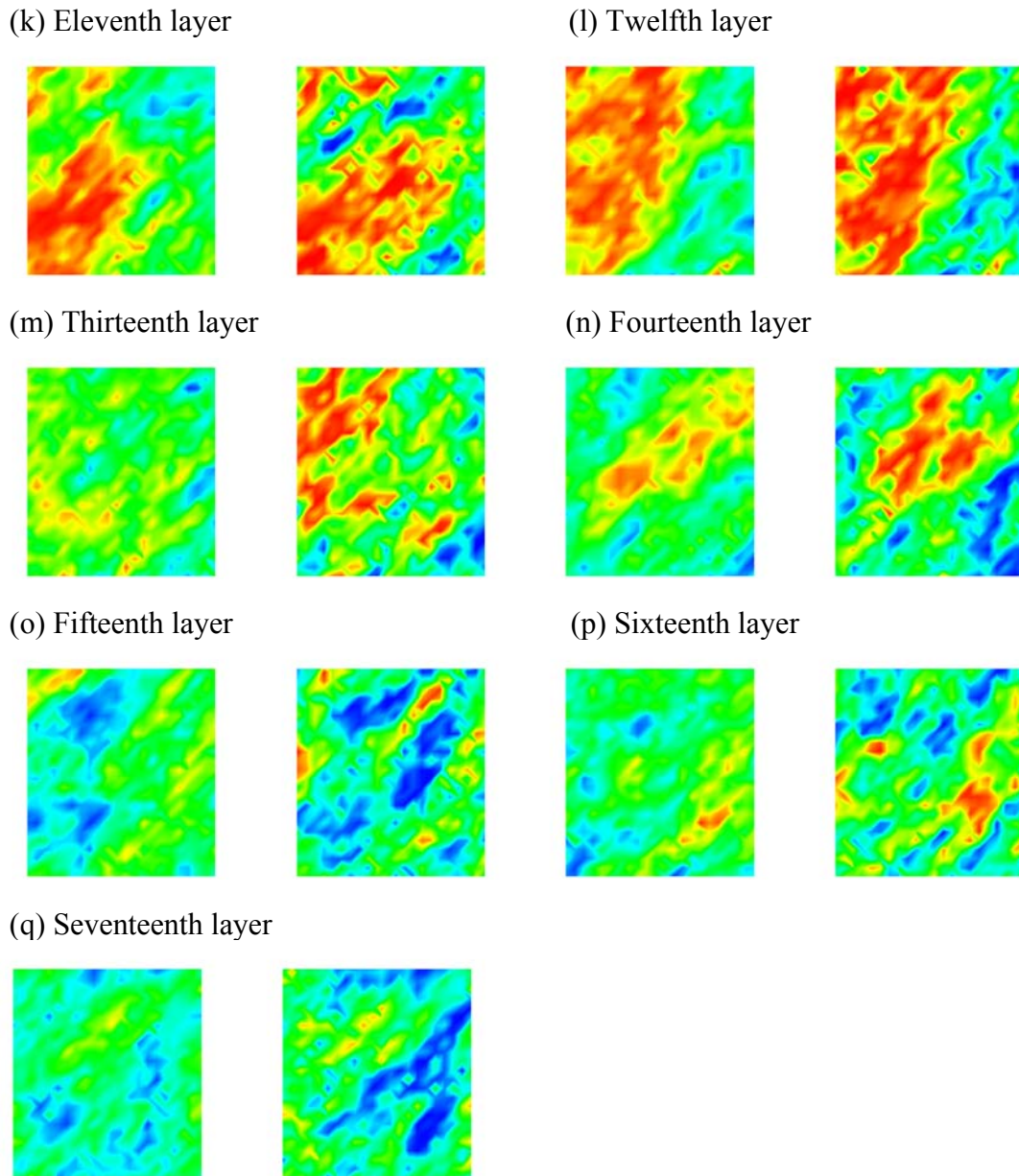
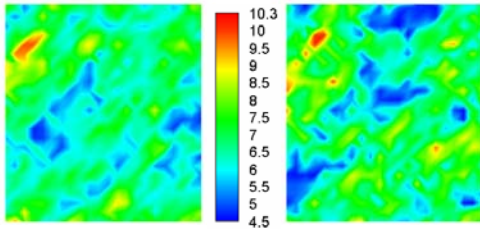
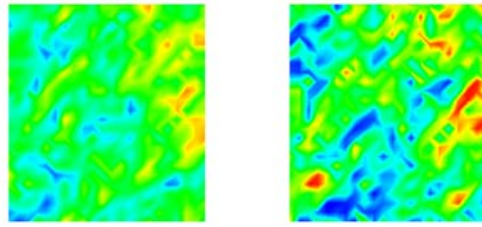


Figure 6-9: At the time of 700 days, logarithm permeability fields of averages of ensemble members from two sampling scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

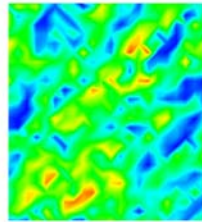
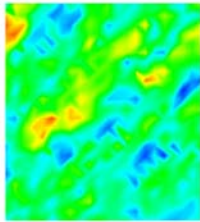
(a) First layer



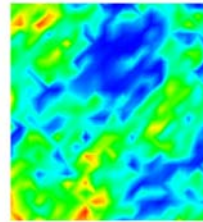
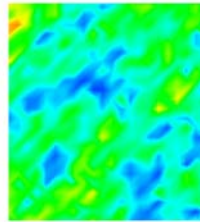
(b) Second layer



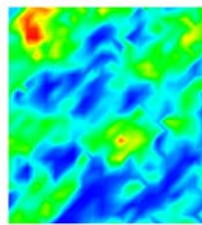
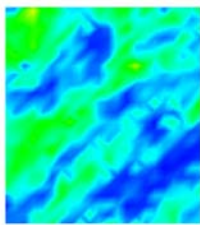
(c) Third layer



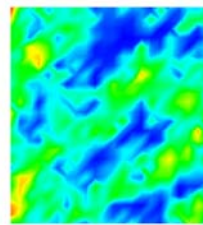
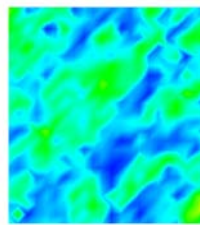
(d) Fourth layer



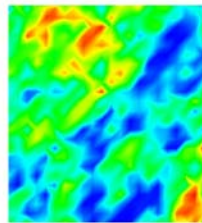
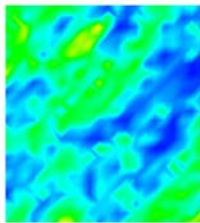
(e) Fifth layer



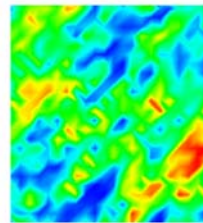
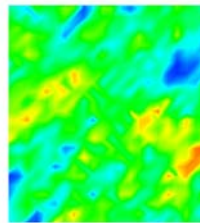
(f) Sixth layer



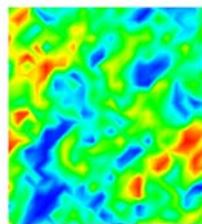
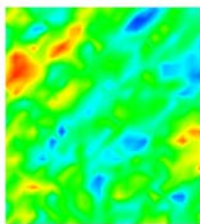
(g) Seventh layer



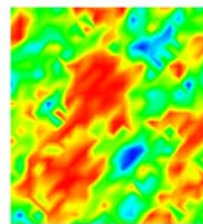
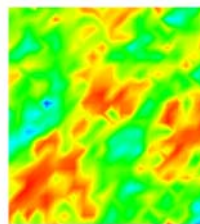
(h) Eighth layer



(i) Ninth layer



(j) Tenth layer



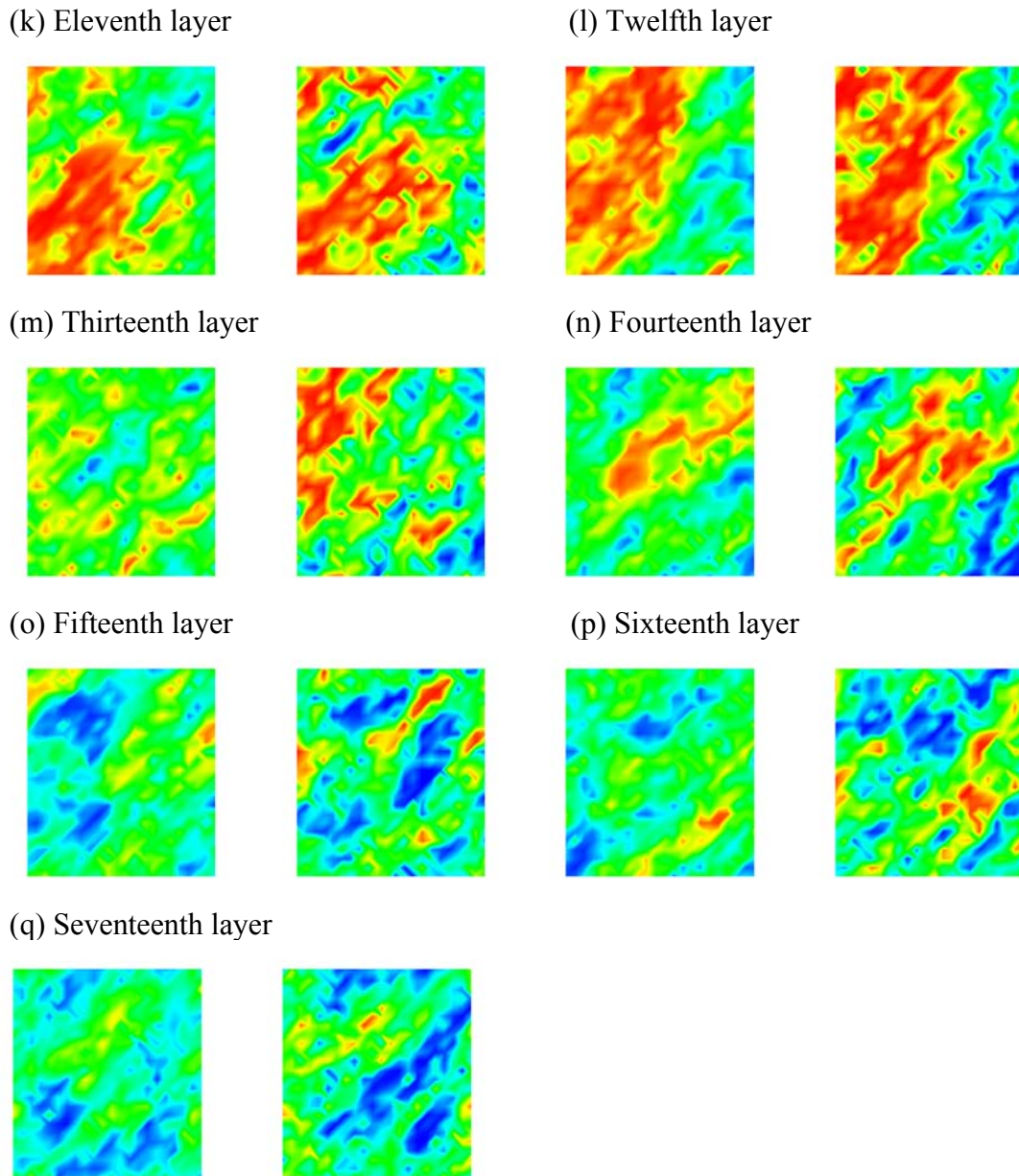


Figure 6-10: At the time of 800 days, logarithm permeability fields of averages of ensemble members from two sampling scenarios by using direct inverse assimilation method. Left hand side is results from 20 measurements while right hand side is results from 130 measurements. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

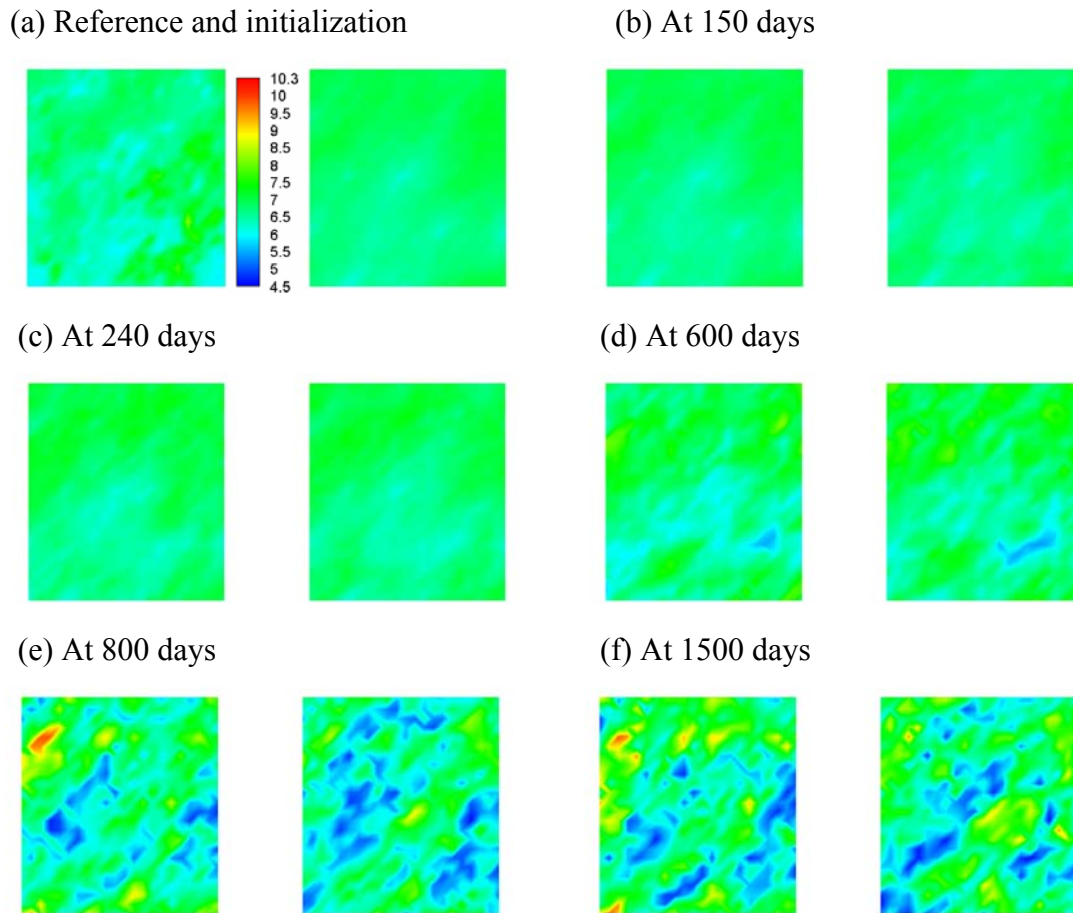


Figure 6-11: First-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

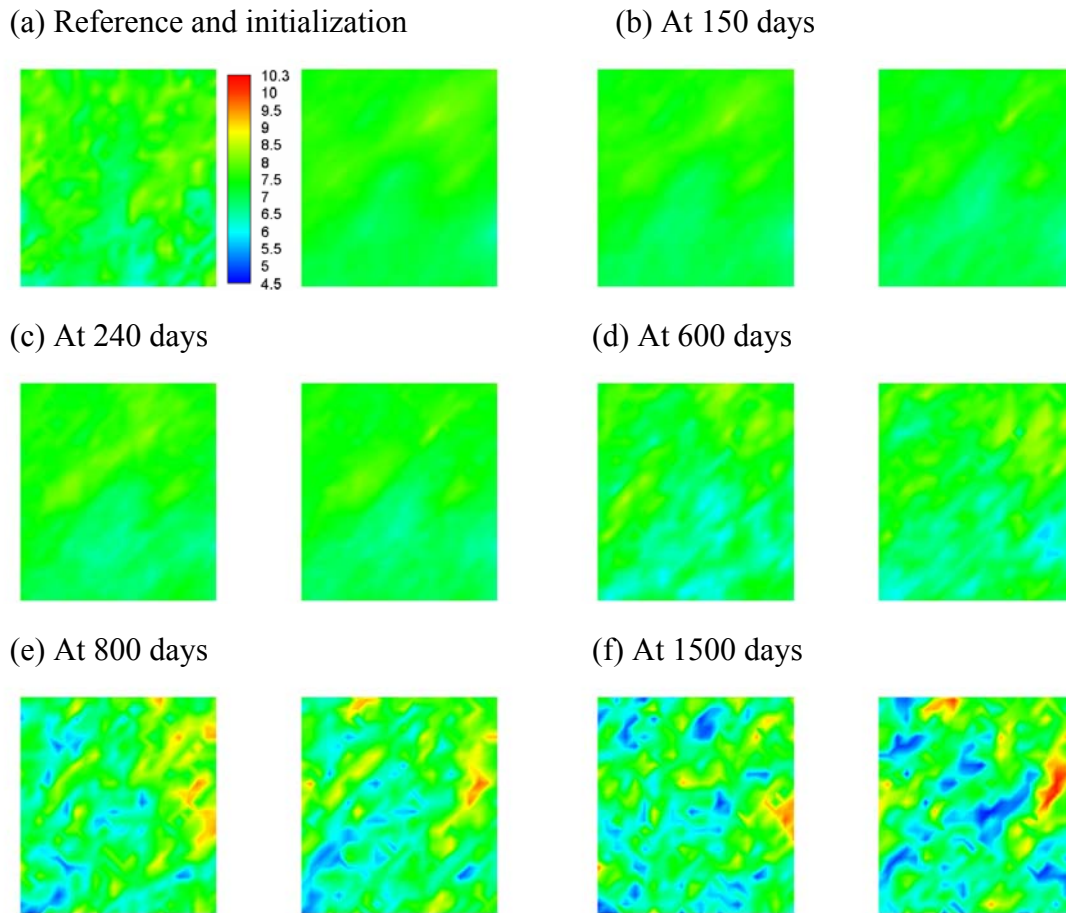


Figure 6-12: Second-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

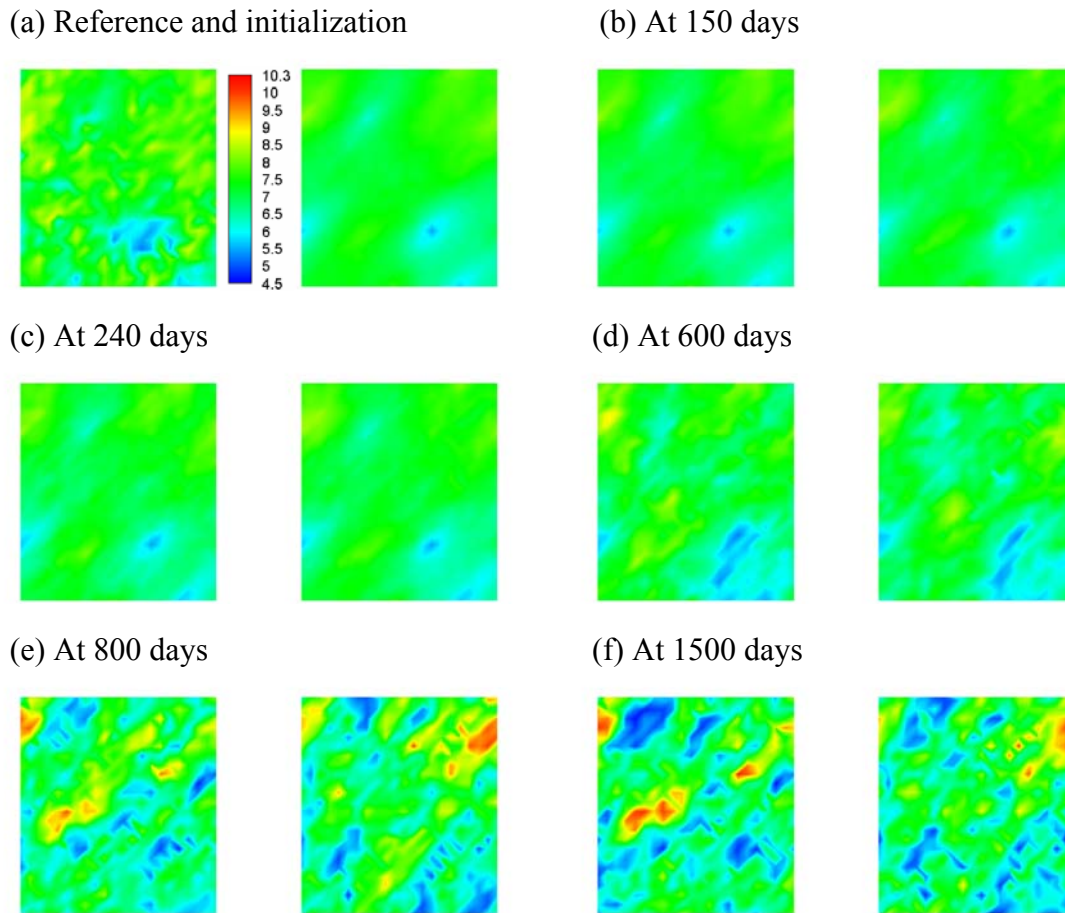


Figure 6-13: Third-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

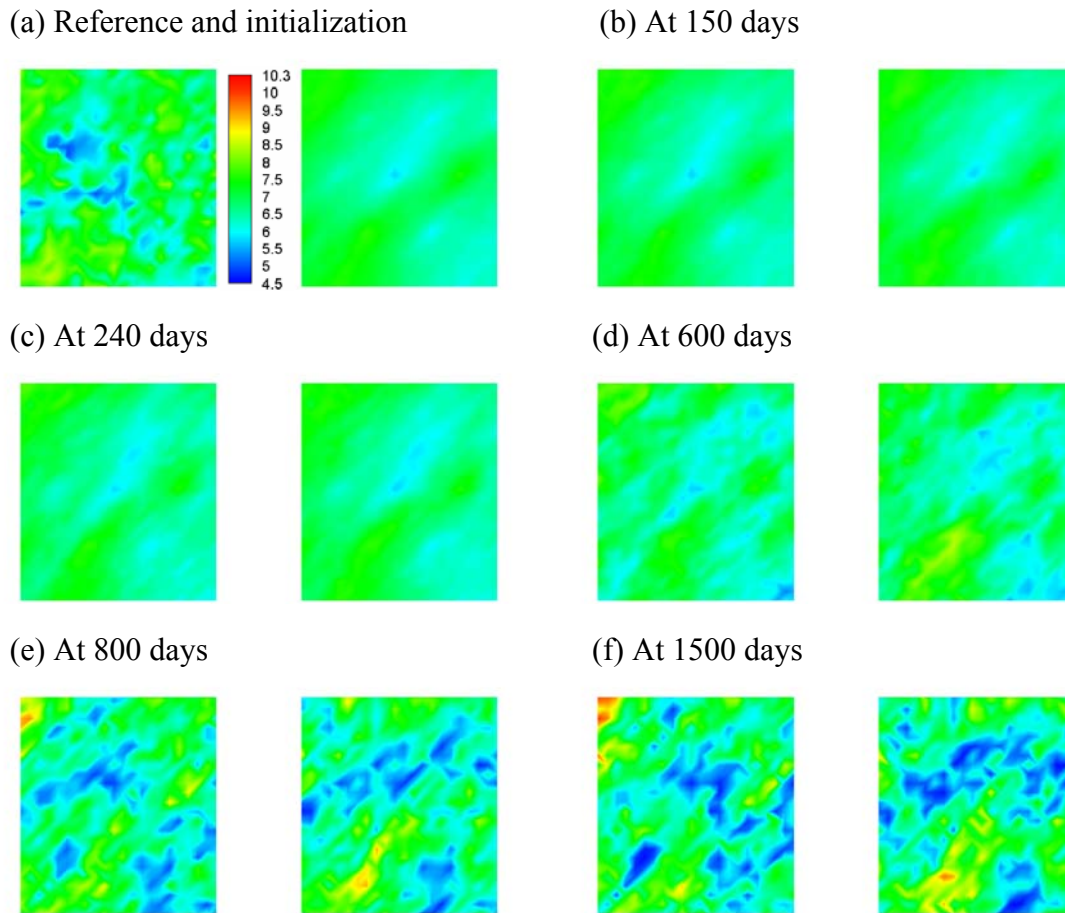


Figure 6-14: Fourth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

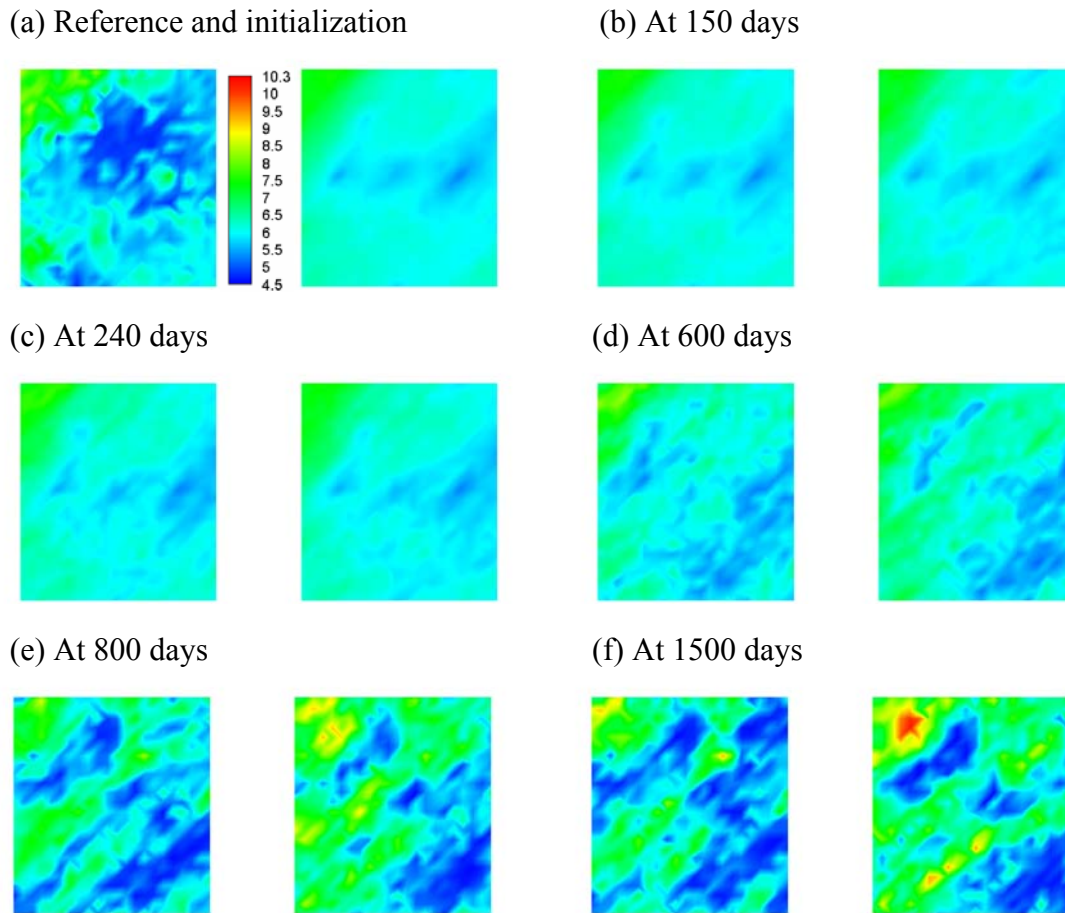


Figure 6-15: Fifth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

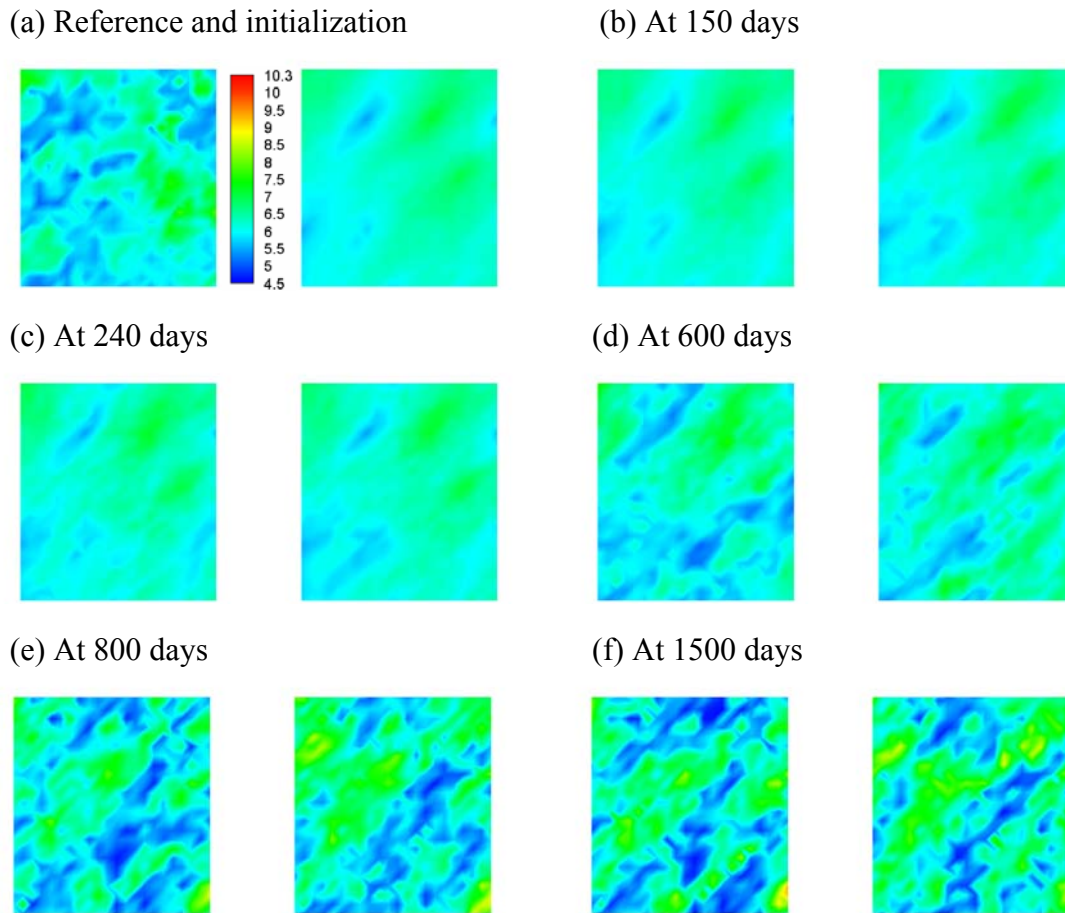


Figure 6-16: Sixth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

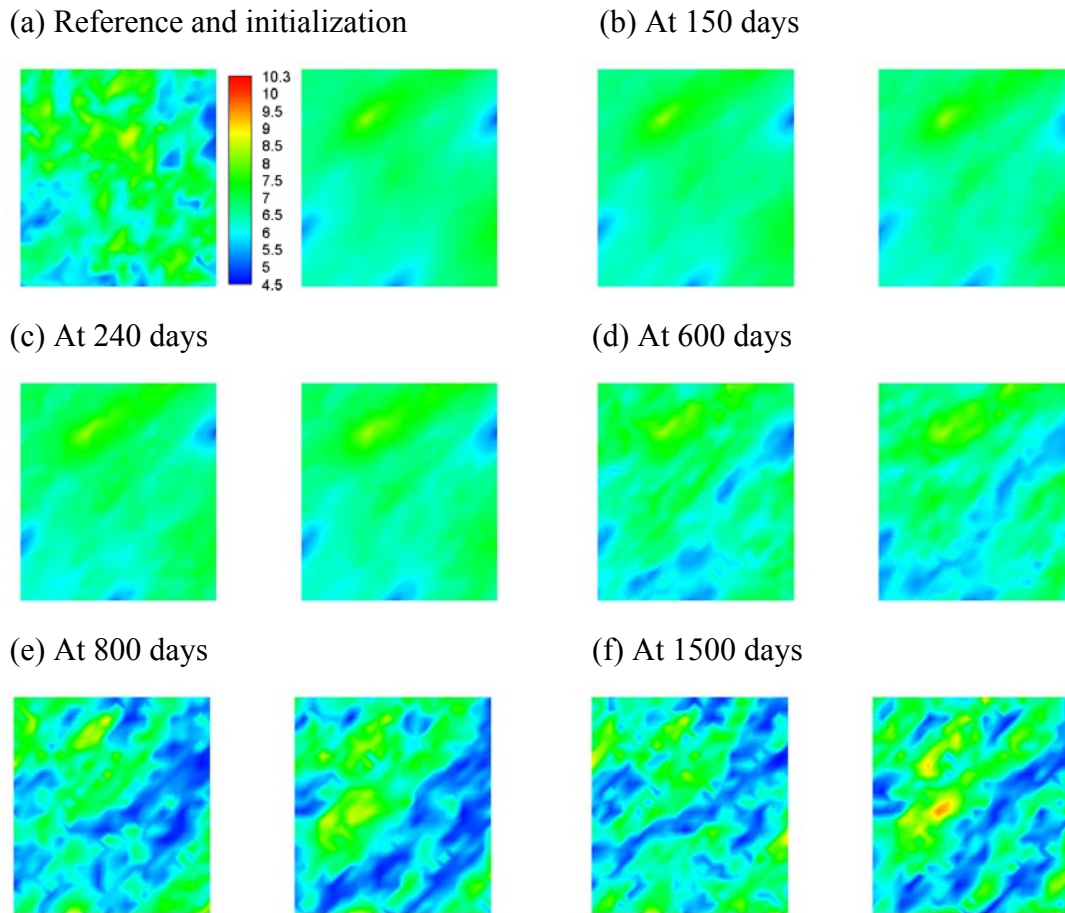


Figure 6-17: Seventh-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

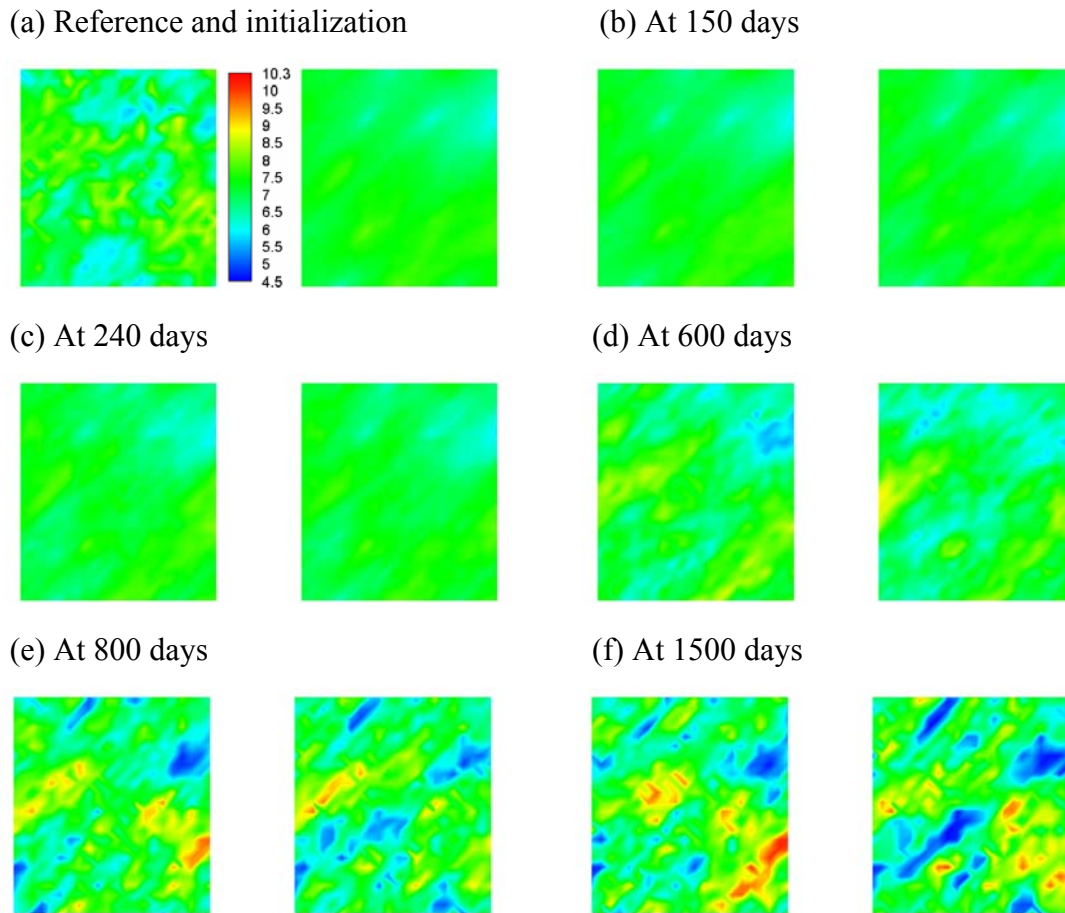


Figure 6-18: Eighth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

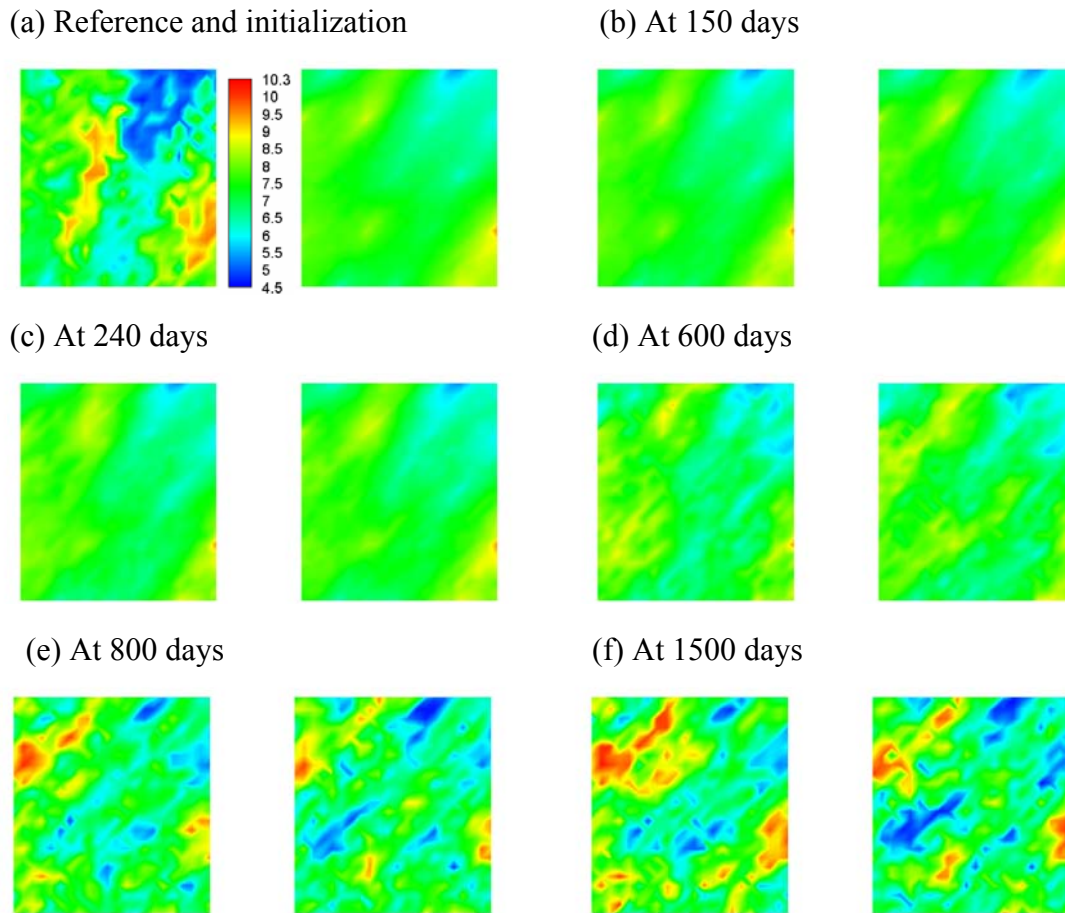


Figure 6-19: Ninth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

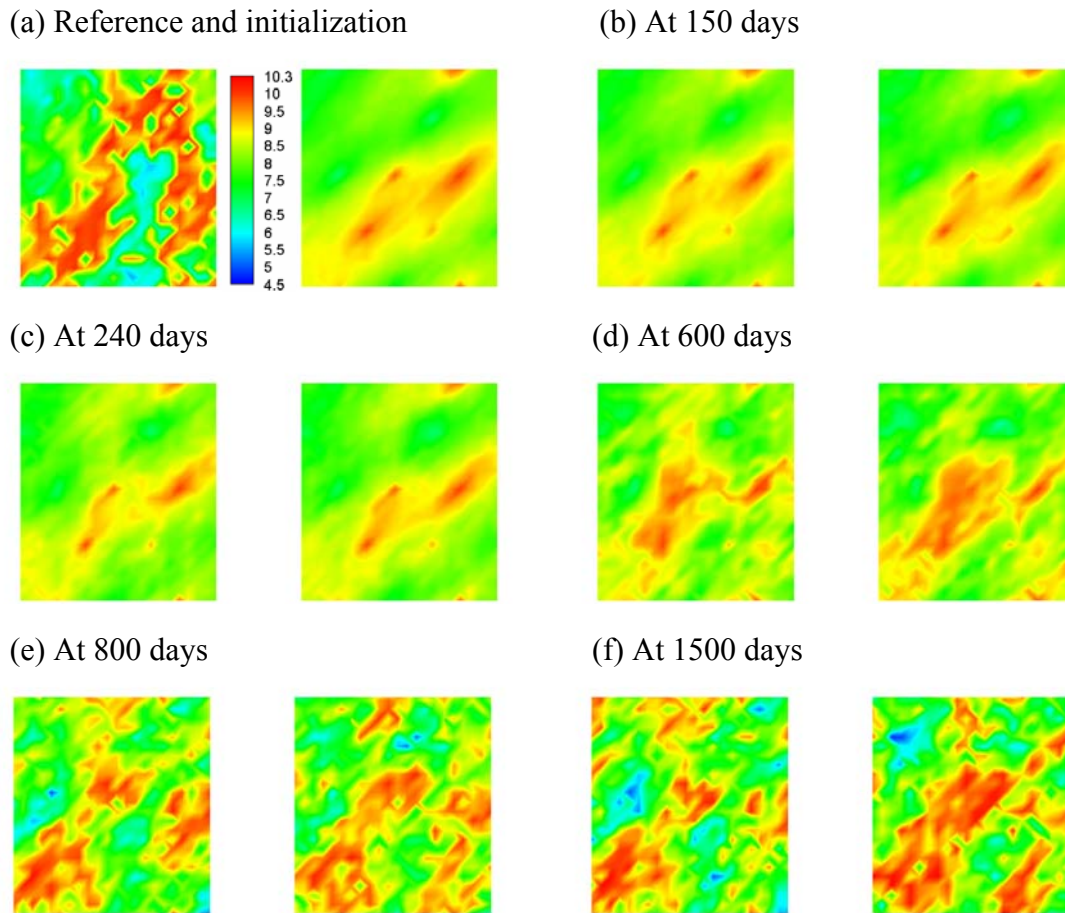


Figure 6-20: Tenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

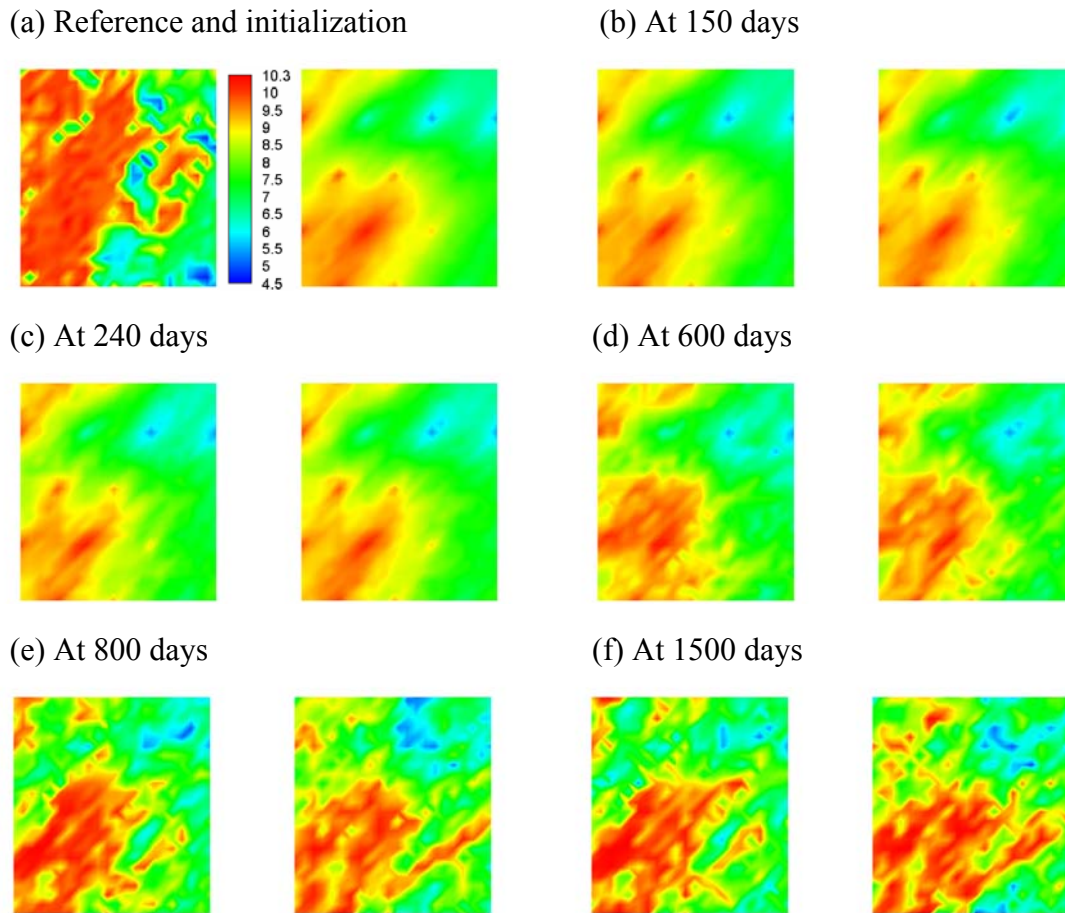


Figure 6-21: Eleventh-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

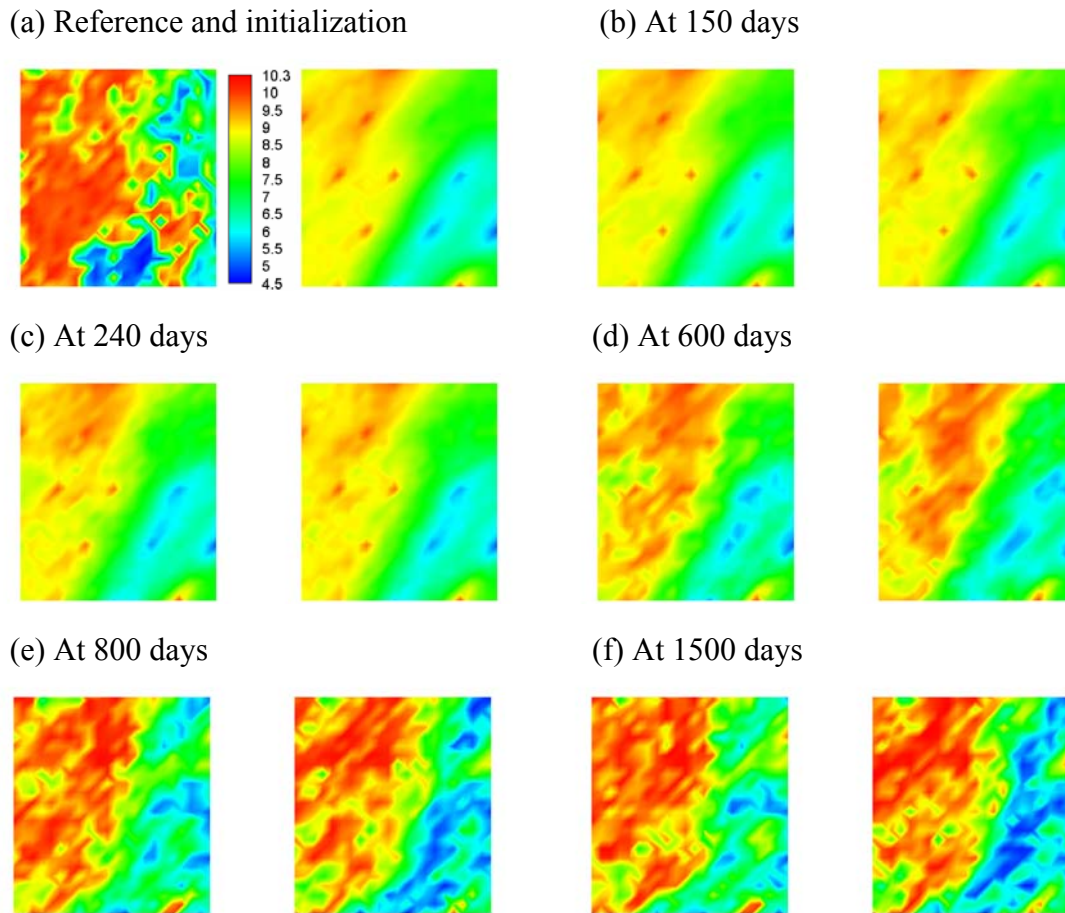


Figure 6-22: Twelfth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

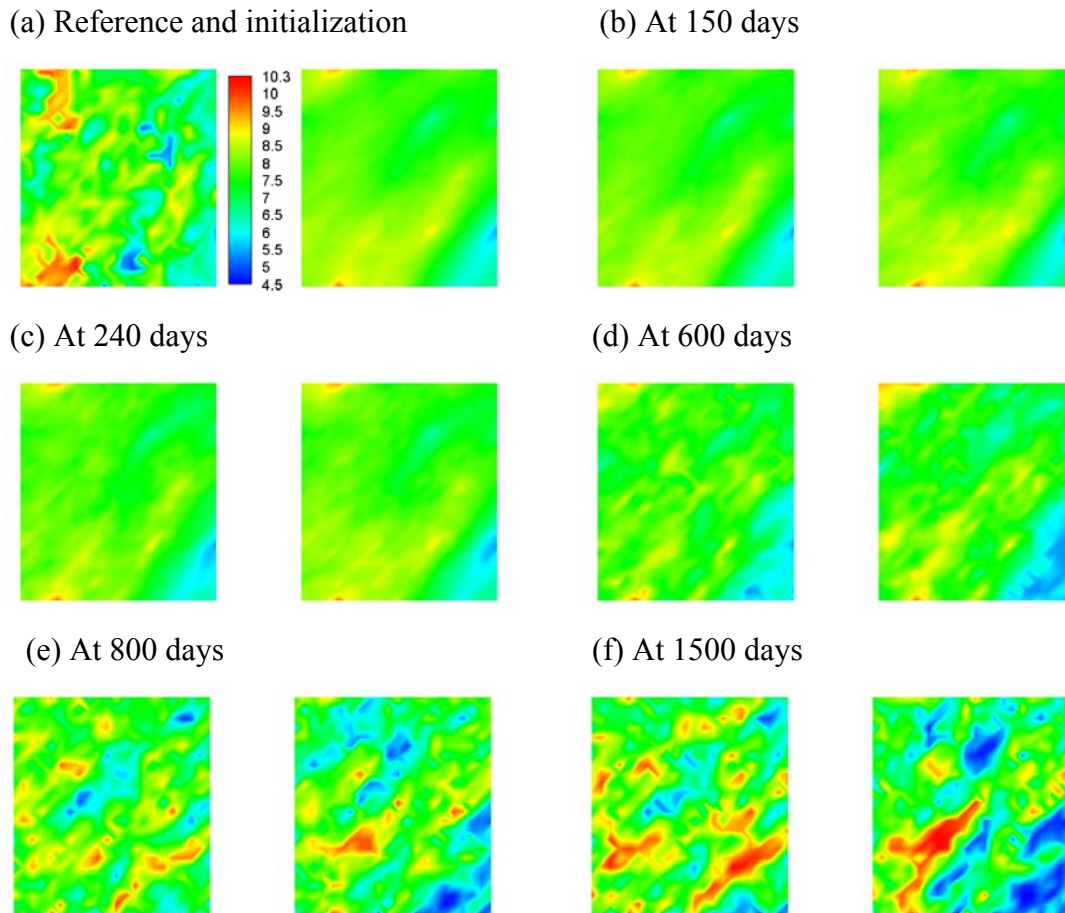


Figure 6-23: Thirteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

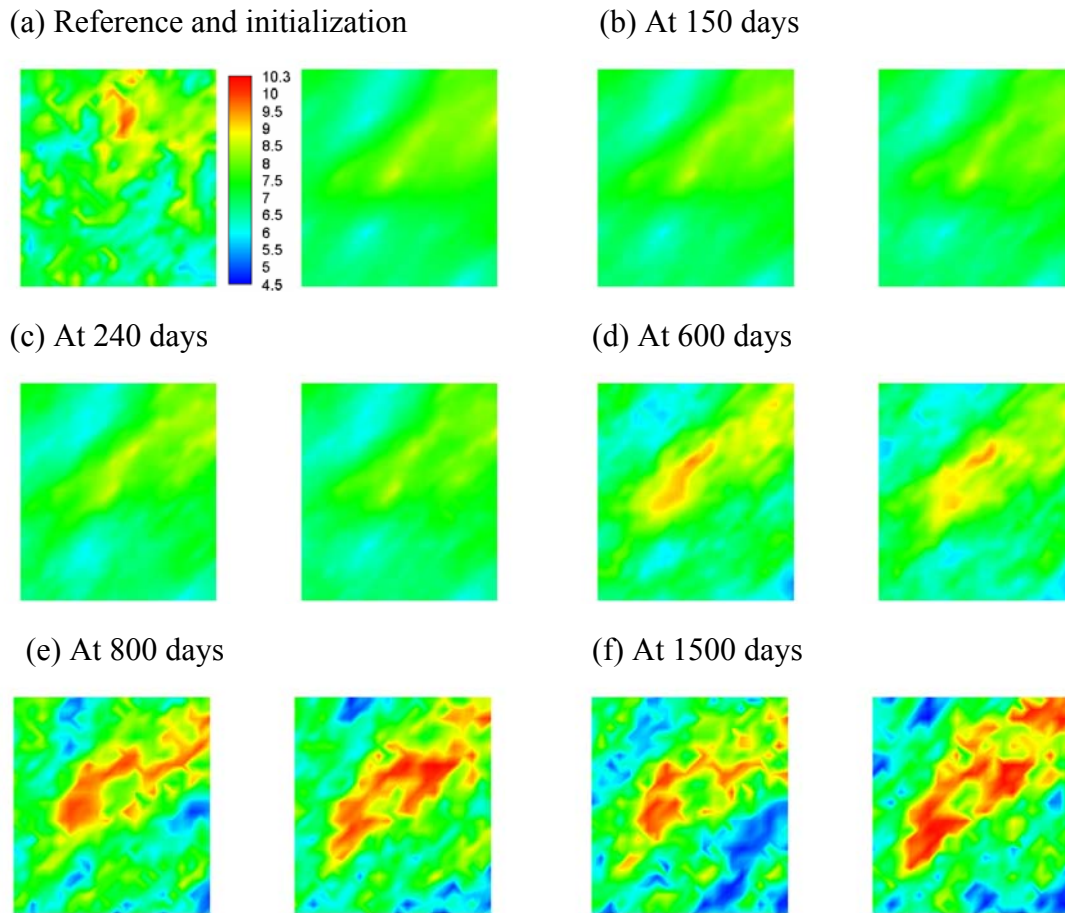


Figure 6-24: Fourteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

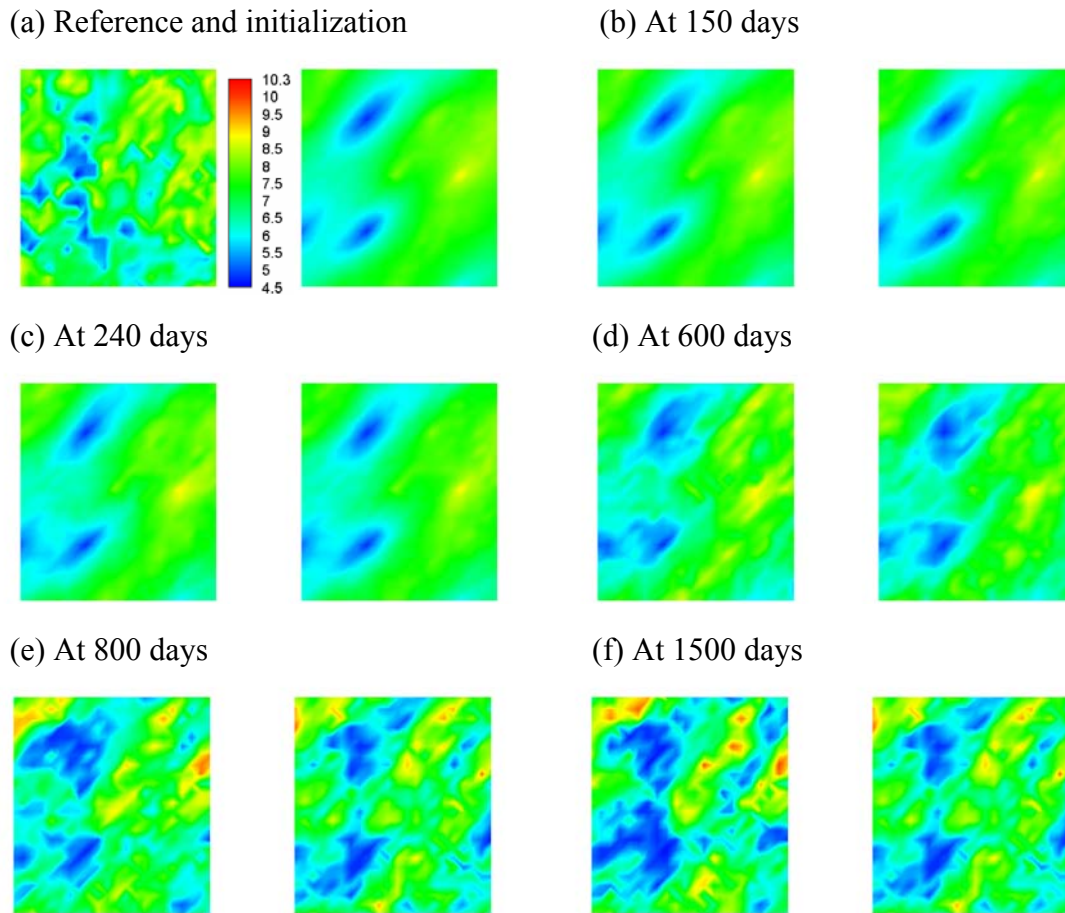


Figure 6-25: Fifteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

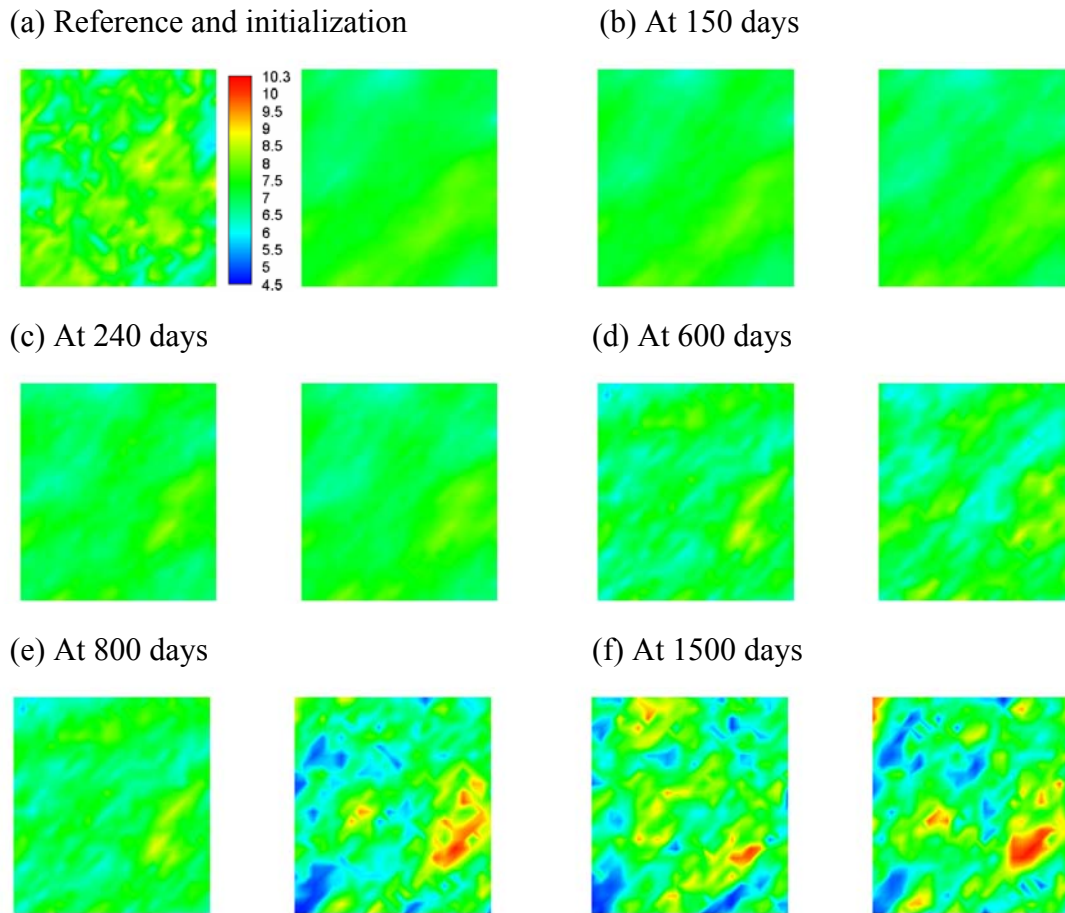


Figure 6-26: Sixteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

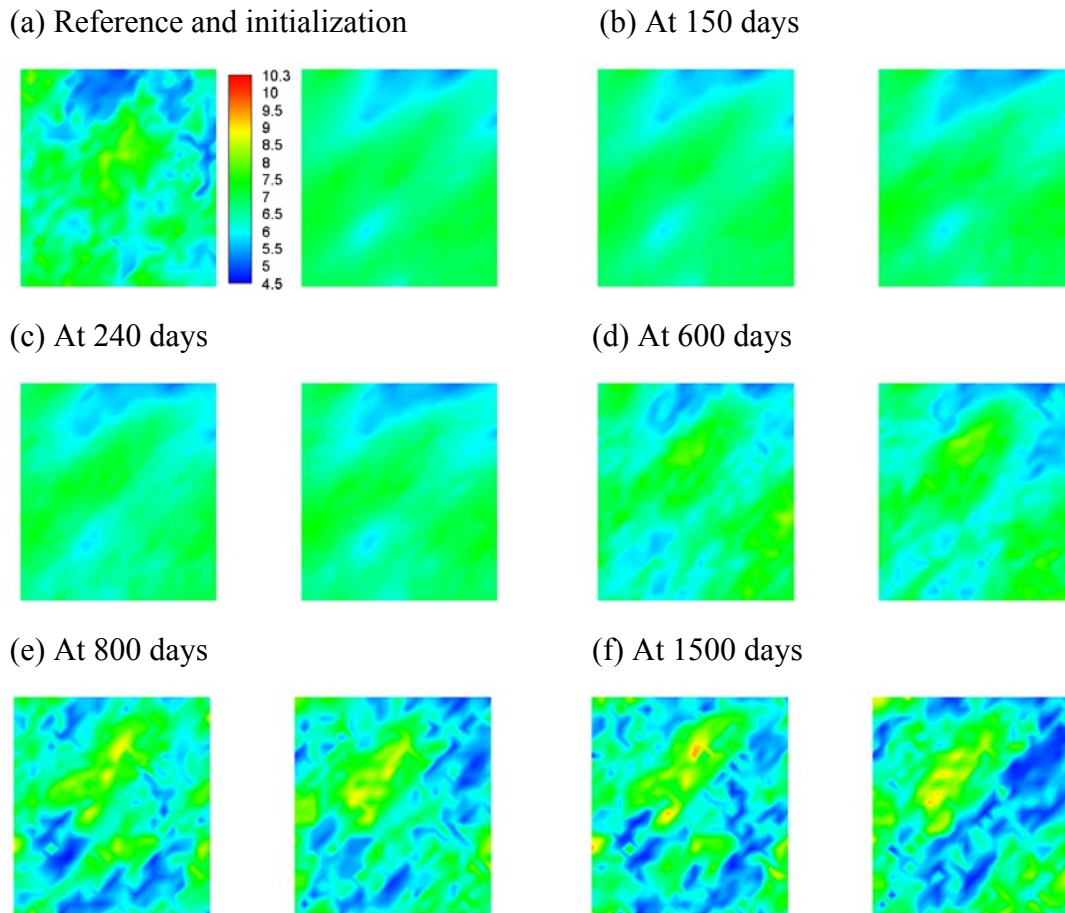
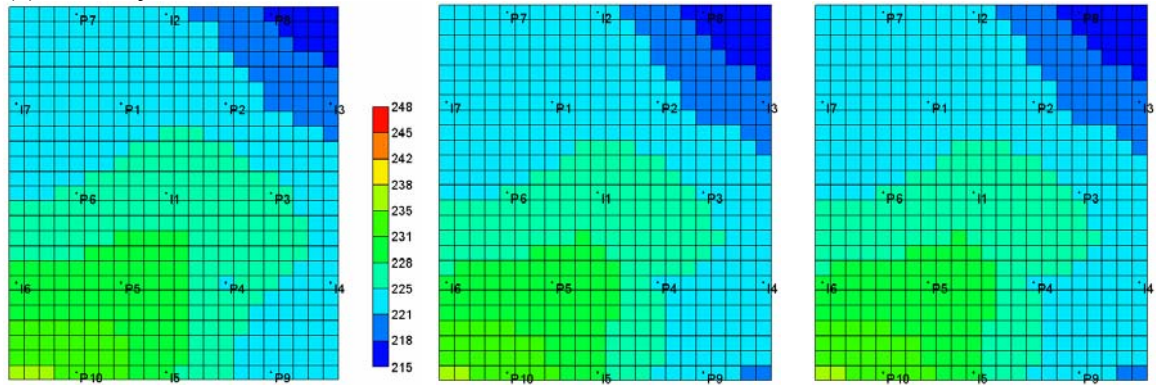
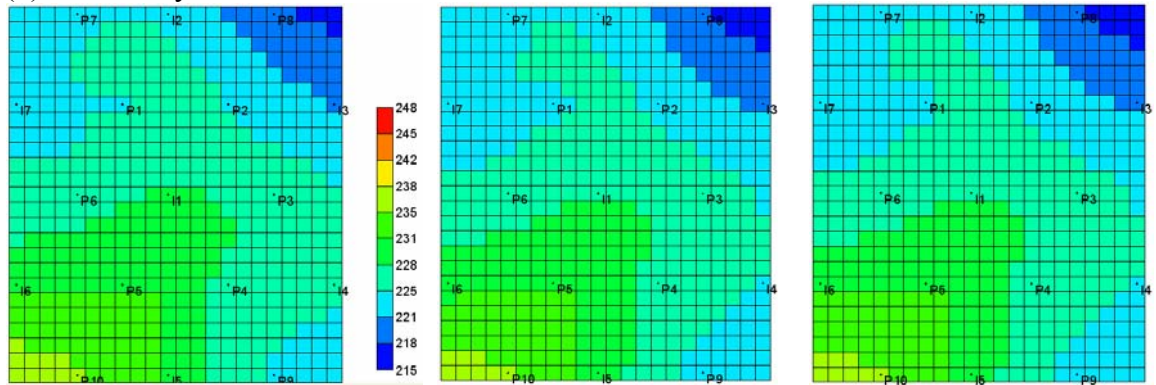


Figure 6-27: Seventeenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

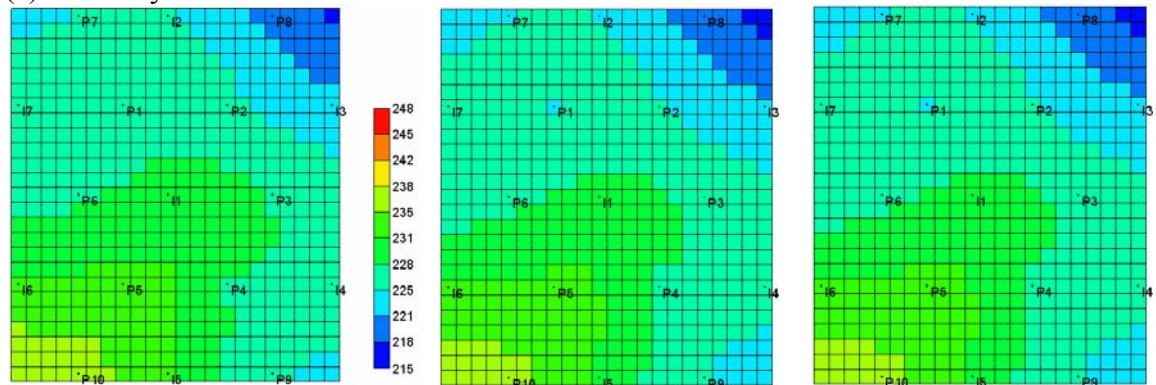
(a) First layer



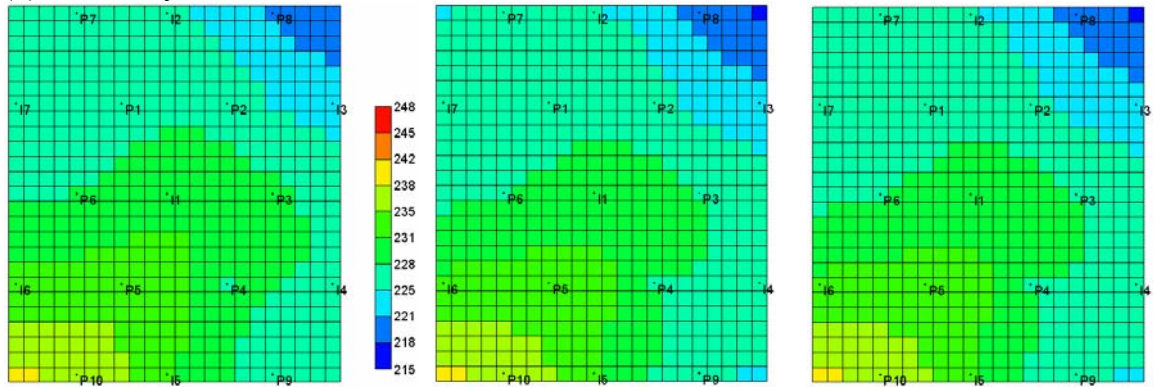
(b) Second layer



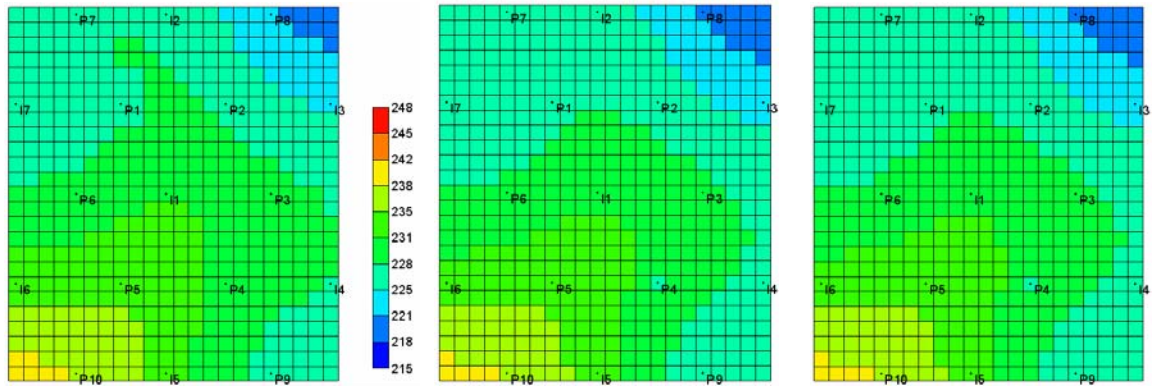
(c) Third layer



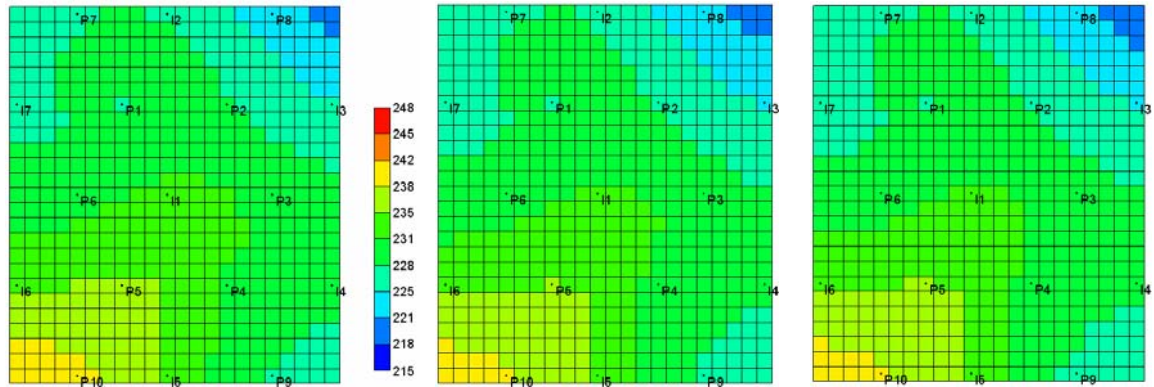
(d) Fourth layer



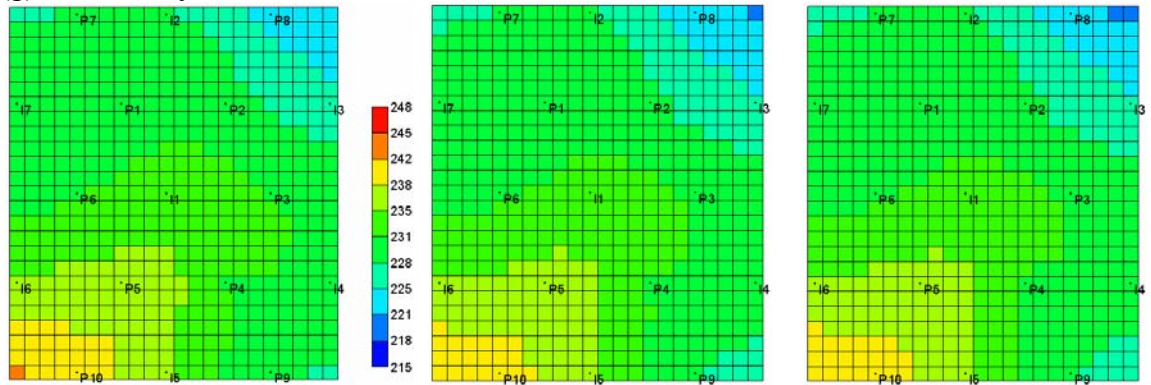
(e) Fifth layer



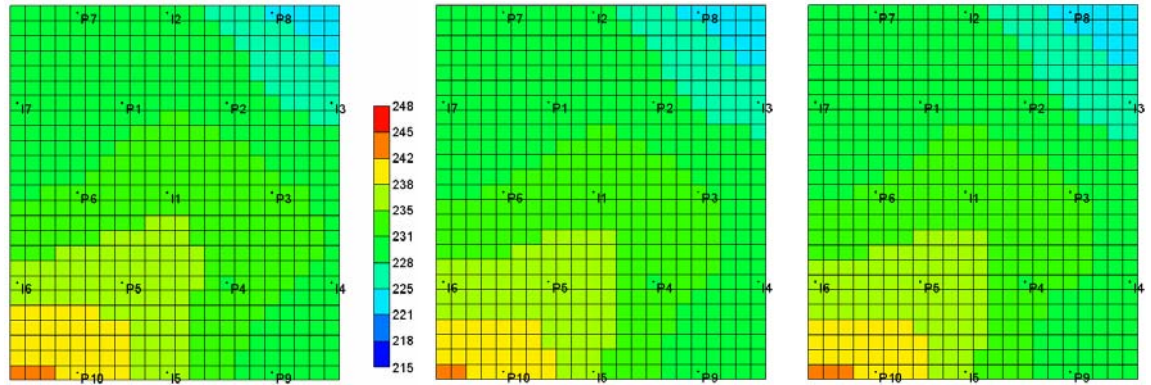
(f) Sixth layer



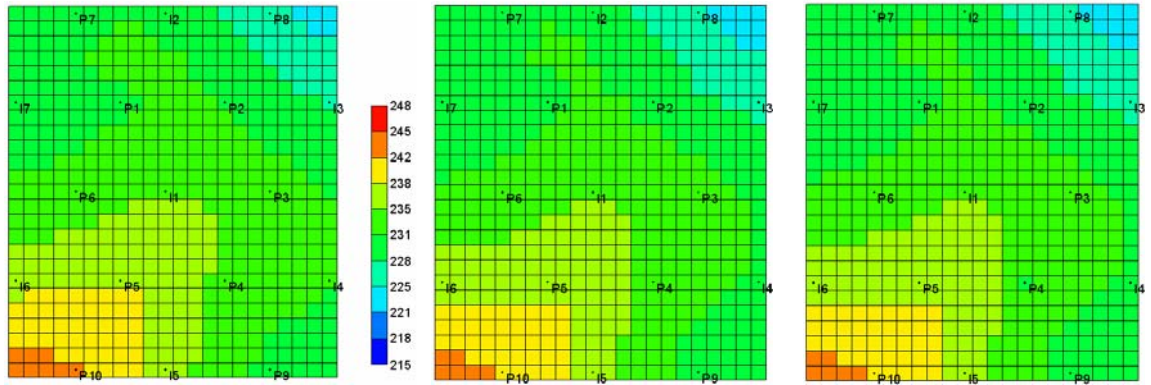
(g) Seventh layer



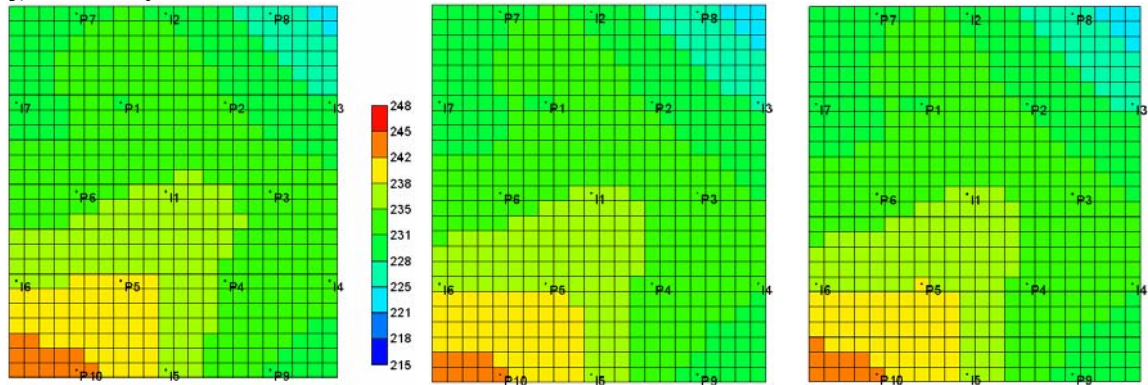
(h) Eighth layer



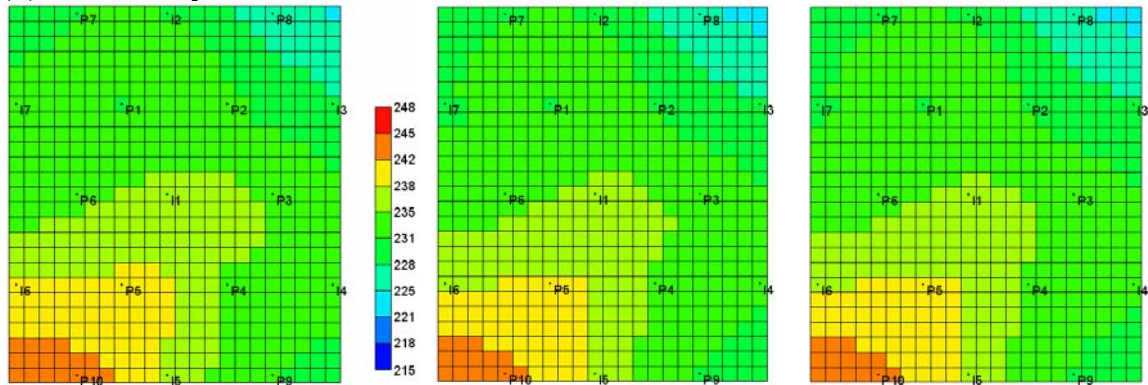
(i) Ninth layer



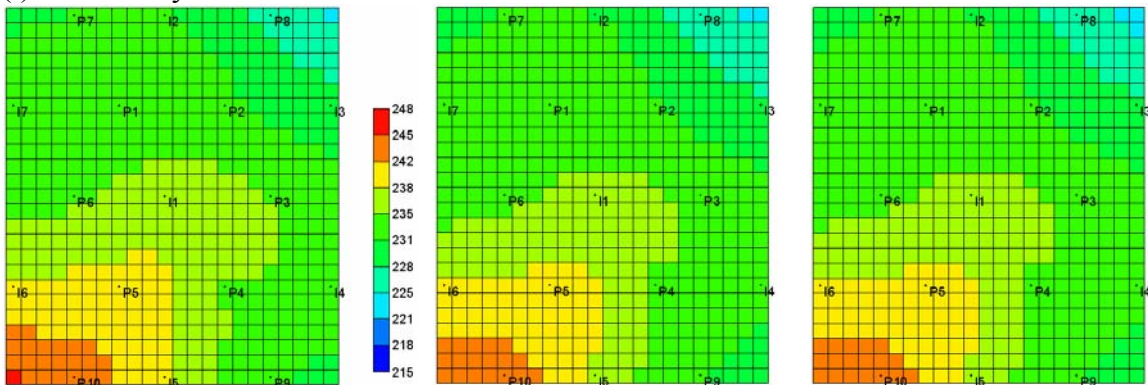
(j) Tenth layer



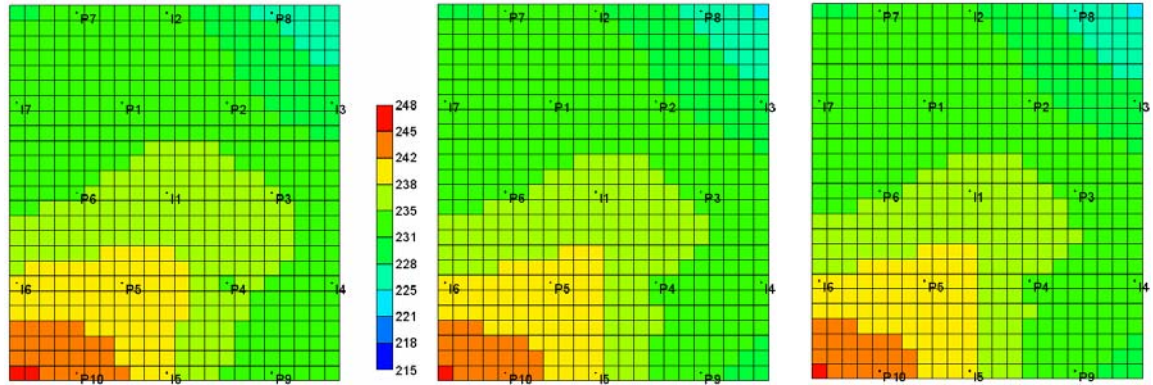
(k) Eleventh layer



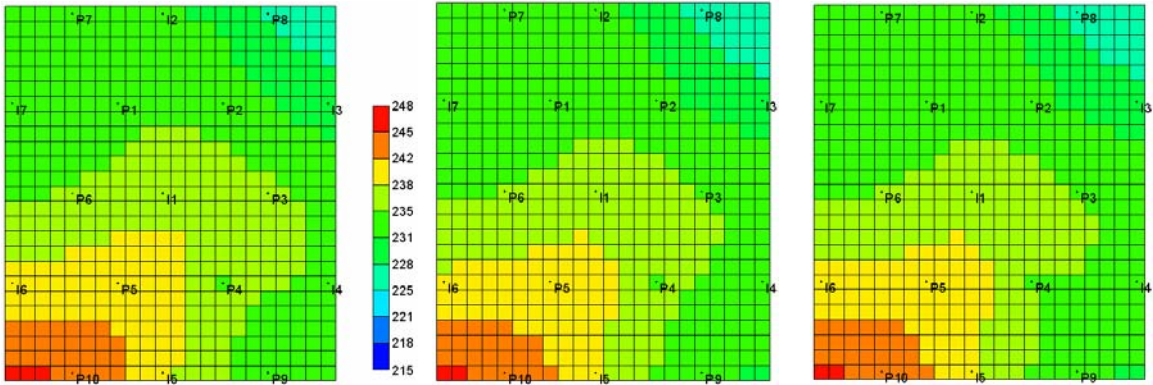
(l) Twelfth layer



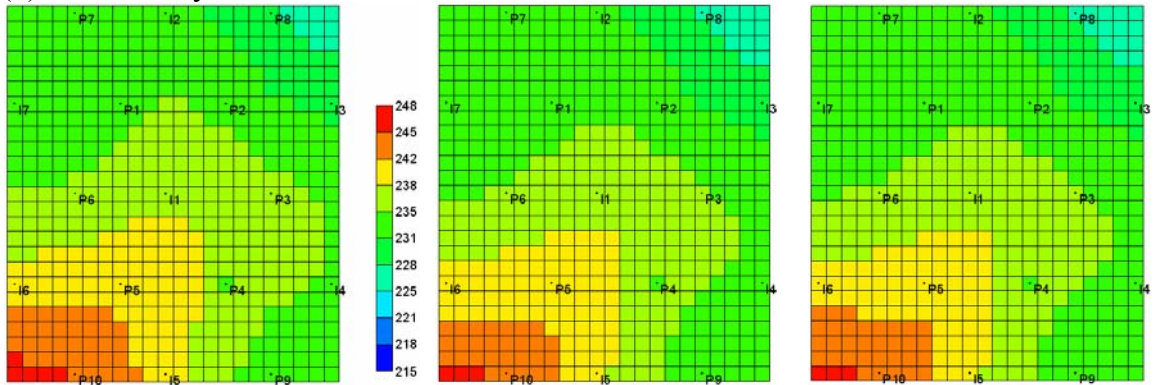
(m) Thirteenth layer



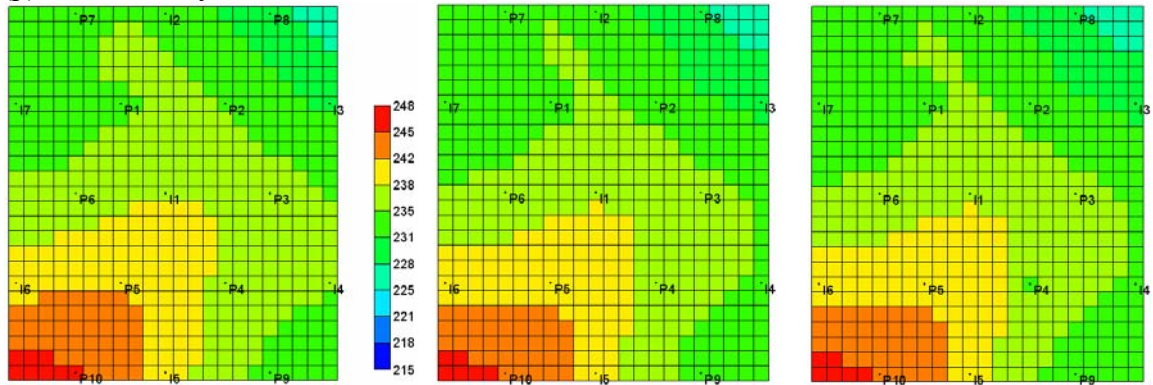
(n) Fourteenth layer



(o) Fifteenth layer



(p) Sixteenth layer



(q) Seventeenth layer

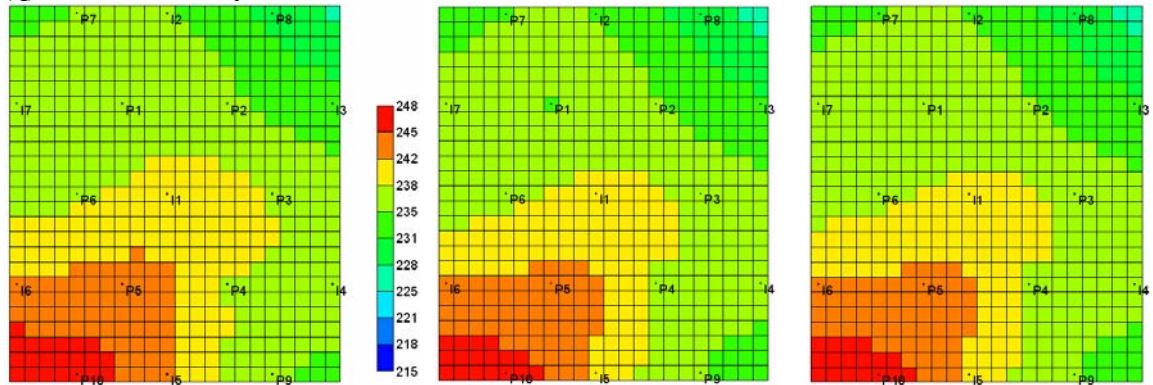


Figure 6-28: At the time of 150 days, from left to right in each layer, pressure maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively. Legend scale is the same for every profile, from 214.5368 psi (blue) through 248.34 psi (red).

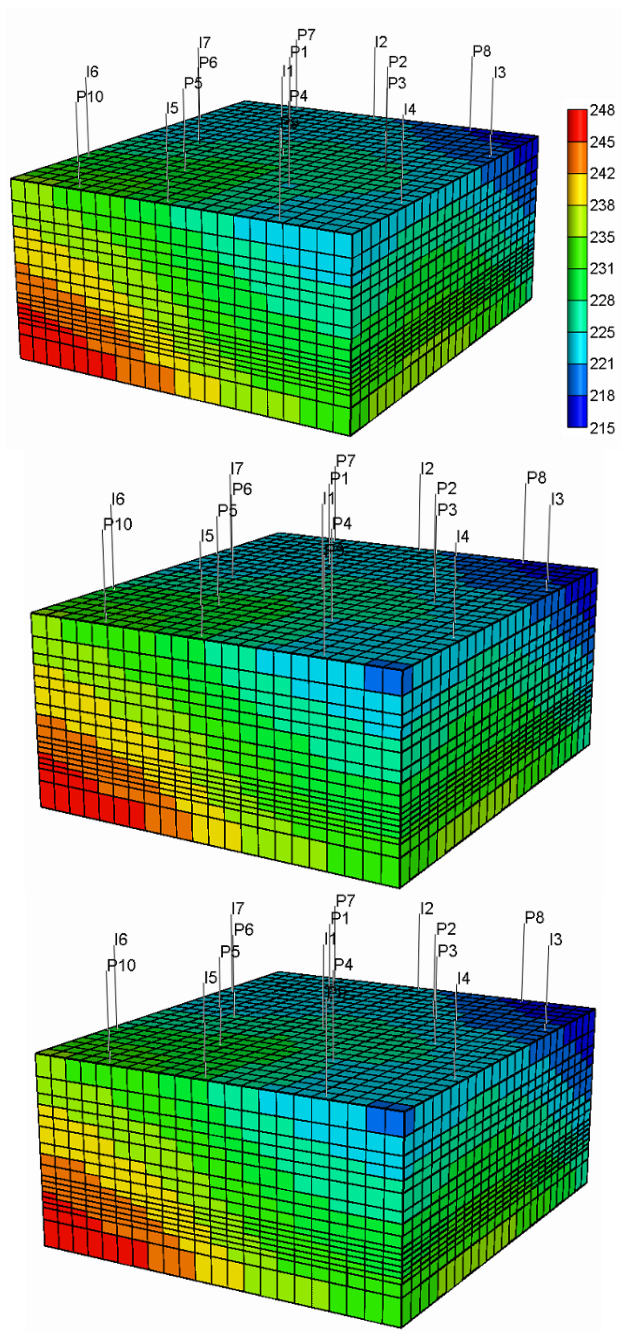
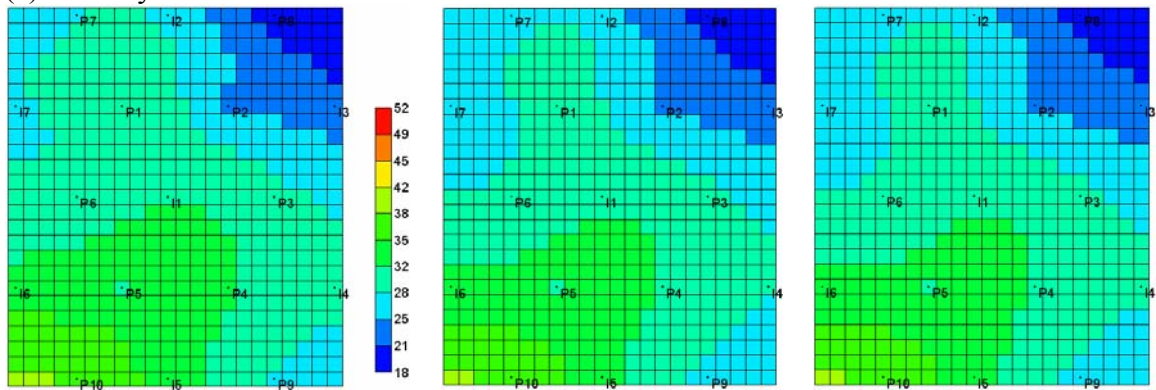
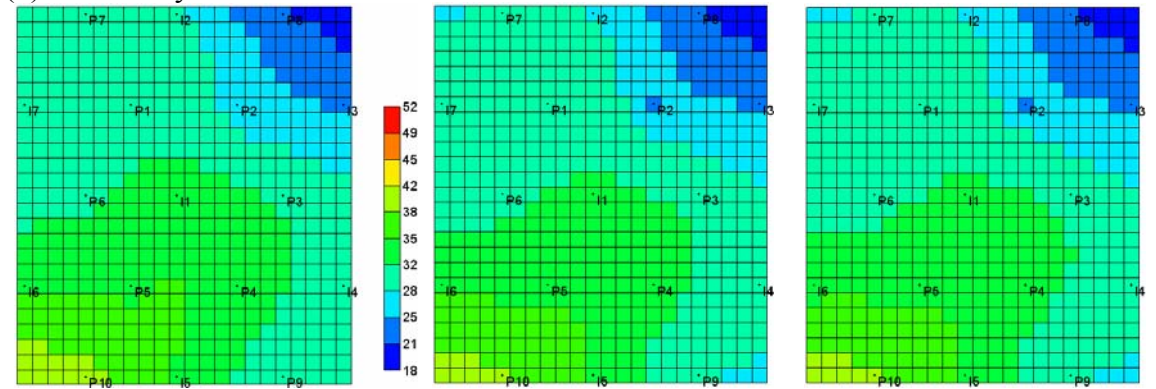


Figure 6-29: At the time of 150 days, from top to bottom, three-dimensional pressure views of the reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively. Legend scale is the same for every profile, from 214.5368 psi (blue) through 248.34 psi (red).

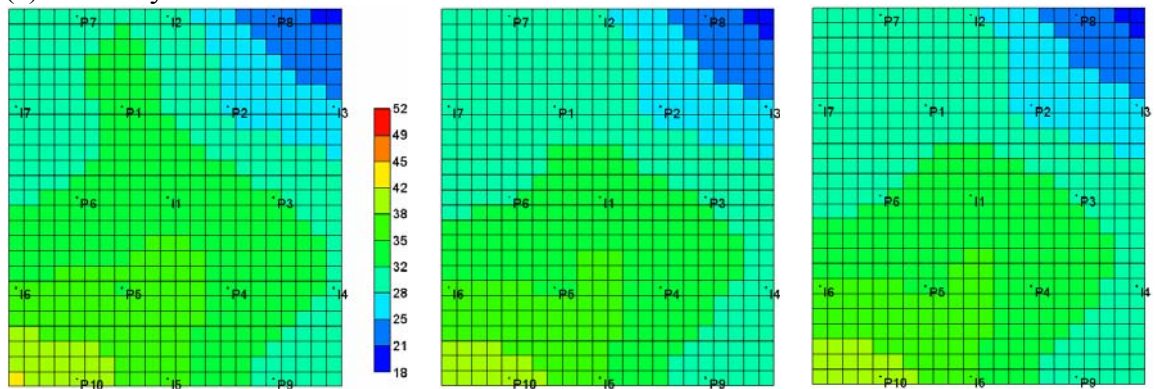
(a) First layer



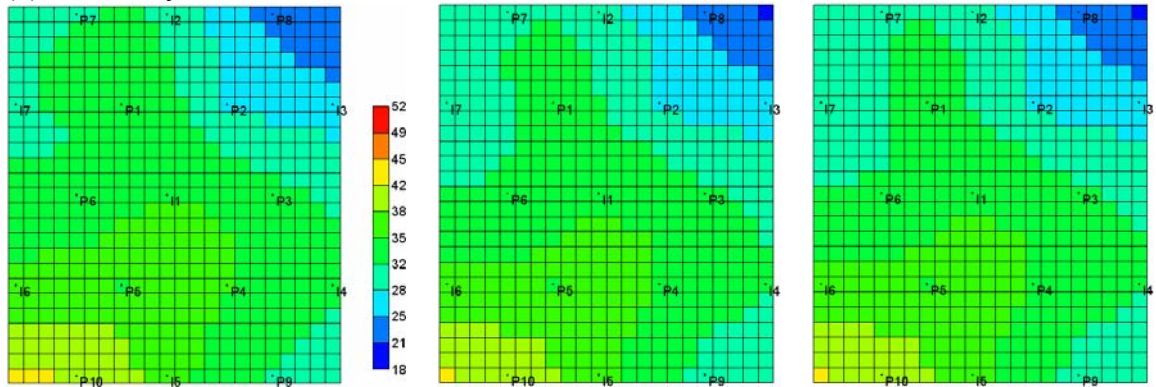
(b) Second layer



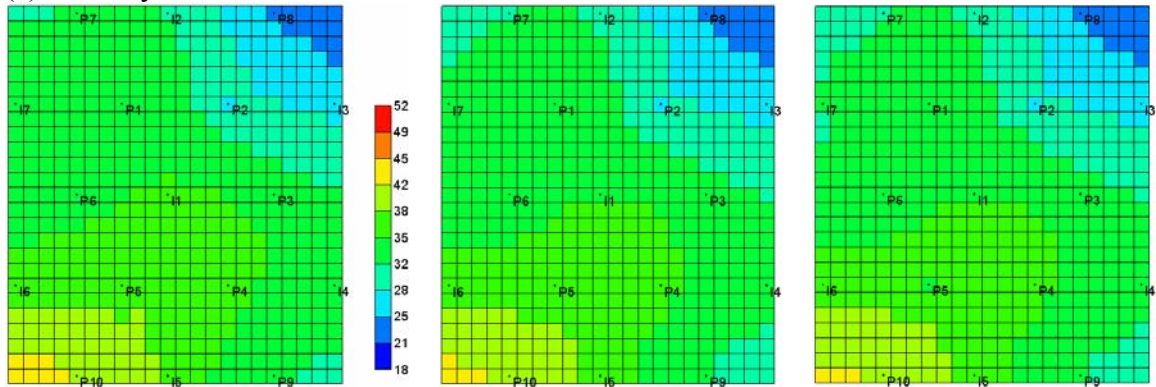
(c) Third layer



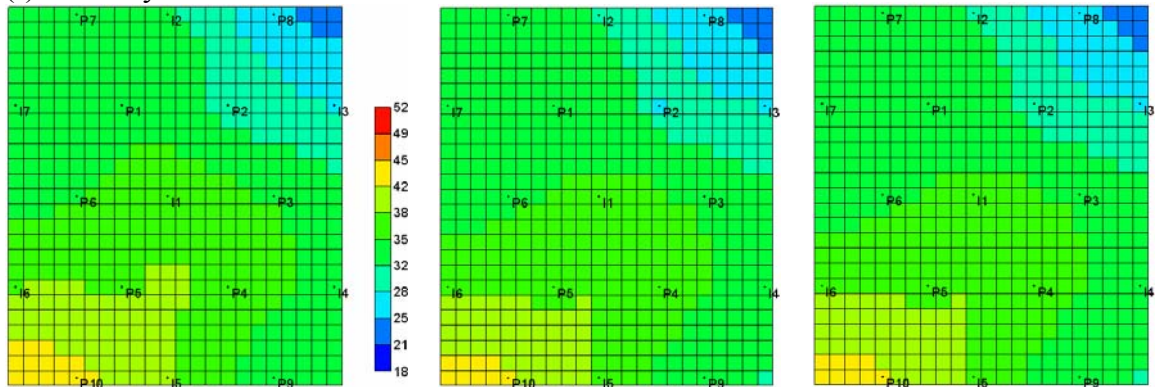
(d) Fourth layer



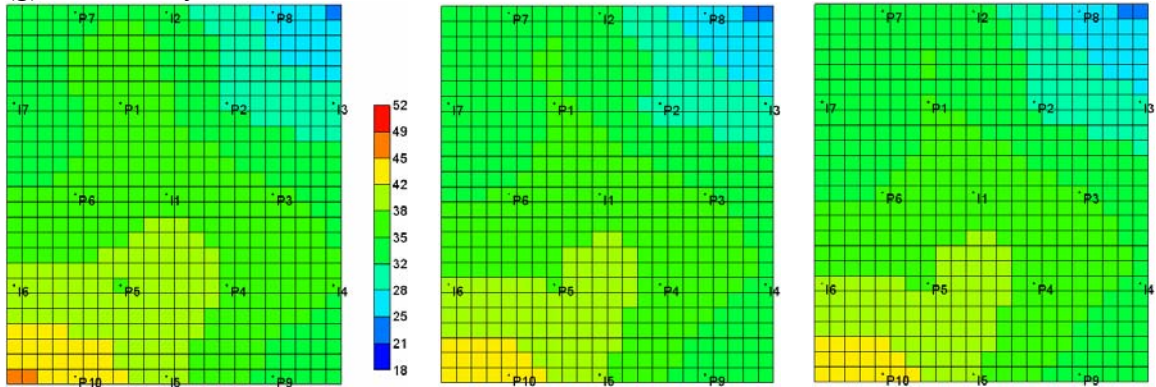
(e) Fifth layer



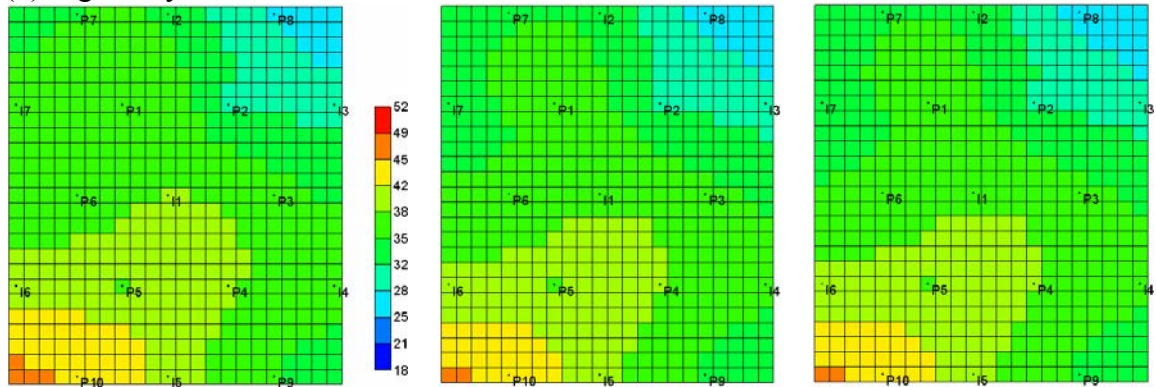
(f) Sixth layer



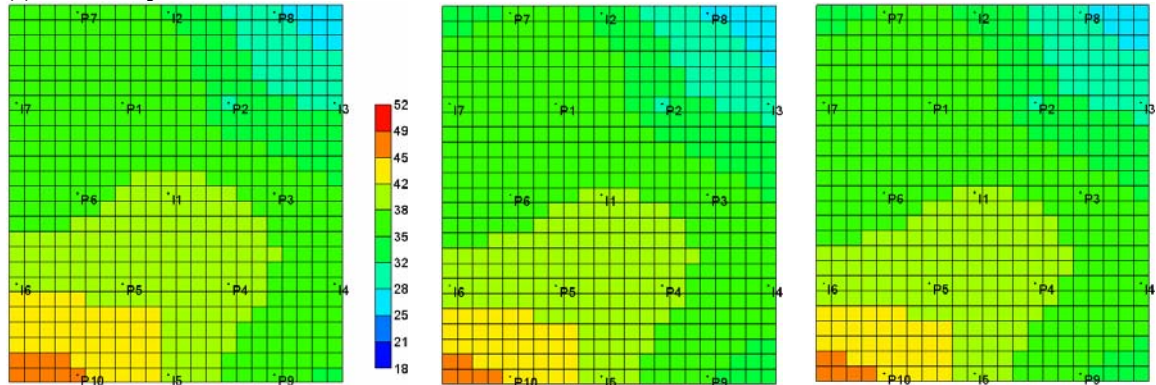
(g) Seventh layer



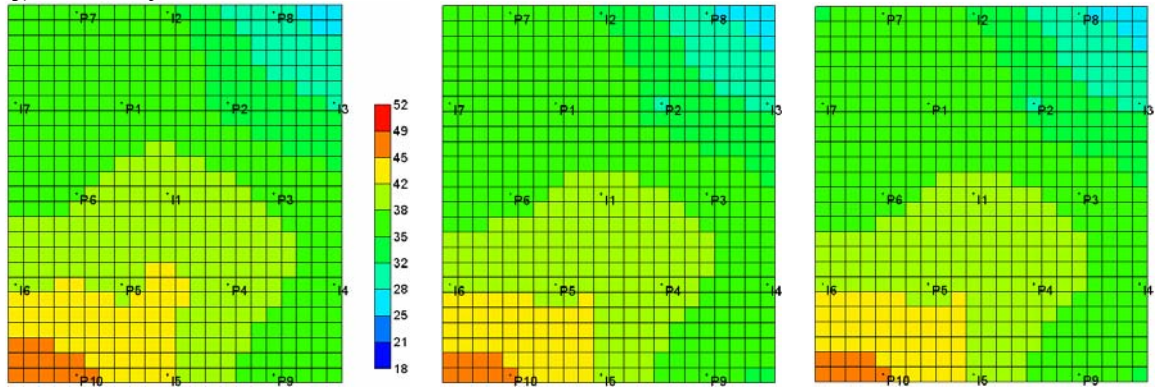
(h) Eighth layer



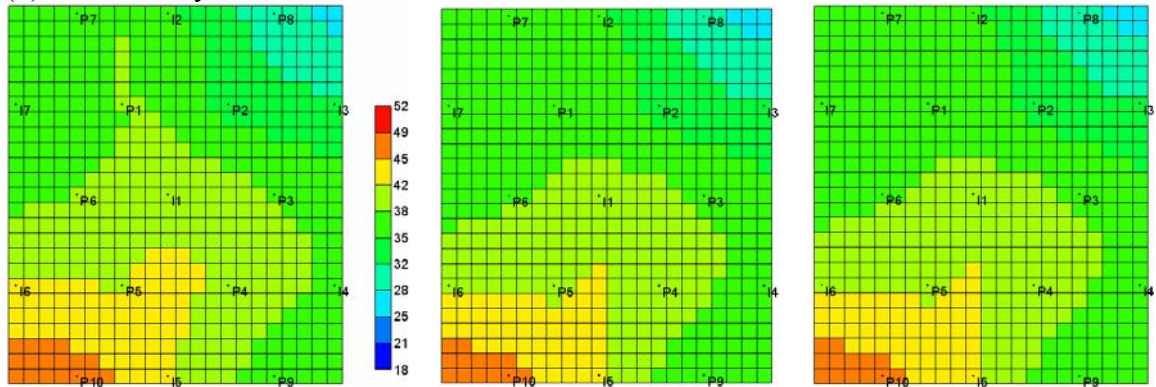
(i) Ninth layer



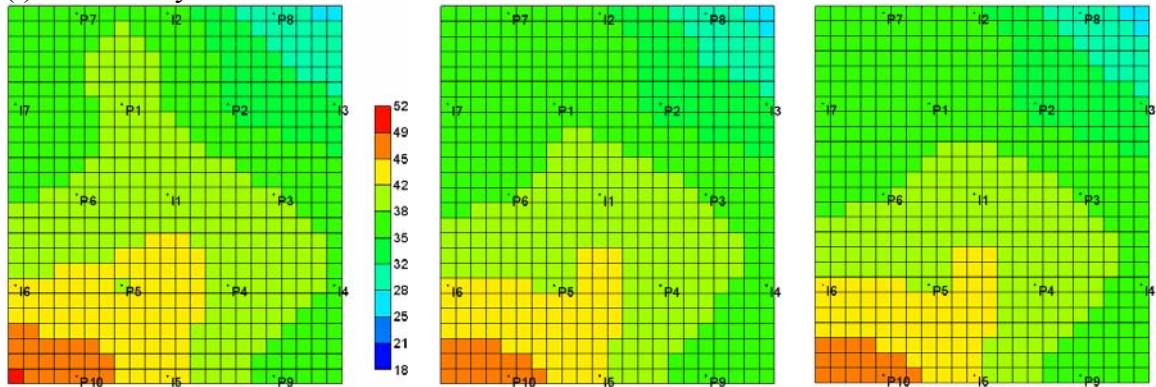
(j) Tenth layer



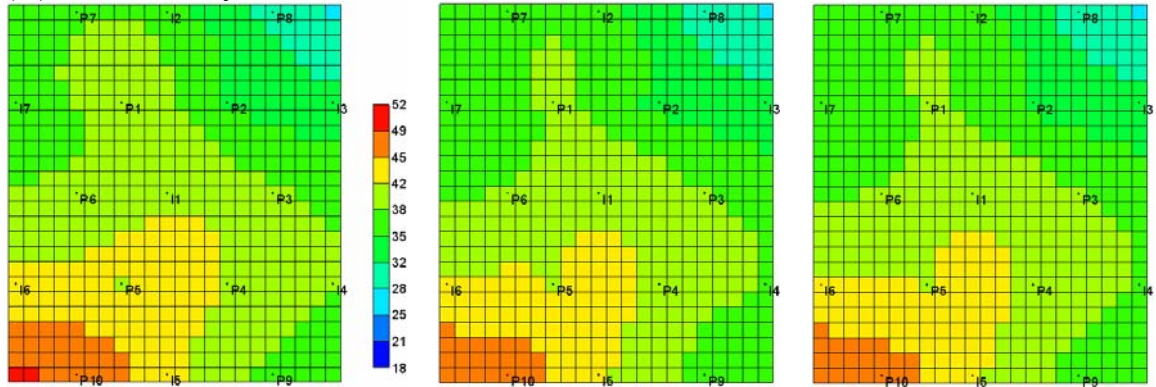
(k) Eleventh layer



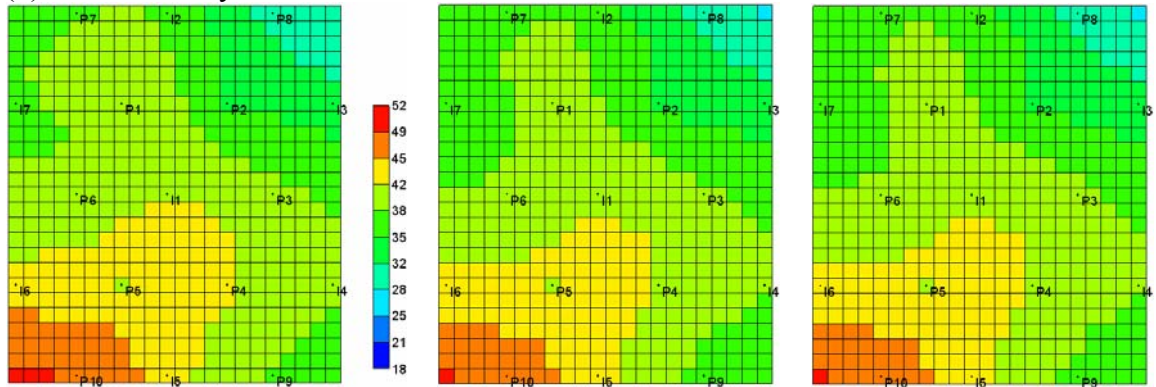
(l) Twelfth layer



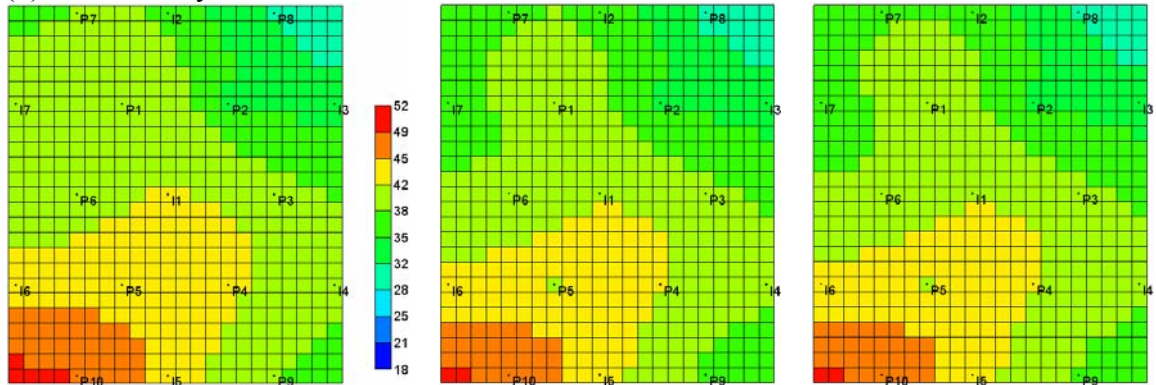
(m) Thirteenth layer



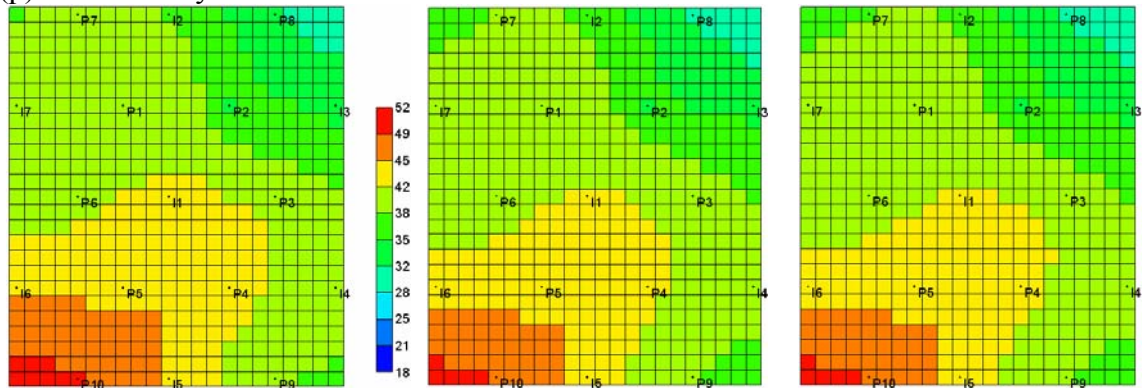
(n) Fourteenth layer



(o) Fifteenth layer



(p) Sixteenth layer



(q) Seventeenth layer

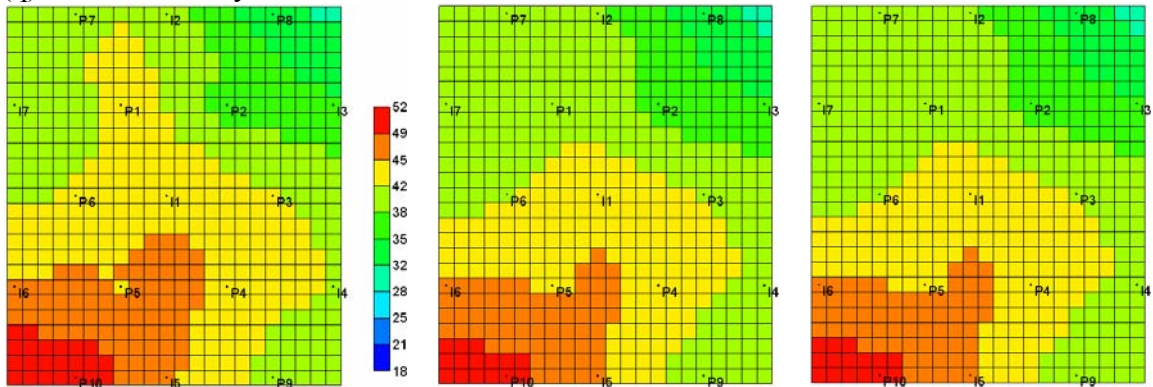


Figure 6-30: At the time of 240 days, from left to right in each layer, pressure maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively. Legend scale is the same for every profile, from 17.886 psi (blue) through 52.013 psi (red).

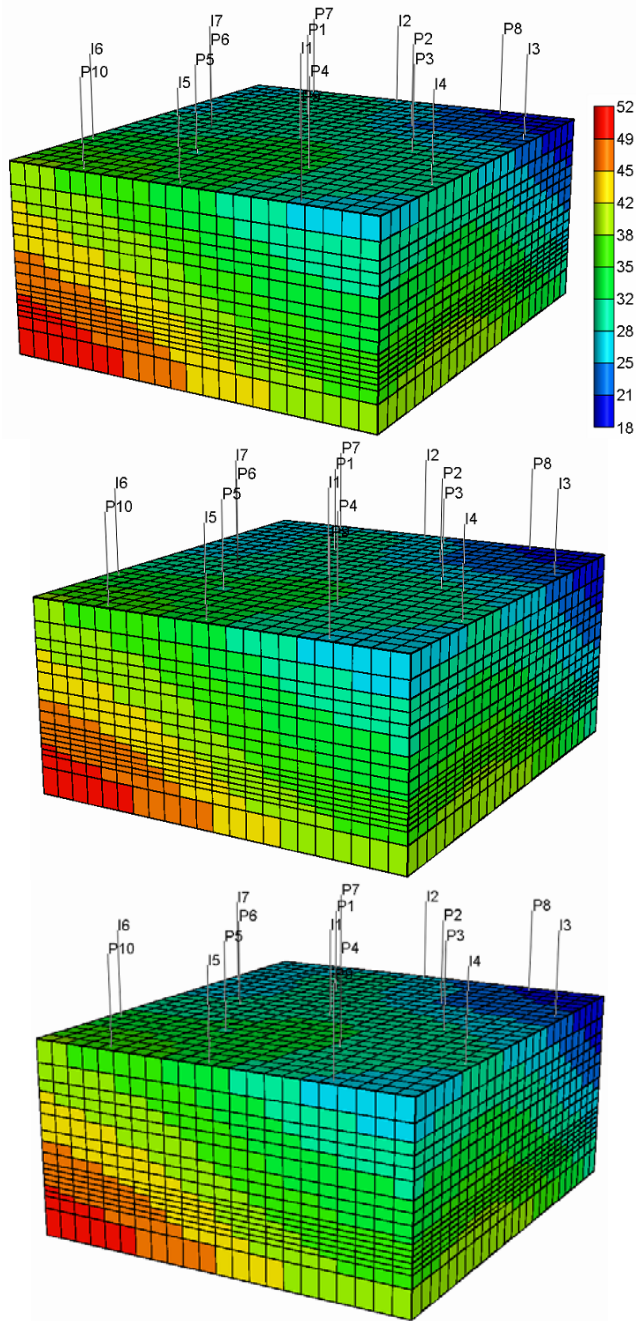
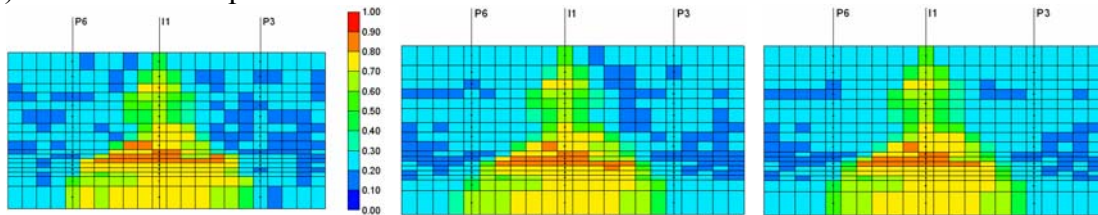
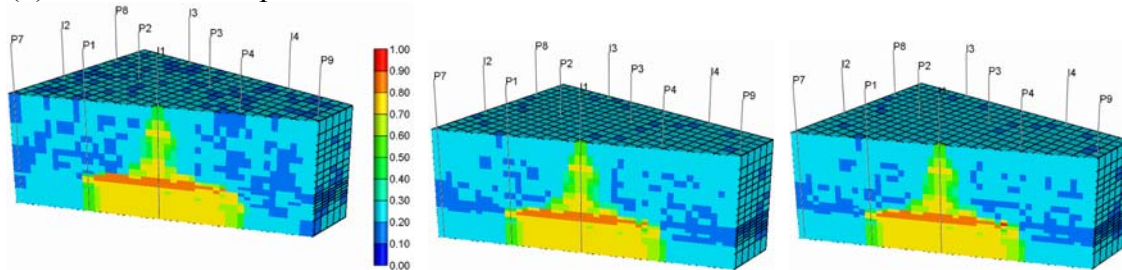


Figure 6-31: At the time of 240 days, from top to bottom, three-dimensional pressure views of the reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively. Legend scale is the same for every profile, from 17.886 psi (blue) through 52.013 psi (red).

(a) Cross-sectional profiles for P6-I1



(b) Cross-sectional profiles for P1-I1



(c) Cross-sectional profiles for P2-I1

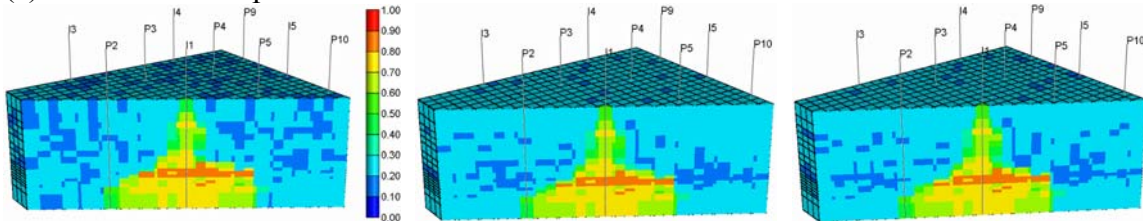
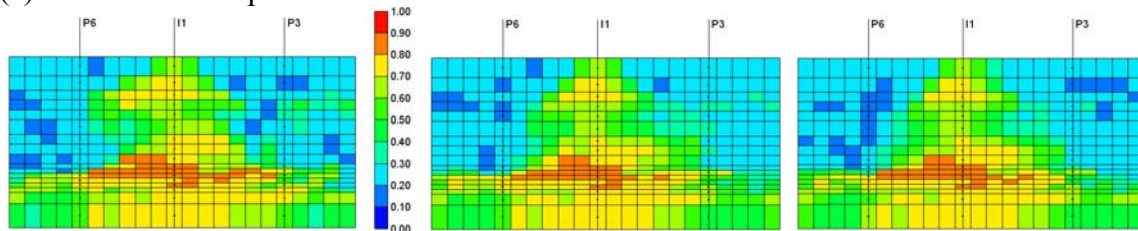
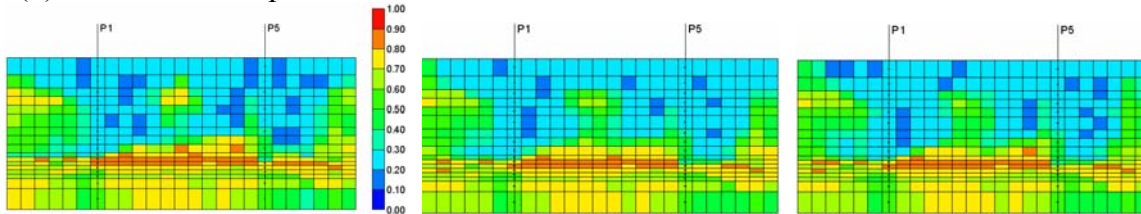


Figure 6-32: At the time of 600 days, water saturation maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively: (a) cross-sectional profiles for P6-I1; (b) cross-sectional profiles for P1-I1; (c) cross-sectional profiles for P2-I1. Legend scale is the same for every profile, from 0.0 (blue) through 1.0 (red).

(a) Cross-sectional profiles for P6-I1



(b) Cross-sectional profiles for P1-P5



(c) Three-dimensional slab views

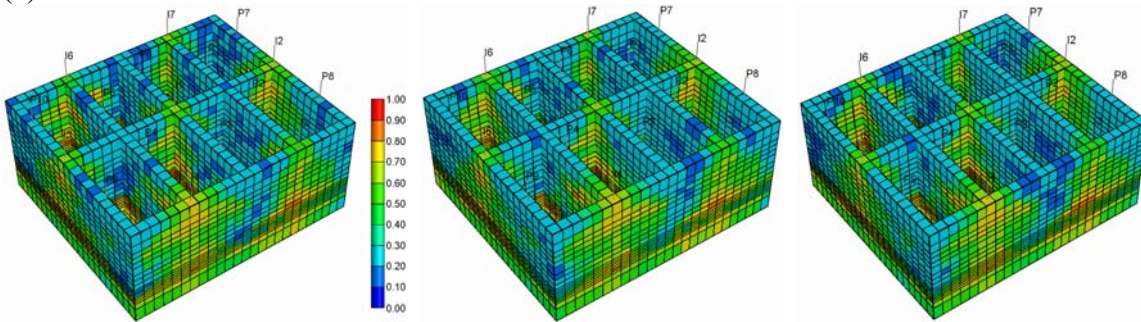
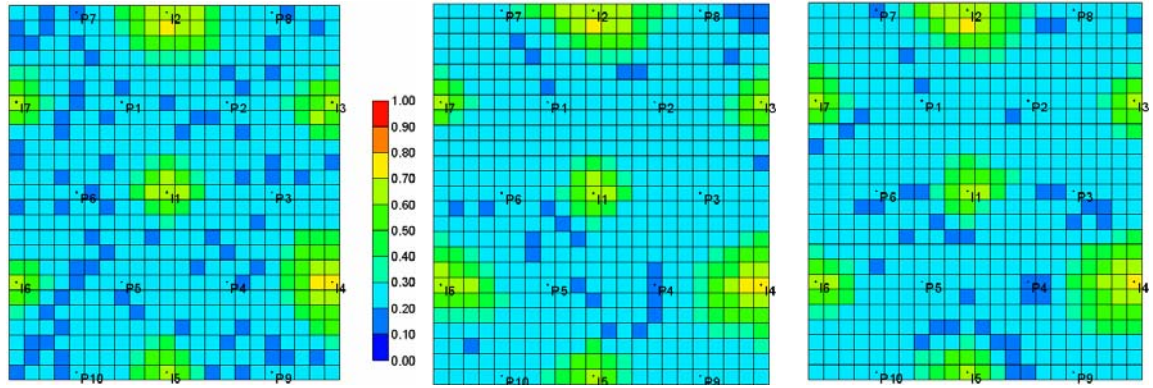
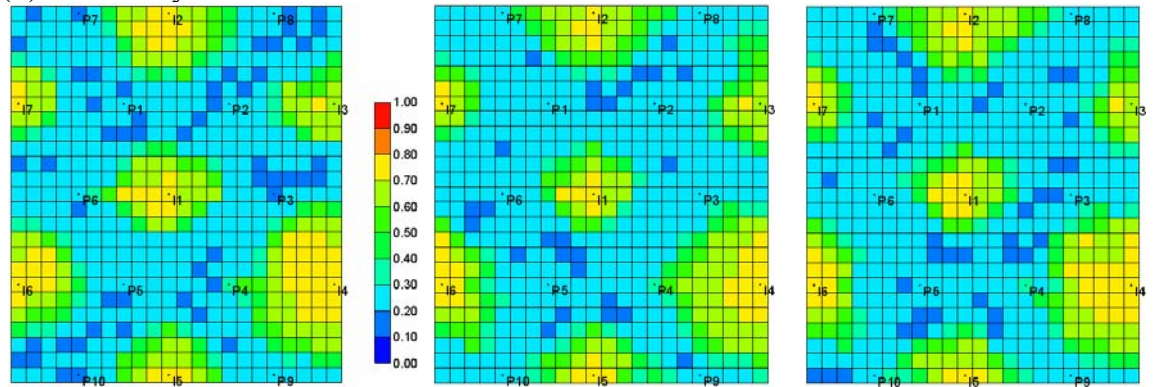


Figure 6-33: At the time of 700 days, water saturation maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively: (a) cross-sectional profiles for P6-I1; (b) cross-sectional profiles for P1-P5; (c) three-dimensional slab views. Legend scale is the same for every profile, from 0.0 (blue) through 1.0 (red).

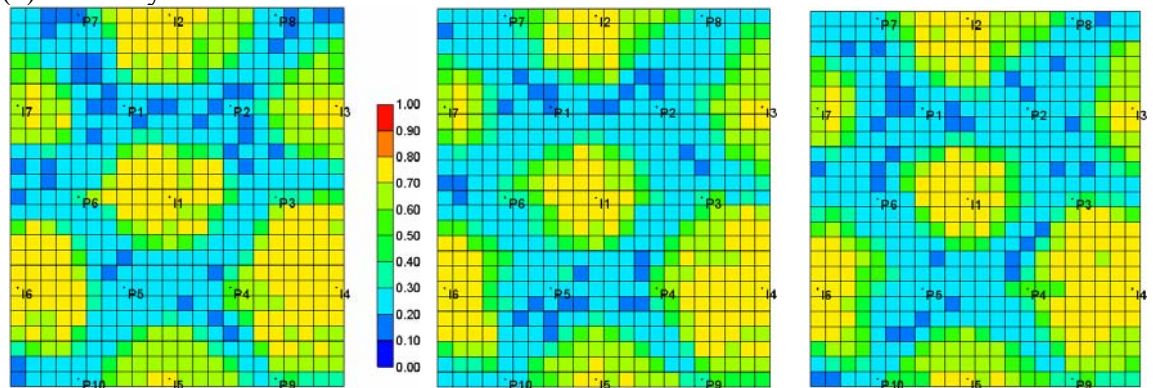
(a) First layer



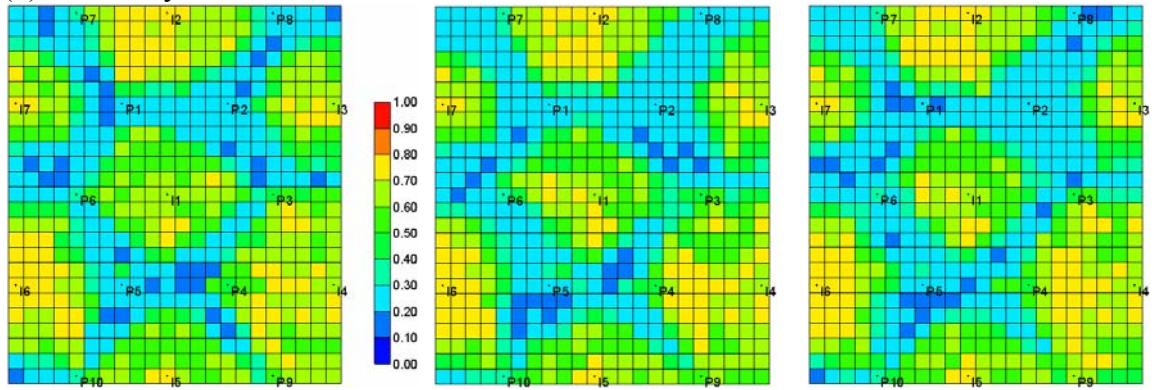
(b) Second layer



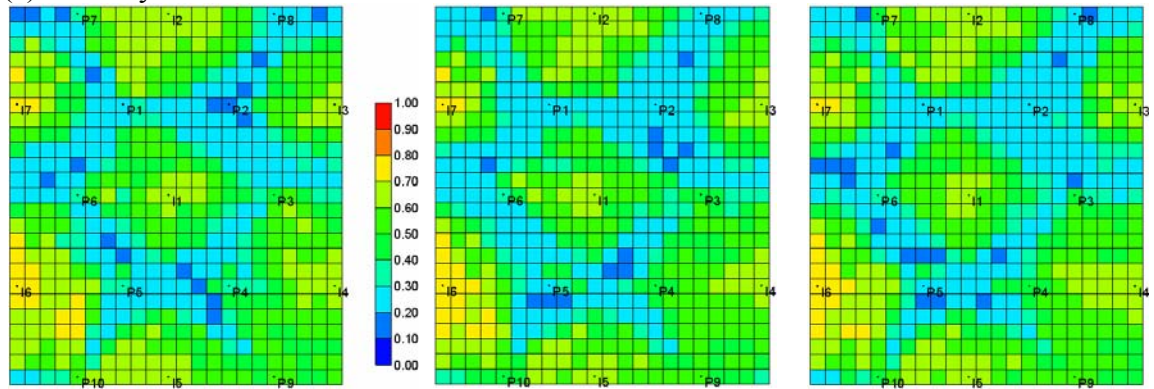
(c) Third layer



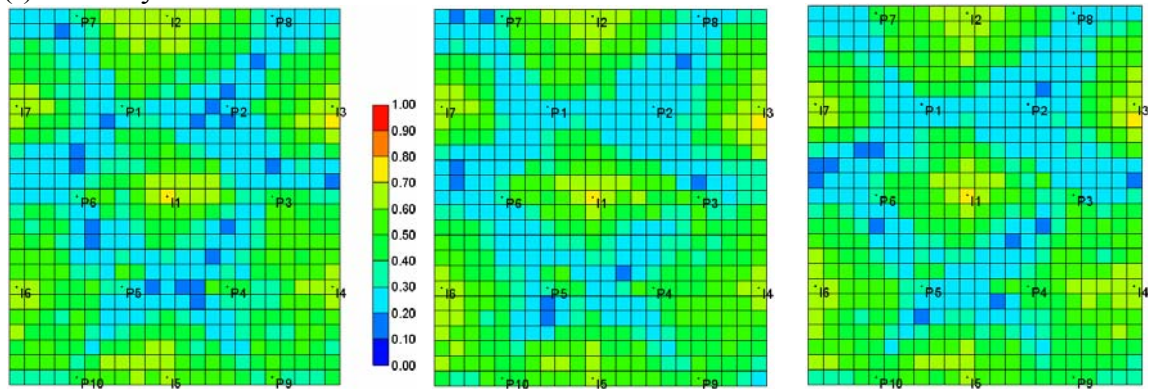
(d) Fourth layer



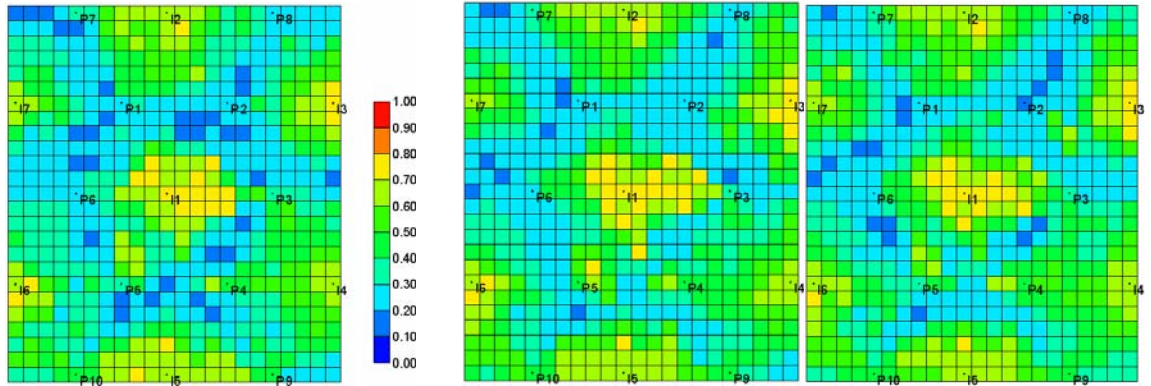
(e) Fifth layer



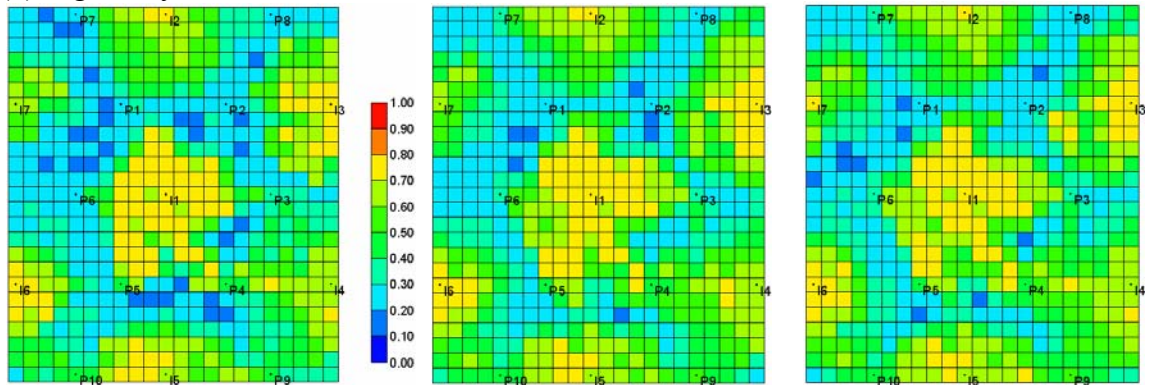
(f) Sixth layer



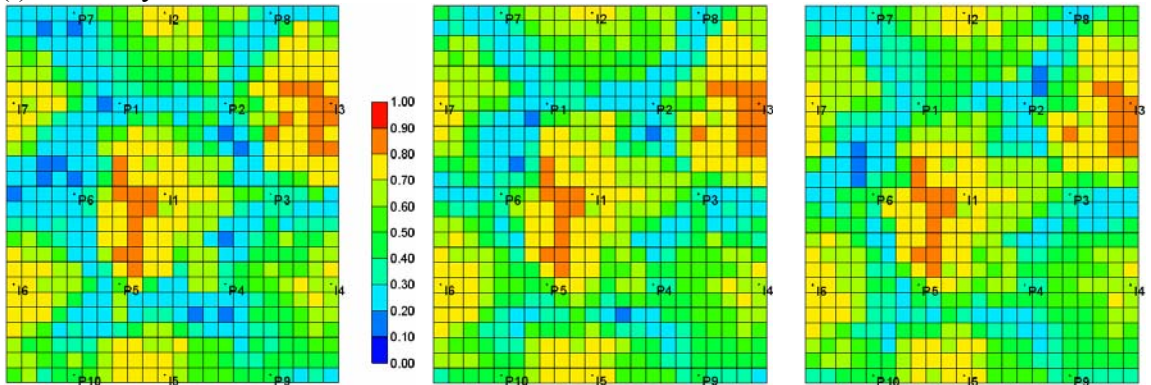
(g) Seventh layer



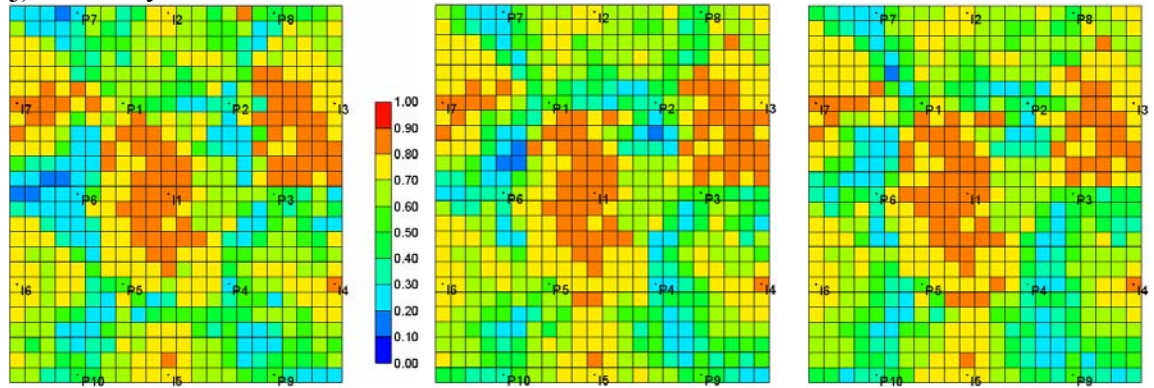
(h) Eighth layer



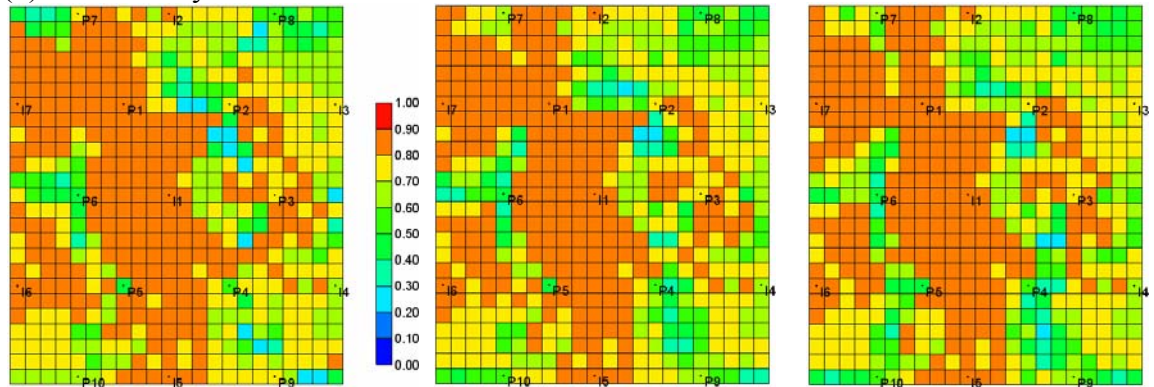
(i) Ninth layer



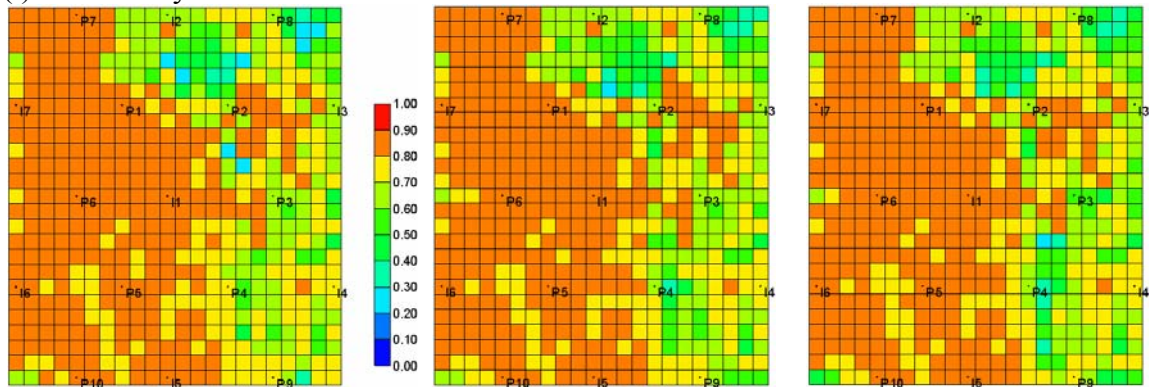
(j) Tenth layer



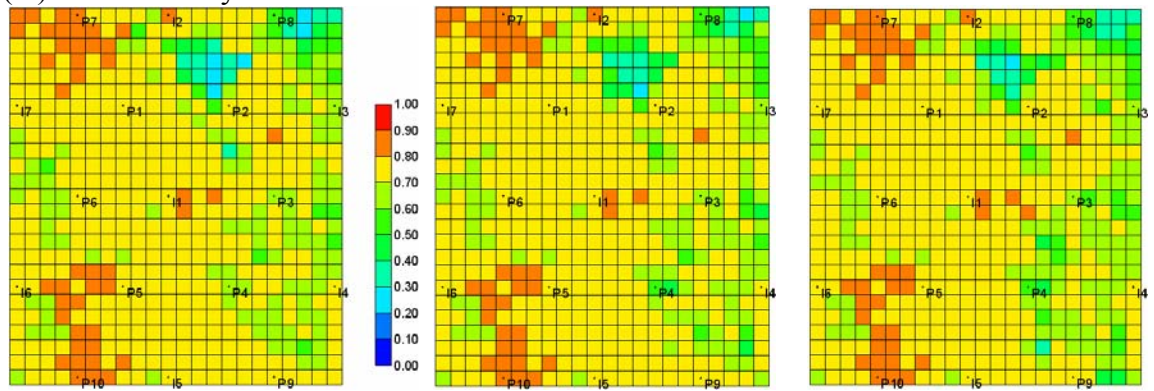
(k) Eleventh layer



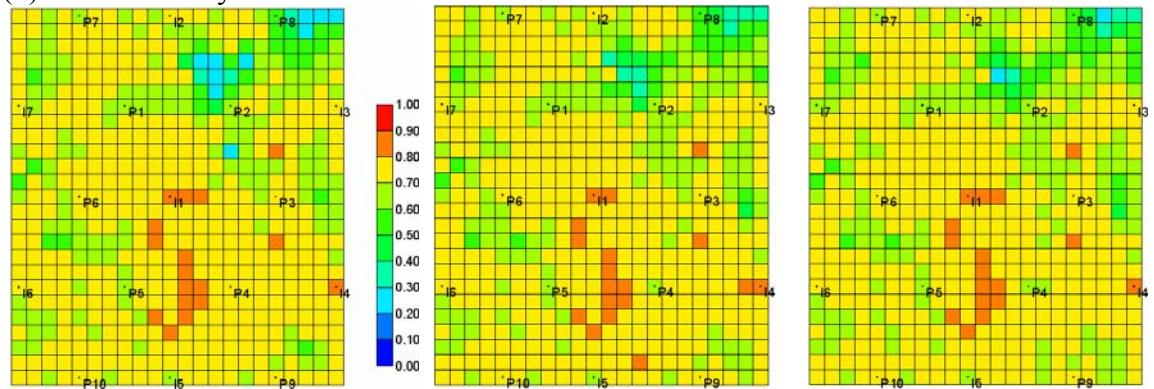
(l) Twelfth layer



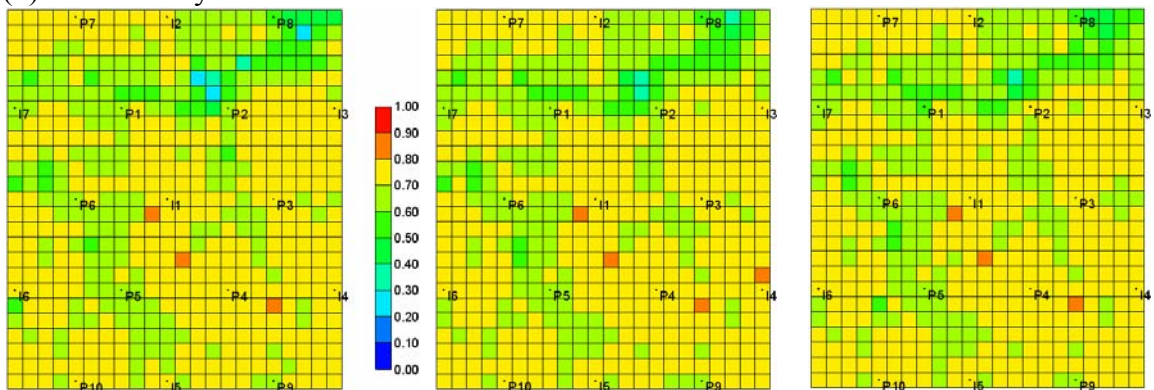
(m) Thirteenth layer



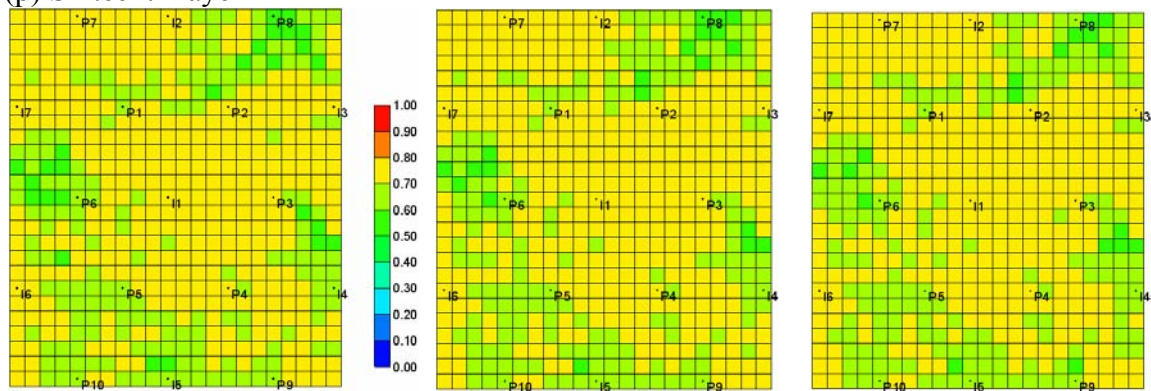
(n) Fourteenth layer



(o) Fifteenth layer



(p) Sixteenth layer



(q) Seventeenth layer

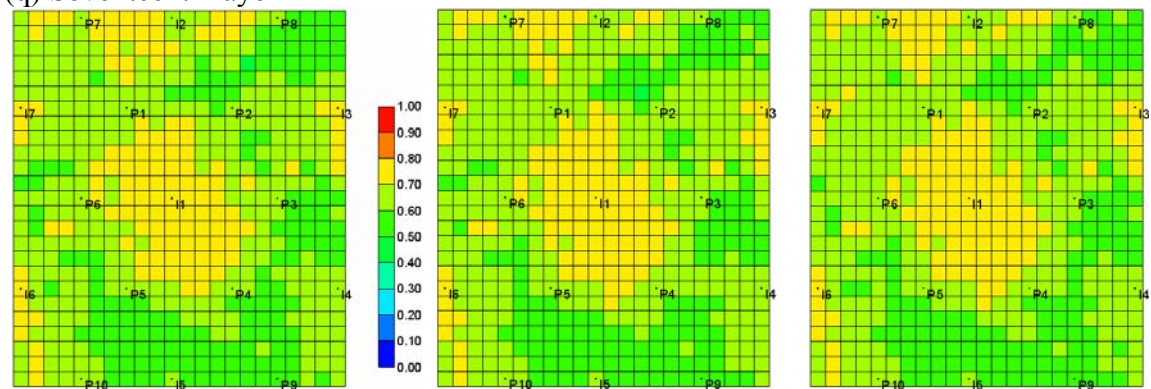
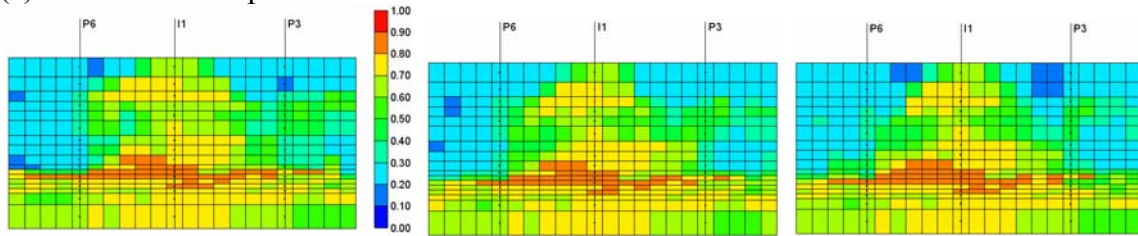
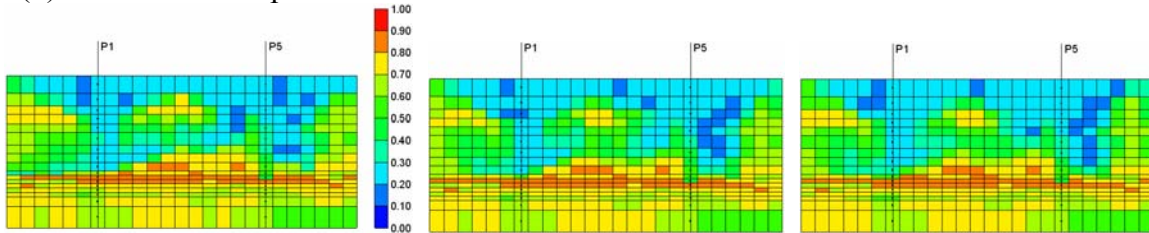


Figure 6-34: At the time of 800 days, from left to right in each layer, water saturation maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively. Legend scale is the same for every profile, from 0.0 (blue) through 1.0 (red).

(a) Cross-sectional profiles for P6-I1



(b) Cross-sectional profiles for P1-P5



(c) Three-dimensional slab views

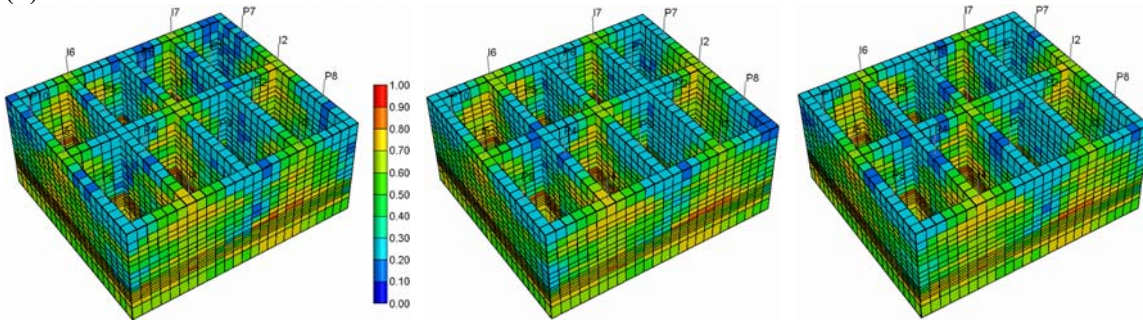
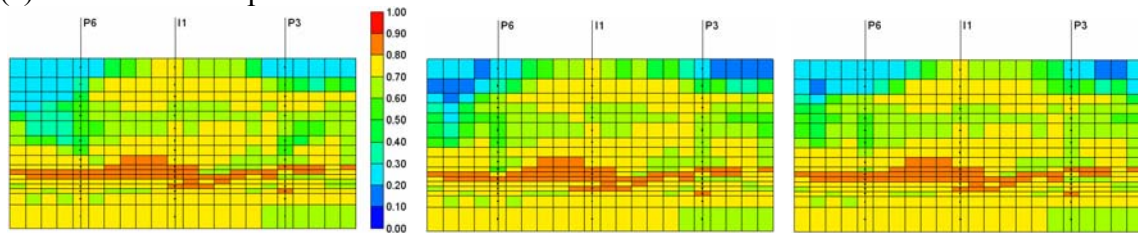
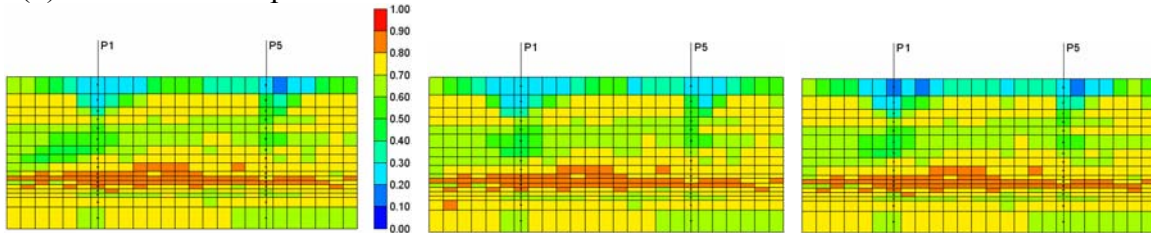


Figure 6-35: At the time of 800 days, water saturation maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively: (a) cross-sectional profiles for P6-I1; (b) cross-sectional profiles for P1-P5; (c) three-dimensional slab views. Legend scale is the same for every profile, from 0.0 (blue) through 1.0 (red).

(a) Cross-sectional profiles for P6-I1



(b) Cross-sectional profiles for P1-P5



(c) Three-dimensional slab views

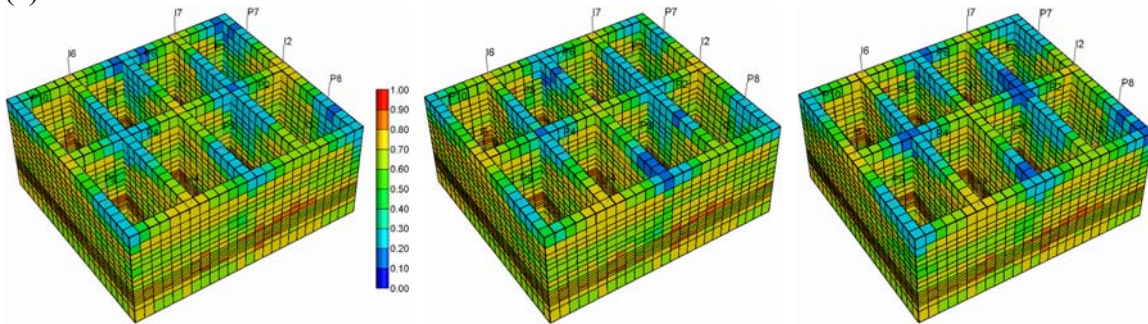
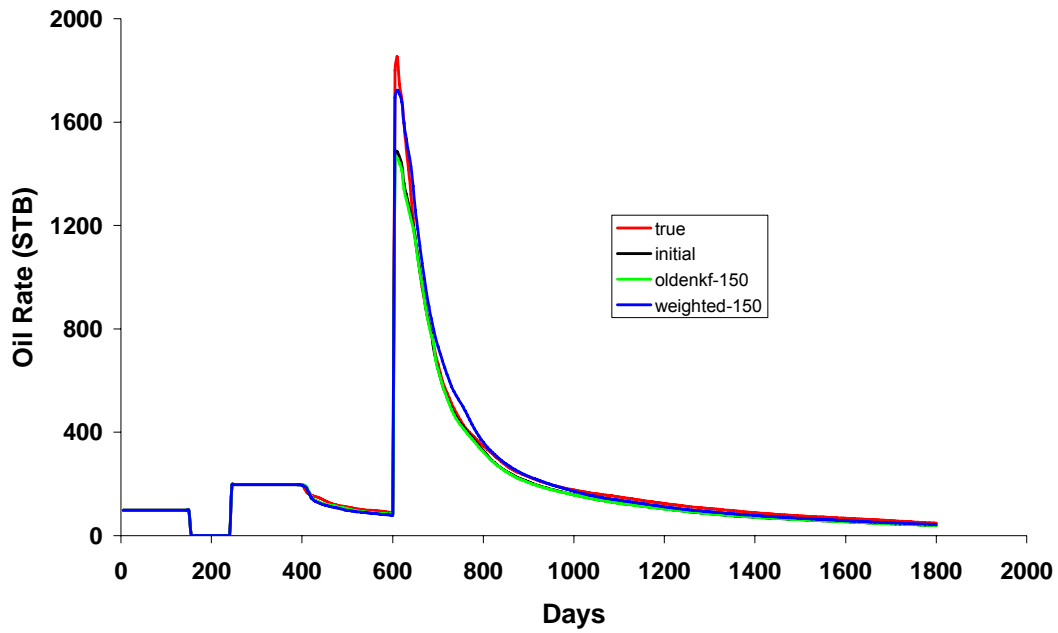
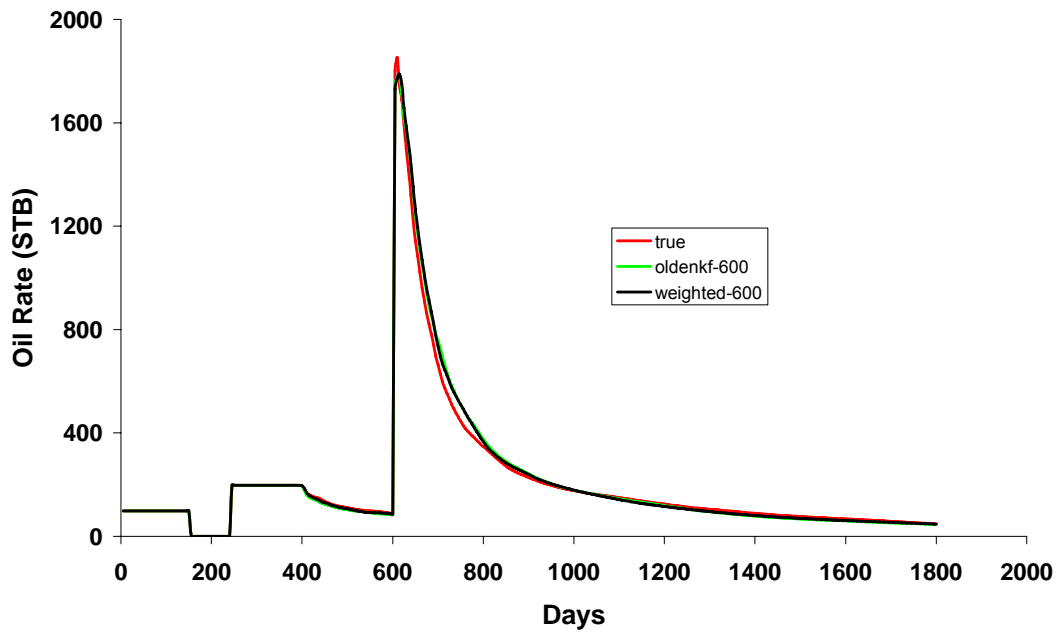


Figure 6-36: At the time of 1500 days, water saturation maps of reference and averages of ensemble members from the conventional EnKF and the weighted EnKF, respectively: (a) cross-sectional profiles for P6-I1; (b) cross-sectional profiles for P1-P5; (c) three-dimensional slab views. Legend scale is the same for every profile, from 0.0 (blue) through 1.0 (red).

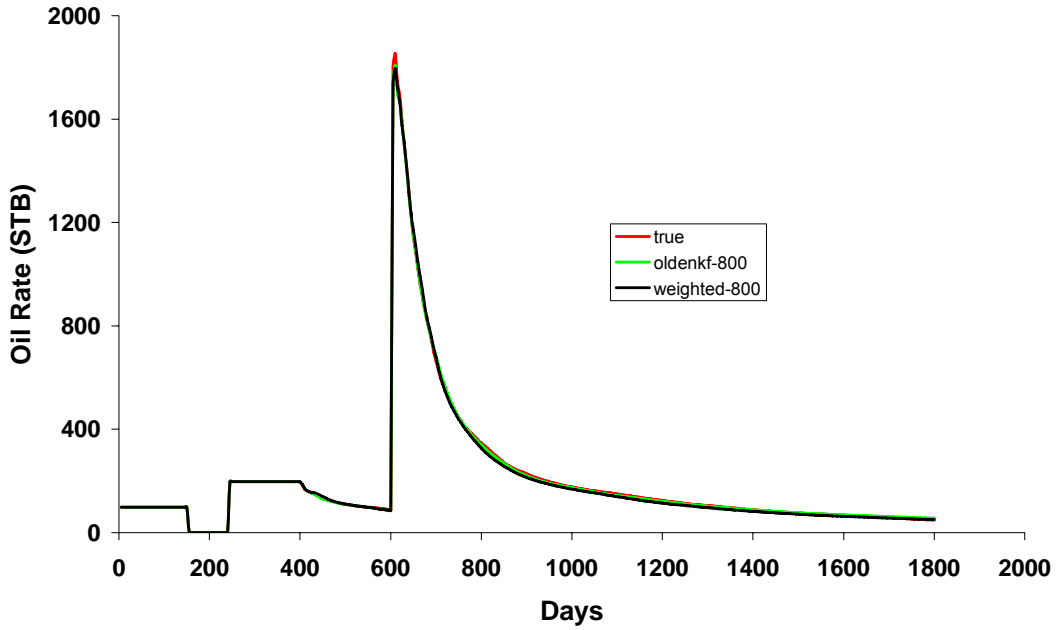
(a) Oil rate plots from reference, initial, and 150 days



(b) Oil rate plots from reference and 600 days



(c) Oil rate plots from reference and 800 days



(d) Oil rate plots from reference and 1500 days

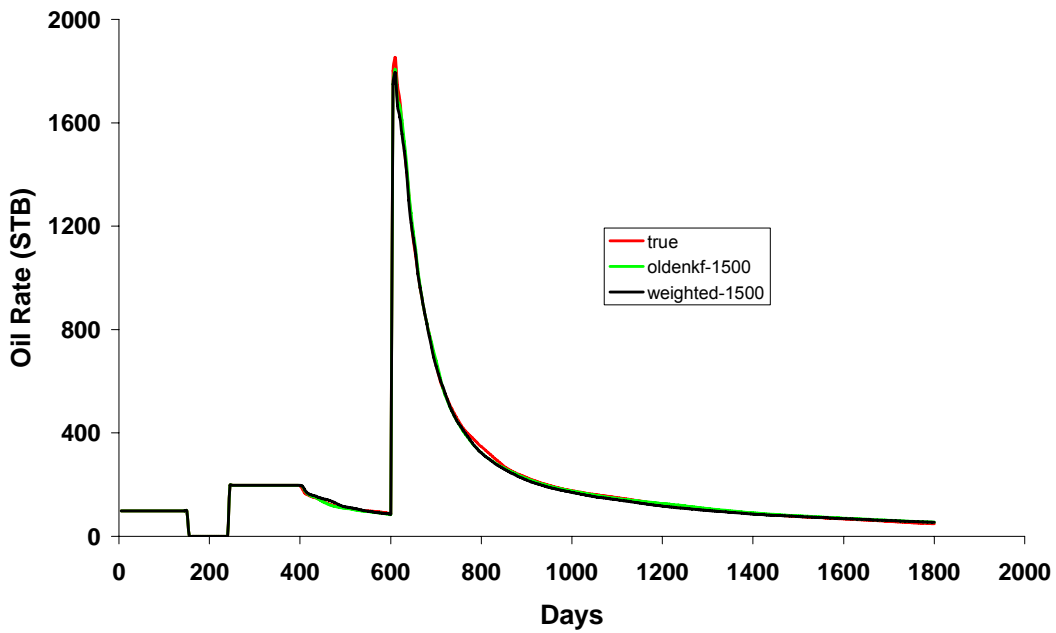
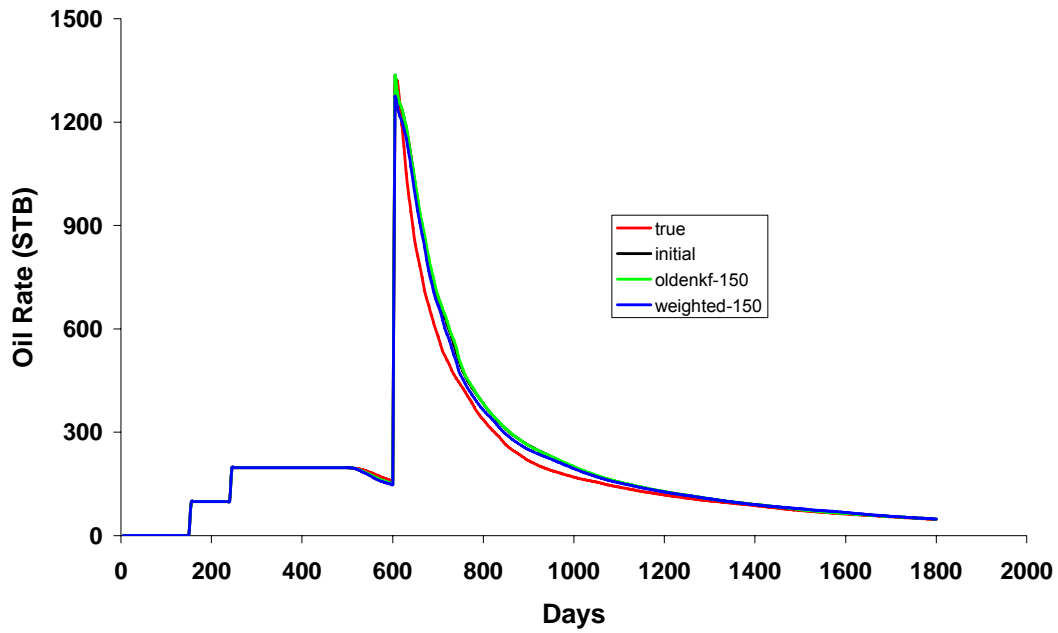
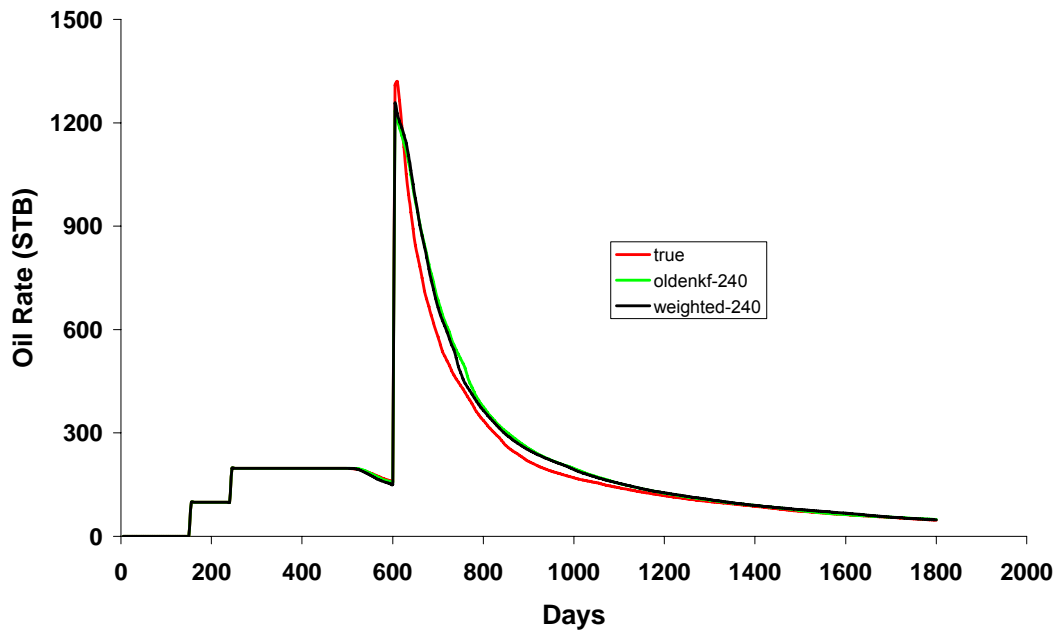


Figure 6-37: Oil rate in well P1 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (denoted as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).

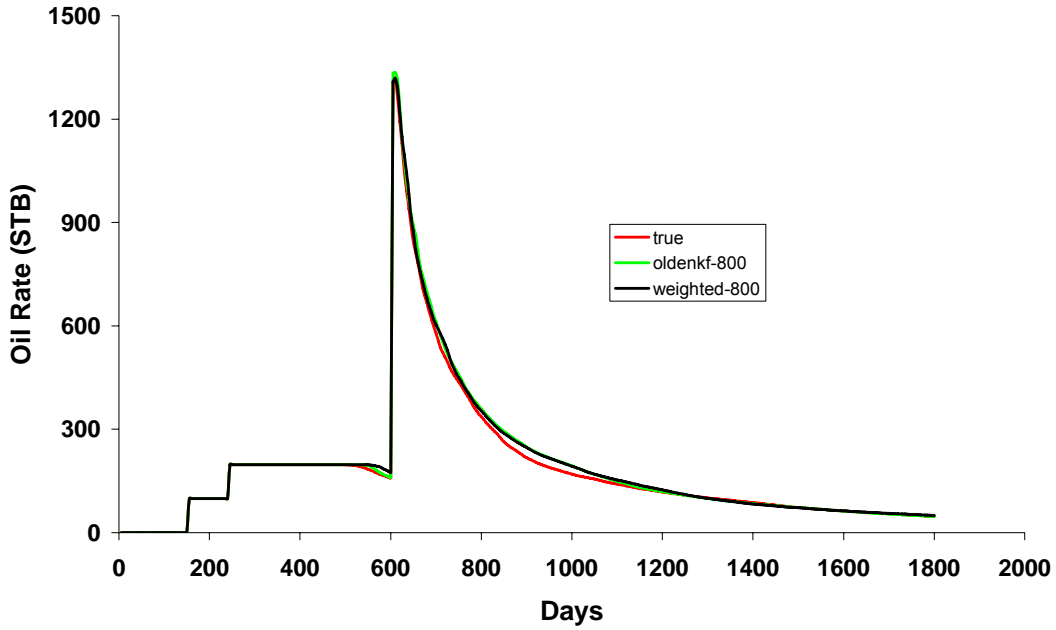
(a) Oil rate plots from reference, initial, and 150 days



(b) Oil rate plots from reference and 240 days



(c) Oil rate plots from reference and 800 days



(d) Oil rate plots from reference and 1500 days

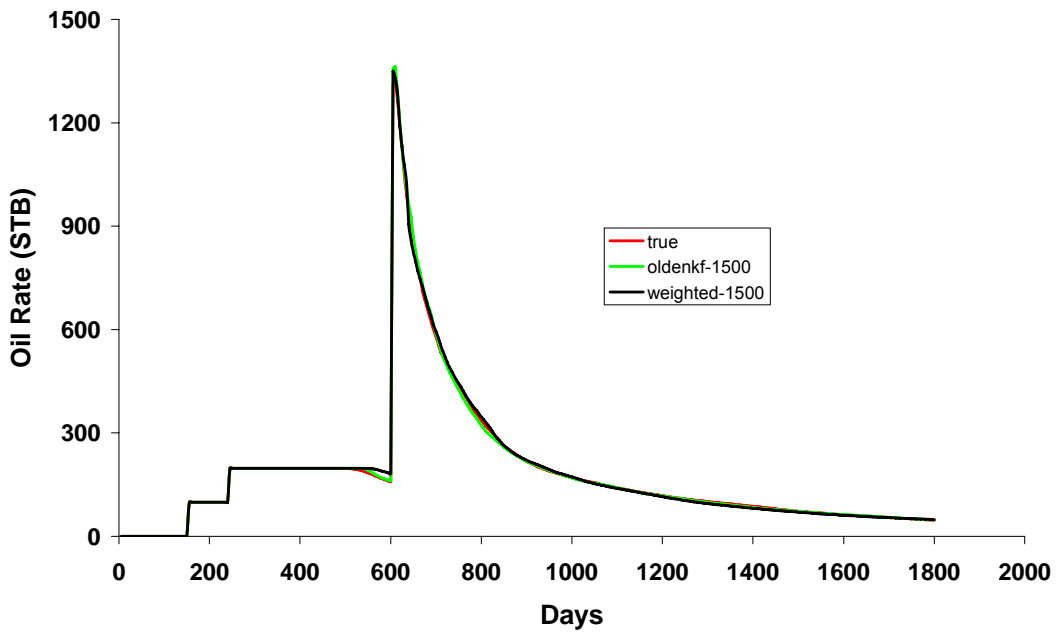
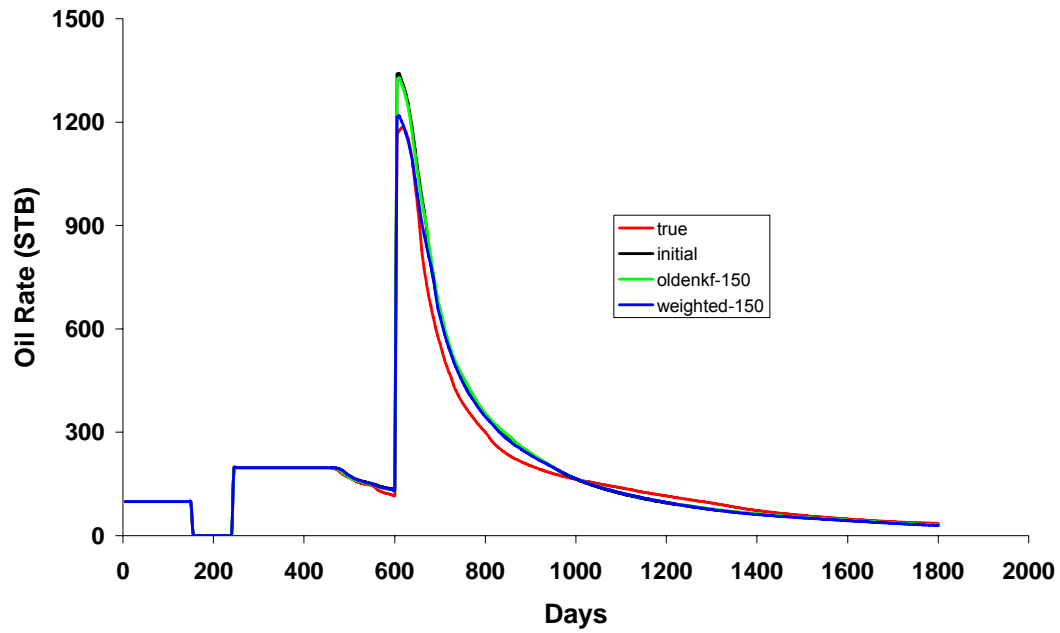
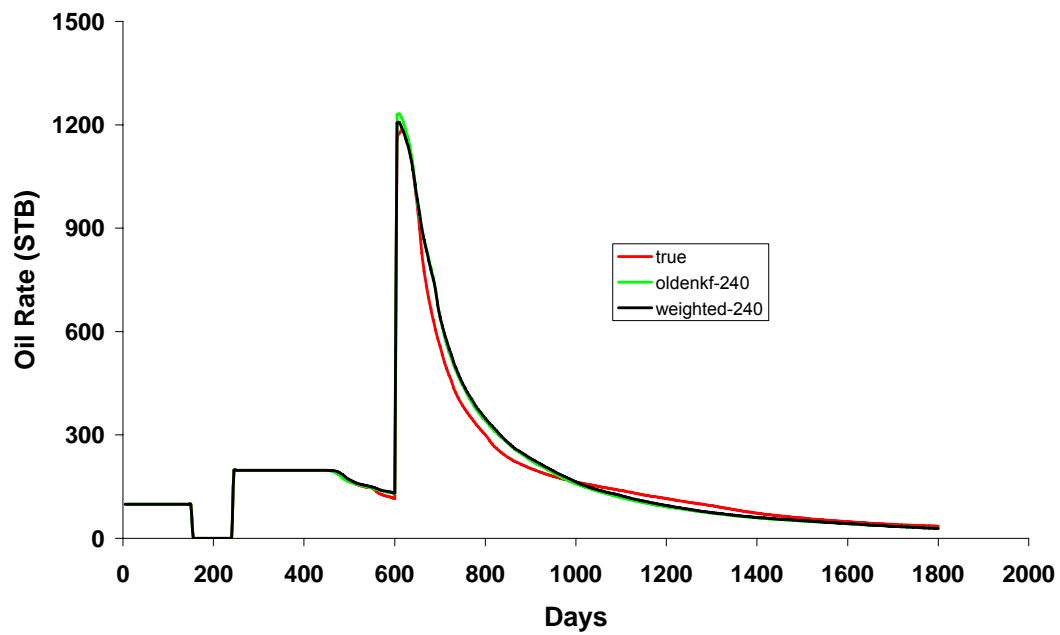


Figure 6-38: Oil rate in well P2 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (denoted as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).

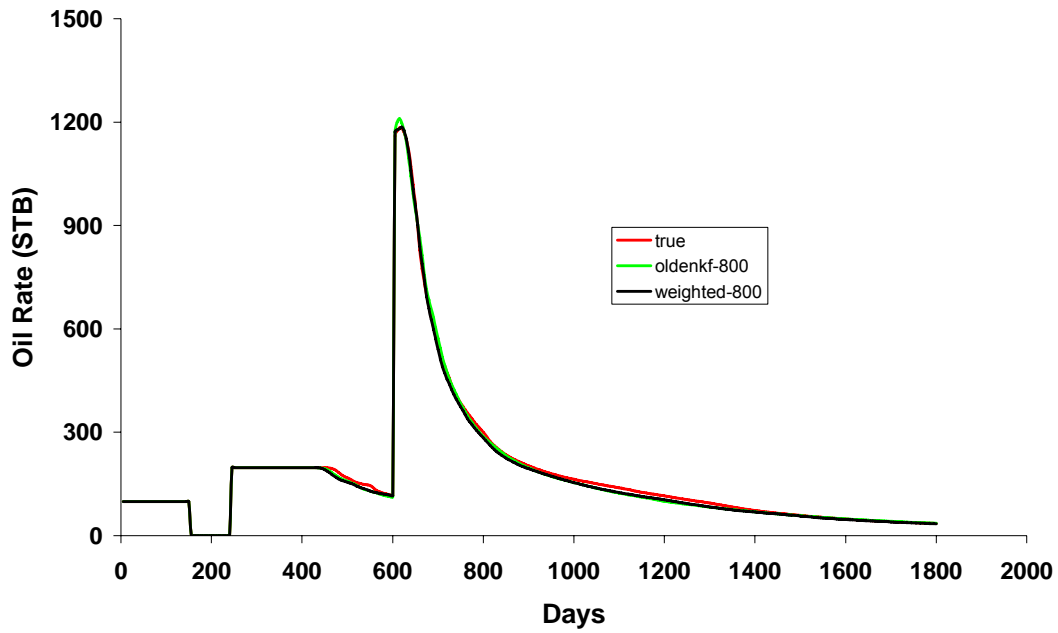
(a) Oil rate plots from reference, initial, and 150 days



(b) Oil rate plots from reference and 240 days



(c) Oil rate plots from reference and 800 days



(d) Oil rate plots from reference and 1500 days

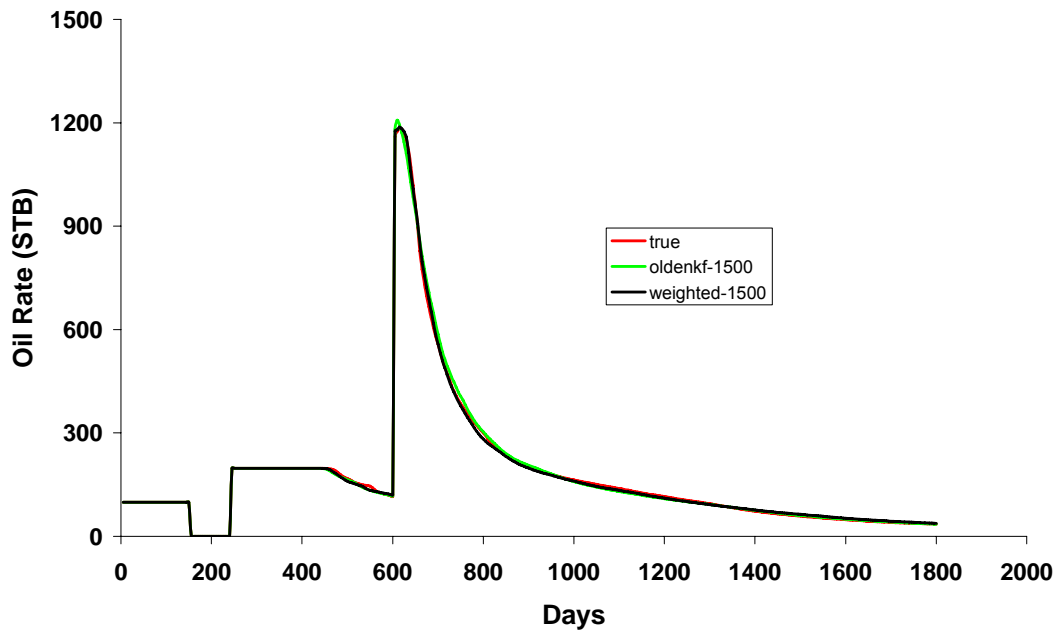
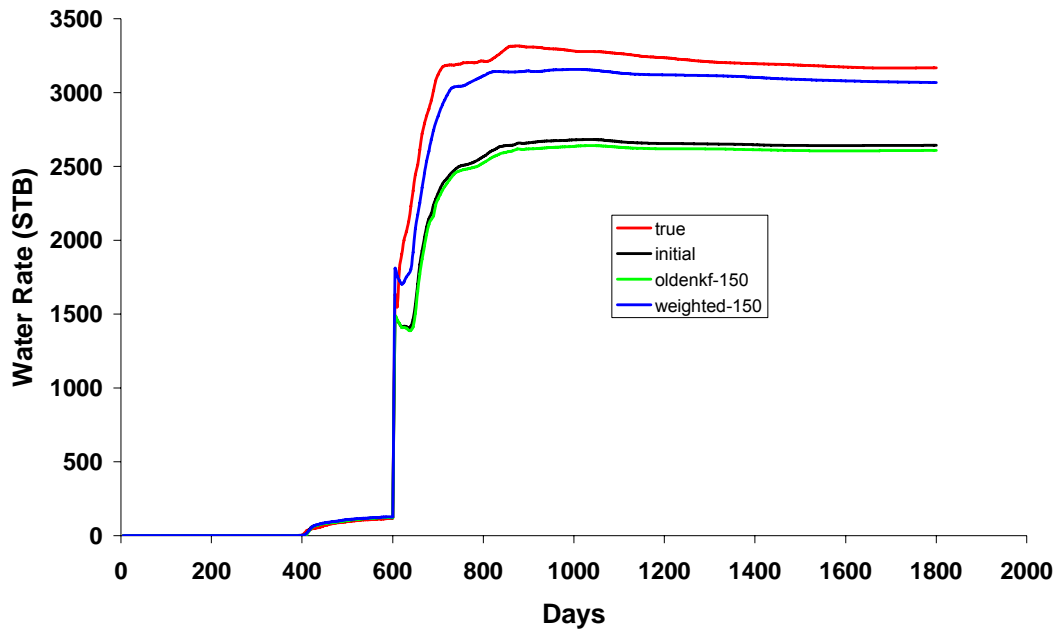
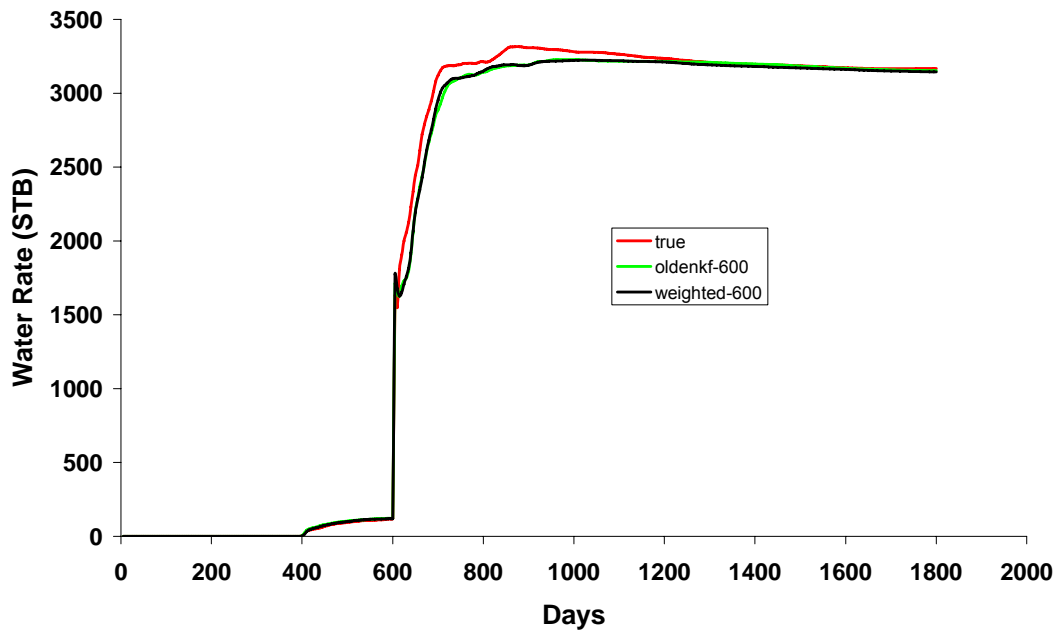


Figure 6-39: Oil rate in well P4 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (denoted as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).

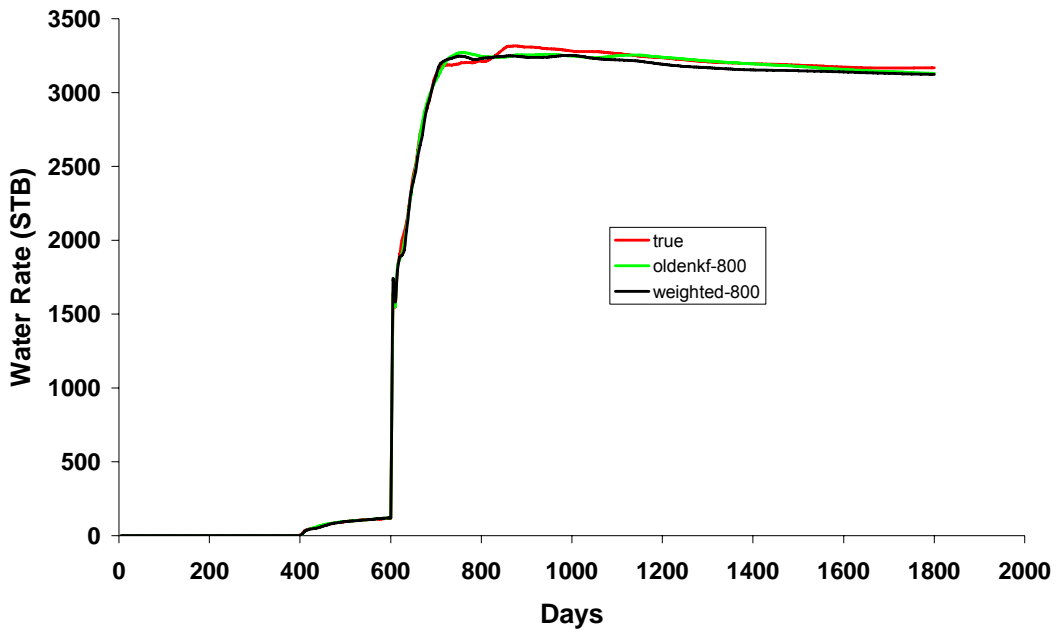
(a) Water rate plots from reference, initial, and 150 days



(b) Water rate plots from reference and 600 days



(c) Water rate plots from reference and 800 days



(d) Water rate plots from reference and 1500 days

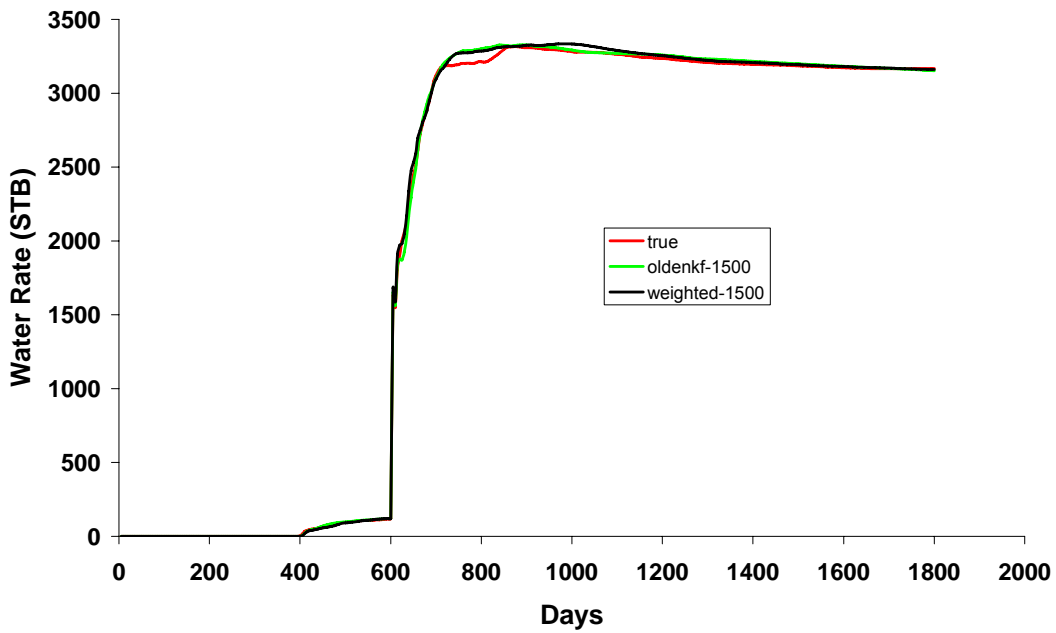
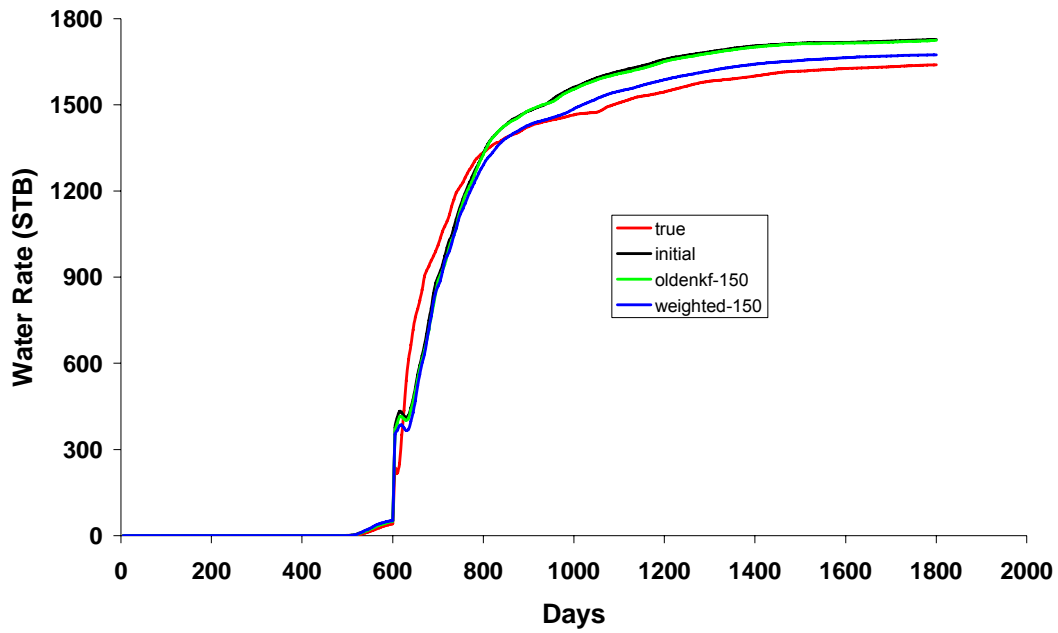
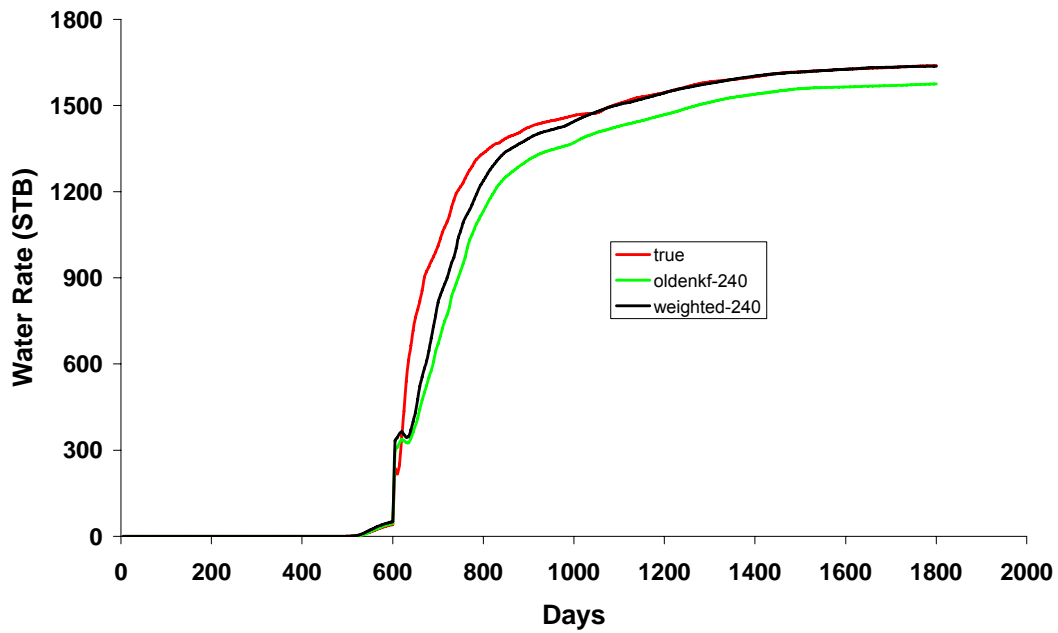


Figure 6-40: Water rate in well P1 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (denoted as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).

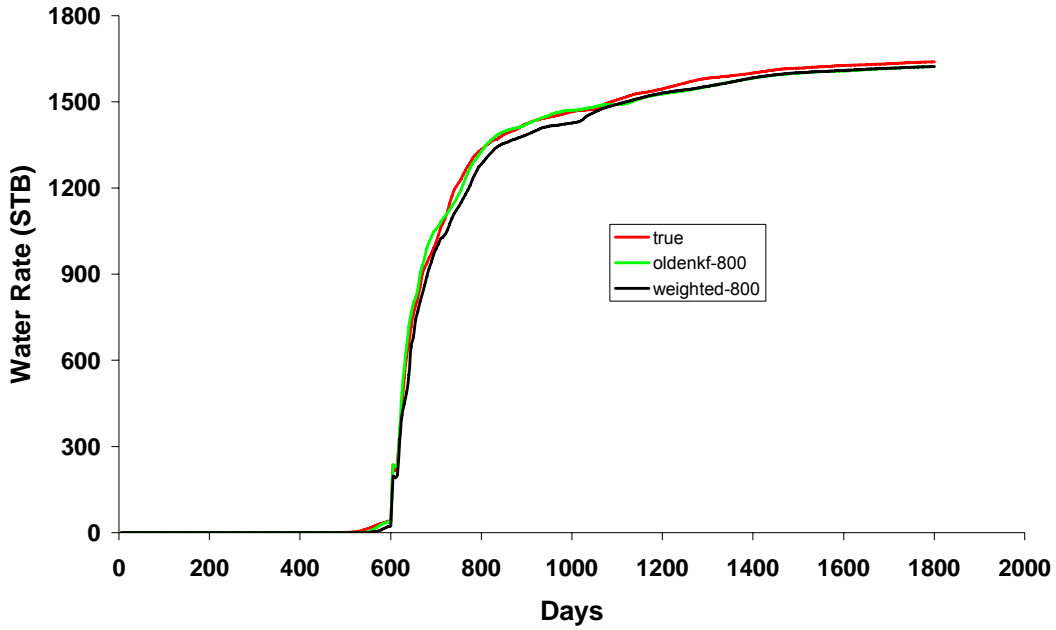
(a) Water rate plots from reference, initial, and 150 days



(b) Water rate plots from reference and 240 days



(c) Water rate plots from reference and 800 days



(d) Water rate plots from reference and 1500 days

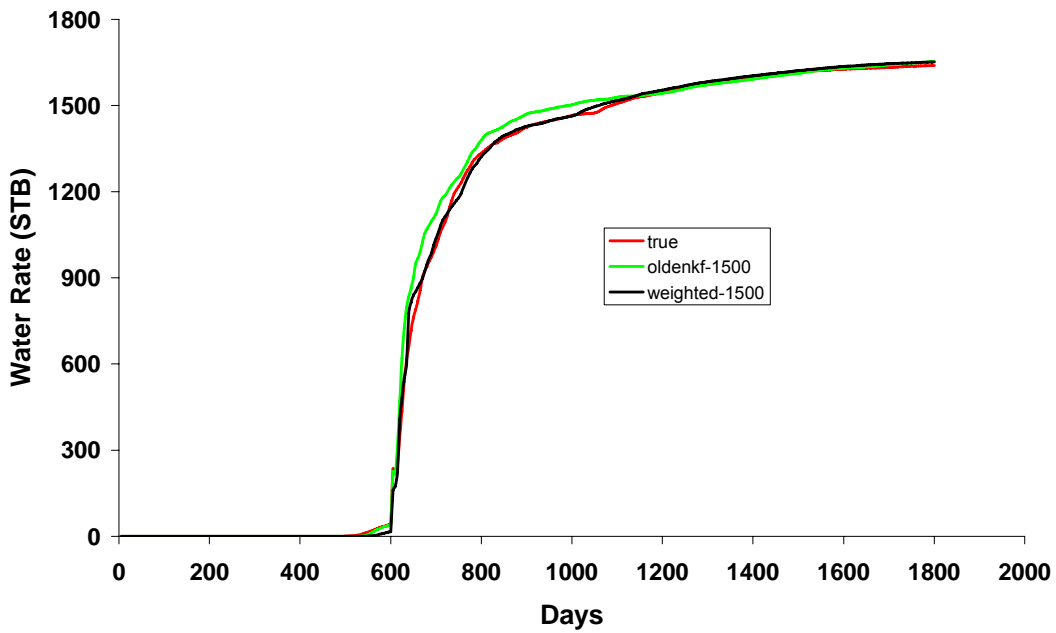
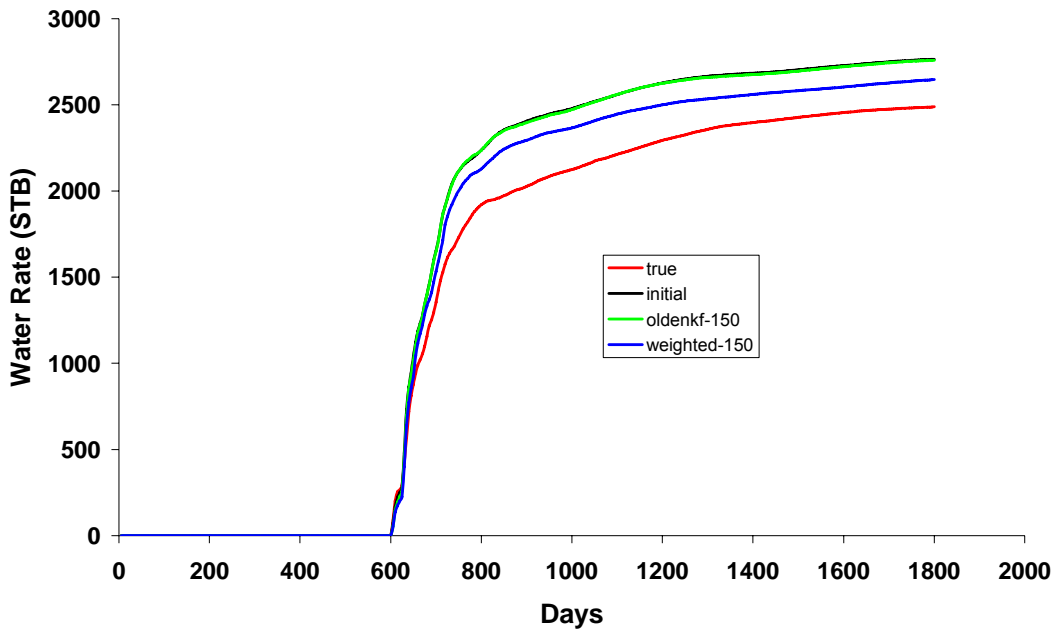
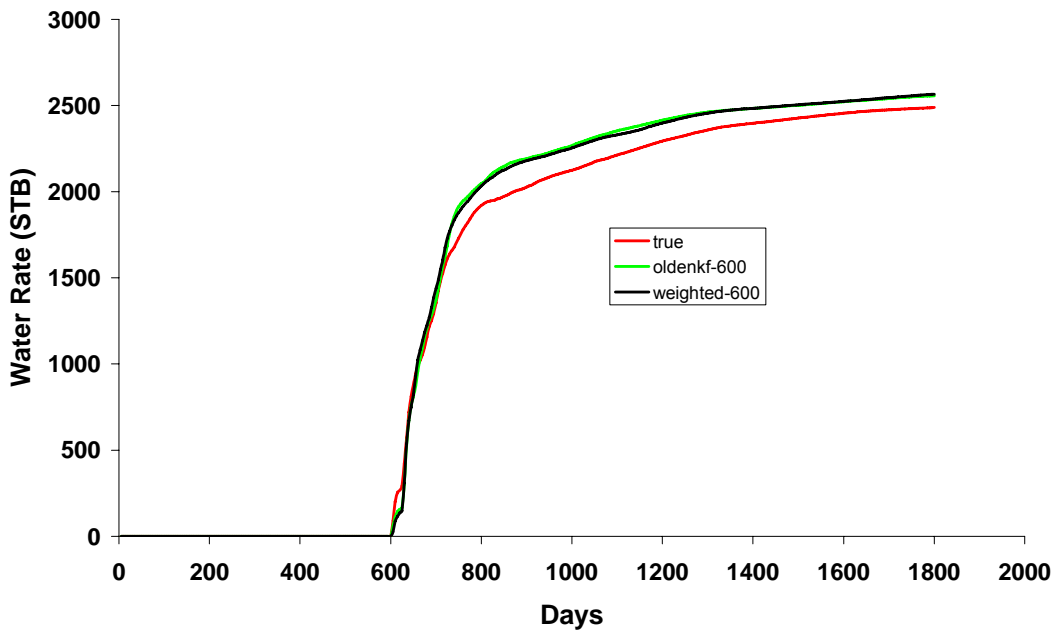


Figure 6-41: Water rate in well P2 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (denoted as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).

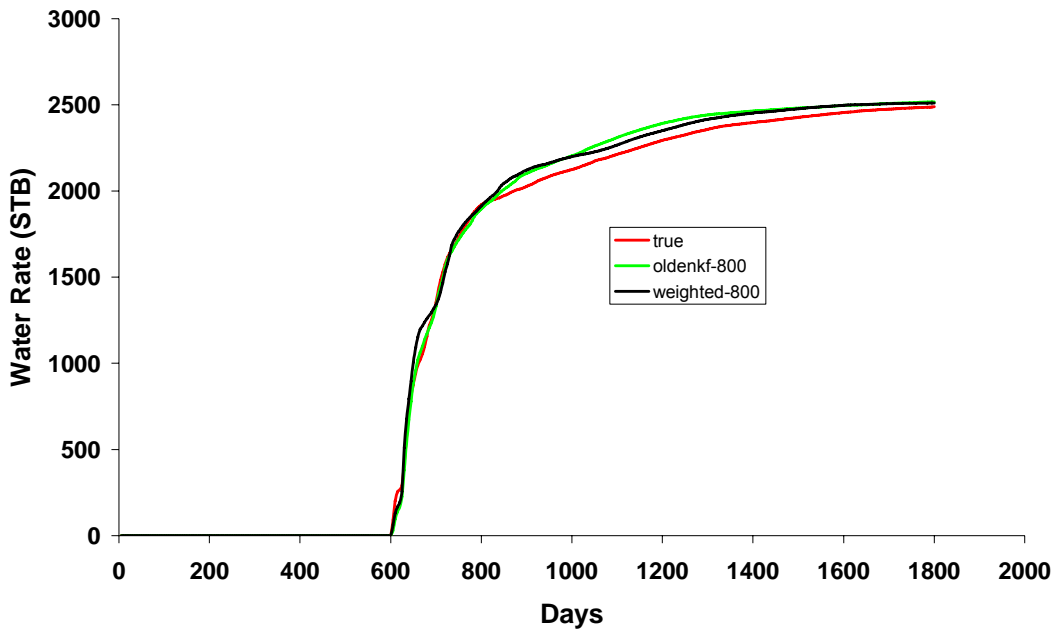
(a) Water rate plots from reference, initial, and 150 days



(b) Water rate plots from reference and 600 days



(c) Water rate plots from reference and 800 days



(d) Water rate plots from reference and 1500 days

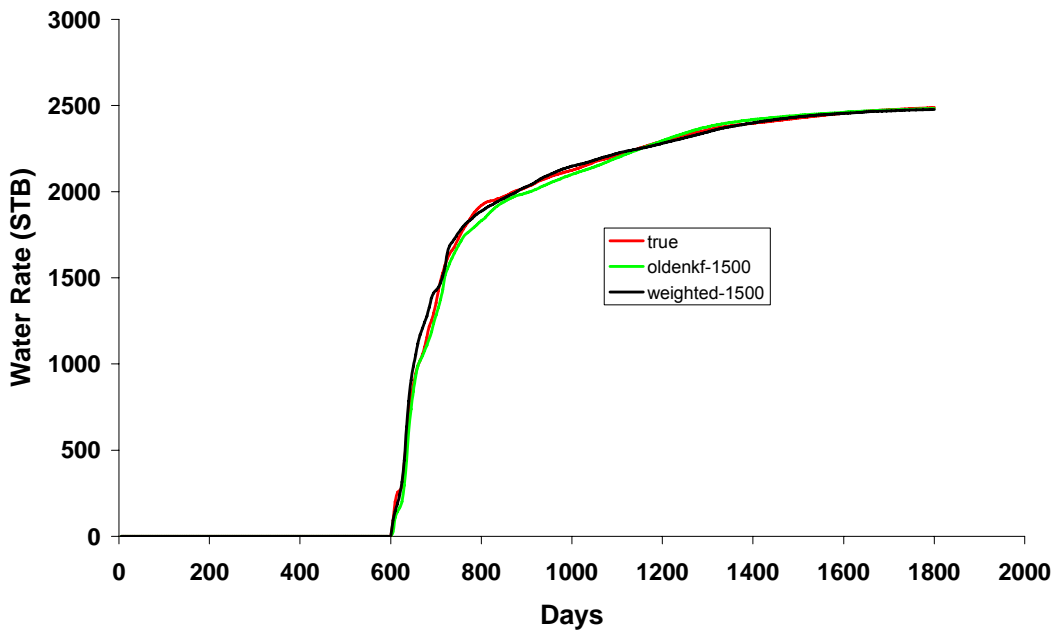
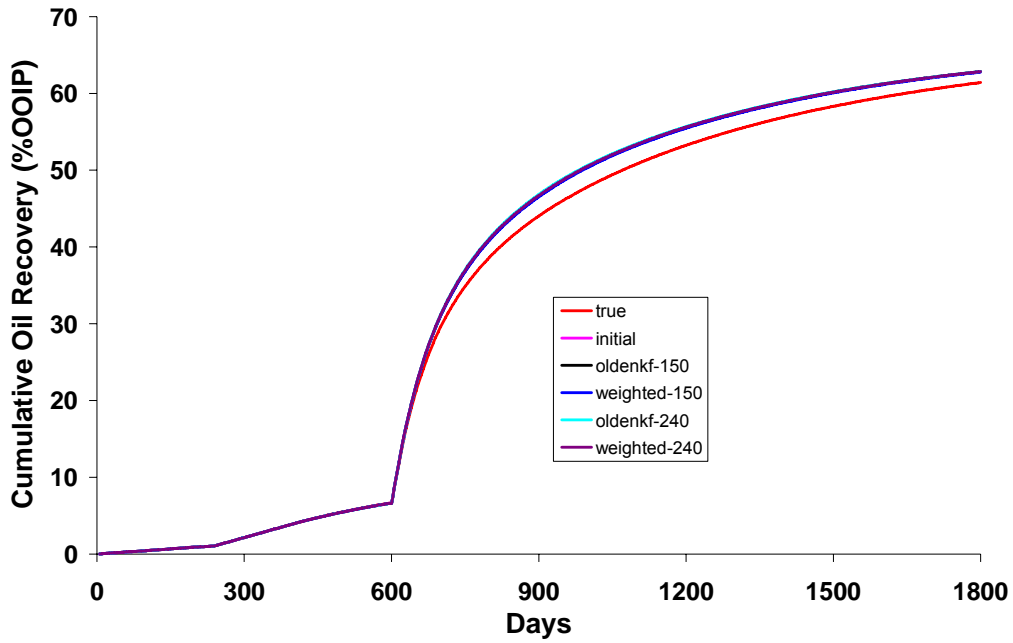


Figure 6-42: Water rate in well P3 vs. time by using reservoir models from different assimilation times in two methods, the conventional EnKF (denoted as “oldenkf”) and the weighted EnKF (denoted as “weighted”), and the reference (denoted as “true”).

(a) Cumulative oil recovery plots from reference, initial, 150 days, and 240 days



(b) Cumulative oil recovery plots from reference and 600 days

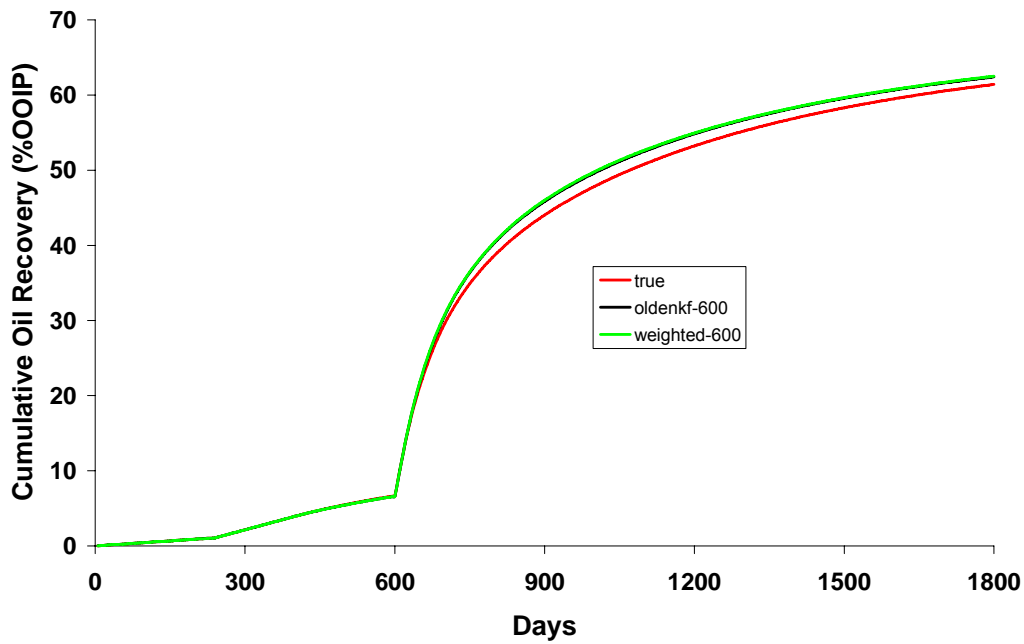
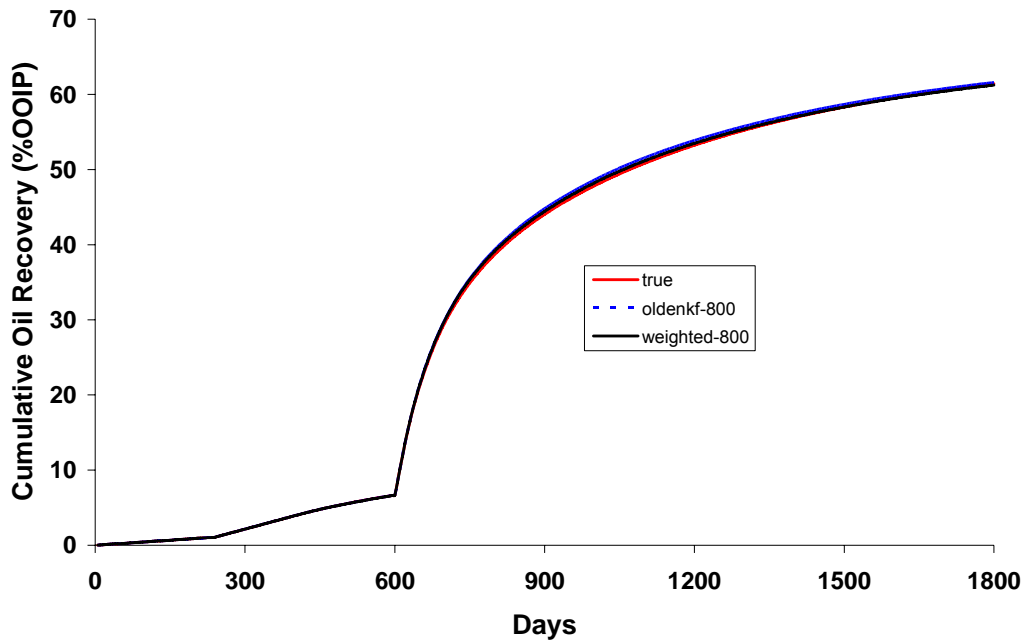


Figure 6-43: Cumulative oil recovery in terms of original oil in place from initial, 150, 240, and 600 days in the conventional EnKF method (denoted as “oldenkf”) and the weighted EnKF method (denoted as “weighted”), and the reference (denoted as “true”).

(a) Cumulative oil recovery plots from reference and 800 days



(b) Enlarged part of plot (a)

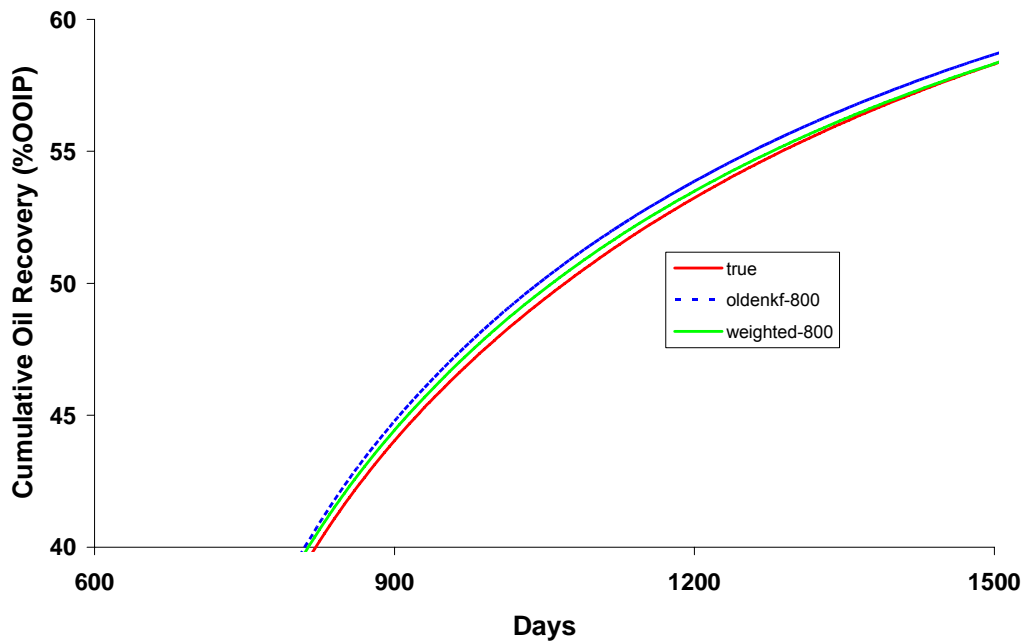
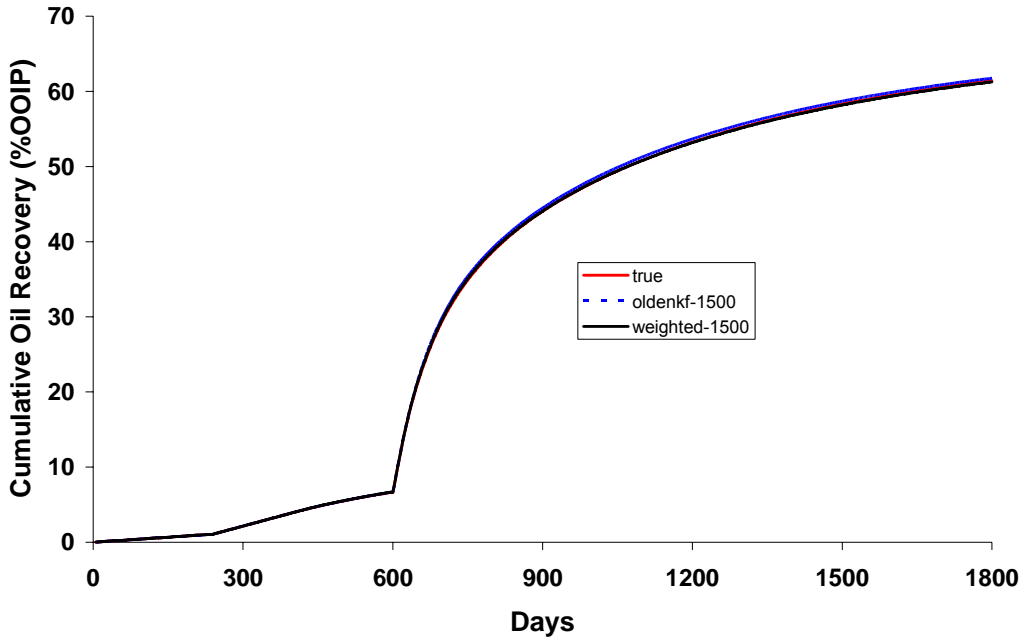


Figure 6-44: Cumulative oil recovery in terms of original oil in place from 800 days in the conventional EnKF method (denoted as “oldenkf”) and the weighted EnKF method (denoted as “weighted”), and the reference (denoted as “true”).

(a) Cumulative oil recovery plots from reference and 1500 days



(b) Enlarged part of plot (a)

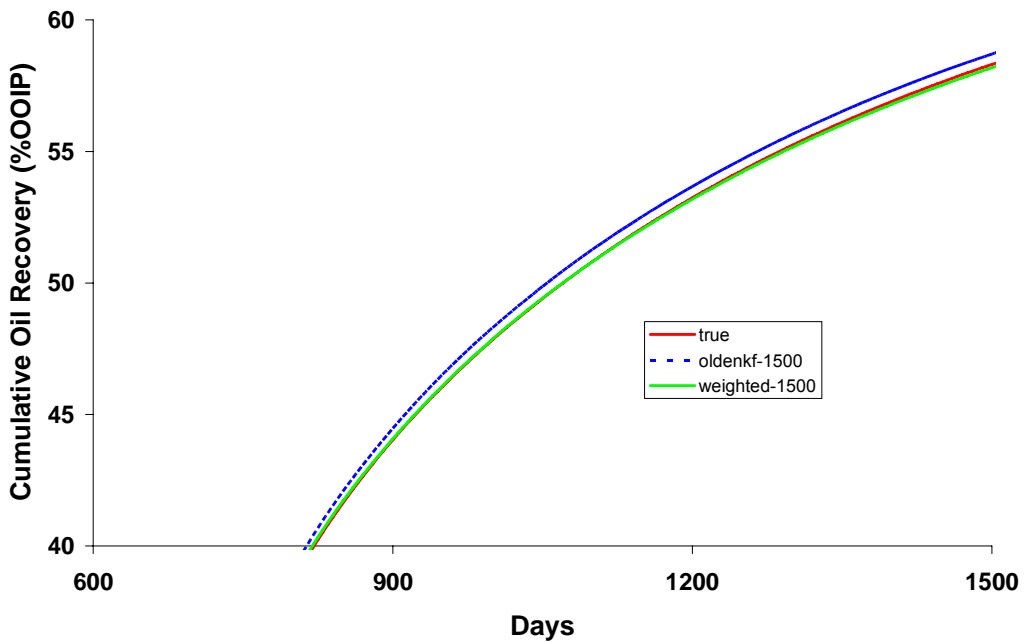


Figure 6-45: Cumulative oil recovery in terms of original oil in place from 1500 days in the conventional EnKF method (denoted as “oldenkf”) and the weighted EnKF method (denoted as “weighted”), and the reference (denoted as “true”).

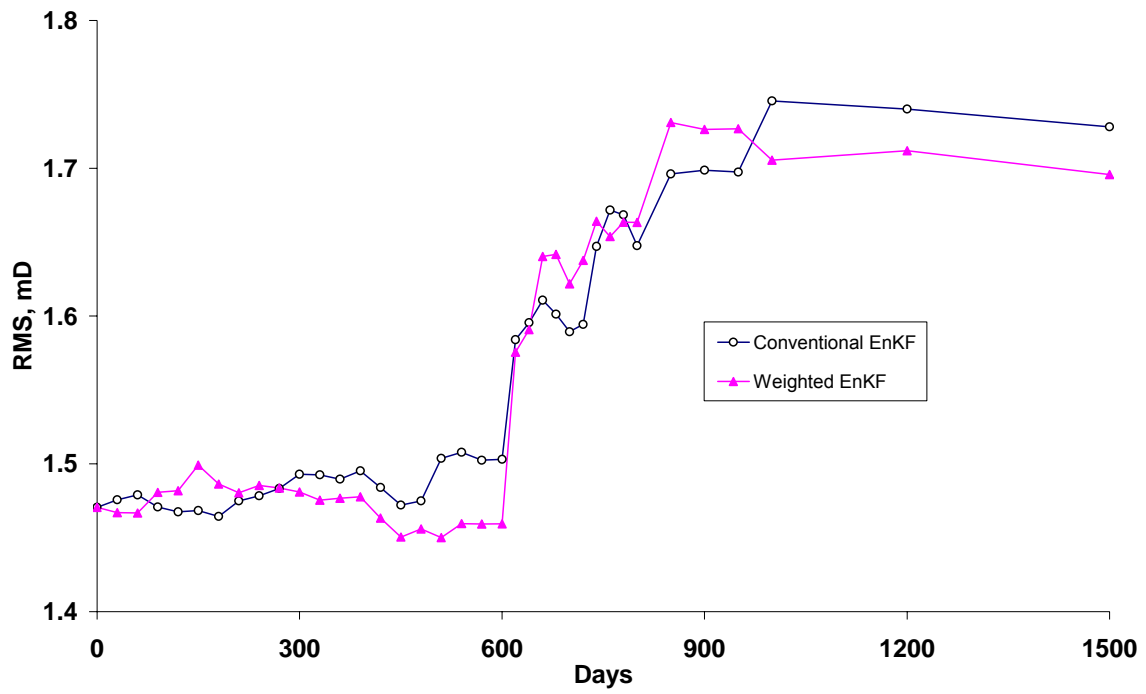


Figure 6-46: Root mean square in the eleventh layer versus time in two methods, the conventional EnKF method (denoted as “Conventional EnKF”) and the weighted EnKF method (denoted as “Weighted EnKF”).

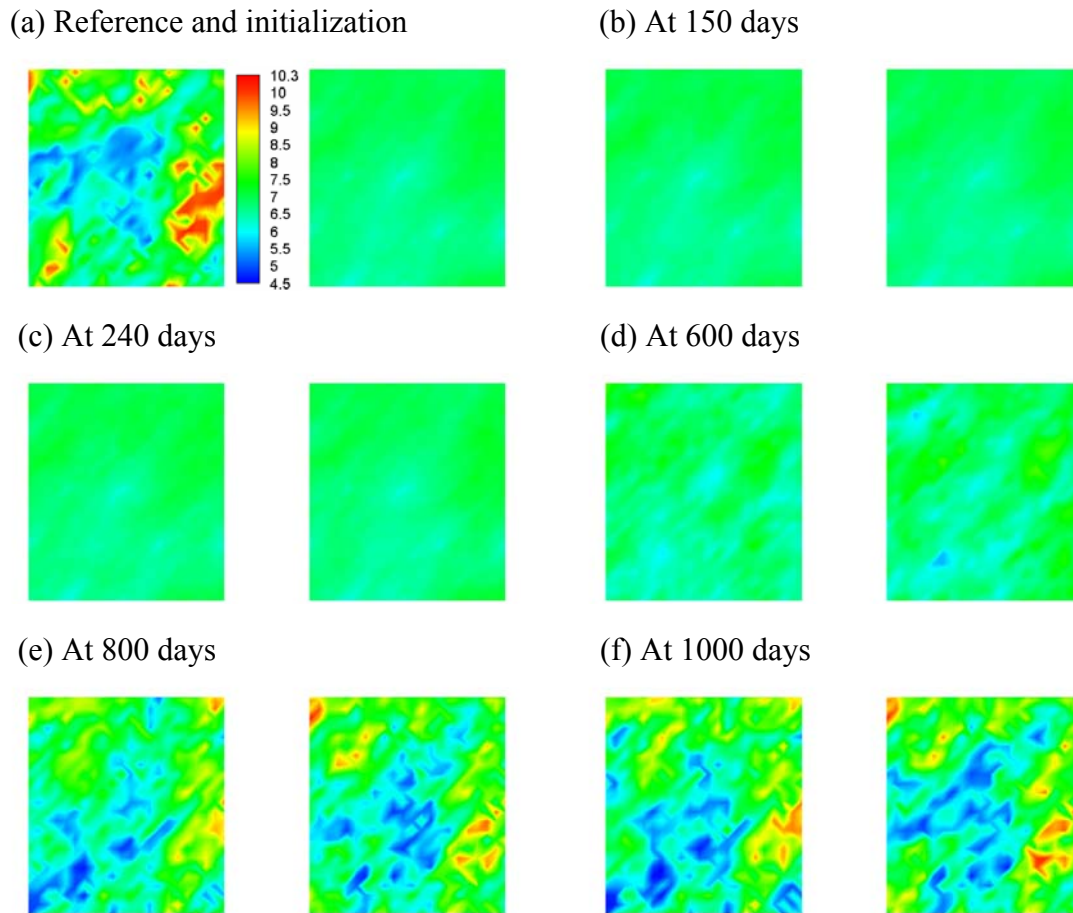


Figure 6-47: First-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

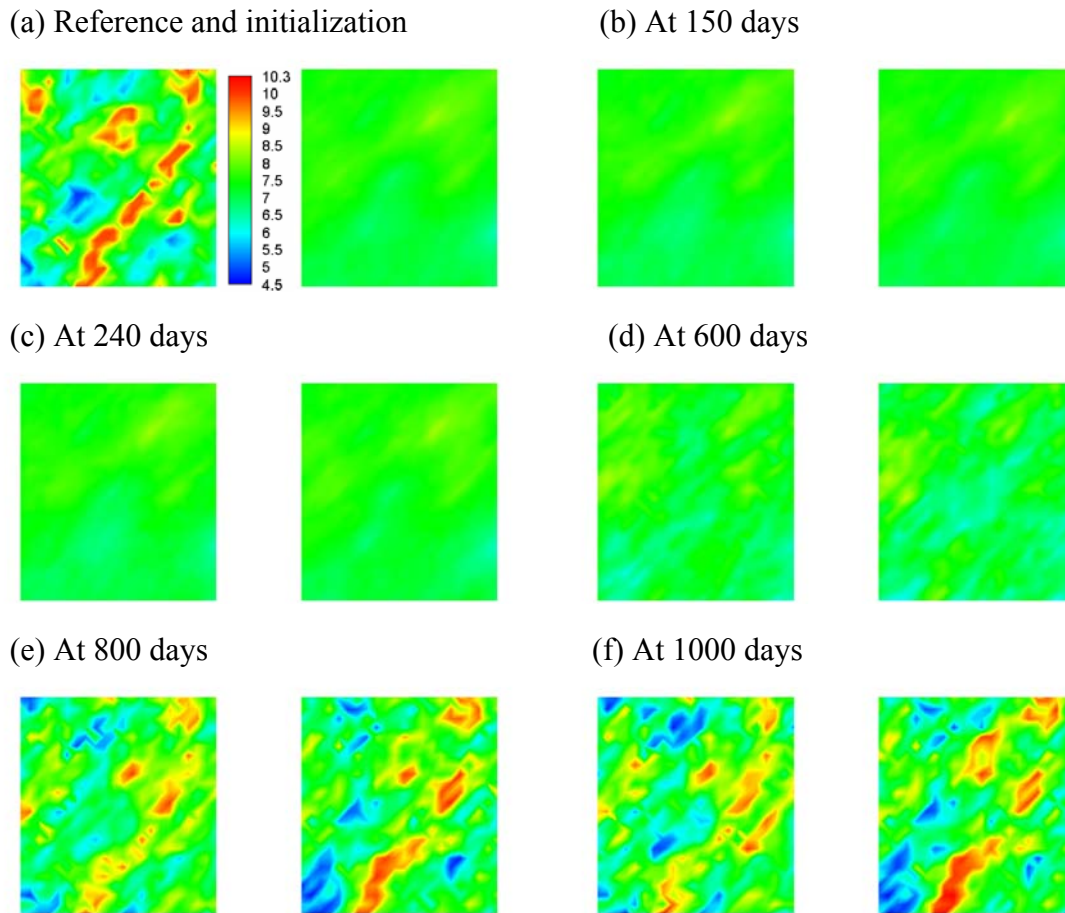


Figure 6-48: Second-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

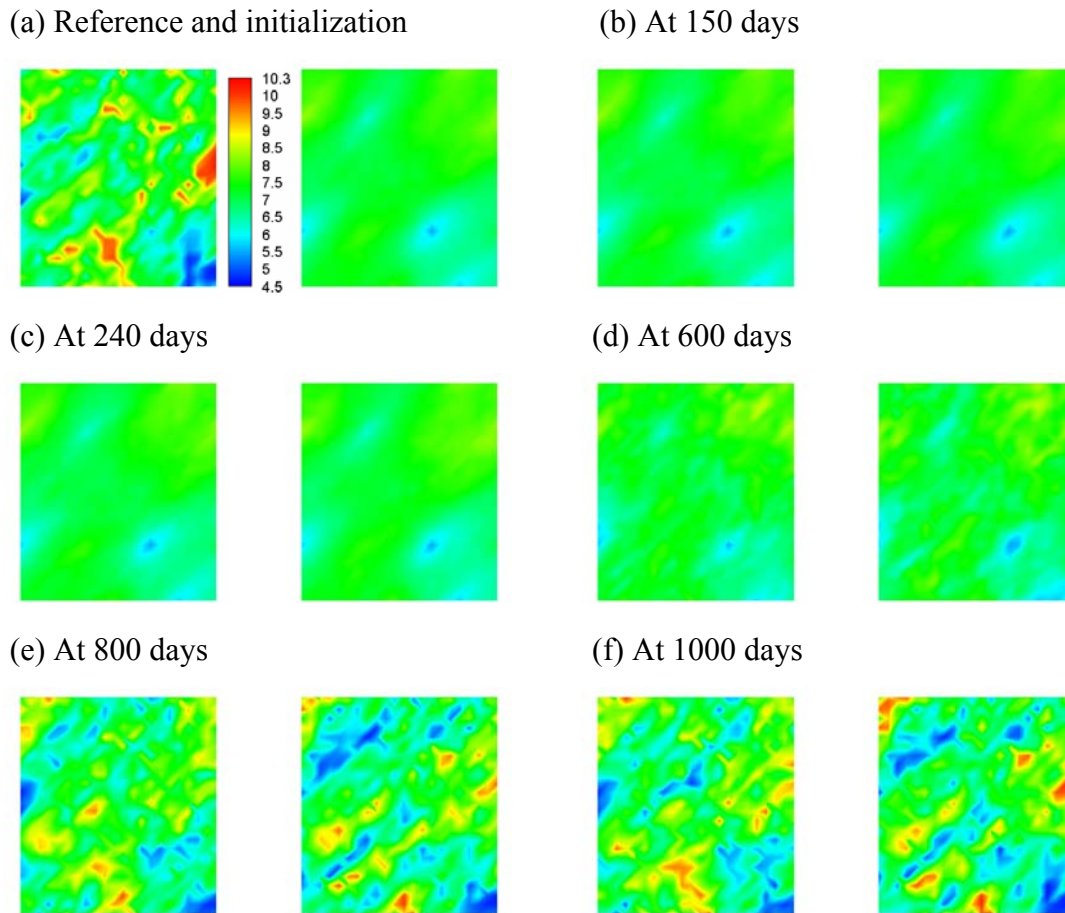


Figure 6-49: Third-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

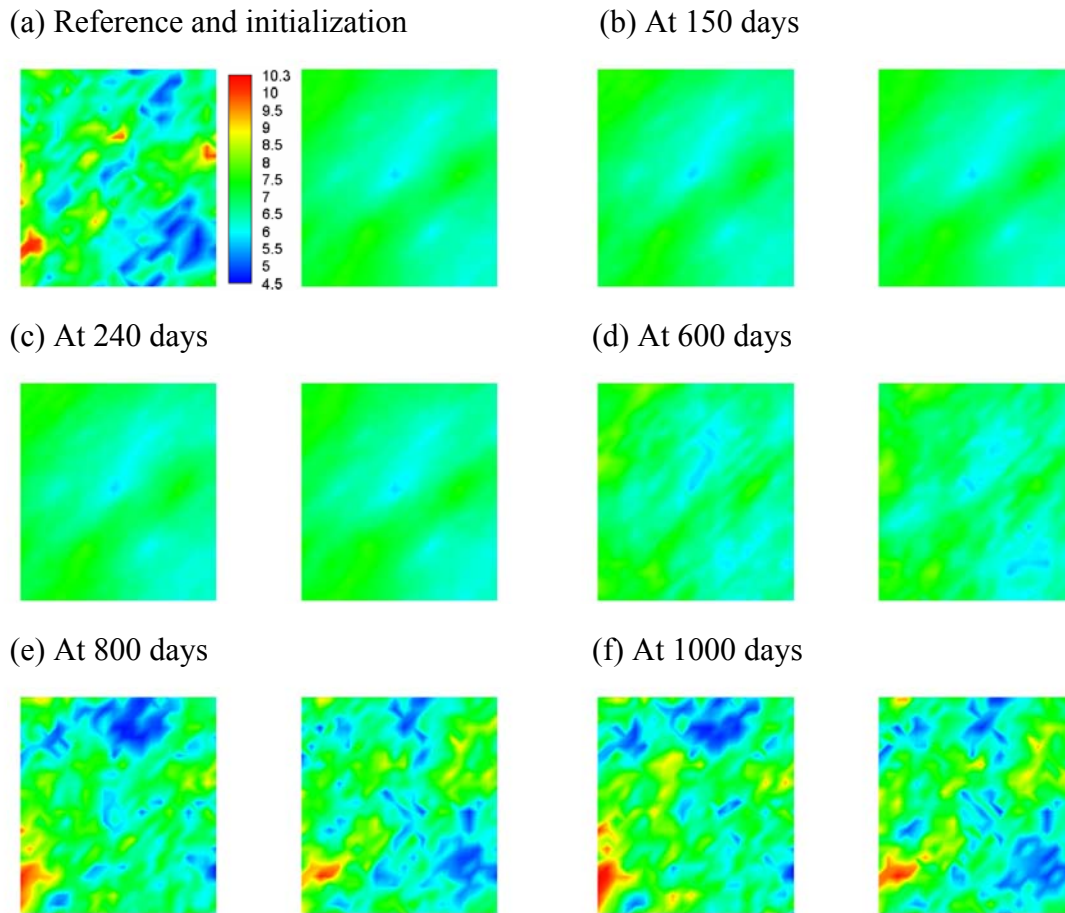


Figure 6-50: Fourth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

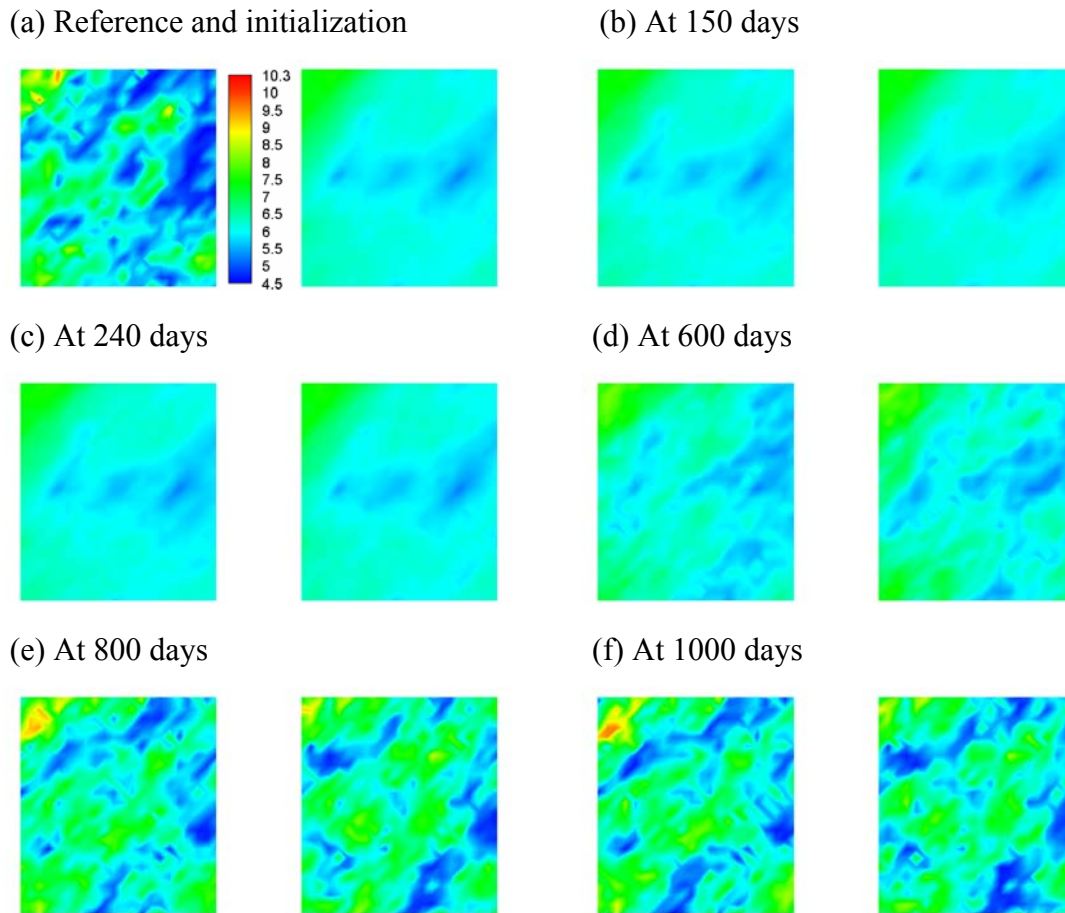


Figure 6-51: Fifth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

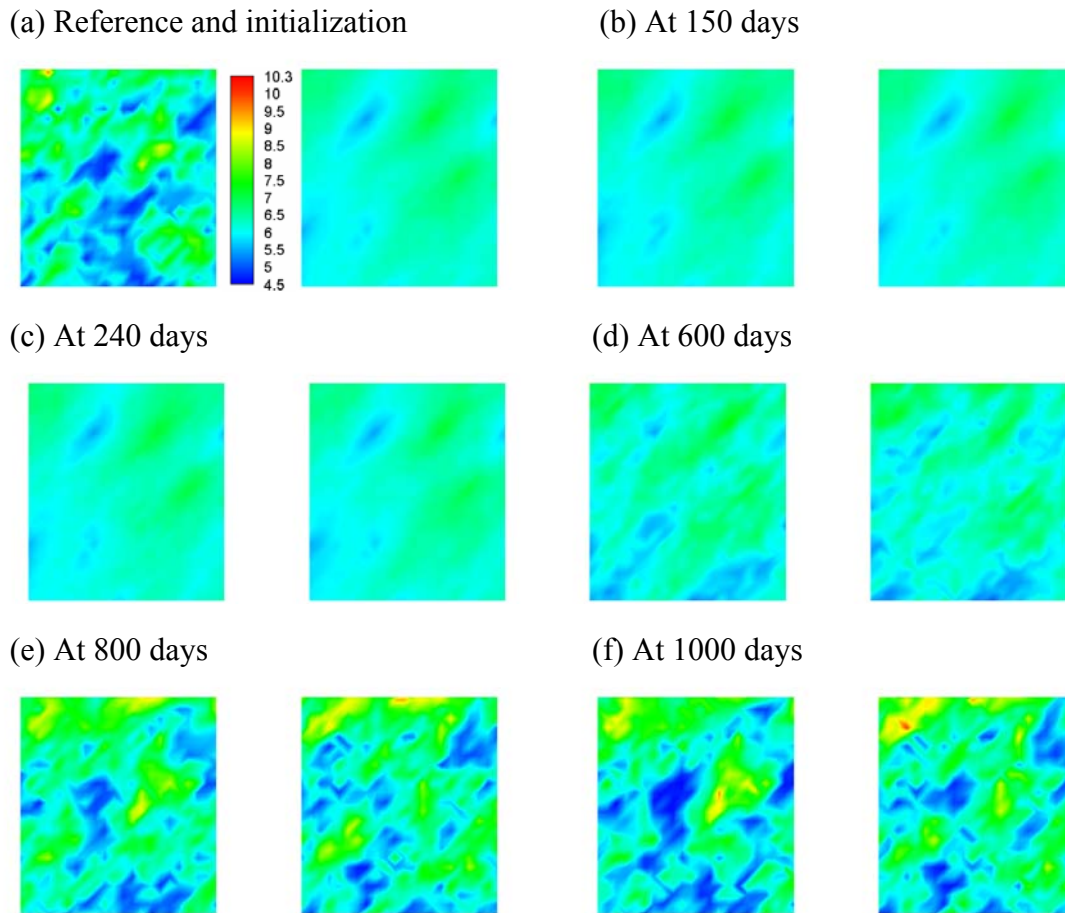


Figure 6-52: Sixth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

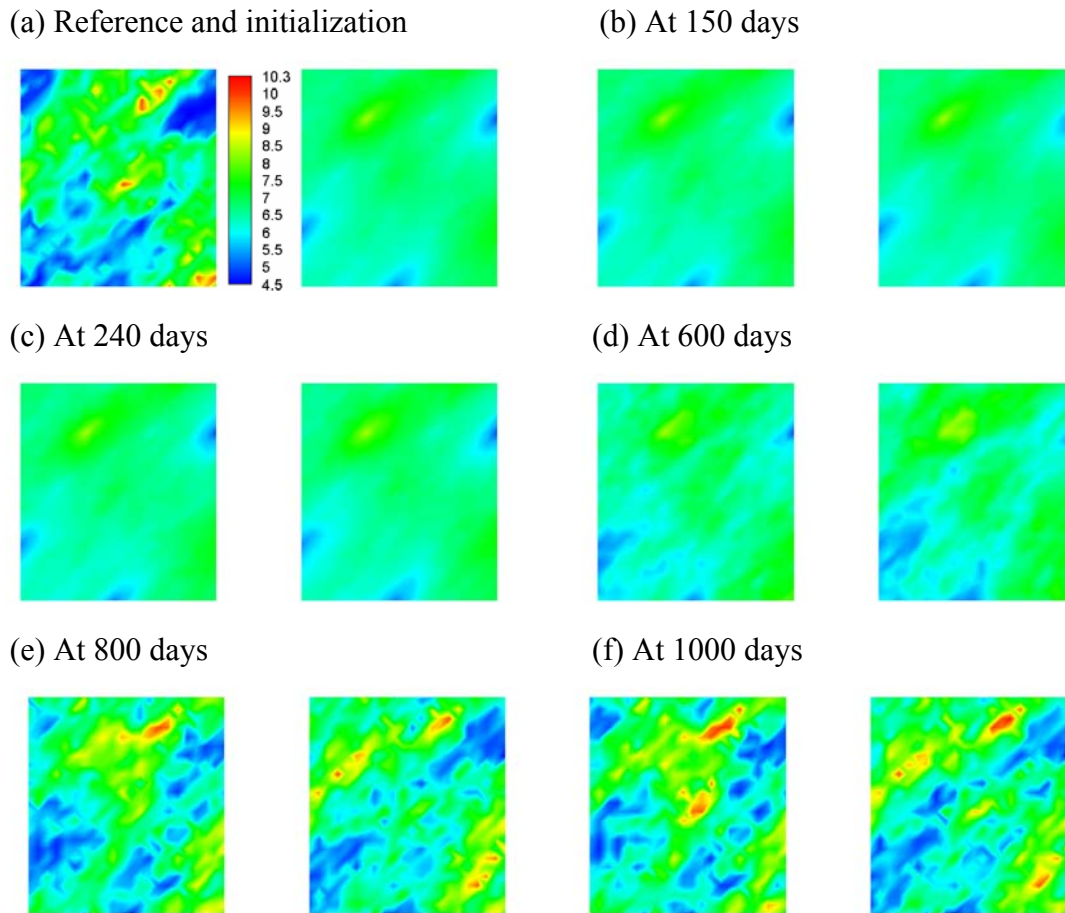


Figure 6-53: Seventh-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

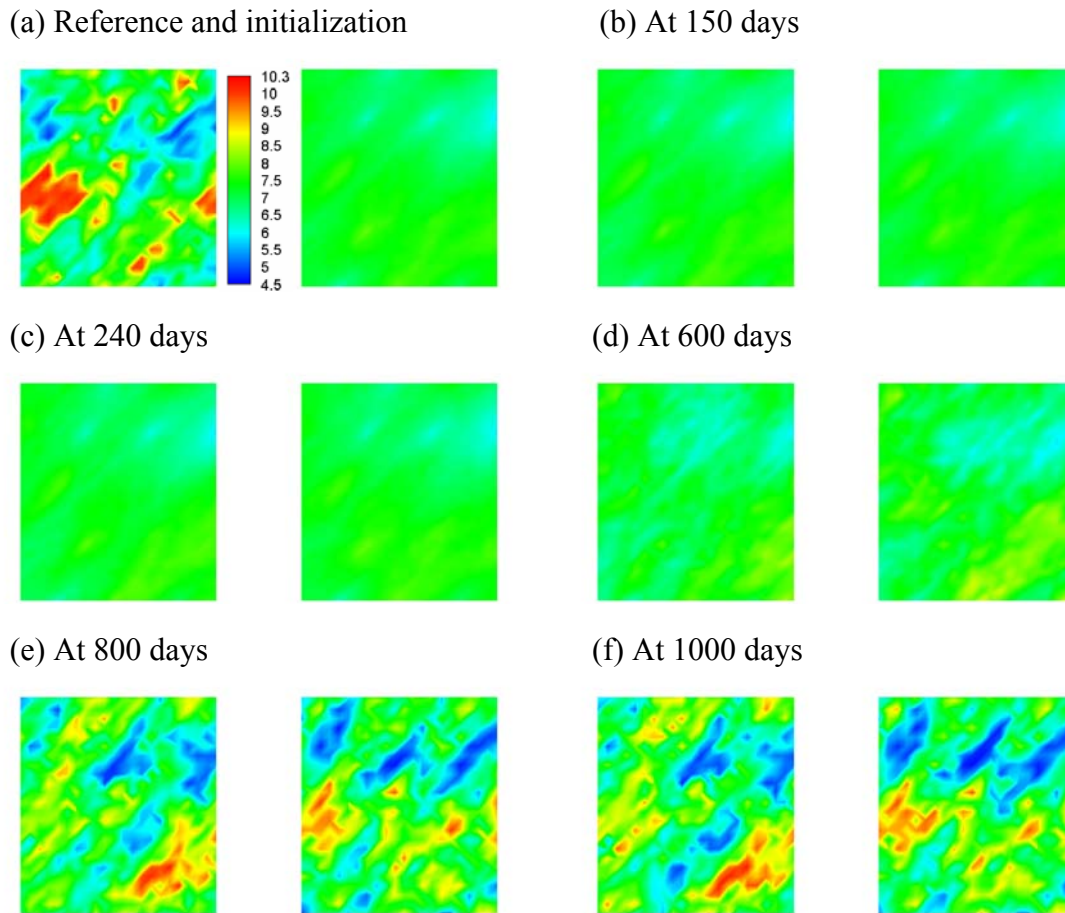


Figure 6-54: Eighth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

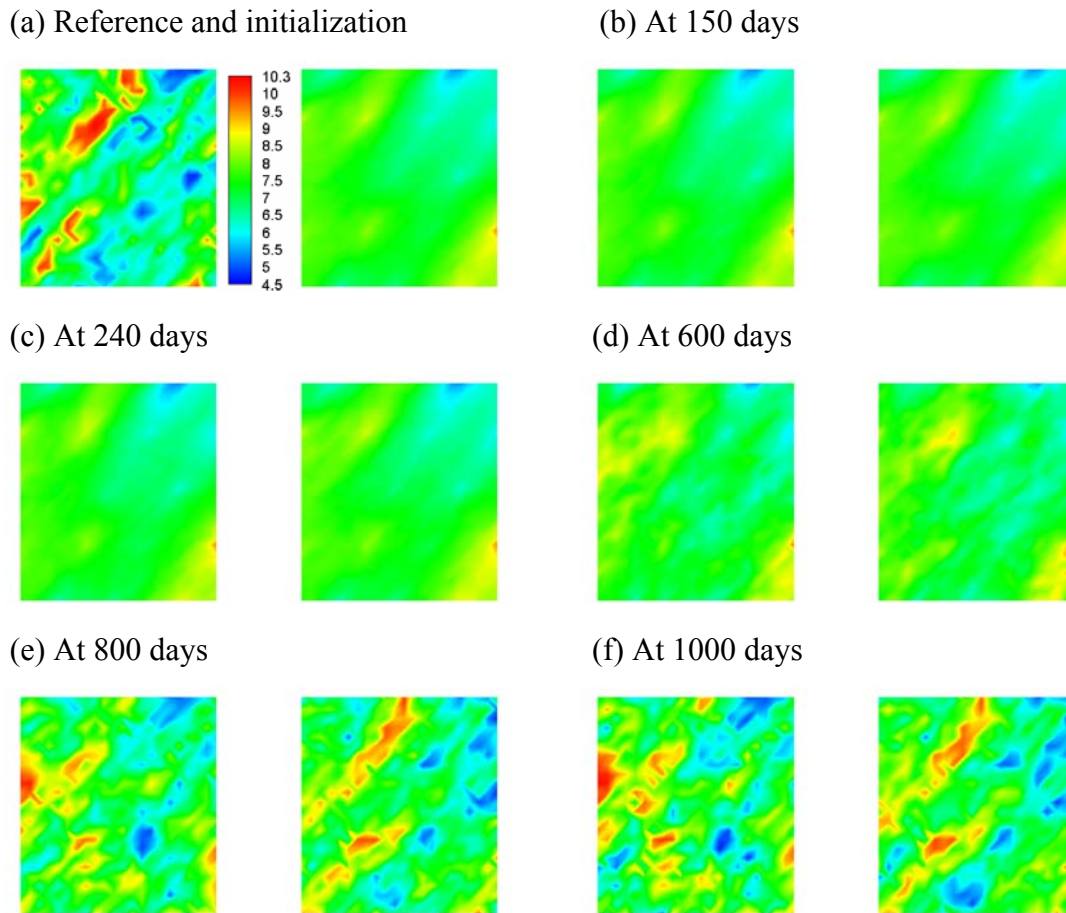


Figure 6-55: Ninth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

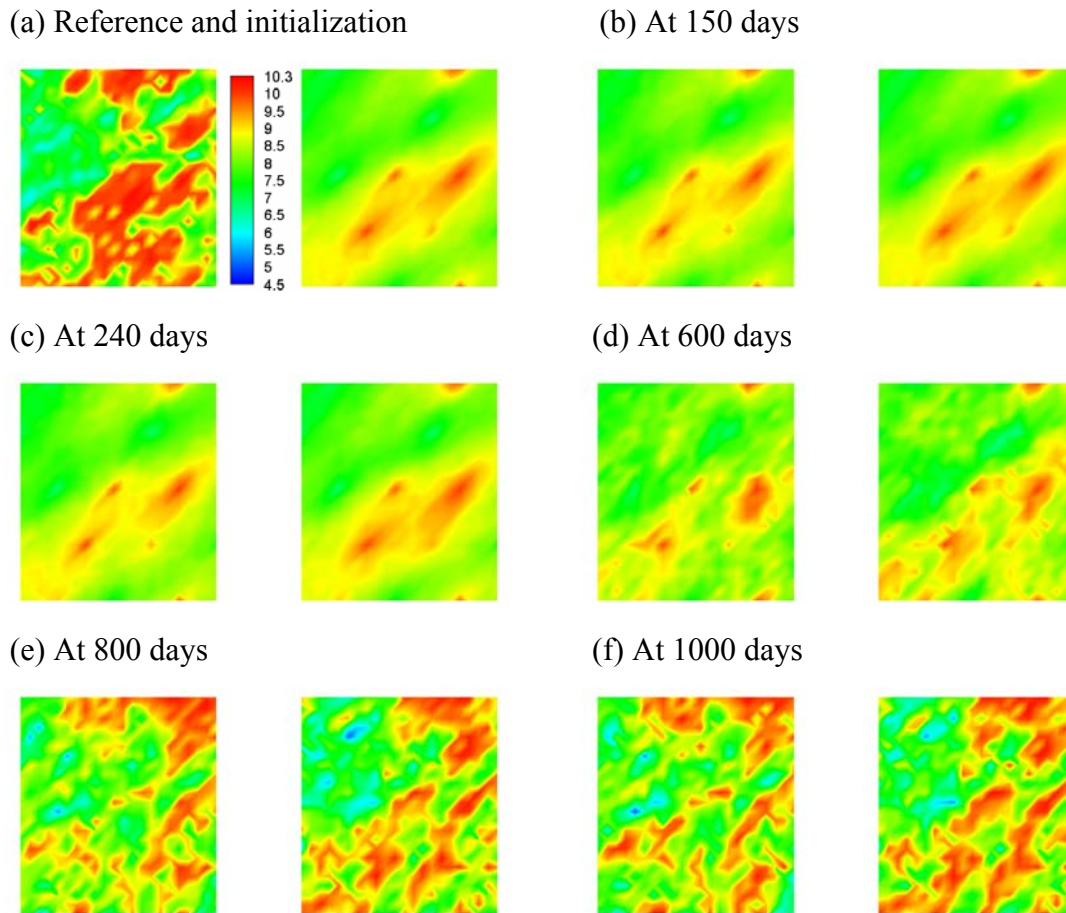


Figure 6-56: Tenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

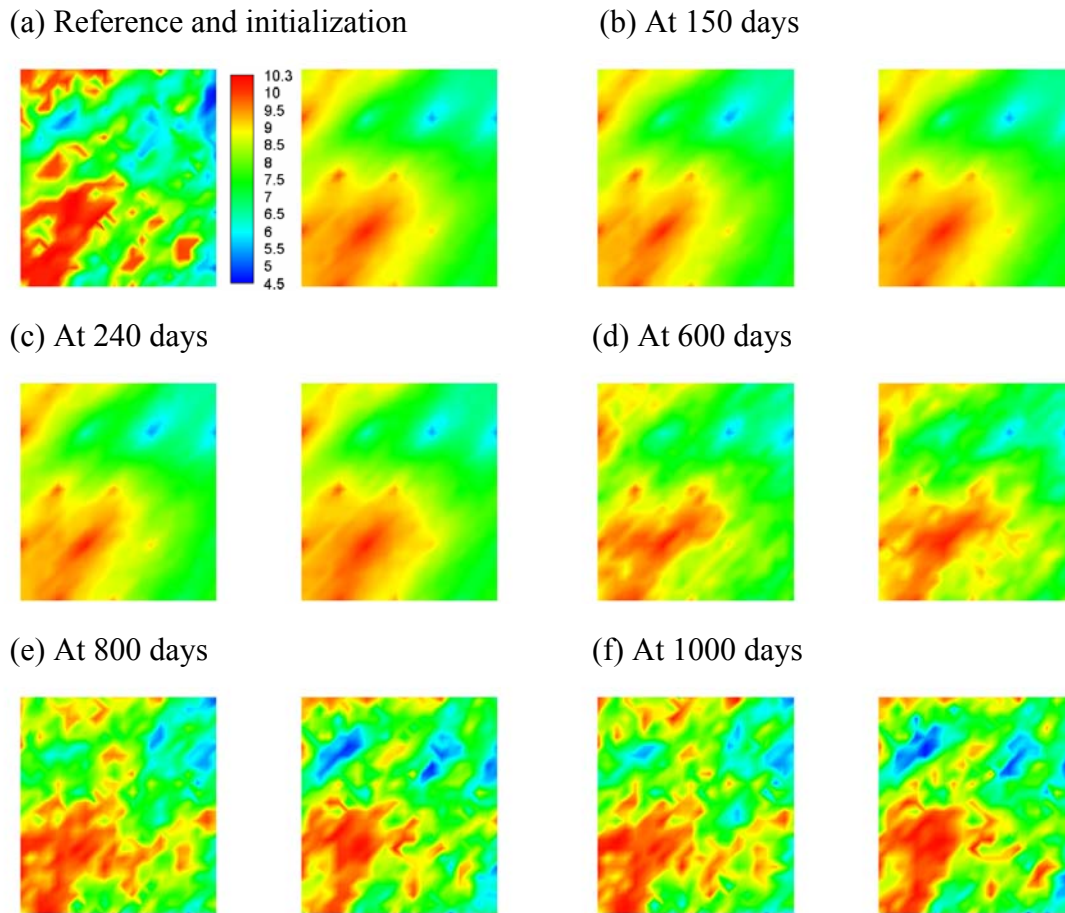


Figure 6-57: Eleventh-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

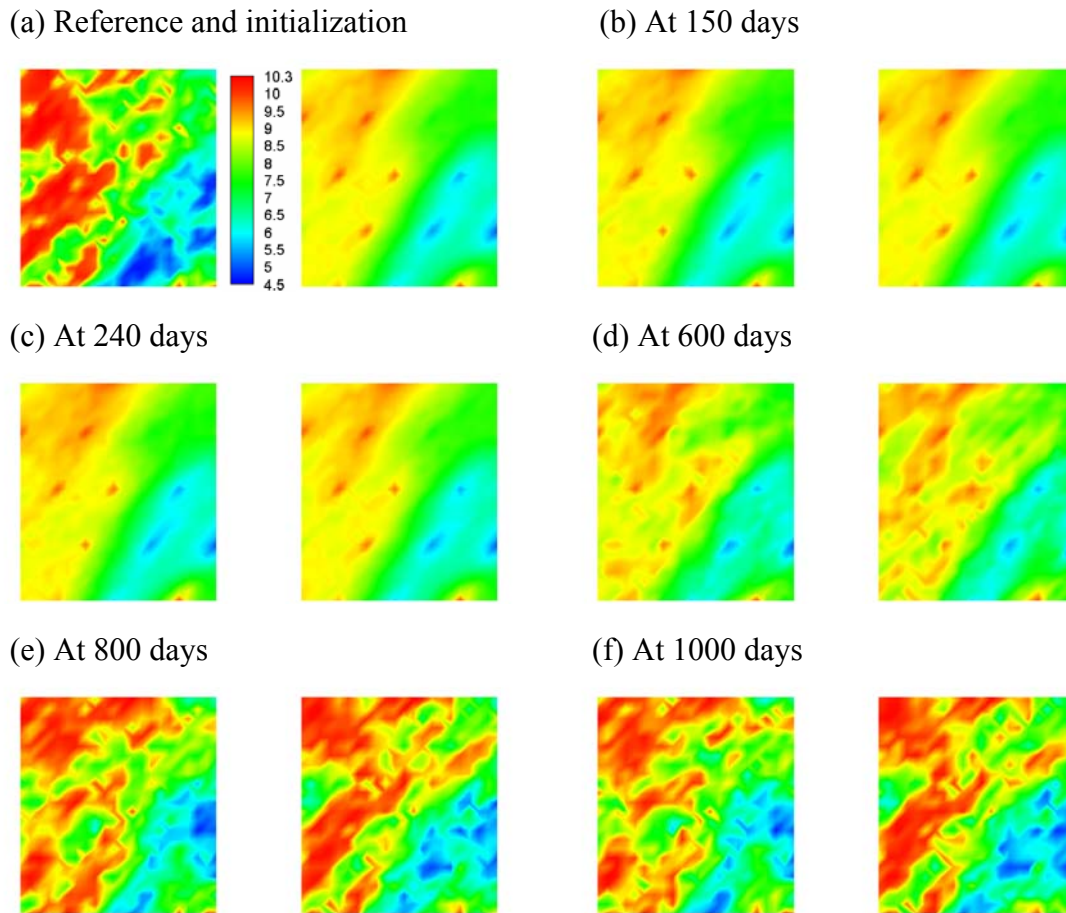


Figure 6-58: Twelfth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

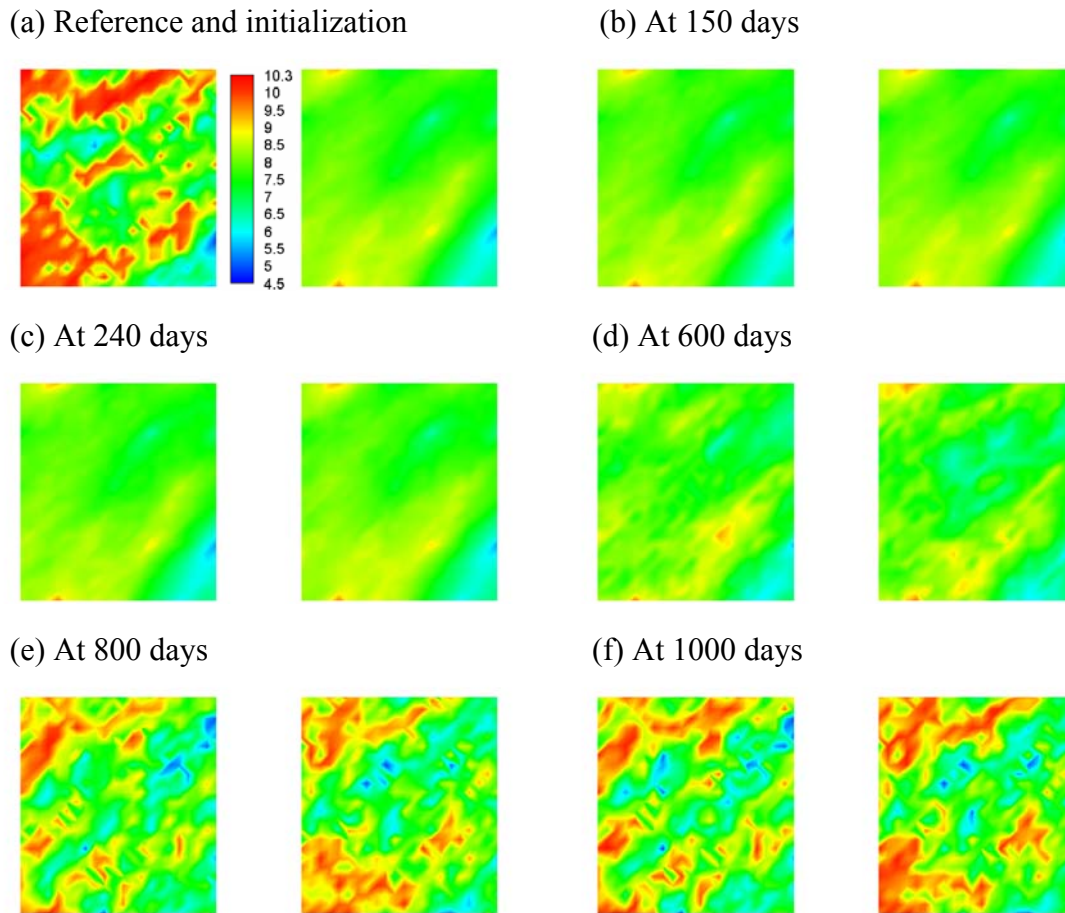


Figure 6-59: Thirteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

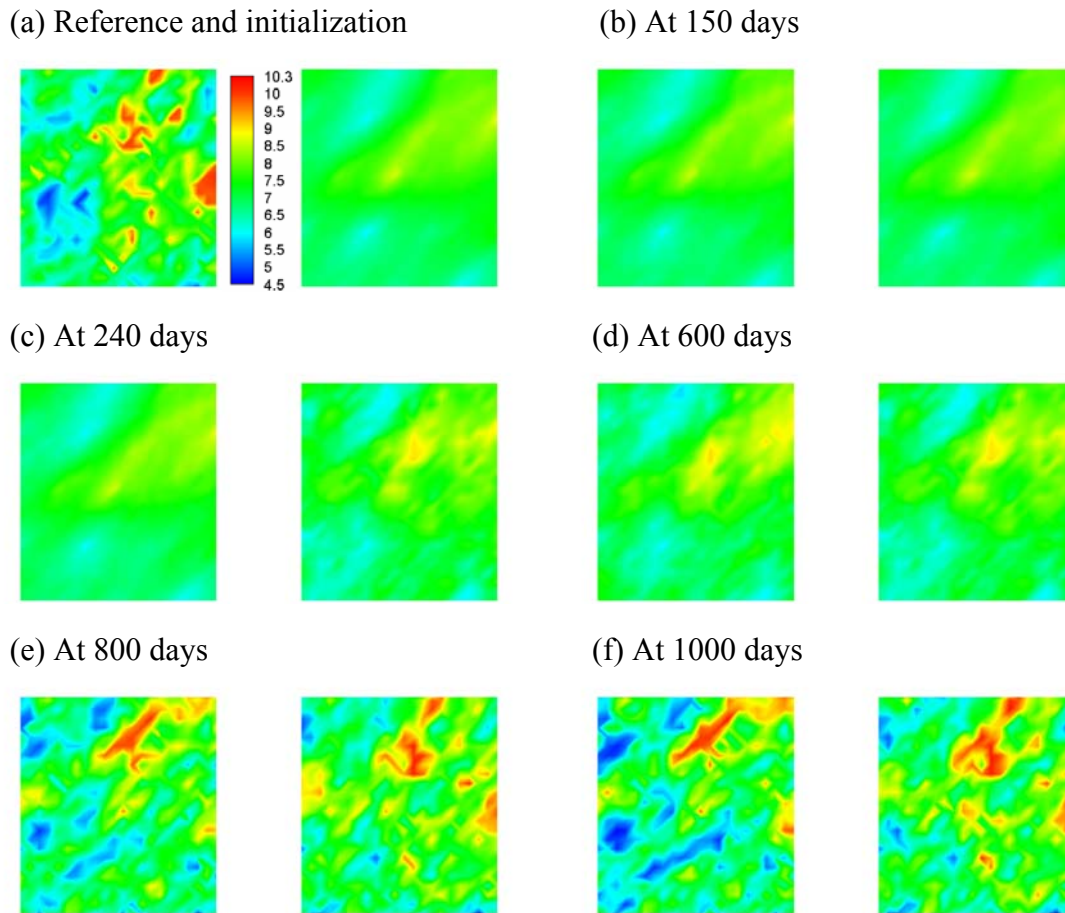


Figure 6-60: Fourteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

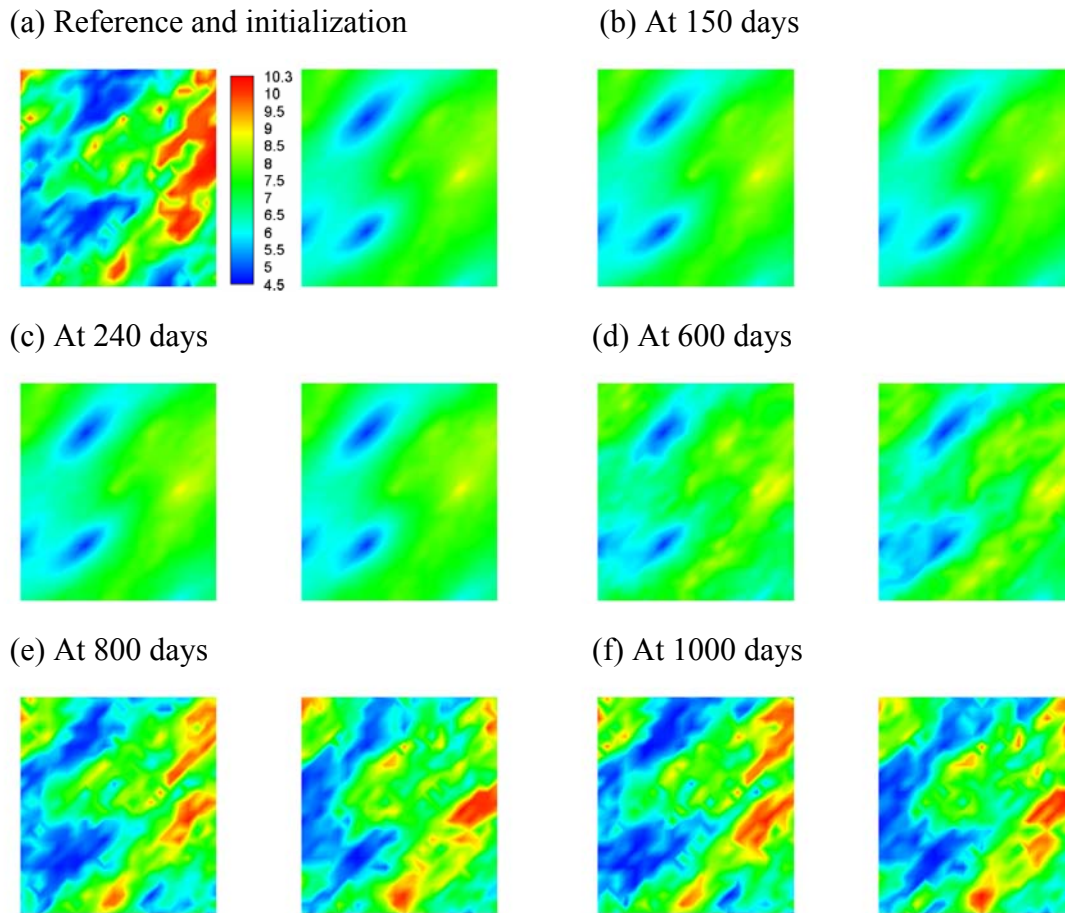


Figure 6-61: Fifteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

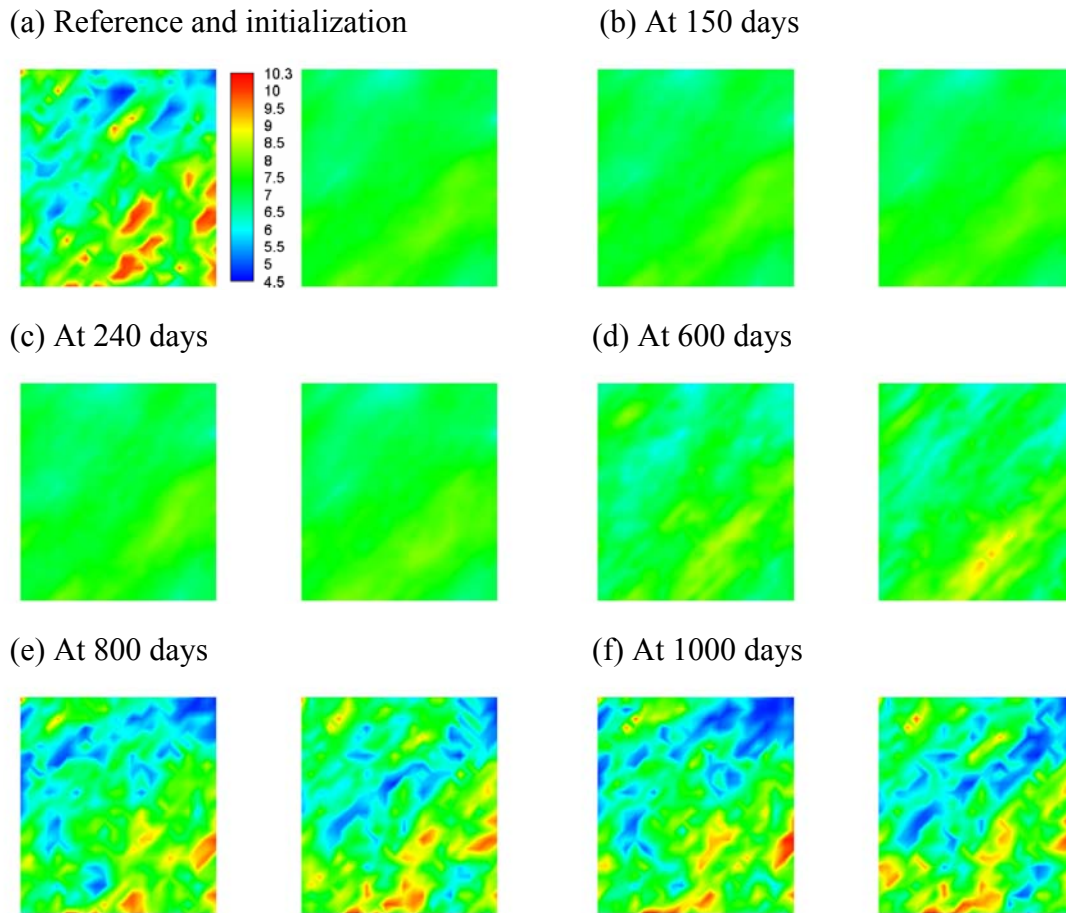


Figure 6-62: Sixteenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

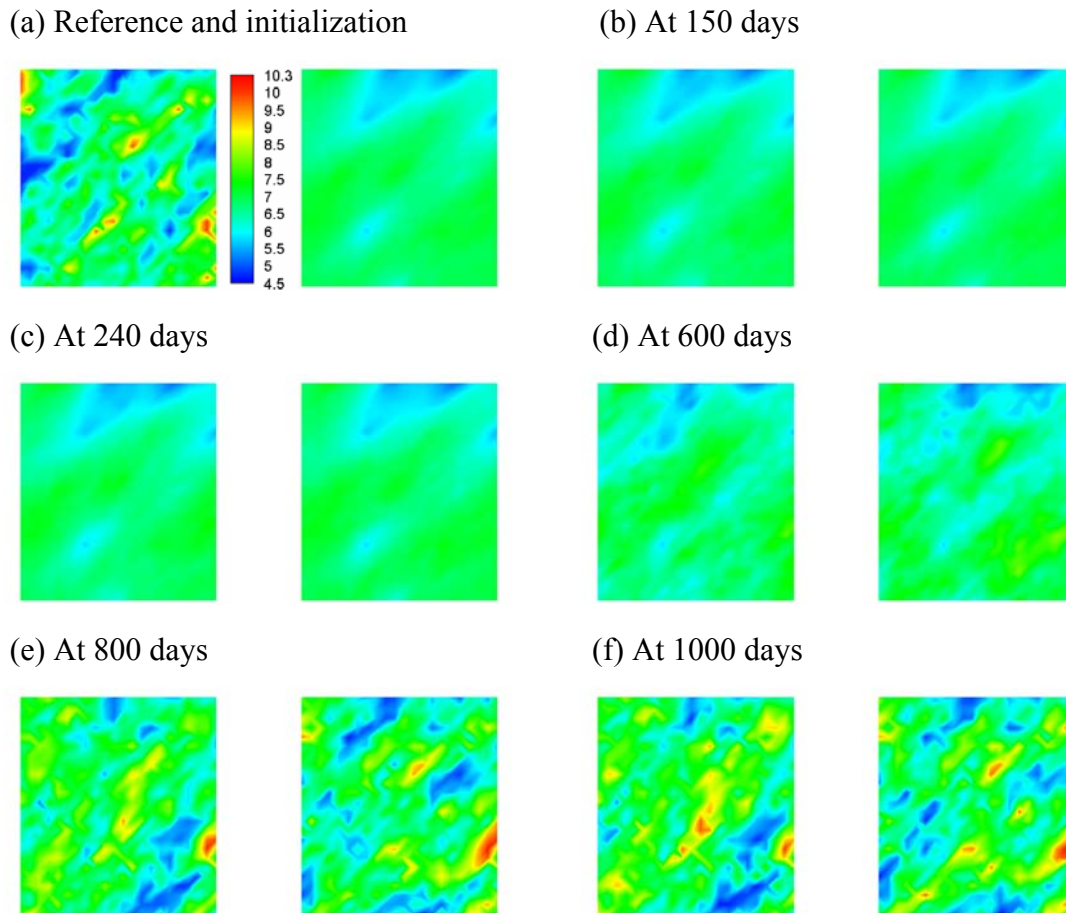


Figure 6-63: Seventeenth-layer logarithm permeability field of averages of ensemble members after several selected assimilation times by using the conventional EnKF and the weighted EnKF, respectively. (a) Left hand side is the reference and right hand side is the initialization; (b)-(f) Left hand side is results from the conventional EnKF while right hand side is results from the weighted EnKF. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

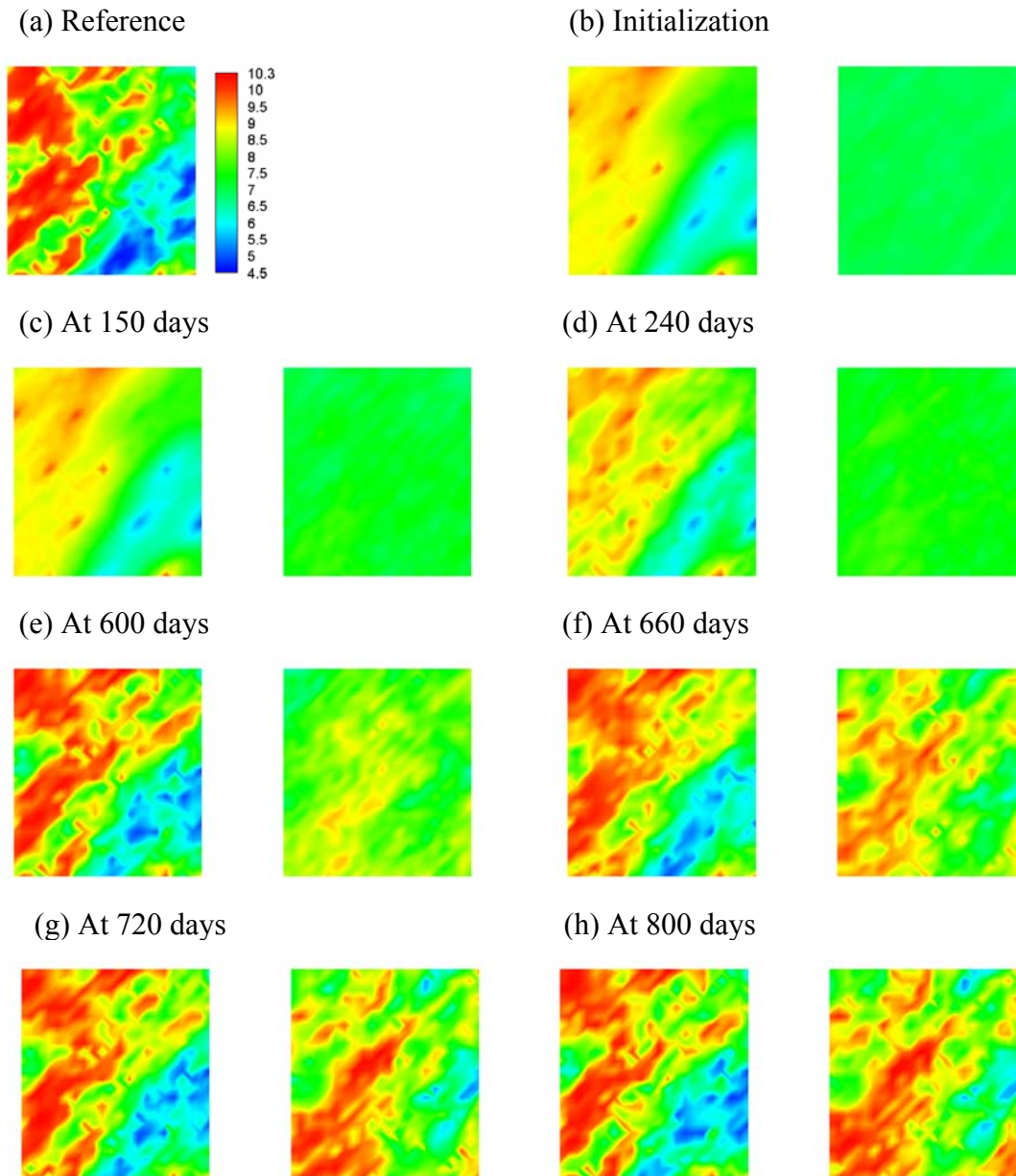


Figure 6-64: Twelfth-layer logarithm permeability field of the averages of ensemble members after several selected assimilation times by using the weighted EnKF, respectively. (a) Reference; (b)-(h) Left hand side is results from conditional realizations while right hand side is results from unconditional realizations. Legend scale is the same for every profile, from 4.5 (blue) through 10.3 (red) with an increment of 0.5.

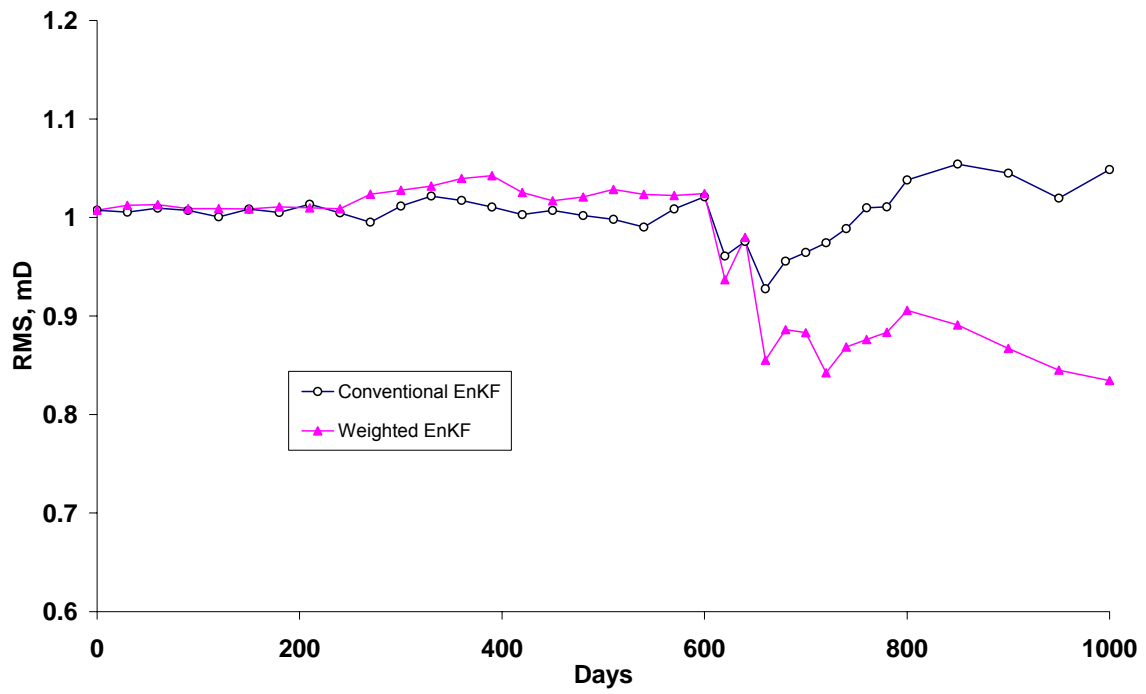


Figure 6-65: Root mean square in the twelfth layer versus time in two methods, the conventional EnKF method (denoted as “Conventional EnKF”) and the weighted EnKF method (denoted as “Weighted EnKF”).

Chapter 7: A Singular Evolutive Interpolated Kalman Filter for Uncertainty Quantification

Inherent data and model uncertainties make the history matching non-unique. Therefore, a reliable uncertainty quantification framework for predicting reservoir dynamic performance requires multiple reservoir models that match field production data. It has been demonstrated that the ensemble Kalman filter technique can be used for this purpose. In this technique, an ensemble of reservoir models is evolved by means of a stochastic nonlinear filtering procedure to agree with the observed production data. An efficient variant of the ensemble Kalman filter, namely, Singular Evolutive Interpolated Kalman Filter (SEIKF) (Pham *et al.*, 1998a) is applied to the multi-model history-matching problem in this work. This novel technique operates in three steps: resampling, forecasting, and assimilation. Unlike the ensemble Kalman filter, where the members of the model ensemble are operated by forecasting and assimilation, in SEIKF the members of the model ensemble are selected in the main orthogonal directions of a functional space described by an approximation of the error-covariance matrix. This enhanced sampling strategy, embedded into the resampling step, improves the filter stability and delivers rapid convergence.

In this chapter, SEIKF is applied to a three-dimensional proof-of-concept waterflooding case where reservoir permeability is calibrated to production data (Liang *et al.*, 2007). Accuracy and convergence of history match, as well as the uncertainty of dynamic predictions yielded by the final model ensemble, are used as criteria to evaluate the performance of SEIKF. In terms of accuracy and uncertainty reduction, SEIKF performs comparably to a conventional ensemble Kalman filter.

The organization of this chapter is as follows: We first provide a brief introduction of uncertainty study and mathematical description of SEIKF. A workflow for the ensemble based history matching with SEIKF will be introduced. In this context, we will point out the main differences and similarities between SEIKF and EnKF workflows. Next, we will describe a three-dimensional synthetic example. This example serves as a laboratory for our uncertainty quantification workflow where we navigate through the entire uncertainty quantification process from real-time data integration to recovery forecasts. In order to establish a direct measure of comparison for our workflow, we also apply EnKF to the proof-of-concept test case. Calibrated reservoir models are subsequently used to quantify the uncertainty in the recovery forecasts. A discussion section will provide a critical evaluation on the results obtained by use of SEIKF and EnKF techniques from the viewpoint of accuracy, robustness, and convergence. Finally, the chapter will close with the summary.

7.1 INTRODUCTION

The optimization of a reservoir development strategy is measured by its robustness under the influence of uncertainty. In addition to economic unknowns, uncertainties in reservoir characterization constitute a large component of the financial risk. The practice of forecasting hydrocarbon recovery performance through dynamic reservoir modeling is therefore an integral component of risk analysis and uncertainty reduction strategies. Emerging technologies such as geophysical reservoir monitoring (i.e., permanent sensors, 4D seismic) and optimal reservoir management (i.e., smart completions) also rely heavily on dynamic modeling. From this perspective, future forecasts of reservoir performance are used to optimize reservoir management decisions. The quality of the oil reservoir model is therefore of essential importance for performing

robust and accurate predictions of recovery potential and, in turn, making decisions based on correct premises.

It is very desirable to constrain the dynamic model to all available data and reduce uncertainties. The most direct information about the physics of fluid flow in the dynamic model is embedded in the production data. The type of production data, in turn, is a function of the recovery mechanism. More precisely, flowing phases and injection/production constraints associated with a given recovery mechanism determine the types of available production data. An arbitrary combination of water, oil, and gas production rate, as well as wellbore pressure, may constitute the individual components of a production data set. As a direct measure of the reservoir response, integration of production data to dynamic reservoir models is the primary driver for history-matching. In a history matching exercise, model parameters are adjusted in such a way that the dynamic simulation response reproduces the historical production record as accurately as possible. This is achieved either by manually adjusting the parameters of the dynamic model or permitting an automatic process to propose adjustments.

It is of central importance to recognize the fact that, from the mathematical perspective, integrating production data to subsurface dynamic models is an ill-posed, inverse problem. Thus, there exists an ensemble of models which satisfy the production measurements to a sufficient degree of accuracy. There are additional factors compounding the non-uniqueness of the history matching problem: (1) Virtually all types of hard (static and dynamic) measurements acquired in hydrocarbon reservoirs carry attached error bars. (2) In many cases, the geologic blueprints of the dynamic models rely on an incomplete, qualitative understanding of the subsurface. (3) Both dynamic and static measurements are spatially sparse. Often dynamic measurements are temporally sparse as well. (4) Different measurements have different resolution. In general, there

exist spatial scales at which the reservoir model is unresolved by any of the available data types. Seeking a single, deterministic best history-matched model may lead to misleading predictions of future recovery. In fact, a perfectly history-matched model may lead to spectacularly erroneous forecasts of recovery (Tavassoli *et al.*, 2004). There is a clear need for history matching algorithms that can efficiently generate multiple history-matched models (ensemble) while retaining geologic consistency. Predictions performed with ensemble members help quantify uncertainty in the recovery forecast. The accuracy of the forecast statistics, in turn, is controlled by the exploratory nature of the history matching algorithm. The more effectively the history matching processes sample the uncertainty space, the more accurate the statistics of the recovery forecast.

In this chapter, we implement an efficient variant of the ensemble Kalman filter. The Kalman filter is a statistically optimal sequential-estimation procedure for linear dynamical systems (Kalman, 1960). In a Kalman filter, observations are fed to a numerical flow model with weights to minimize error variance. The information content of observations is advected from data-rich areas to data-poor areas with the help of an optimally estimated error-covariance matrix. The latter is propagated in time together with the flow model. The ensemble Kalman filter (EnKF), introduced by Evensen (1994), handles nonlinear problems by integrating an ensemble of model trajectories from which error-covariance estimates (and thus a gain matrix) can be calculated. EnKF has found widespread applications in the areas of weather forecasting, oceanography and hydrology, because of its simple formulation and relative ease of implementation.

The singular evolutive extended Kalman filter (SEEKF) is proposed by Pham *et al.* (1998b) as an alternative to EnKF. A singular low-rank matrix is used to approximate the error covariance matrix. This strategy introduces corrections only in the directions for which the error is not sufficiently attenuated by the nonlinear system. These directions

evolve in time consistent with the underlying dynamics of the system. However, as in the case of the extended Kalman filter, strong model nonlinearities may lead to instabilities and eventually impede convergence, as noted by Evensen (1992). Pham *et al.* (1998a) introduced a variant called singular evolutive interpolated Kalman filter in which the linearization used in SEEKF is replaced by a linear interpolation. The implication of this strategy is the mitigation of error for large deviations. The resulting filter has been applied to realistic problems of oceanography with satisfactory results (Hoteit *et al.*, 2002; Triantafyllou *et al.*, 2003; Nerger *et al.*, 2007). In this paper, we compare the accuracy and convergence behavior of SEIKF and EnKF with an improved assimilation strategy on a three-dimensional history matching problem. Multiple history-matched models are used to forecast future oil recovery.

7.2 SINGULAR EVOLUTIVE INTERPOLATED KALMAN FILTER

The overall uncertainty quantification workflow is shown in Figure 7-1. The amount of available dynamic data volume increases as a function of the progress in recovery. Knowledge embedded in the dynamic data needs to be integrated to the subsurface model in real time to close the gap between the model predictions and the actual response of the reservoir. This is the point where dynamic filter based history matching or data integration techniques such as SEIKF and EnKF enter into the workflow. A number of equally probable reservoir models could match the production data. Thus, ideally a large ensemble of subsurface models needs to be evolved in time. Advancing a large ensemble of subsurface models through the entire lifecycle of the reservoir is a computationally demanding task. As such, it emerges as one of the key blockers for the widespread use of nonlinear filters in history matching problems. Additionally, carrying along a large number of ensemble members that do not exhibit consistency with the observed data can lead to a low rate of convergence and reduce the

efficiency of the history matching (data integration) process. Thus, it is very desirable to devise an evolution strategy that readily allows retaining only the ensemble members that lead to rapid convergence. As will be elaborated later in this paper, SEIKF possesses promising features in this context. Our objective is to evaluate SEIKF in a controlled numerical experiment and shed light on its strengths and weaknesses. Prior to this, however, we would like to provide a formal mathematical introduction to SEIKF.

Initialization process aside, SEIKF proceeds in three stages: resampling, forecasting and assimilation. The resampling step is to generate the ensemble realizations, namely permeability fields. This resampling procedure makes use of a minimum second-order exact sampling technique (Triantafyllou *et al.*, 2003). The forecast step advances the state vectors from the current time step to the next time step. In reservoir simulation, the forecast step is performed by use of a reservoir simulator. The state vector contains the variables required to describe the system. It typically includes the values of permeability, oil pressure, and oil saturation in each gridblock. Well-by-well measurements such as bottomhole pressures, fluid rate, gas oil ratio, and water cut are also included in the state vector. To that end, if the number of grid blocks equals N_m , and the number of computed data equals N_d , the dimension of the state vector Y is given by $3 \times N_m + N_d$ for the immiscible flow of two fluid phases such as the one that arises in waterflooding applications. The assimilation step introduces corrections to the variables in the state vectors to honor the new observed data.

A schematic diagram describing the history matching procedure with SEIKF is shown in Figure 7-2. A large ensemble of equiprobable realizations of permeability maps is generated using a geostatistical modeling algorithm. The large initial ensemble is distilled into a reduced ensemble with a low degree of inter-member correlation. This is accomplished by means of the singular value decomposition (SVD) technique. The

ensemble is advanced forward in time by means of a reservoir simulator until the moment when the first set of observations (production data) is acquired. In the assimilation step, the discrepancy between observed data and the simulation response is quantified. In turn, the variables that describe the state of the dynamic system are calibrated to render the system response consistent with the observed data. Multiple realizations of model parameters (permeability maps) are then resampled. We shall elaborate more on the resampling step further in the paper. The resampling step is followed by the forecasting and assimilation steps. The sequence of resampling, forecasting, and assimilation steps is repeated until the final time of history matching is reached. Multiple history-matched permeability maps constitute a subset of the ensemble state variables at the final time and can be extracted from the filter in straightforward fashion. In turn, simulation based forecasts can be performed using multiple history-matched permeability maps. Outcome of such forecasts can be used to obtain a first-order description of the uncertainty in future hydrocarbon recovery.

The innovative feature of SEIKF lies in the resampling step. This step involves an interpolation procedure applied to the randomly drawn state vectors at every filtering step. Subsequent to an assimilation step, ensemble state vectors are used to compute an analysis state vector and its error covariance matrix. By using this information, interpolated states are generated. Interpolated states are then fed into the forecast step. The analysis state vector and its error covariance matrix are used to retain the central tendency and the covariance matrix over the course of the interpolation process of the resampling step.

Let $Y_1^a(t_k), Y_2^a(t_k), \dots, Y_N^a(t_k)$ denote the interpolating states. Here, N_e is the total number of interpolating states (or the total number of ensemble members). The analysis state vector and its error covariance matrix are described by the following equations:

$$Y^a(t_k) = \frac{1}{N_e} \sum_{i=1}^{N_e} Y_i^a(t_k), \quad (7.1)$$

$$P^a(t_k) = \frac{1}{N_e} \sum_{i=1}^{N_e} [Y_i^a(t_k) - Y^a(t_k)][Y_i^a(t_k) - Y^a(t_k)]^T. \quad (7.2)$$

Use of ensemble states in SEIKF is analogous to EnKF. There are, however, important differences. SEIKF attempts to use the smallest possible number of interpolating states. Also, a constrained drawing process is applied honor Eq. (7.1) and Eq. (7.2). SEIKF consists of three processes for each time step: resampling, forecasting, and assimilation. An initialization step replaces the resampling step at the beginning of the data integration process.

7.2.1 Initialization

In the absence of available production data at the time of initialization, an ensemble of permeability fields is randomly generated subject to geostatistical constraints. Dynamic variables of the initial ensemble such as initial pressures and saturations are assumed known with absolute accuracy and imposed to be the same for all ensemble members. Quantitative information on model statistics is derived from this initial ensemble of realizations.

It is worthwhile to highlight the fact that sole use of random realizations may give rise to an ensemble with highly correlated members. This can potentially lead to an inaccurate representation of the uncertainty space. Our experience also indicates that use of an ensemble with highly correlated members creates a significant hurdle for filter convergence. To circumvent this predicament, in this paper, we apply an improved sampling strategy for generating the initial ensemble (Evensen, 2004). We first generate a

large number of equiprobable realizations. Subsequently, by use of SVD a small number of significant realizations are selected for inclusion in the initial ensemble. This procedure ensures that the initial ensemble retains geological soft-knowledge and ensemble members exhibit a low degree of correlation.

7.2.2 Resampling

SEIKF marches a dynamic system forward in time while assimilating observed information and calibrating the state variables to be consistent with the observations. The resampling step of SEIKF operates between the assimilation step of the previous point in time, t_{k-1} , and the forecast step of the current point in time, t_k . Unlike EnKF, SEIKF does not make a direct use of the outcome of the assimilation step of t_{k-1} . Rather, the resampling step generates perturbations of the t_{k-1} assimilation results to locally widen the exploratory characteristics of the filter. Here, information derived from previous interpolation states, namely mean and the covariance of the previous interpolating states, plays a central role in enforcing consistency. At t_{k-1} , an analysis state $Y^a(t_{k-1})$ and its pertinent error covariance matrix $P^a(t_{k-1})$ is computed. The error covariance matrix is stored in a factorized form described by $L_{k-1}U_{k-1}L_{k-1}^T$. Here, L_{k-1} and U_{k-1} are matrices derived from the assimilation step and the superscript T denotes matrix transpose. The interpolating states are expressed via

$$Y_i^a(t_{k-1}) = Y^a(t_{k-1}) + \sqrt{r+1} L_{k-1}(\Omega_{k-1,i}C_{k-1}^{-1})^T, \quad (7.3)$$

where $1 \leq i \leq r+1$, and r denotes the rank of the error covariance matrix. The term $r+1$ represents the smallest possible number in the ensemble. Here, C_{k-1} denotes the Cholesky decomposition of U_{k-1}^{-1} , and Ω_k represents any $(r+1) \times r$ matrix with orthonormal columns and zero column sums, randomly selected following the procedure

described by Hoteit *et al.* (2002). In Eq. (7.3) $\Omega_{k-1,i}$ denotes the i -th row of Ω_{k-1} . The above expression reinforces that the error covariance matrix of the state, $P^a(t_{k-1})$, is honored.

7.2.3 Forecasting

Based on the current ensemble state vectors, a simulation is performed for each realization. The simulation is run up to the time of the next measurement acquisition. In this step, dynamic simulation and well-by-well production data are rendered consistent with the model parameters of the ensemble state vector. The forecasting step can be stated via

$$Y_j^p(t_k) = f(Y_j^a(t_{k-1})), \quad j = 1, 2, \dots, r+1. \quad (7.4)$$

where p denotes “predicted”, a denotes “assimilated”, f denotes the reservoir simulator, k is the time step index, j is the ensemble member index, $(r+1)$ is the number of ensemble members, $Y_j^a(t_{k-1})$ is the j -th assimilated state vector after the data assimilation at the timestep $k-1$, and $Y_j^p(t_k)$ is the predicted state vector based on all available information prior to the time step k . Note that only dynamic variables, i.e. gridblock pressures and saturations, and production data are represented by indices $k-1$ and k . The static variables, i.e. gridblock permeabilities, remain unchanged. Adjustment of static variables occurs at the assimilation step together with the dynamic variables. In the forecasting step, the state forecast $Y^a(t_k)$ will be taken as the average of $Y_j^a(t_k)$. Thus, the prediction error covariance matrix could be approximated by

$$P^p(t_k) = \frac{1}{r+1} \sum_{i=1}^{r+1} [Y_i^p(t_k) - Y^p(t_k)] [Y_i^p(t_k) - Y^p(t_k)]^T. \quad (7.5)$$

This matrix could also be represented by

$$P^p(t_k) = S_k \left[(r+1)W^T W \right]^{-1} S_k^T, \quad (7.6)$$

where,

$$S_k = \left[Y_1^p(t_k), \dots, Y_{r+1}^p(t_k) \right] W. \quad (7.7)$$

In Eq. (7.7) W is a $(r+1) \times r$ full rank matrix with zero column sums. A suitable choice for W can be found in Hoteit *et al* (2002).

In our implementation of SEIKF, the forecasting step makes use of UTCHEM simulator (Delshad, 1996). UTCHEM is a three-dimensional, multiphase, multicomponent simulator, particularly suitable for water and chemical flooding applications, which has been used extensively and validated with laboratory and field data.

7.2.4 Assimilation

The assimilation step of SEIKF exhibits noteworthy differences when compared to EnKF. Instead of the state vectors of each ensemble member, the analysis state vector and its associated covariance matrix is propagated from the forecast step. Newly acquired observations are assimilated to correct the analysis state vector and guide the dynamic system to the next resampling step. The new observation d_{obs} and observation error covariance R at time t_k are used to correct the forecast according to

$$Y^a(t_k) = Y^p(t_k) + S_k V_k (H S_k)^T R_k^{-1} \left[d_{obs} - H_k Y^p(t_k) \right], \quad (7.8)$$

where,

$$(H S)_k = [H_k Y_1^p(t_k), \dots, H_k Y_{r+1}^p(t_k)] W \quad (7.9)$$

and V_k computed from

$$V_k^{-1} = (r+1)T^T T + (H S)_k^T R_k^{-1} (H S)_k. \quad (7.10)$$

7.3 A THREE-DIMENSIONAL NUMERICAL EXAMPLE

In this section, we compare the quality of history matching and forecasting results obtained with SEIKF and EnKF. The impact of different sampling and assimilation strategies pertinent to these nonlinear filters is investigated from the viewpoints of (1) accuracy of the reconstructed permeability fields, (2) history of filter convergence, and (3) uncertainty reduction.

7.3.1 Model Description

A simple three-dimensional reservoir model is constructed as a virtual laboratory to conduct proof-of-concept numerical examples. The reservoir simulation model encompasses 1,083 ($19 \times 19 \times 3$) gridblocks. The permeability field is generated using of a geostatistical reservoir modeling algorithm called Matrix Decomposition Method (MDM), (Yang, 1990). In simple terms, MDM operates as a weighted-averaging method in which the weighting matrix is related to a covariance matrix. In our case, the covariance matrix is derived by use of a spherical variogram and a lognormal permeability distribution. The correlation length is 460 ft along both horizontal directions (X and Y). Along the vertical direction (Z) a shorter correlation length of 46 ft is

enforced. The mean value for permeability is 350 mD and the vertical-to-horizontal permeability ratio equals to 0.1. Dykstra-Parsons coefficient, a measure of heterogeneity, is 0.8 indicating a considerably heterogeneous porous medium. Table 7-1 provides the remaining rock, fluid, and geometrical properties. Injector and producer wells are both operated with a constant rate constraint. Over the course of the production both injection and production rates are varied.

Layer-by-layer permeability maps of the numerical example are shown in Figure 7-3. Hereafter, we will refer to these permeability maps as the reference maps. Injection and production well locations are indicated on the permeability map of Layer #1 (top layer). Permeabilities cover the range from 1 mD to 4313 mD and are transformed to the logarithmic domain. Red color indicates high and blue color low values of permeability. The waterflooding project is operated via five injection and nine production wells. Producers are denoted with black dots and injectors are denoted with both black dots and a through-going line. All wells are fully penetrated through the 40 ft-thick reservoir. The production data set is derived by simulating a reference case for 500 days. Well-by-well oil and water production rates and wellbore pressures are recorded. The data set is reduced so that each data record will contain measurements acquired every 50 days. Subsequently, all data records are contaminated with varying levels of zero-mean random Gaussian noise.

7.3.2 Accuracy and Convergence of Model Domain Reconstruction

One hundred initial realizations of the permeability field are selected through SVD from a pool of two hundred geostatistically constrained realizations. Let us first discuss the results obtained with SEIKF. The evolution of the mean permeability field populating Layer #1 is illustrated in Figure 7-4 along with the reference map. Snapshots are shown for the initial time, 50, 150, 300, and 500 days (final time for history

matching). Similar plots for Layer #2 and Layer #3 are shown in Figures 7-5 and 7-6, respectively.

Main features of the mean permeability field are successfully captured by the SEIKF history-matching workflow. Accuracy of model domain reconstruction is remarkably higher for Layers #2 and #3. Geologic features start to emerge between 50 to 150 days of data integration. After 500 days of history-matching the highest quality of mean-field reconstruction is attained for Layer #3 and the lowest for Layer #1. The history-matching process appears to encounter difficulties in extracting permeable features in the upper-left corner of Layer #1. Simulated model responses in general exhibit good agreement with the noise-contaminated synthetic production measurements.

7.3.3 Comparison of SEIKF and EnKF Results

In order to validate the application of SEIKF to history matching problems, the proof-of-concept numerical test is extended to include EnKF. Both SEIKF and EnKF methods are put into equal footing by use of the same initial realizations of the permeability field and the same time-interval for production data acquisition. An example of model domain convergence with EnKF is shown in the panels of Figure 7-7 for Layer #3. The comparison of Figures 7-6 and 7-7 indicates that, for the investigated proof-of-concept example, SEIKF protocol yields a more rapid and stable model domain convergence than EnKF.

Figure 7-8 depicts a comparison of permeability maps obtained after 500 days of data integration using SEIKF and EnKF for each layer. Pertinent reference permeability maps are also shown. For Layer #1 both methods encounter difficulties in capturing the mean permeability structure, especially the portion shown in the left-half of the reference map. Nevertheless, EnKF appears to yield a slightly more accurate reconstruction of the mean permeability populating Layer #1. For Layer #2, with regard to the macro-scale

mean permeability structure, both methods perform almost equally well. The needle of the balance, however, tips slightly in favor of SEIKF. For Layer #3, again, both methods successfully capture the large-scale features of the mean permeability. Root Mean Square (RMS), which is the distance between the mean of multiple matched models and the reference, is an effective tool to measure the match performance. RMS is defined as

$$\text{RMS} = \sqrt{\frac{1}{N_m} \sum_{i=1}^{N_m} (Y_{mean,i} - Y_{ref,i})^2}, \quad (7.11)$$

where N_m is the number of gridblocks. $Y_{mean,i}$ and $Y_{ref,i}$ are the matched average value and the reference value of the i -th gridblock, respectively. With the mean permeability fields at the end of the history-match from SEIKF and EnKF, Table 7-2 gives the RMS results, which confirm our observations from Figure 7-8. SEIKF yields a more accurate reconstruction than EnKF. Results of the proof-of-concept example confirm that SEIKF offers a viable alternative to EnKF in terms of accuracy of model reconstruction.

7.3.4 Quantification of Forecast Uncertainty

Going once through the ensemble history-matching workflow yields multiple reservoir models that honor the up-to-date production data. In turn, these multiple models may be deployed for conducting future forecasts of hydrocarbon recovery. Recovering a family of multiple history-matched models within one automatized and non-repetitive workflow is a characteristic feature of nonlinear filter-based history-matching techniques, i.e. SEIKF and EnKF. Clearly, this ubiquitous feature streamlines the uncertainty quantification process.

At this point, it is worthwhile to emphasize that in terms of history-matching and quantification of forecast uncertainty, we make a number of critical assumptions. First

and foremost, we assume that the only source of uncertainty is the permeability field. In real applications, there are almost always a number of other rock, fluid, stratigraphic, and structural parameters that can strongly influence the production behavior. Ideally, these parameters should be included in the entire uncertainty quantification workflow. Our objective is, however, to understand, evaluate, and validate a novel history-matching technique within the framework of an uncertainty quantification workflow. Therefore, we justifiably work on a simplified proof-of-concept problem.

With regard to forecast uncertainty, we assume that 100 history-matched realizations of the permeability field provide an accurate coverage of the a posteriori uncertainty space. In accordance, using the dynamic responses of these permeability fields, we assume that we can construct a reliable statistical description for the progress of cumulative oil recovery. In theory, however, there may be a significant number of permeability fields that can honor the production data and geostatistical soft-constraints but that are left out due to the random nature of the sampling process. Therefore, dynamic recovery responses of such permeability fields will never be accounted for. We attempt to mitigate the impact of this fact by selectively choosing the members of the initial ensemble so that each member will bring a separate piece of information to the history-matching process. This is accomplished by the use of SVD as elaborated earlier.

Recovery forecasts are conducted for the next 1500 days by simulating multiple realizations of the permeability field obtained from 500 days of data integration. Thus, the total time of forecasting is 2000 days. Simulations are performed using all 100 realizations obtained via SEIKF. The same process is repeated for realizations that emanate from the use of EnKF. Outcomes of future recovery prediction are quantified in terms of the cumulative oil production volume. Forecast results are reported in the panels of Figure 7- 9 for models stemming from SEIKF and EnKF workflows. The history

matching period and the final spread in the recovery forecast are clearly marked. Predicted progress of oil recovery for the ground-truth reference model is also shown in the panels of Figure 7- 9.

A comparison of the panels in Figure 7-9 reveals a number of intriguing results. For both SEIKF and EnKF at the later times of the history matching period, cumulative oil production derived from some of the history-matched models diverges from the reference model response. The amount of spread at the final time of the history match appears to be slightly larger for EnKF than for SEIKF. This behavior appears to translate consistently into the forecast period. As a matter of fact, in the forecast mode, the spread in the recovery predictions grows rather rapidly for both methods consistent with the non-unique nature of the history matching problem. However, the magnitude of the growth in this spread is different for models obtained via SEIKF and EnKF. For the models stemming from the application of SEIKF, the cumulative oil recovery curves tend to cluster more around the reference model response. For models inherited from the use of EnKF, the overall as well as the final spread in the recovery predictions is slightly larger than their counterparts derived from the application of SEIKF. We attribute this behavior to the resampling step of SEIKF. Resampling appears to guide the filter to assess a widened spectrum of possibilities in its progress towards convergence. This, in turn, plays a crucial role in giving an opportunity to the assimilation step for executing a more accurate calibration of the state parameters. In turn, the final ensemble is a product of a diverse assessment and efficient calibration process.

Forecast simulations are analyzed statistically to identify the central tendency and the probabilistic range for the low- and high-case scenarios. Recovery statistics are derived from 100 dynamic simulations performed with models derived via SEIKF and EnKF. At any time, there exist 100 realizations of cumulative oil recovery for a given

family of models which honor the production data. The central tendency is described by the mean (P50) value of the cumulative oil recovery time-function. While an interval of confidence for the recovery predictions is described by the “subjectively imposed” P10 to P90 range, in our notation P10 corresponds to the low-risk case and P90 describes the high-risk case. Results of this probabilistic assessment of recovery forecasts are shown in the panels of Figure 7-10. For the investigated proof-of-concept example, SEIKF does a comparable job to EnKF in reducing the forecast uncertainty. The central tendency of cumulative oil recovery derived by use of SEIKF is slightly more accurate than the one obtained via use of EnKF.

7.4 SUMMARY

An efficient variant of the ensemble Kalman filter, namely, Singular Evolutive Interpolated Kalman filter (SEIKF) is applied to the multimodel history-matching problem. SEIKF operates with an enhanced sampling strategy embedded into its resampling step, which appears to improve the filter stability and help the nonlinear filter to deliver rapid convergence both in model and data domains.

SEIKF is applied to a three-dimensional proof-of-concept waterflooding test example. Multiple history-matched models are generated for the reservoir permeability field. Model domain accuracy and convergence of the history matches as well as the uncertainty of dynamic predictions rendered by the final model ensemble are used as criteria to evaluate the performance of SEIKF. The outcome of the proof-of-concept studies quantitatively demonstrates that SEIKF exhibits a rapid convergence behavior. In terms of accuracy and uncertainty reduction, SEIKF performs comparably to EnKF. SEIKF is validated as a rapid and reliable framework for automatic multimodel history matching. Our proof-of-concept numerical test case quantitatively demonstrated that SEIKF has considerable potential in streamlining uncertainty quantification workflows.

Our analysis is based on one application. Further study on SEIKF is needed. The results also demonstrate that EnKF is still an effective tool for forecasting uncertainty quantification.

The only concern about this method is the computational cost of covariance matrix during the resampling step.

Table 7-1 Reservoir rock, fluid, and geometrical properties

Property	Value
Reservoir size	623.2×623.2×40 [ft]
Reservoir gridblock sizes	
Layer #1	32.8×32.8×10 [ft]
Layer #2	32.8×32.8×20 [ft]
Layer #3	32.8×32.8×10 [ft]
Number of gridblocks	19×19×3
Porosity	0.3
Permeability	X and Y directions: $k_x = k_y$ Z direction: $0.1 \times k_x$
Dykstra-Parsons coefficient	0.8
Initial pressure	Variable on a gridblock basis
Initial water saturation	Variable on a gridblock basis
Water viscosity	0.46 [cp]
Oil viscosity	40 [cp]
Water specific gravity	0.433 [psi/ft]
Oil specific gravity	0.368 [psi/ft]
Relative permeability model	Modified Corey model
Endpoint relative permeabilities	0.2 [water], 0.95 [oil]
Phase relative permeability exponents	3.0 [water], 2.0 [oil]

Table 7-2 RMS of permeability fields from SEIKF and EnKF

Layer	SEIKF	EnKF
#1	0.81	0.80
#2	1.33	1.58
#3	0.79	0.83

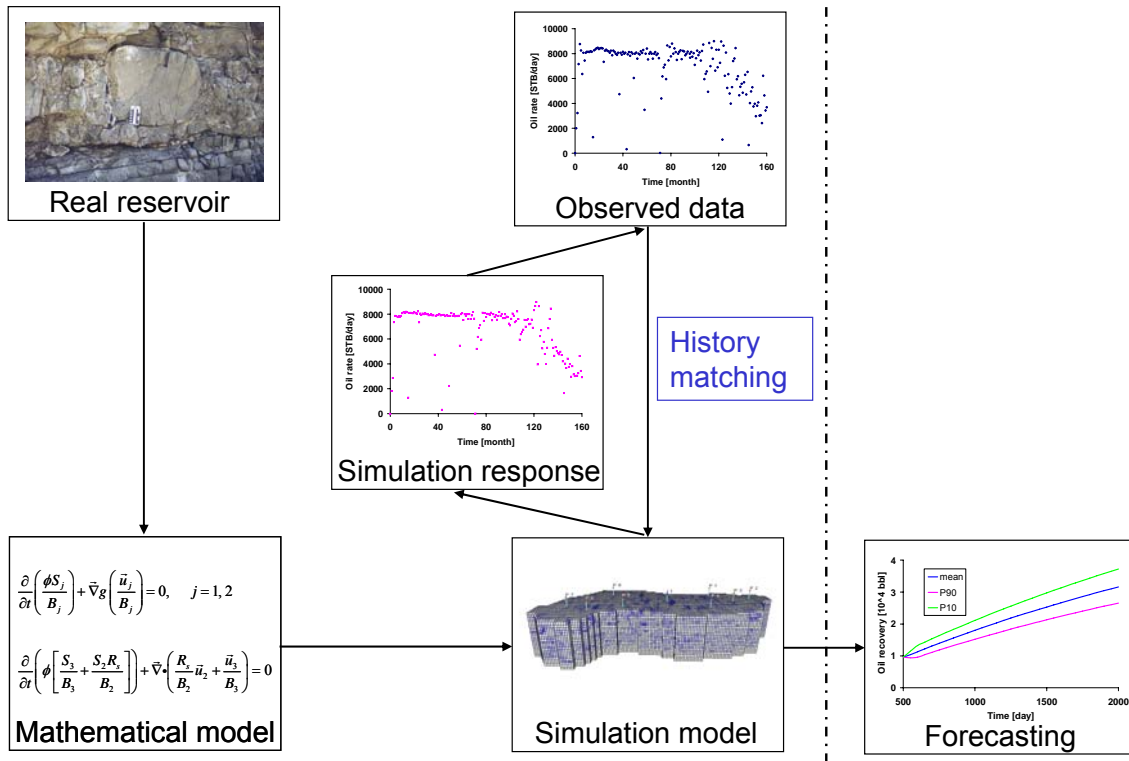


Figure 7-1: Uncertainty quantification workflow.

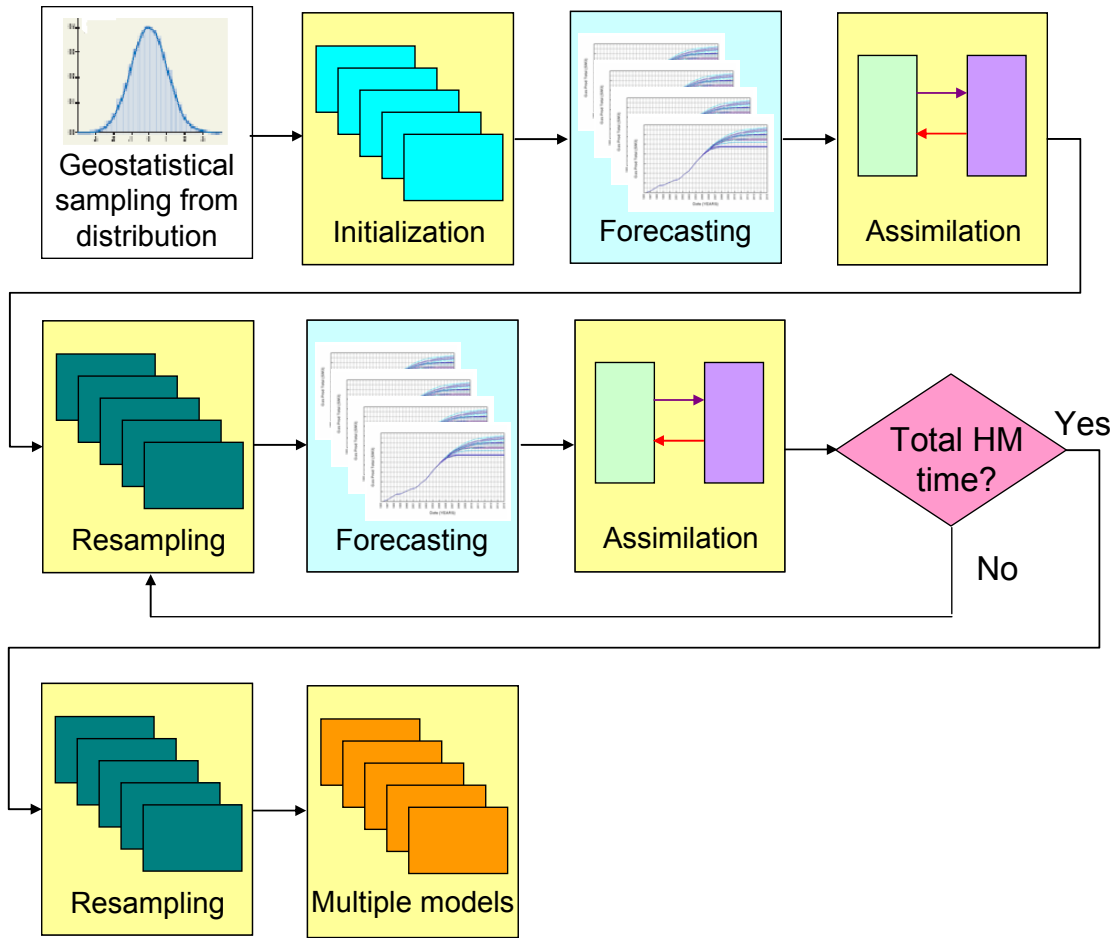
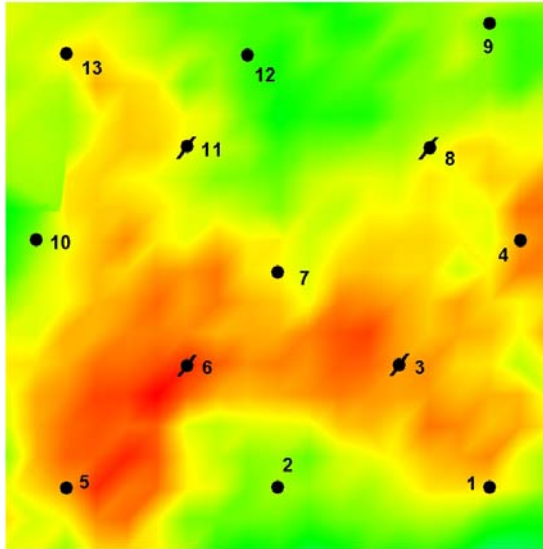
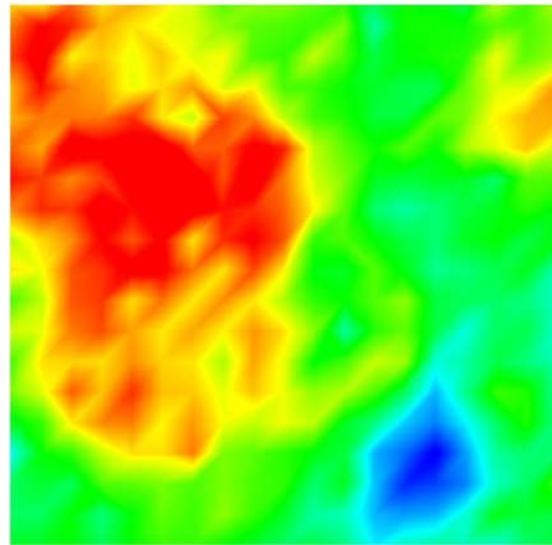


Figure 7-2: Workflow of real-time data integration using SEIKF.

Layer #1



Layer #2



Layer #3

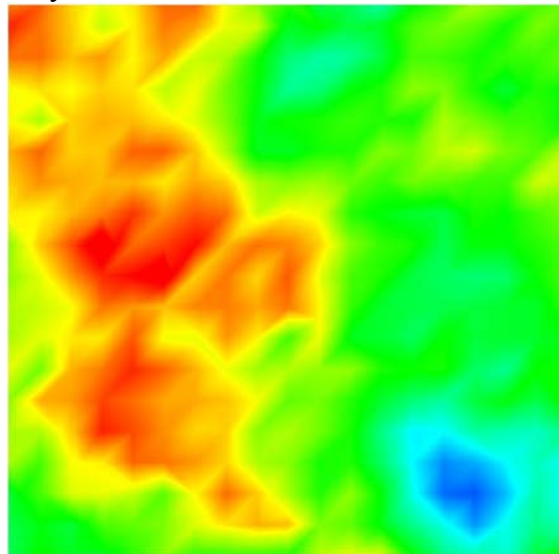


Figure 7-3: Layer-by-layer permeability maps for the reference reservoir. Permeabilities are transformed into the logarithmic domain. Areal locations of injector (dot + through-going line) and producer (dot) wells are shown on the permeability map Layer #1. Permeabilities range from 1 mD (blue) to 4313 mD (red).

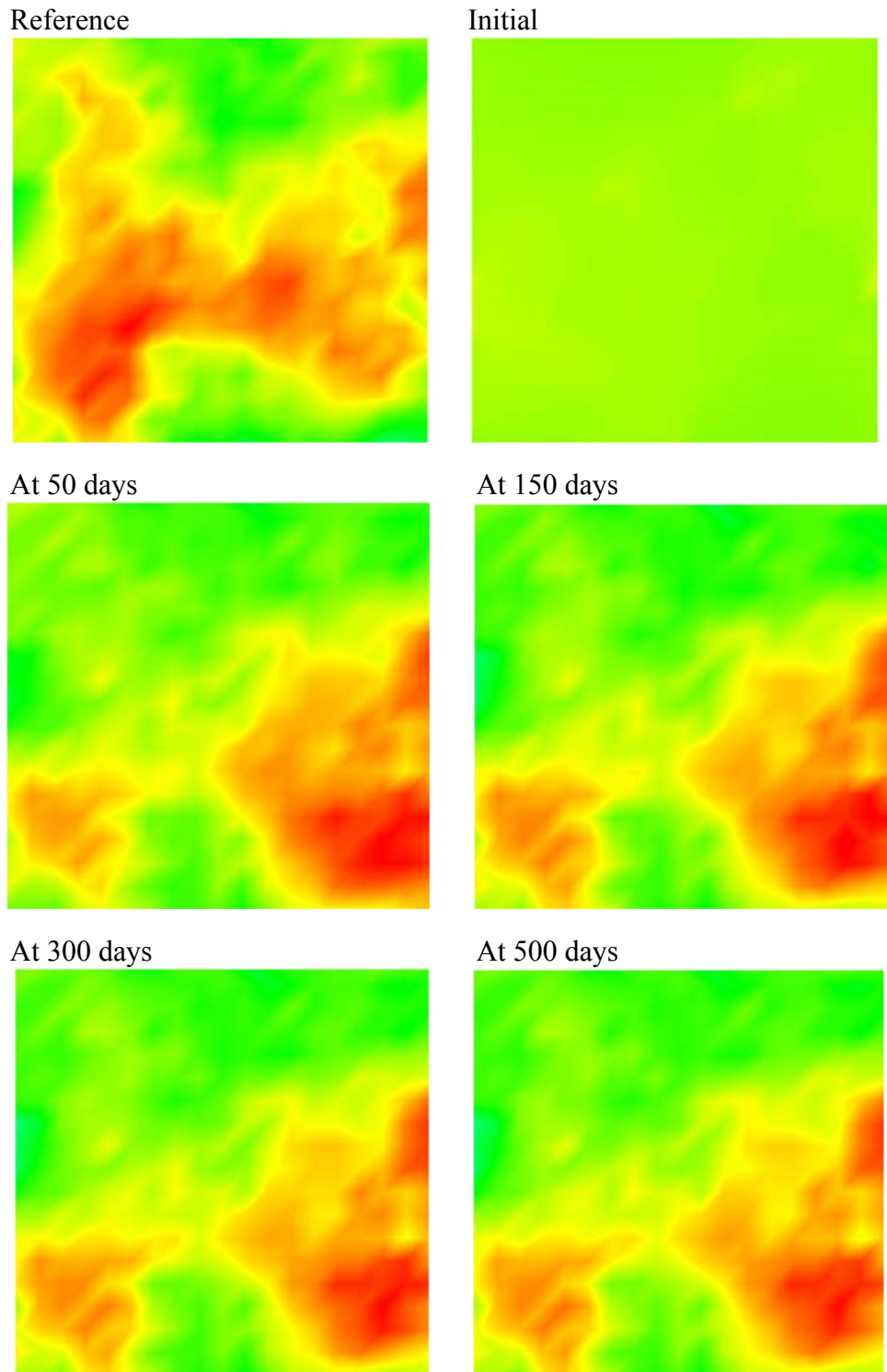


Figure 7-4: Evolution of the mean permeability field in Layer #1. Results obtained by use of SEIKF.

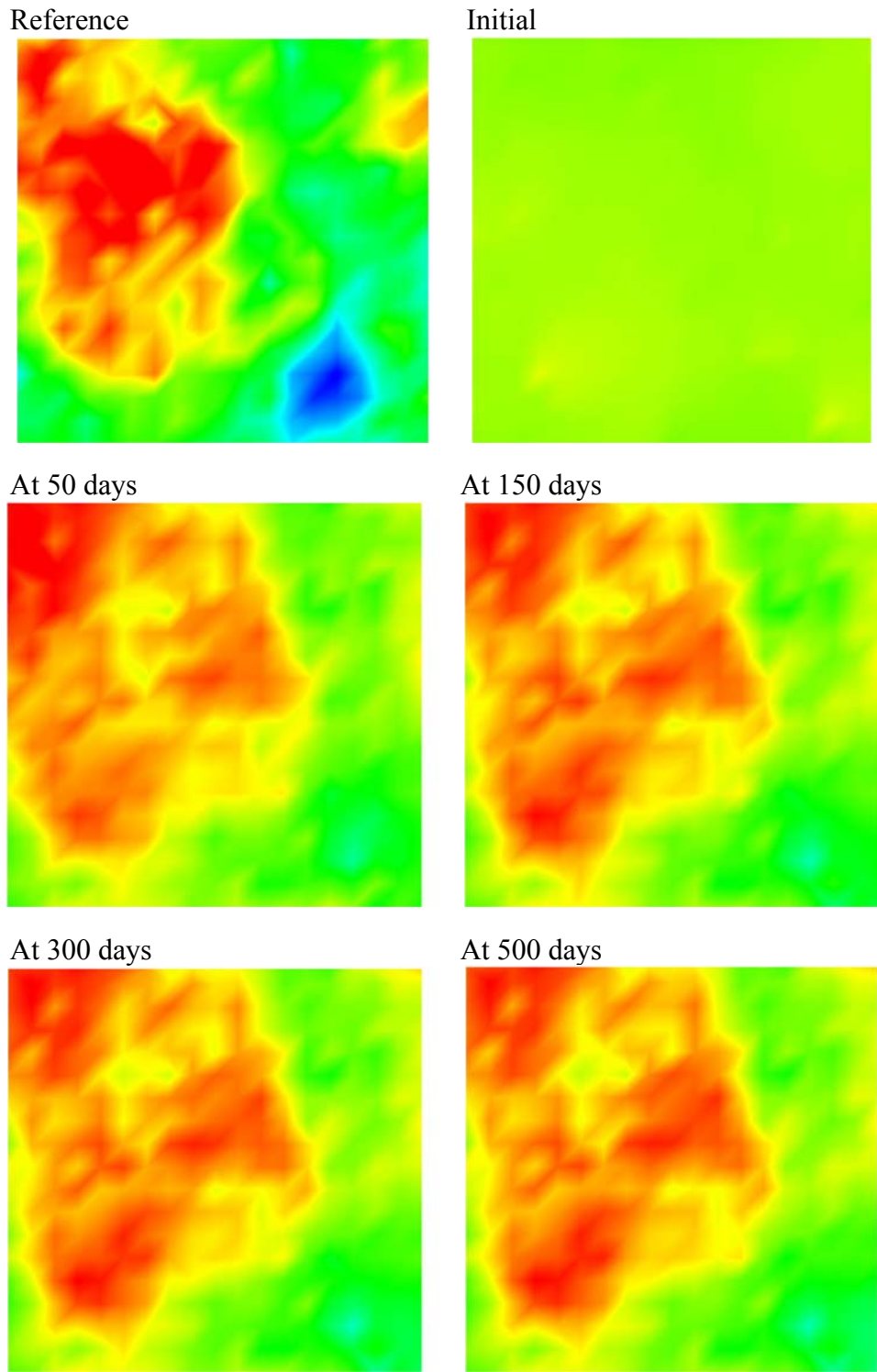


Figure 7-5: Evolution of the mean permeability field in Layer #2. Results obtained by use of SEIKF.

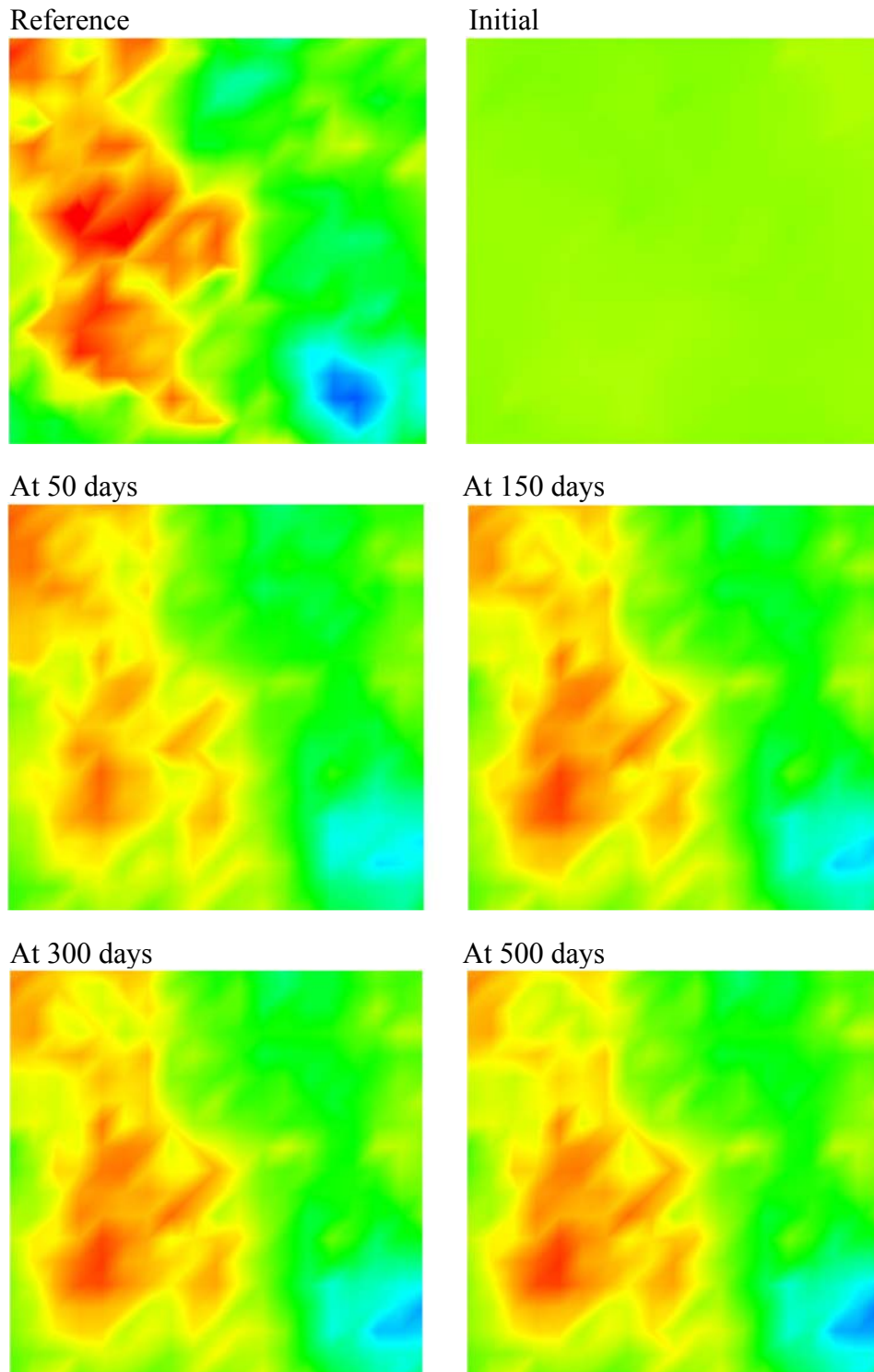


Figure 7-6: Evolution of the mean permeability field in Layer #3. Results obtained by use of SEIKF.

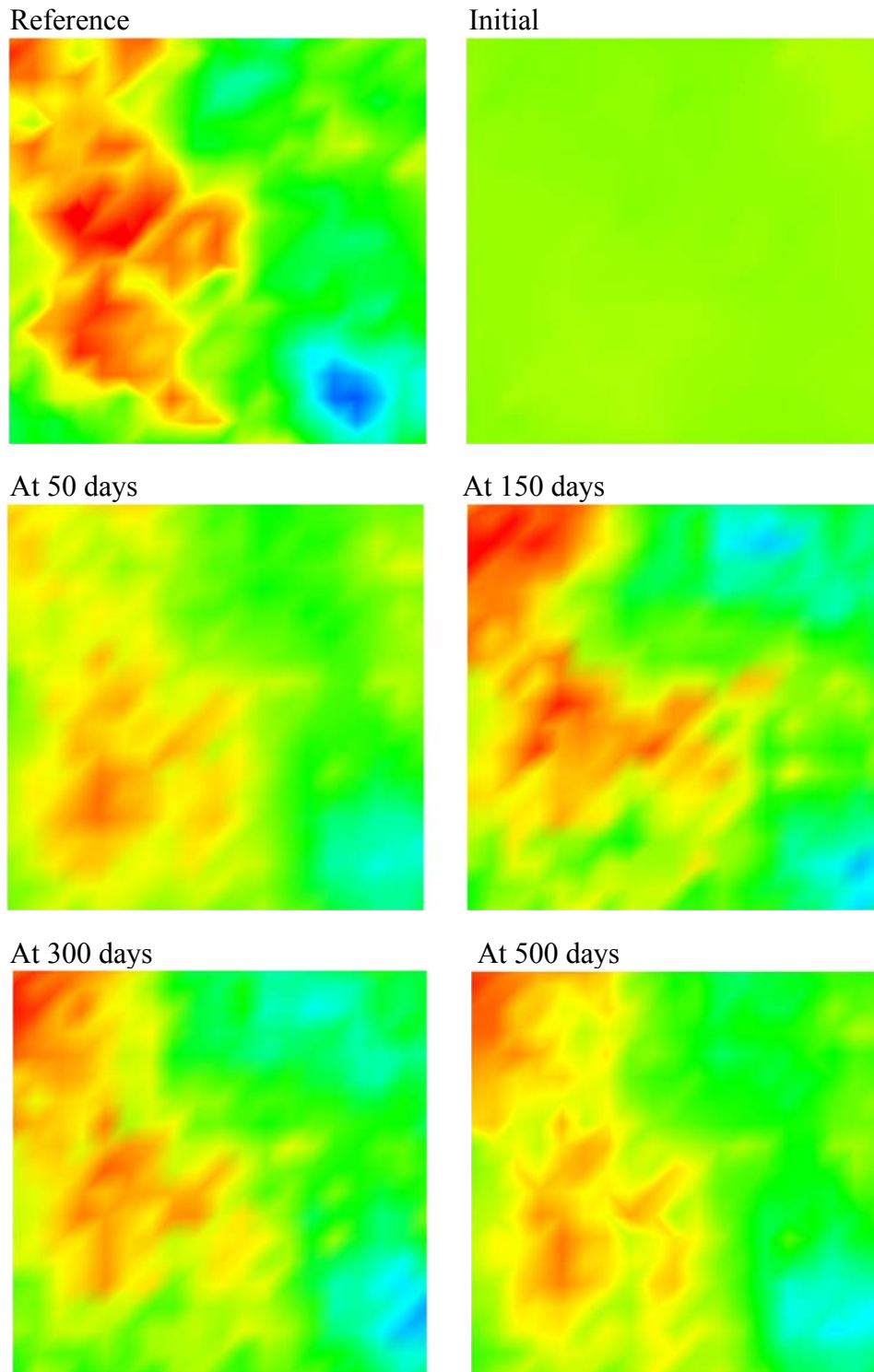
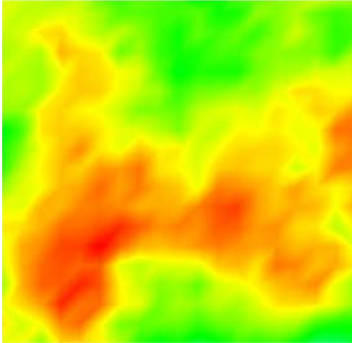


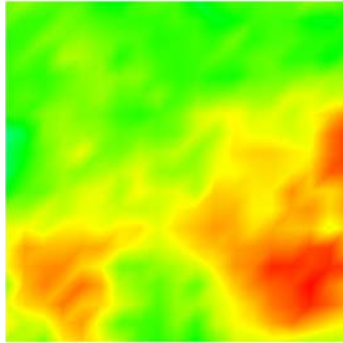
Figure 7-7: Evolution of the mean permeability field in Layer #3. Results obtained by use of EnKF.

Layer #1

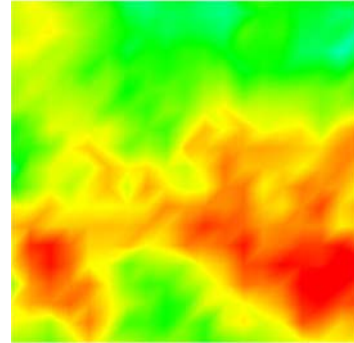
Reference



SEIKF

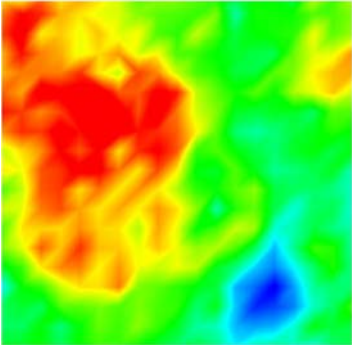


EnKF

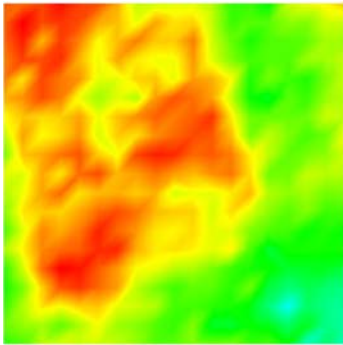


Layer #2

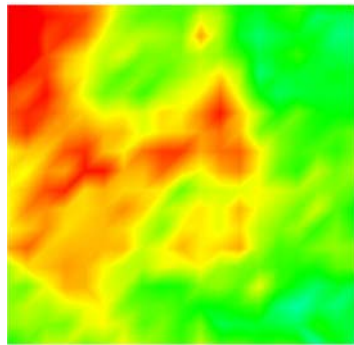
Reference



SEIKF

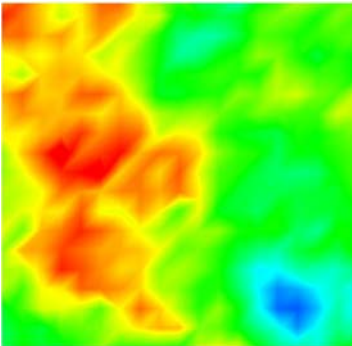


EnKF

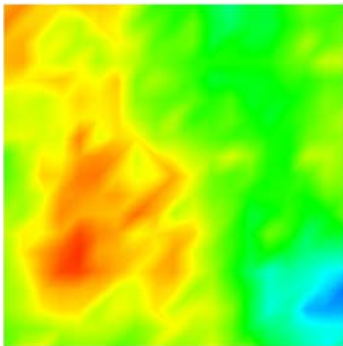


Layer #3

Reference



SEIKF



EnKF

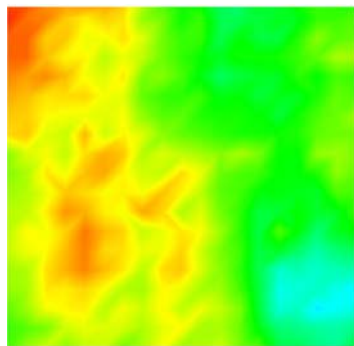
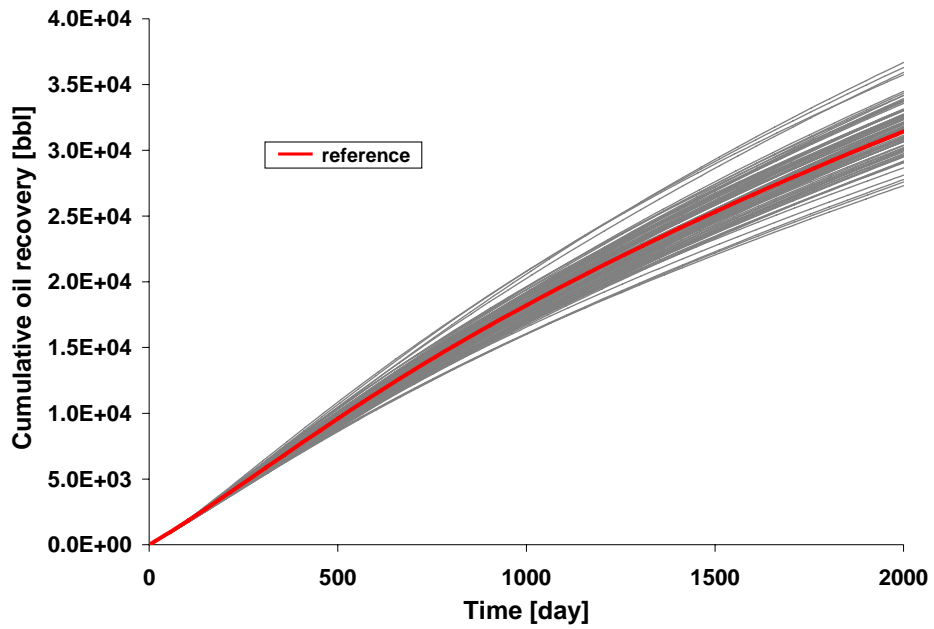


Figure 7-8: Layer-by-layer comparison of mean permeability fields at the end of the history-match.

Forecast results with permeability models obtained via SEIKF



Forecast results with permeability models obtained via EnKF

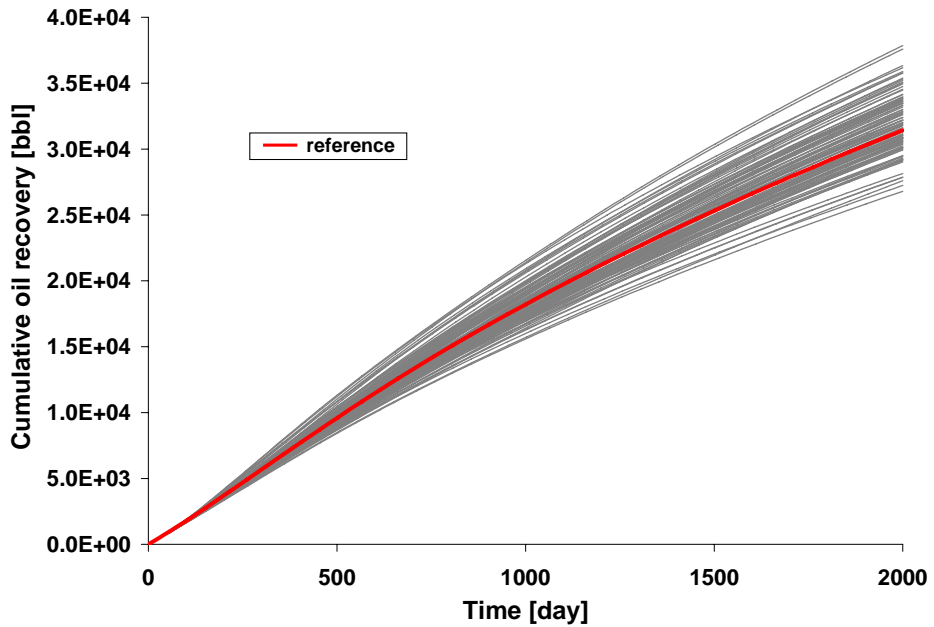
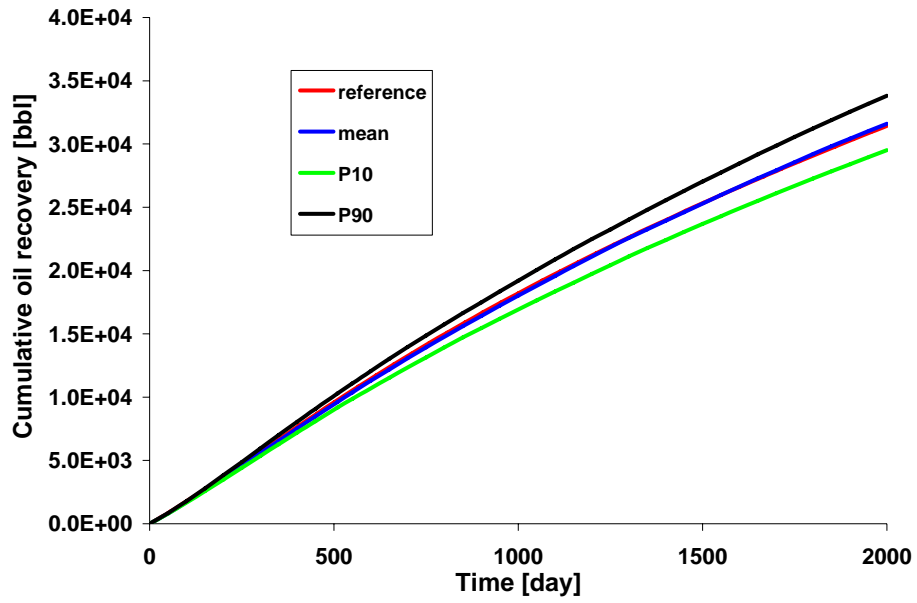


Figure 7-9: Comparison of cumulative oil production forecasts conducted using 100 history-matched models obtained via SEIKF and EnKF. The red curve signifies the reference model response.

Forecast results with models obtained via SEIKF



Forecast results with models obtained via EnKF

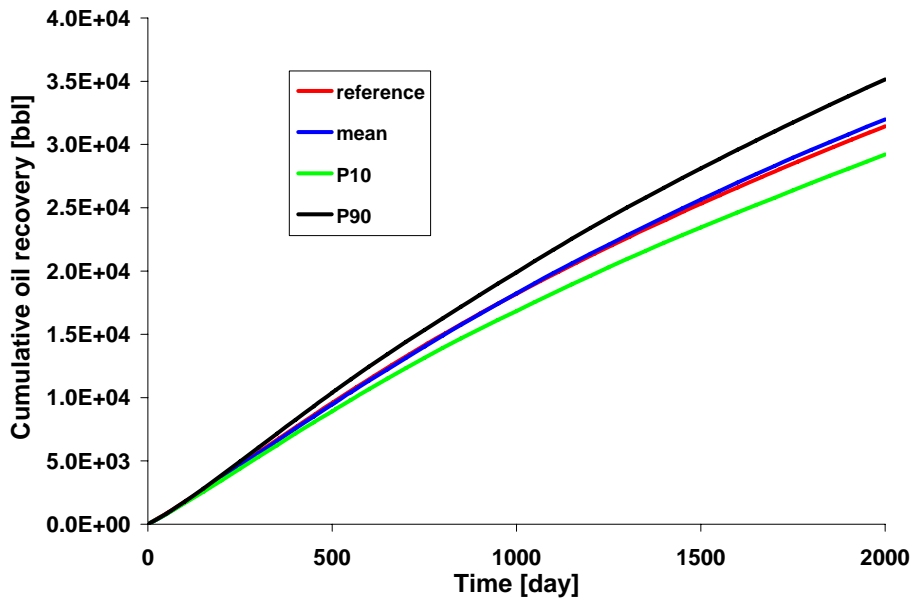


Figure 7-10: Quantification of forecast uncertainty.

Chapter 8: Automatic History Matching with Distributed and Parallel Computing

Over the past two decades, computer performance has advanced tremendously. However, current performance is limited by speed-of-light and material-size limitations. Automatic history matching is still time consuming through the single-processor sequential computing because of the large number of variables, high degree of uncertainties, high-resolution mega-cell reservoir models, modeling complexity and long production history. Fortunately, the independence of the ensemble members during the forecasting steps provides the possibility of distributed processing for the ensemble Kalman filter (EnKF) such that the computational cost is significantly reduced. In addition, parallel reservoir simulation has gradually moved from research to industry application. Commercial reservoir simulators with parallel capability are available. The purpose of this chapter is to efficiently implement automatic history matching through high-performance computing. Two-level computation is adopted, distributing ensemble members simultaneously while simulating each member in a parallel style.

We first give an introduction to distributed and parallel computing, followed by a synthetic case operating from primary, waterflooding to water-alternating-gas flooding. Under different scenarios, the ensemble numbers are distributed and/or parallelized. The history matching results and the computation costs are then analyzed. The last section presents a summary.

8.1 DISTRIBUTED, PARALLEL COMPUTING AND PORTABLE BATCH SYSTEM

Generally, distributed computing is a method of computer processing in which parts of a program run simultaneously on two or more computers that are communicating with each other over a network. A cluster consists of multiple stand-alone machines

working in parallel across a local-high-speed network. In our history matching work, each forward reservoir simulation required by each ensemble member is independent, without inter-communication. Therefore, all the required simulation jobs can be submitted to the cluster at the same time. But the limited number of nodes in the cluster is commonly insufficient to provide for all the submitted jobs. The availability of these nodes is determined by a portable batch system (PBS) in the cluster. PBS is a workload management system for the Linux cluster, which supplies commands to submit, monitor and delete jobs. PBS queues all the submitted jobs and adds them sequentially once a certain node is detected free. All the waiting jobs are pending in the queue. More details about PBS components, function, and script implementation are given in Appendix B.

Parallel computing is the simultaneous execution of the same task (split up and specially adapted) on multiple processors in order to obtain faster results. The key idea is based on the fact that the process of solving a problem can usually be divided into smaller tasks, which are carried out simultaneously with some coordination. Some terminologies are frequently used in parallel computing: efficiency, parallel overhead, and speedup. Efficiency is the ratio of the execution time using one processor and the time using a multiprocessor and the number of processors. Some extra time and memory space are required in parallel computing. This extra work compared to its sequential code is parallel overhead. Parallel speedup is defined as wall-clock time of serial execution divided by wall-clock time of parallel execution.

There are roughly two categories for parallel models: shared memory and distributed memory. The area of parallel reservoir simulation has been extensively investigated to reduce the elapse running time. Scott *et al.* (1987), Chien and Northrup (1993), and Li *et al.* (1995) presented the parallel versions of reservoir simulators on shared memory computers. Wheeler and Smith (1989), Killough and Bhogosvera (1991),

Kaarstad (1995), Rame and Delshad (1995), Shiralkar *et al.* (1997), Chien *et al.* (1997), Killough *et al.* (1997), Parashar *et al.* (1997), Dogru *et al.* (1999), and Wang *et al.* (1999) explored the parallel reservoir simulators on distributed memory computers.

Recently, both companies and universities have been working on the next generation of parallel reservoir simulators. Chevron, with Schlumberger, are developing a highly scalable simulator INTERSECT (DeBaun *et al.*, 2005; Cao *et al.*, 2005; Fjerstad *et al.*, 2005). ExxonMobil developed a simulator called EM^{power} (Beckner *et al.*, 2001). Landmark Graphics is working on a new parallel simulator called NEXUS (Al-Matar *et al.*, 2007). Computer Modelling Group is working on a shared-memory parallel simulator (Collins *et al.*, 2003). The University of Texas at Austin is developing a fully implicit, compositional simulator called GPAS (General Purpose Adaptive Simulator; Han *et al.*, 2005) and an implicit parallel accurate reservoir simulator called IPARS (Lacroix *et al.*, 2001). Stanford University is working on the GPRS (General Purpose Research Simulator; Jiang, 2007).

For the application of parallel simulation on history matching, Ouenens *et al.* (1995) parallelized the simulated annealing algorithm for automatic history matching. Schiozer and Sousa (1997) used the external parallelization to improve history matching through distributing multiple simulations to a network station. Leitao and Schiozer (1999) presented an application example for a history matching problem, where the gradient-based method is used to find the best values for certain reservoir parameters to match pressure and water production data through distributed reservoir simulation. Schulze-Riegert *et al.* (2002) distributed the evolutionary algorithm for history matching reservoir layer permeability, fault transmissibilities and relative permeabilities. Landa *et al.* (2005) presented a distributed high performance computing using a Linux cluster for history matching.

8.2 STRUCTURE OF DISTRIBUTED AND PARALLEL COMPUTING

The computation procedure for the EnKF contains two steps: forecasting and assimilation. The assimilation step requires only a little computational time since the amount of observation data at each measurement time is small. The forecasting step makes it straightforward to take the benefit of parallel and distributed high-performance computing; because multiple simulation jobs are operated independently, these jobs can be distributed to multiple nodes simultaneously. For each job, a parallel version of the reservoir simulator can be applied to occupy multiple processors.

In implementation, to efficiently apply PBS to process multiple jobs, multiple shell scripts are used: submission script, PBS job scripts in the “pbsJobs” folder, and simulation execution scripts in the “realJobs” folder. The submission script determines the host directory and contains all the command lines for PBS job submissions. The PBS job script covers the resource requirements (e.g. memory and CPU time), job attributes, and the set of commands to execute. Each PBS directive starts from “#PBS”. Simulation execution is a shell script containing input link, simulator execution, and some basic processes of output files. Each PBS job execution generates two files: one is echo data and the other is any possible error message. All these files are stored in the “log” folder.

A group of commands is used to detect whether all the submitted PBS jobs have been finished: list all the jobs of the specific user, filter and count all the running and waiting PBS jobs. If the number is zero, start the next step; otherwise, sleep a while and recount again. Chapter 9 gives more details about programming and running a job.

In the input file, the last line is running mode: “0” for sequential, “1” for distributed and “2” for parallel and distributed. Correspondingly, the structure for sequential, parallel, distributed and parallel modes is illustrated in Figure 8-1. In the figure, n is the number of ensemble members (i.e., the number of simulation jobs at each

forecasting step in the EnKF), d is the number of distributed nodes and p is the number of parallel processors in each simulation job.

8.3 CASE STUDY AND DISCUSSION

A three-dimensional quarter-five-spot compositional reservoir simulation problem is used to investigate the performance of the developed history matching module in a parallel and distributed processing platform.

8.3.1 Case Description

The reservoir is discretized by $15 \times 15 \times 20$ with grid sizes 22ft, 22ft, and 7.5ft, in X, Y, and Z directions, respectively. The top of the reservoir has a depth of about 5000ft. In total, there are seven pseudocomponents, i.e., CO₂, C1, C2-3, C4-6, C7-16, C17-29, and C30+. Initial water saturation is 0.35 and initial pressure is 2000 psi.

The porosity field is known for each gridblock. Permeability field is generated by sequential Gaussian simulation with correlation lengths 150ft, 150ft and 30ft in X, Y, and Z directions, respectively. The Dykstra-Parsons coefficient is 0.8.

This is a small reservoir block with two fully-perforated vertical wells. The entire simulation time is 1040 days. Two producers, P0115 and P1501, operate with 1000 psi bottomhole pressure from 0 day to 10 days. The maximum oil surface rate in each well is 400 STB/day. Permeability field associated with well locations is shown in Figure 8-2. Well P0115 then switched into a water injector, named Jw0115. This injector has a constant bottomhole pressure with 2000 psi. At the end of 1000 days, the water injector turns into the solvent injector, named Jg0115, with constant bottomhole pressure of 2100 psi. The producer P1501 is primarily constrained by 1000 psi constant bottomhole pressure and secondarily conditioned by a 400 STB/day surface oil rate.

The observation time is irregular. Since all the wells are constrained by bottomhole pressure, surface rates of oil, gas, and water are obtained from producers at each observation time. The gas or water injection rates in the injectors are also collected. Hence, from 0 day to 10 days, the observation data are oil, water and gas rates in surface condition from both P0115 and P1501. From 11 days to 1000 days, oil, water and gas rates in surface condition from P1501 and water injection rate from Jw0115 are observed. After 1000 days, gas injection rate from Jg0115 is collected instead of water injection rate from Jw0115. The observation times and the amount of data at each time are listed in Table 8-1.

8.3.2 History Matching Results

The weighted EnKF, addressed in Chapter 6, is applied in this history matching case. The twelfth and twentieth layers are shown in Figures 8-3 and 8-4. With time increasing, the permeability map in each layer updates and approaches to the reference. The matched field finally captures the main structure. However, the permeability field does not match very well with the reference model. The reason for the difficult recovery of the reservoir permeability field is of the insufficient information obtained from wells. First, there are 4500 gridblocks in total and each gridblock has an individual permeability value. We only collect 4 or 6 data points at totally 25 observation times. Second, there are 20 layers while the observed data are surface rates at two well locations. The information from each layer, or at the various locations, is not included.

Because history matching is not unique, especially when the measurements are far from the sufficiency, the matched permeability fields are used in the reservoir model to evaluate well performances. The whole simulation period is from 0 day to 1250 days. Figure 8-5 gives the plots of surface oil rate and surface gas oil ratio with time in the producing well P1501. The figures demonstrate that with time increasing and more

information collected, better matching and prediction are achieved. The results from the matched field at the end of 1040 days have good agreements with the reference plots. Both figures also indicate that the observed information is insufficient to recover the reference reservoir field. More constraints, such as using the conditional reservoir field generation at the beginning and adding observation wells, are needed to improve the history matching performance.

8.3.3 High-Performance Computing

With a Dell cluster and commercial reservoir simulator CMG parallel version, we implement the above compositional case using different numbers of processors.

The cluster hardware information is as follows:

- 8 nodes are all Dell PowerEdge 1750s:
 - (2x) Xeon 3.06 GHz, 1.0 MB Cache; i.e., two processors per node
 - 2.0 GB RAM
- Frontend is a Dell PowerEdge 2950:
 - (2x) Dual-Core Xeon, Woodcrest 5130, 2.0 GHz, 4.0 MB Cache; i.e., dual-core looks like 4 processors
 - 2.0 GB RAM (4x512 MB 667MHz)

For software, the SMP version of RedHAT is used for the running kernel. Commercial reservoir simulator GEM parallel version (compositional module of Computer Modelling Group Ltd.) is selected. Note that GEM parallel version uses shared memory architecture computers. Also note that the Linux32 version of GEM 2006.10 runs on Linux_x64 while it does not support parallel. The Linux_x64 version of GEM 2006.12 supports parallel. Both Linux32 and Linux_x64 of GEM 2007.10 versions support parallel. We use the version 2007.10 GEM in this study since our cluster is Linux32.

We first test the parallel efficiency, which is the performance improved as more processing elements are employed. The speedup is one of the ways to measure the performance efficiency of parallel processing and is defined as

$$\text{Speedup} = \frac{t_1}{t_n}$$

where t_1 is the execution time on a single processor and t_n is the execution time on n processors. Speedup performance is largely determined by the ratio of computation cost and communication cost. The ratio can be dependent on both the parallel algorithms and computer hardware/software performance. The ideal speedup of parallel simulation with n processors is n , which means the program runs n times faster. However, as the number of processors becomes larger, the speedup less than n is observed in reality because of the overheads that are not encountered in a single processor.

Our reservoir reference model is run in a single processor and two processors shared memory in one node. For comparison, the node in these two running modes is the same. The total simulation time is 1250 days. The elapsed time in the single-processor case is 2371.58 seconds while the elapsed time in the two-processor parallel case is 1443.69 seconds. Hence, the speedup is about 1.643 while the ideal is 2.0. The speedup is not very good and one possible reason is the short simulation running time.

We then apply the EnKF module to the history matching case using sequential, parallel, and distributed and parallel computing performances. The case and results are described in Sections 8.3.1 and 8.3.2. Because of the short periods in the primary depletion and waterflooding, the simulation times in these sections are very fast. Therefore, we only consider the last three matching steps in the input file to demonstrate the parallel and distributed computing: 1015 days, 1030 days and 1040 days. In our

study, 200 ensemble members are used. After each assimilation step, the permeability field is updated. We need to rerun the case from the beginning because the restart option in GEM does not support such a change in reservoir property. To finish the last three steps in the input file, we study five execution modes:

- Mode 1: One single processor, shown in Figure 8-1(a), needs 93.4 hours.
- Mode 2: Each ensemble member is run in two-processor parallel and only one node is used. The structure is shown in Figure 8-1(b), where $p = 2$. It needs 61.0 hours.
- Mode 3: Each ensemble member is run in two-processor parallel and two nodes are used. The structure is shown in Figure 8-1(c), where $p = 2$ and $d = 2$. It needs 30.5 hours.
- Mode 4: Each ensemble member is run in two-processor parallel and four nodes are used. The structure is shown in Figure 8-1(c), where $p = 2$ and $d = 4$. It needs 15.3 hours.
- Mode 5: Each ensemble member is run in two-processor parallel and eight nodes are used. The structure is shown in Figure 8-1(c), where $p = 2$ and $d = 8$. It only needs 7.6 hours.

The above simulation times are plotted in Figure 8-6. With parallel computing, more distributed nodes bring about a faster execution time. We reduced 93.4 hours (about 4 days) in the single-processor sequential running to only 7.6 hours in two-processor parallel eight-node parallel running. The benefit of using high performance computing is clearly demonstrated.

Here we show only a concept of time saving through high-performance computing. In reality, each simulation model has millions of gridblocks and needs several days, even several weeks. For history matching, if two hundred models are running, there

will be a heavy computational burden. The hardware is relatively cheap and the time is more valuable. We can apply distributed and parallel computing to the automatic history matching process and can greatly save time and improve work efficiency.

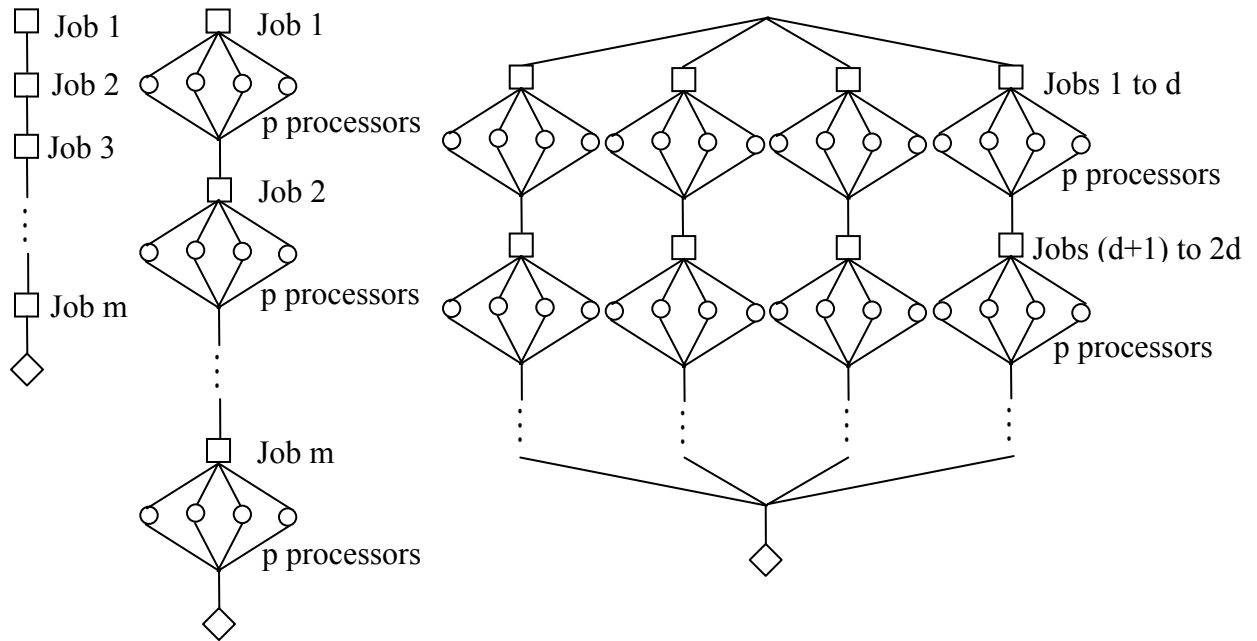
8.4 SUMMARY

Automatic history matching with distributed and parallel computing was investigated in this chapter. Through a compositional case study, the efficiency of distributed and parallel computing during history matching, particularly in the EnKF methodology, was demonstrated. In this case, 4 days of computational time was necessary to perform the simulations using a single-processor computer. This time was reduced to about 7.6 hours using distributed/parallel processing (eight distributed nodes with two parallel shared memory processors).

The EnKF based history matching has the straightforward advantage of distributed computing. The parallel version of the reservoir simulator can be used for additional gain in reducing the actual time for history matching. Therefore, distributed and parallel computing techniques are recommended for the automatic history matching process and can significantly reduce the execution time.

Table 8-1 Observation data during the simulation time

Time (days)	Surface Oil Production Rate	Surface Water Production Rate	Surface Gas Production Rate	Injected Water Rate	Injected Gas Rate	Total Number of Data
1	2×1	2×1	2×1	0	0	6
4	2×1	2×1	2×1	0	0	6
7	2×1	2×1	2×1	0	0	6
10	2×1	2×1	2×1	0	0	6
11	1×1	1×1	1×1	1	0	4
15	1×1	1×1	1×1	1	0	4
20	1×1	1×1	1×1	1	0	4
30	1×1	1×1	1×1	1	0	4
40	1×1	1×1	1×1	1	0	4
60	1×1	1×1	1×1	1	0	4
100	1×1	1×1	1×1	1	0	4
200	1×1	1×1	1×1	1	0	4
300	1×1	1×1	1×1	1	0	4
400	1×1	1×1	1×1	1	0	4
460	1×1	1×1	1×1	1	0	4
560	1×1	1×1	1×1	1	0	4
660	1×1	1×1	1×1	1	0	4
780	1×1	1×1	1×1	1	0	4
830	1×1	1×1	1×1	1	0	4
920	1×1	1×1	1×1	1	0	4
1000	1×1	1×1	1×1	1	0	4
1001	1×1	1×1	1×1	0	1	4
1015	1×1	1×1	1×1	0	1	4
1030	1×1	1×1	1×1	0	1	4
1040	1×1	1×1	1×1	0	1	4



(a) Sequential (b) Parallel (c) Distributed and Parallel

- Simulation job
- Processor in the parallel computing
- ◇ Assimilation step to update reservoir prosperities

Figure 8-1: Schematic structure of sequential, parallel, and distributed and parallel computing in the EnKF process.

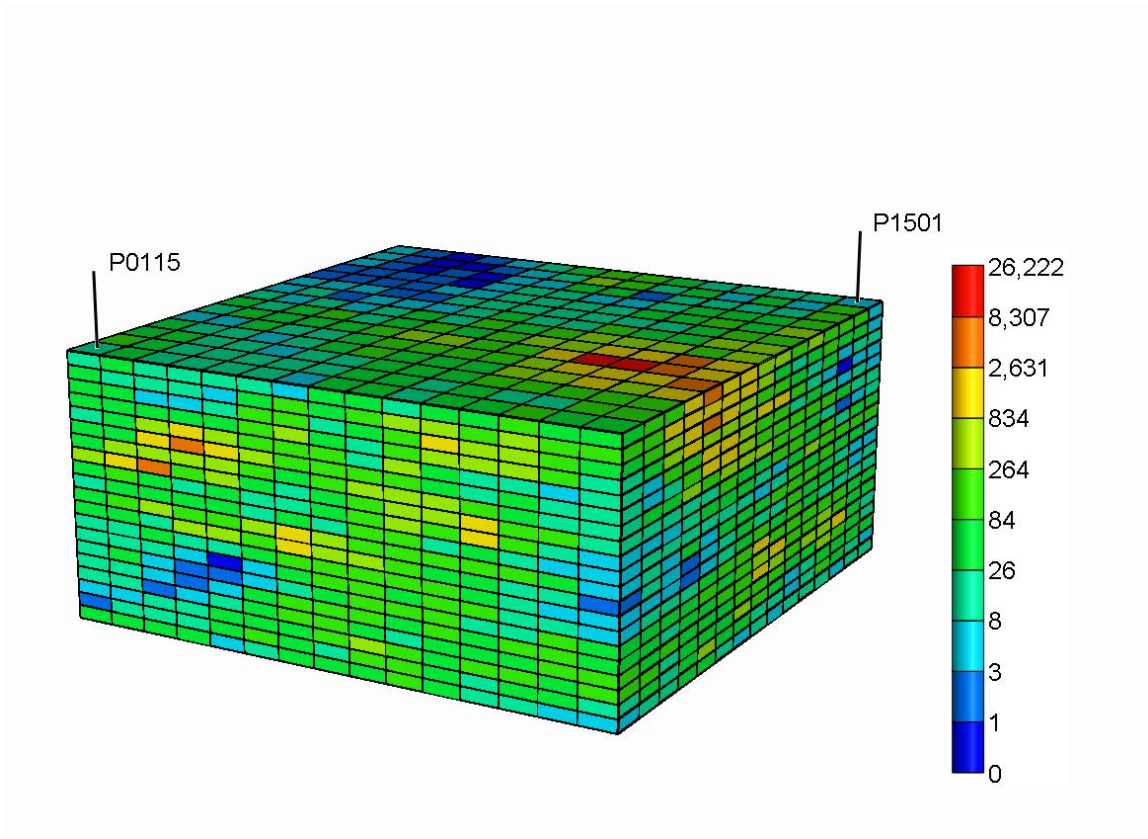


Figure 8-2: Three-dimensional view of reservoir permeability.

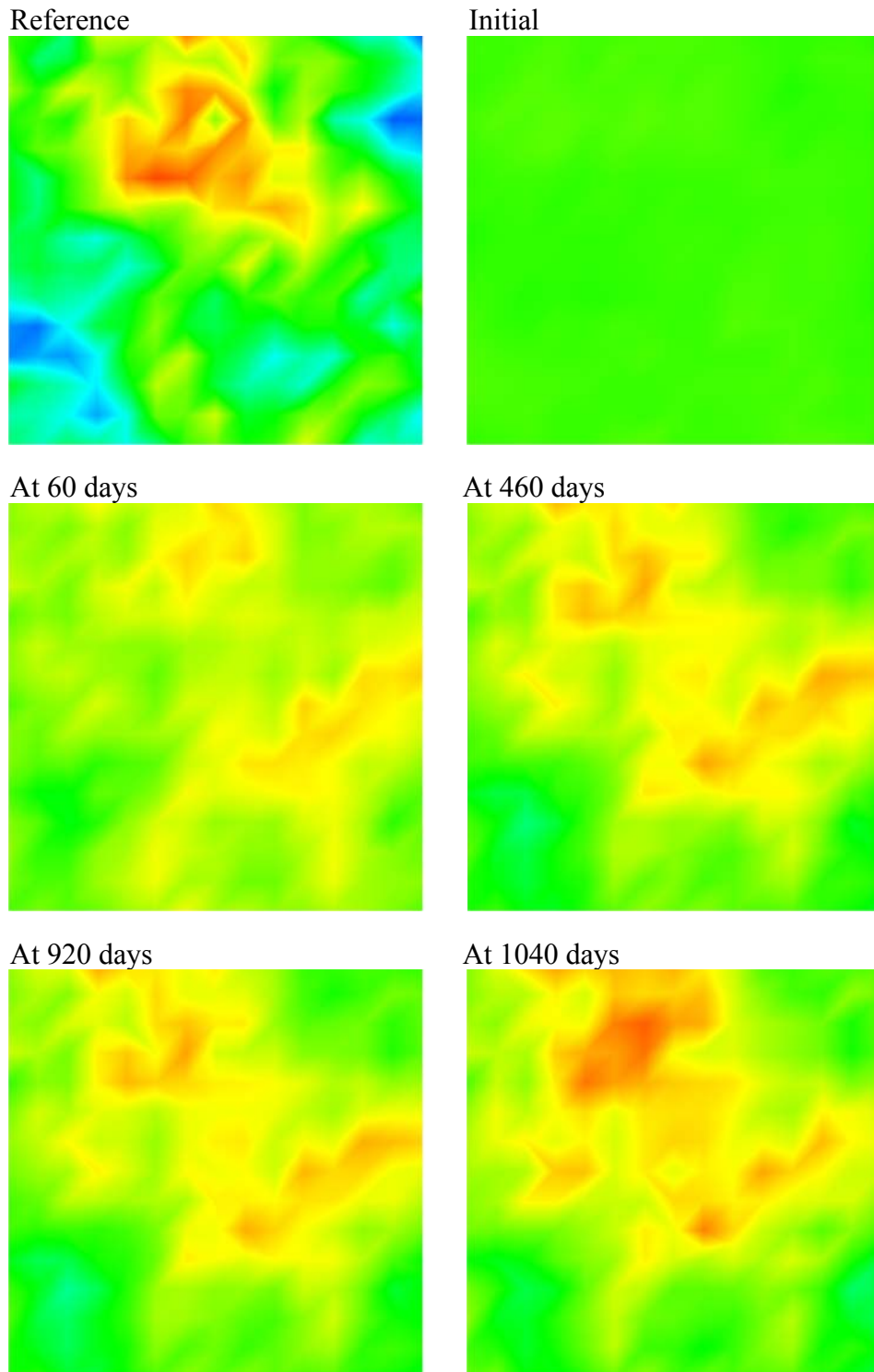


Figure 8-3: Evolution of the mean permeability field in Layer #12 after logarithm transformation. Permeabilities range from 1 mD (blue) to 1100 mD (red).

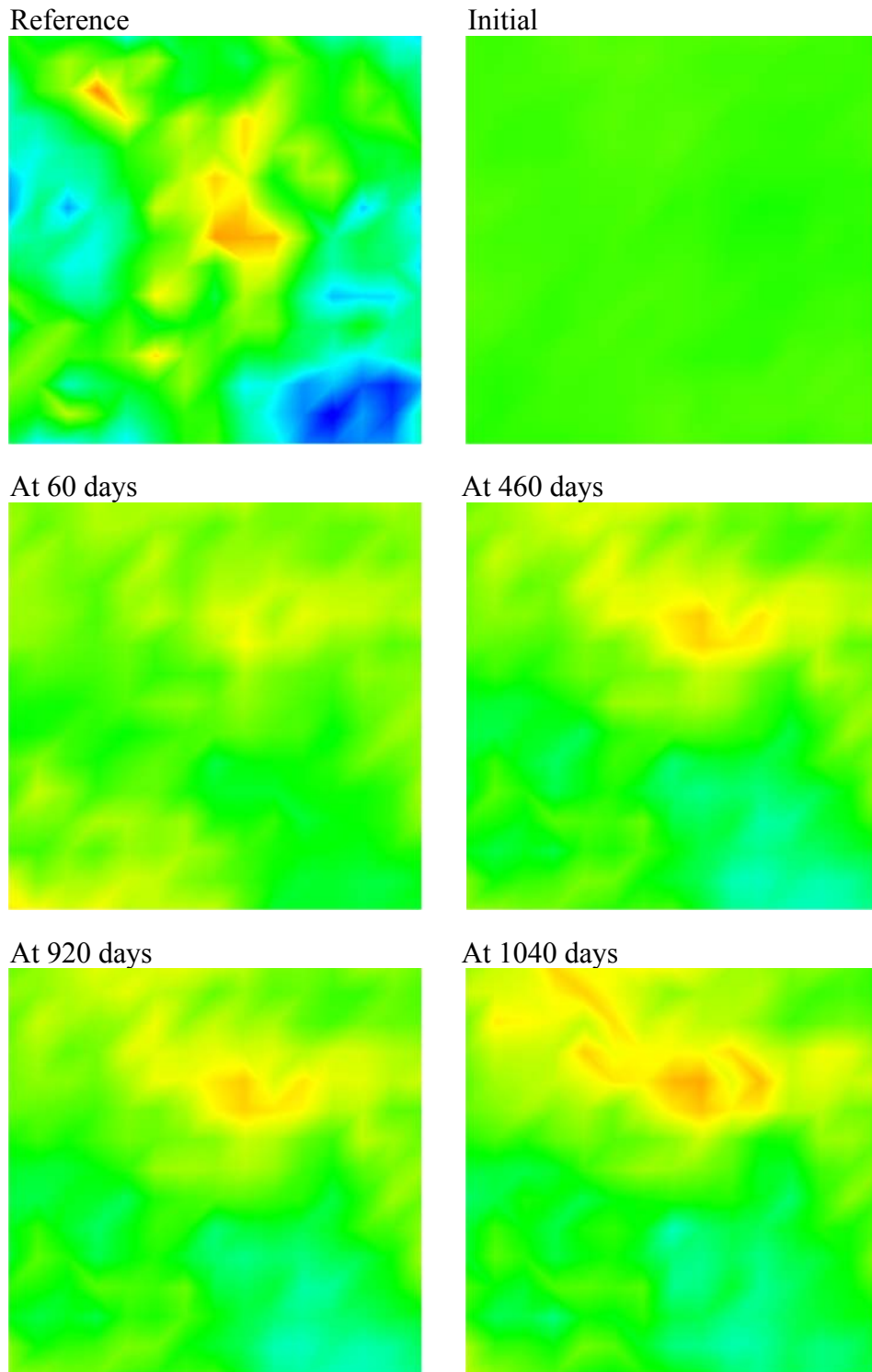
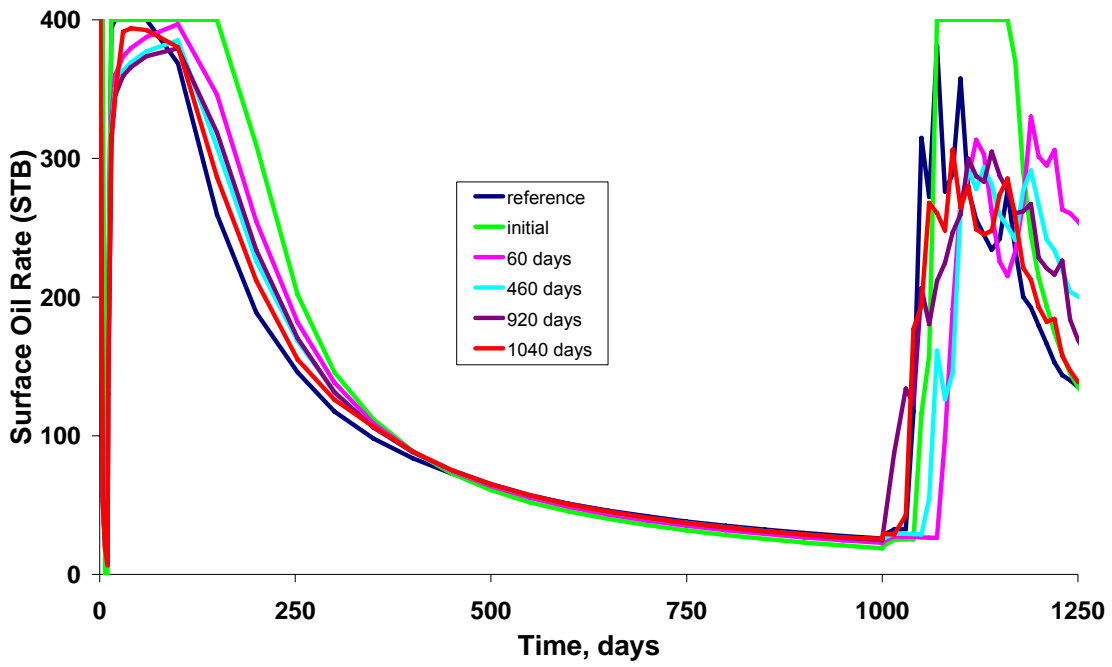


Figure 8-4: Evolution of the mean permeability field in Layer #20 after logarithm transformation. Permeabilities range from 1 mD (blue) to 1100 mD (red).

(a) Surface Oil Rate at Well P1501



(b) Surface Gas Oil Ratio at Well P1501

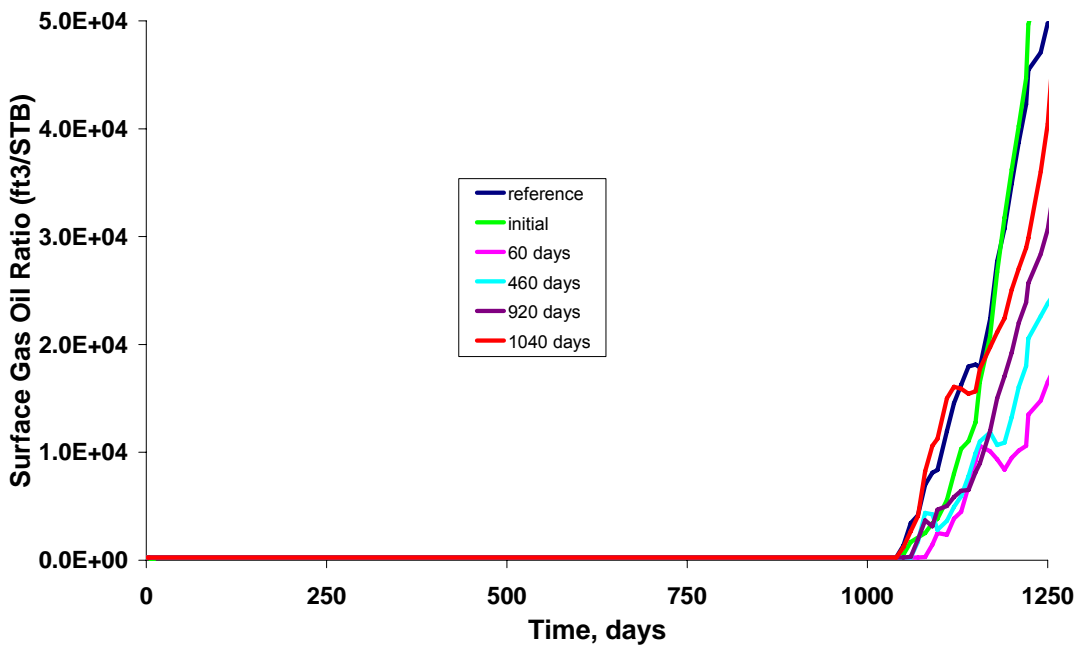


Figure 8-5: History matching and forecasting of surface oil rate and surface gas oil ratio in well P1501 by using the reservoir field obtained at the different time.

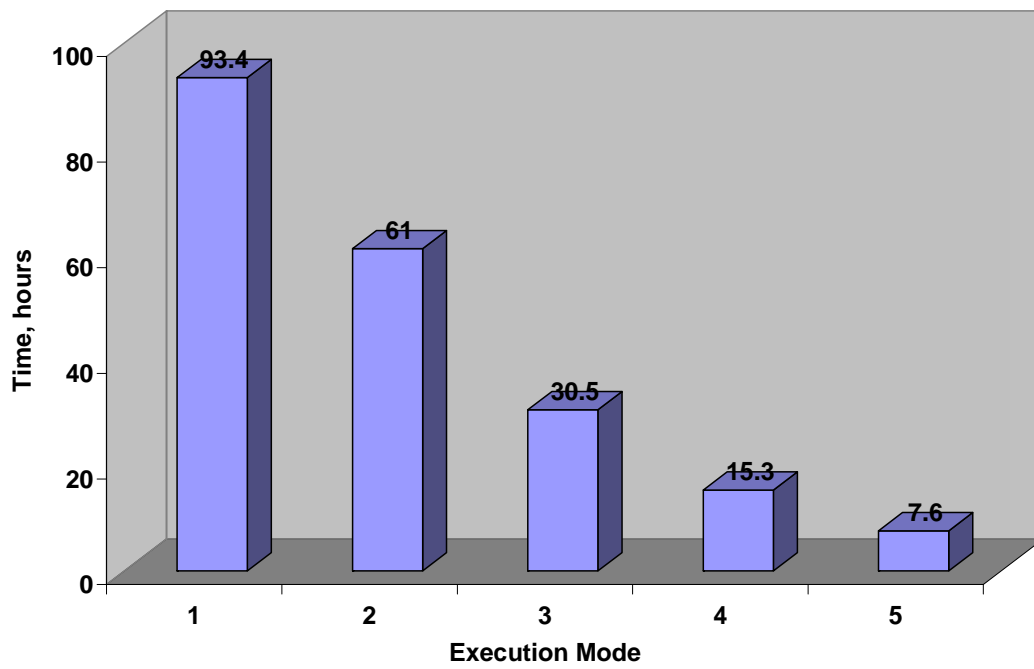


Figure 8-6: Elapsed time in five different execution modes.

Chapter 9: Automatic History Matching Module

This Chapter is served as a guideline for using the automatic history match module developed and discussed in previous chapters. The platform UT_IRSP is first introduced followed by an outline for the automatic history module. The main program and a procedure for compilation and execution are then discussed. Example input files for UTCHEM, ECLIPSE, and CMG reservoir simulators are described and provided.

9.1 INTRODUCTION TO UT_IRSP

Increasing hydrocarbon production via advanced technologies commonly involves the use of numerical simulation of the associated processes to minimize the risk involved in development decisions. The oil industry today requires much more detailed analysis with a greater demand for reservoir simulations with more detailed geological, physical, and chemical models than in the past. Without detailed simulations it is very unlikely that cost effective recovery processes can be developed and applied economically. Although reservoir simulation software is currently available, there are still many obstacles to the widespread and effective use in the upstream oil and gas industry, such as time-consuming data preparation and output analysis, large uncertainties associated with the petrophysical properties and methods for performance predictions, and a large number of scenarios required for performance optimization in reservoir simulations.

The UT_IRSP (Integrated Reservoir Simulation Platform), developed by Jiang Zhang (2005), is a user-friendly framework to promote the routine application of reservoir simulation for design and optimization. In particular, it facilitates the problem solving during well location optimization, sensitivity studies to rank the important

reservoir parameters, stochastic simulation to gauge risk, the design and optimization of chemical flooding processes, and a composite study that combines the above.

Using the concept of the objected-oriented design, the UT_IRSP has integrated three reservoir simulators, two spatial stochastic field generators, and two job schedulers for performing distributed and/or parallel computing. Three reservoir simulators are UTCHEM, ECLIPSE, and VIP. Two spatial stochastic field generators are MDM and SGSIM of GSLIB for generating initial realizations of reservoir properties. Job schedulers are Portable Batch System (PBS) and Load Sharing Facility (LSF). There are several windows-based commercial software packages used to design, analyze, and optimize the results from reservoir simulation studies. For experimental design and response surface, Design-Expert (Stat-Ease, Inc., 2003) is used. Crystal Ball (Decisioneering, Inc., 2001, 2004) is used for optimization with uncertainties through Monte Carlo simulation. Tecplot RS (Tecplot, Inc., 2004) is used for visualization of UTCHEM and ECLIPSE output maps, whereas VIP's 3DView is used for visualization of VIP maps. Microsoft Excel is used for processing well data for UTCHEM. Surfer (Golden Software, Inc.), can be used for variogram analysis of the geostatistical data.

The UT_IRSP first prepares and executes multiple reservoir model cases as specified in the user's instruction files denoted as preprocessing step. Post-processing step of the UT_IRSP extracts desired simulation results from each simulation run, facilitates the three-dimensional visualization, and generates statistical information on certain variables such as reservoir recovery. The automatic history matching module with EnKF methodologies is now added to the UT_IRSP. When the UT_IRSP is launched, the program will prompt the user to select either backward history matching or forward prediction and optimization.

Figure 9-1 shows the working environment of the UT_IRSP.

9.2 REQUIREMENTS OF THE AUTOMATIC HISTORY MATCHING MODULE

The history matching module is implemented in Linux environment. The computation platform can either be a single-processor Linux computer or a cluster of them. Petros cluster located in the Department of Petroleum and Geosystems Engineering is used. This cluster has eight nodes, from “compute-0-0” to “compute-0-7”, each of which has two CPUs sharing 2G of memory. To login, type “petros.cpge.utexas.edu” in SSH secure shell. The job can be submitted in any of these eight nodes with/without a script and run in sequential style, or submitted from Petros I/O node and run in either distributed or distributed and parallel environment.

Linux uses a PATH environment variable to tell the operating system where to look for files or programs to execute. The PATH environment variable is stored in an operating system table, along with several other environment variables. In a Linux command line, you can type “env” to view the contents. In Petros, PATH is defined in the “.bash_profile” file. The “.bash_profile” file is one of several scripts that are always run upon log in. The user can add a directory to the PATH editing the “.bash_profile”.

For matrix computation and random seed generation of the history match module, four extra libraries are used: LAPACK (LAPACK website), BLAS (BLAS website), EISPACK (EISPACK website), and FFTW (FFTW website). LAPACK, Linear Algebra PACKage, is written in Fortran 77 and provides routines for solving systems of linear equations, least-squares solutions of linear systems of equations, eigenvalue problems, and singular value problems. The associated matrix factorizations (LU, Cholesky, SVD, QR) are also provided. BLAS, Basic Linear Algebra Subprograms, are routines that provide standard building blocks for performing basic vector and matrix operations. Because the BLAS are efficient, portable, and widely available, they are commonly used in the development of high quality linear algebra software. EISPACK is a collection of

FORTRAN subroutines that compute the matrix eigenvalues and eigenvectors. In addition, two routines are included that use singular value decomposition to solve certain least-squares problems. FFTW is a free collection of fast C routines for computing the discrete Fourier transform in one or more dimensions. It includes complex, real, symmetric, and parallel transforms, and can handle arbitrary array sizes efficiently. All the above packages are free to download from their websites. In our Petros cluster, LAPACK and BLAS have been installed. The user needs to install FFTW and EISPACK, and then put their absolute paths in the makefile.

Three simulators, UTCHEM, ECLIPSE and CMG, are installed on the cluster. UTCHEM, developed at The University of Texas at Austin, is a three-dimensional compositional chemical flooding simulator. The solution scheme is analogous to IMPES, where pressure is solved for implicitly, but concentrations rather than saturations are then solved for explicitly. Phase saturations and concentrations are then solved in a flash routine. The UTCHEM distribution package is free and contains FORTRAN 90 source files, two files used for compiling the code (Commodule and Makev9). To compile UTCHEM in Petros cluster, first issue the command “Commodule”, which will compile the modules; then issue the command “make -f Makev9 FC=ifort” to build the executable file called “utchem93.exe”; last, change to the executable model “chmod 700 utchem93.exe” and move it to “bin” folder in the user’s home directory. ECLIPSE, installed in “/opt/eclipse” in Petros, is already set in the user’s path and can be used directly. CMG software, installed in “/share/apps/cm/gem/2007.10/Linux32/exe” folder, needs to be set in the user’s path: in user’s home directory, type “vi .bash_profile” and add “PATH=\$PATH:\$HOME/bin:/share/apps/cm/gem/2007.10/Linux32/exe” under the line of “# User specific environment and startup programs”. In this way, “bin” folder and CMG directory are added to user’s path.

MDM and SGSIM and the source codes are available to generate initial geostatistical realizations, i.e., permeability field. MDM source code has only one file “mdm.f” and can be compiled by “ifort -o mdm.exe mdm.f”. FORTRAN 90 source codes for SGSIM can be compiled by using makefile. Under directory of source codes, type “make” and it will generate “sgsim.exe” executable file. Then, move “sgsim.exe” to user’s “bin” folder.

To assist high performance computing, Portable Batch System (PBS) is installed in Petros. “qsub”, “qstat” and “pbsnodes” are three commands most frequently used to submit a PBS job, to check the status of jobs, queues and the PBS server, and to show all the PBS nodes’ status.

9.3 STRUCTURE OF THE AUTOMATIC HISTORY MATCHING MODULE

The automatic history matching module, written in FORTRAN 90, contains one main program and twelve important modules. Module design, similar to the object-oriented design in C++, conveniently encapsulates one function within one module and can freely use other modules. In addition, one source file “source.files”, one dependent file “depends.file” and one makefile “makefile” are used to build the executable file.

File “main.F90” is the program engine and works as the front end. Once the history matching module is launched, it will first read the input file “infile.in” with the information provided by the user. These data include number of gridblocks, number of realizations, the choice of software for initial realization i.e. MDM or SGSIM, the number of observations and the corresponding time of the observation, and the choice of sequential or distributed computing.

The program generates multiple initial realizations using the user-specified software such as MDM and SGSIM and also creates folders as many as the number of realizations by modules “m_SGSIM.F90”, “m_MDM.F90” and

“m_folderGeneration.F90”. Simulator input file(s) are automatically copied into each folder. All the realization results then are pasted in the appropriate location in the input file in each folder, i.e., permeability field.

The program enters into a big time step loop, starting from the beginning to the final time given by the user. Within the time loop, the matching parameters are updated: if it is the first time, the permeability, pressure, and saturation data from the initial stochastic results are used; otherwise, the results of Kalman filter methods in the last observation data are used. The program updates the permeability fields in each folder if it is not the first time step. Meanwhile, restart options are also implemented according to different requirements in different reservoir simulators. If the distributed computing is invoked, distributed files for PBS submission are then created. The reservoir simulator then runs the input case in each folder in a sequential or distributed style. All the above steps are performed using module “m_changeSimTime.F90”.

Once all the jobs are completed, the program reads the observation data prepared by the user and extracts the corresponding response of pressure and water/oil saturation in each gridblock from the simulation outputs in the folders of different cases. These tasks are formed in modules “m_extractTrue.F90” and “m_extractEnsemble.F90”. To keep variability between the ensemble members (Burgers, 1998), white noises randomly perturbing the observation data are added in the module “m_measurements.F90” to create an ensemble of observation datasets for the ensemble reservoir cases.

Since permeability fields are assumed to have lognormal distribution and the Kalman-filter-based methods are implemented in Gaussian space, a module “m_logexp.F90” is used to transform the permeability data from lognormal to normal spaces.

For the assimilation step, more than ten different assimilation schemes are implemented in the EnKF method, including direct inverse, standard computation algorithm, square-root algorithm, and weighted EnKF. Direct inverse and weighted EnKF schemes are coded separately in the module “m_olddenkf.F90” and “m_weightedenkf.F90”, while all the other EnKF schemes are in the module “m_enkf.F90”. The module “m_seikf.F90” is used for the singular evolutive interpreted Kalman filter discussed in more details in Chapter 7.

After the assimilation step, the module “m_tecplot.F90” is used to prepare the average and variance of permeability data calculated from the ensemble members. Finally, the saturation and pressure data in each ensemble member before assimilation step are written to output files in folders “pressure” and “saturation”. The CPU time is given in “cputime.dat” file. The response data of each ensemble member during all the observation times are given in “observation.dat” file. For easy comparison and error detection, the true observation data and the corresponding average data from the results in ensemble members along with assimilation time and CPU time cost for each assimilation step are given in the file “overall.dat”.

The flowchart of the automatic history matching module is illustrated in Figure 9-2.

9.4 COMPILATION PROCEDURE FOR THE AUTOMATIC HISTORY MATCHING MODULE

Intel® Fortran Compiler 9.1.040 for Linux is used in Petros cluster to compile the automatic history matching module. The compiler processes FORTRAN language source and generates object files. There are four steps in compiling the program: preprocess, compile, assemble, and link. The Intel® Fortran Compiler can be invoked in either of two ways: using the “ifort” command or using the “make” command to specify a makefile.

The following command requests a file name “mdm.exe” for the source file “mdm.f”:
“ifort -o mdm.exe mdm.f”.

A makefile associated with “depends.file” and “source.files” is used for the automatic history matching module. In the file “source.files”, we list all the modules and F90 files. File “depends.file” gives the relationships of the files. The implementation is straightforward. First, place all the source files in one folder and create a folder “TMP” in this folder where the source files are located. Second, type “make” to generate the executable program. If you want to change the name of the executable program, edit “makefile” and change the character string of “TARGET”. Third, if you update a routine and need to recompile, type “make clean; make” to ensure deleting the previous objective files. The following shows the content of the “makefile”:

```
VPATH = .:RCS:TMP

.SUFFIXES:
.SUFFIXES: .o .F90 .f90 .F .f .H .h

LD = ifort
CF90 = ifort
CF77 = ifort

FFLAGS = -c -r8
PAR =
DEBUG_FLAGS =
F77FLG =
F90FLG =
LINKFLAGS = -r8

CPPARCH = -DLINUX
CPPFLAGS = -P $(CPPARCH)

LIBS = -llapack -lblas -lfftw3 eispack_linux.a

CPP = /usr/bin/cpp
# Rules for running cpp and updating files in TMP directory
```

```

.H.h:
    rm -f ./TMP/$*.h
    cat MODEL.CPP $*.H | $(CPP) $(CPPFLAGS) > ./TMP/$*.h
.F90.o:
    rm -f ./TMP/$*.f90
    cat MODEL.CPP $*.F90 | $(CPP) $(CPPFLAGS)
> ./TMP/$*.f90
    cd ./TMP ; $(CF90) $(FFLAGS) $(F90FLG) -o $*.o $*.f90
.F.o:
    rm -f ./TMP/$*.f
    cat MODEL.CPP $*.F | $(CPP) $(CPPFLAGS) > ./TMP/$*.f
    cd ./TMP ; $(CF77) $(FFLAGS) $(F77FLG) -o $*.o $*.f

TARGET = 3DHIST

include source.files

INC2 =$(INC1:.H=.h)
FILES =$(F90FILES) $(F77FILES) $(MODULES)
FFILES =$(F90FILES:.F90=.f90) $(F77FILES:.F=.f)
$(MODULES:.F90=.f90)
OBJECTS = $(F90FILES:.F90=.o) $(F77FILES:.F=.o)
OMOD = $(MODULES:.F90=.o) $(MODULES77:.F=.o)

all: $(TARGET)

$(TARGET): $(INC2) $(OMOD) $(OBJECTS)
    cd ./TMP ; $(LD) $(LINKFLAGS) -o ../$(TARGET) $(OMOD)
$(OBJECTS) $(LIBS)
install:
    cp $(TARGET) $(HOME)/bin
clean:
    cd ./TMP ; rm -f *.f *.o *.f90 *.h *.mod
include depends.file

```

9.5 DESCRIPTION OF THE INPUT FILE

An input file “infile.in” is required to provide the basic information, including the number of gridblocks, selection of geostatistic software for initial ensemble members, the mean and variance of the field if MDM or the unconditional SGSIM is used, number of realization, variances of observation data, total number of observation time and

corresponding number of observation data, study name, and running mode. An example of the “infile.in” file for an input case for UTCHEM simulator discussed in Chapter 6 is given as follows:

```

22 25 17          ! numbers of gridblocks in X, Y and Z directions, respectively
1                ! 1 for SGSIM; 2 for MDM to generate permeability field
3                ! variance of ln(k), if MDM is used, just put any number
6                ! mean of ln(k), if MDM used, any number will be OK
200              ! number of realizations
1                ! generate initial realizations. if =1 for normal; if >1 for SVD
1.0              ! data errors for oil rate
3.0              ! data errors for water rate
3.0              ! data errors for bottomhole pressure (BHP)
4                ! EnKF: 1-standard; 2-square root; 3-direct inverse; 4-weighted
34 10 7          ! # of times where data available; max # of wells for rate
                 ! and max # of wells for BHP measurement
30 38 2 2        ! time, # of data, # of wells for rate and # of wells for BHP
1 4              ! if # of wells with rate data > 0, well ID for rate data
1 4              ! if # of wells with BHP data > 0, well ID for BHP data
60 38 2 2
1 4
1 4
90 38 2 2
1 4
1 4
120 38 2 2
1 4
1 4
150.05 38 2 2
1 4
1 4
180.05 72 2 4
2 5

```

1	2	4	5
210.05	72	2	4
2	5		
1	2	4	5
240.10	72	2	4
2	5		
1	2	4	5
270.10	77	4	4
1	2	4	5
1	2	4	5
300.10	77	4	4
1	2	4	5
1	2	4	5
330.10	77	4	4
1	2	4	5
1	2	4	5
360.10	77	4	4
1	2	4	5
1	2	4	5
390.10	77	4	4
1	2	4	5
1	2	4	5
420.10	77	4	4
1	2	4	5
1	2	4	5
450.10	77	4	4
1	2	4	5
1	2	4	5
480.10	77	4	4
1	2	4	5
1	2	4	5
510.10	77	4	4
1	2	4	5
1	2	4	5

540.10	77	4	4						
1	2	4	5						
1	2	4	5						
570.10	77	4	4						
1	2	4	5						
1	2	4	5						
600.10	77	4	4						
1	2	4	5						
1	2	4	5						
620.10	139	10	7						
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17			
640.10	139	10	7						
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17			
660.10	139	10	7						
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17			
680.10	139	10	7						
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17			
700.10	139	10	7						
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17			
720.10	139	10	7						
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17			
740.10	139	10	7						
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17			
760.10	139	10	7						
1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17			
780.10	139	10	7						

```

1      2  3  4  5  6  7  8  9  10
11     12 13 14 15 16 17
800.10 139 10 7
1      2  3  4  5  6  7  8  9  10
11     12 13 14 15 16 17
850.10 139 10 7
1      2  3  4  5  6  7  8  9  10
11     12 13 14 15 16 17
900.10 139 10 7
1      2  3  4  5  6  7  8  9  10
11     12 13 14 15 16 17
950.10 139 10 7
1      2  3  4  5  6  7  8  9  10
11     12 13 14 15 16 17
1000.10 139 10 7
1      2  3  4  5  6  7  8  9  10
11     12 13 14 15 16 17

```

```

CHAPT6          ! study name
PERMX           ! keyword of parameters for matching
TMAX           ! keyword of simulation time
IMODE          ! keyword of simulation running mode: 1-first run; 2-restart
1              ! running mode: 0 for sequential; 1 for PBS distribution
0              ! water saturation: 0-uniform; 1-per layer; 2-per gridblock
2              ! reservoir pressure: 0-uniform; 1-per layer; 2-per gridblock

```

Note that the units for date errors for rate and pressure should be consistent with the units given in the observation data files.

9.6 STORAGE HIERARCHY

Before running the history matching module, the observation data is stored in one folder. Besides, a folder “SGSIM” for SGSIM software or folder “MDM” for MDM

software is required. Inside of such folder, the file for SGSIM or MDM is already prepared.

Once the history matching module is executed, multiple folders which are equal to the number of realizations are created. Each folder contains one independent reservoir simulation case, whose permeability field is automatically updated after each assimilation step. An empty folder “Tecplt-match” is created to store the matched results of each layer in the format of the Tecplot software. In this folder, the permeability mean and variance of ensemble realizations in each gridblock from the beginning to the end of history matching time are stored in the Tecplot format.

If PBS is used, a batch file “subPBSJobs.job” is created to submit multiple PBS jobs. In addition, four additional folders are generated: “log”, “pbsJobs”, “pbsOut” and “realJob”. “pbsJobs” contains the batch file to implement PBS. The echo information during PBS job running on different nodes is stored in the folder “log”. “realJob” folder contains the batch files to run simulation in each realization folder. The information of PBS job identifications and any possible error message is stored in the “pbsOuts” folder.

9.7 MDM AND SGSIM INPUT FILES

The evolution of understanding reservoir structure has been a complex interaction between quantitative and qualitative judgment. Geostatistics combines the empirical conceptual ideas that are implicitly subject to degrees of uncertainty with the rigor of mathematics and formal statistics. It has found its way into the field of reservoir characterization and dynamic history matching. In our history matching module, we use MDM and SGSIM to generate initial permeability realizations based on prior geological knowledge and hard data measured in specific locations if possible.

9.7.1 MDM

The current version of MDM has the same coordinate system as the commonly used reservoir coordinate system: in areal view, X direction goes toward East, Y direction goes toward South; Z direction goes downward. The input file “INPUT” includes the number of divisions and gridblock sizes in three directions, the number of variograms, the number of realizations, starting random seed number, output option selection. In addition, the user needs to specify the mean and variance (or Dykstra-Parsons coefficient) of permeability field, semivariogram model, correlation length in X direction and its ratios to Y and Z directions, respectively. If hard data in the well location is available, a separate file “INDAT” is needed, in which each line has X, Y, Z grid number locations and the measured permeability values. If hard data is unavailable, just put “0” in “INDAT” file. The relationship between variance after lognormal transformation $\sigma_{\ln k}^2$ and the Dykstra-Parsons coefficient V_{DP} is given by the following formula (Brown, 1993):

$$\sigma_{\ln k}^2 = [\ln(1 - V_{DP})]^2. \quad (9.1)$$

To execute the MDM program, just type “mdm.exe” in Linux environment. An MDM example used in Chapter 7 is given as follows:

```
perm-1
NX, NUMBER OF X DIVISIONS?
19.00
NY, NUMBER OF Y DIVISIONS?
19.0
NZ, NUMBER OF Z DIVISIONS?
3.0
NV, NUMBER OF VARIOGRAMS (IN THE NESTED MODEL)?
1.0000
```

```

NR, NUMBER OF REALIZATIONS, <1 AS 1 EXCEPT O-FORM.?
1.0000
NS, NUMBER FOR STARTING RANDOM NUMBER (SEED)?
3500
MO, OUTPUT (INPUT) OPTIONS, SUM OF ALL OPTIONS?
5130
DX, GRID SIZE IN X DIRECTION, CONSISTENT UNIT?
32.8
DY, GRID SIZE IN Y DIRECTION, SAME UNIT AS DX?
32.8
DZ, GRID SIZE IN Z DIRECTION, SAME UNIT AS DX?
10
#1: P OF P-NORMAL FOR THIS TERM OF VARIOGRAM, 0=LOG-N.?
0.
#1: MEAN VALUE FOR THIS TERM, CONSISTENT UNIT AS S.D.?
350
#1: CORRELATION LENGTH, X MAJOR AXIS, SAME UNIT AS DX?
460
#1: CO.MODEL, -1=EXP, -2=DEXP, -3=SPH, -5=INPUTACF, +=POWER?
-3.
#1: RATIO OF CORRELATION LENGTHES IN MAJOR AXES, LX/LY?
1.0
#1: RATIO OF CORRELATION LENGTHES IN MAJOR AXES, LX/LZ?
10
#1: VDP FOR THIS TERM, SAME UNIT AS MEAN?
0.8

```

The deficiency of MDM is that it is very slow for large number of gridblocks and it does not support the option that reservoir trend has an angle with the grid coordinate system. Note that the value 5130 after the line “MO, OUTPUT (INPUT) OPTIONS, SUM OF ALL OPTIONS?” is used in the above file. This number is a summation of several input and output options, which are represented by specific numbers: 1024 means using sample mean; 4096 means using Dykstra-Parsons coefficient instead of standard deviation; 2 means using external storage for big arrays; 8 means an output format with reduced variables.

9.7.2 SGSIM

SGSIM requires similar information to MDM program. SGSIM has great advantages. It is much faster than MDM and offers more options, such as six parameters to define the geometric anisotropy of a semivariogram structure in three dimensions, multiple grid search, and maximum search radii.

However, SGSIM has different grid definition: the X axis is associated to the east direction; the Y axis is associated to the north direction; and the Z axis is associated to the elevation, upward. Rather than block-centered oriented, the grid is node-point oriented. A careful transformation of the output is needed to be consistent with the reservoir coordinate system. The history matching module has the capability of completing such automatic system transform.

If the unconditional simulation is implemented, SGSIM generates the output with the standard normal distribution with zero mean and one variance. The history matching module implements the transformation from normal distribution to lognormal distribution by the following transform formula:

$$K_{new} = \exp(K_{old} \times \sigma_{\ln k} + \mu_{\ln}), \quad (9.2)$$

where, K_{new} is the desired permeability data with the lognormal distribution; K_{old} is the data directly from SGSIM output with the normal distribution; $\sigma_{\ln k}$ and $\mu_{\ln k}$ are the standard deviation and mean of K_{old} .

The default file of SGSIM is “sgsim.par”. The execution command is “sgsim.exe filename”. “filename” is the input file name. As an example, SGSIM file used in Chapter 8 is given. For the meaning of each parameter in each line, refer to Deutsch and Journal (1998).

Parameters for SGSIM

START OF PARAMETERS:

```

cluster.dat      - file with data
1 2 0 0 0 0     - columns for X,Y,Z,vr,wt,sec.var.
-1.0          1.0e21 - trimming limits
0              - transform the data (0=no, 1=yes)
sgsim.trn       - file for output trans table
0              - consider ref. dist (0=no, 1=yes)
histsmth.out    - file with ref. dist distribution
1 2            - columns for vr and wt
0.0   10.0     - zmin,zmax(tail extrapolation)
1      0.0     - lower tail option, parameter
1     10.0     - upper tail option, parameter
1              - debugging level: 0,1,2,3
sgsim.dbg       - file for debugging output
sgsim.out       - file for simulation output
200            - number of realizations
15   11.0   22.0 - nx,xmn,xsiz
15   11.0   22.0 - ny,ymn,ysiz
20   3.75   7.5  - nz,zmn,zsiz
69069          - random number seed
0   8          - min and max original data
12           - number of simulated nodes to use
1             - assign data to nodes
1   3         - multiple grid search
0            - maximum data per octant
300 300.0 75.0 - maximum search radii
0.0  0.0  0.0 - angles for search ellipsoid
51   51   11   - size of covariance lookup table
0    3.912  1.0 - 0=SK,1=OK,2=LVM,3=EXDR,4=COLC
ydata.dat      - file with LVM, EXDR, or COLC
4              - column for secondary variable
1   0.0        - nst, nugget effect
1   1.0  0.0  0.0  0.0 - it,cc,ang1,ang2,ang3
      150.0 150.0 30.0 - a_hmax, a_hmin, a_vert

```

9.8 CASE DESCRIPTIONS FOR DIFFERENT RESERVOIR SIMULATORS

The history matching program is currently linked with three reservoir simulators: UTCHEM, ECLISPE, and CMG. Different reservoir simulators have different input and output formats. The following will discuss the main issues when different reservoir simulators are applied.

9.8.1 Procedure for UTCHEM

UTCHEM reservoir simulator requires two input files (the files names are “HEAD” and “INPUT”) for non-restart runs. The “HEAD” input file gives UTCHEM information regarding the name and size of the problem to be run.

To fulfill our purpose of matching permeability in X direction, which is located after the line “*---- PERMX” in INPUT file, several steps are needed as discussed.

1. Prepare a “HEAD” file and an “INPUT” file, except leaving permeability values empty after the line “*---- PERMX” in INPUT file. As for the maximum simulation time after the line “*---- TMAX” in INPUT, any value is acceptable since the program will change the value according to the user’s “infile.in” file.
2. Check the lines “*---- IMODE”, “*---- TMAX” and “*---- PERMX” in INPUT file. Make sure they start with an asterisk, four hyphens, and one space, followed by the keyword in Capital letters. The program will determine the location of the input variables based on these keywords and specifications.
3. Suppose the study name is CHAPT6 and there are total of 34 observation times, we then need to prepare a folder named “CHAPT6-true”. Inside this folder, the user needs to create files named “CHAPT601.dat” to “CHAPT634.dat”, each of which contains the observation data. The file

names correspond to the number of observation times. For example, “CHAPT601.dat” has the observation data for the first specified time. The data is in the same sequence with the well names in the input file. The remaining files contain the data for other times for the total of 34 times. As an example, “CHAPT601.dat” corresponding to Section 9.5 is given in Appendix C.

4. If initial water saturation and pressure are uniform or constant in each layer, the value(s) is given in the input file, described in Section 9.5. Otherwise, separate files are needed to assign values to initial ensemble members.
5. For a restart simulation, the output of the first simulation, named “CHAPT6.RESTAR”, is renamed to “INPUT2”. For the restart, the running mode IMODE flag in INPUT is changed from a value of 1 to 2. The program will then update the permeability values in INPUT file. Please note that the program will automatically perform these steps. UTCHEM recalculates properties such as transmissibilities at the beginning of each run even for a restart simulation.
6. Water saturations and pressures from output files of “*.SATP” and “*.PRES” will be stored. Well data are stored in well history files. For example, data for well 1 is in the file “*.HIST01”. Here, “*” represent the case name. Linux commands “sed”, “awk”, and pipeline are used to extract specific data from these files which can be input case specific.
7. Unlike commercial reservoir simulators, UTCHEM could not guarantee to generate outputs at exact user’s specified time. The user then needs to check the value for TINJ in the input line following “TINJ”. Slightly

increasing simulation time (TMAX) or reducing output frequency can solve this problem.

9.8.2 Procedure for ECLIPSE

The input format for ECLIPSE simulator is flexible. The permeability file is recommended to be an include file so that the program can easily read and update. Two special points are needed to pay attention to.

The information about reservoir pressure and saturation in each gridblock and well pressure and rate are all in the file “*.RSM”, where “*” represents the case name. It first gives well data and then reservoir properties. Take the reservoir pressure section as an example, the format is: first line is case name; second line starts from time and the type of reservoir information, i.e., reservoir pressure; third line is the units for time and pressure; fourth line is blank; fifth line is the gridblock, expressed by I, J, and K; sixth line is the pressure values corresponding to the gridblock number and the time. In addition, if the pressure values are complete, reservoir saturation information will continue and does not start in a new line. Such format gives the challenge to the automatic data collection. For different input cases, the user needs to check the file format and make sure to extract the correct data.

The other issue is the restart option. There are two methods of restarting an ECLIPSE 100 run: fast restart and flexible restart. Fast restart option directly reads reservoir properties and grid data from a save file. Hence, the transmissibilities are not recalculated. This option does not meet the history matching requirement, since permeability fields are sequentially changed. Flexible restart reads a complete data file. Though slower than fast restart, it recalculates the reservoir information. In the history matching program, flexible restart is required. To do this, the RESTART keyword is

needed to insert in the SOLUTION section while all equilibration or enumeration keywords and any analytic aquifer keywords must be deleted.

9.8.3 Procedure for CMG

GEM 2007.10 compositional simulator is used. Make sure to include CMG path to the PATH environmental variable. In each node, the Linux execution command is “gm200710.exe -f model.dat -log”, where “model.dat” is the GEM input file.

Since GEM is flexible for the keyword location and is not case sensitive, a separate file for permeability field is recommended in the input data. Reservoir pressure, saturation, and well data are stored in “*.out” file, where “*” denotes the input data file name. GEM output format is easy to handle. First, it gives the output time, then pressure or saturation title and data. The data is given by layer and ordered in the table style in each layer. As for well data, GEM separates injector and producer information. It is very helpful that the locations of time, pressure and saturation titles, and well names are fixed. The history matching program uses these fixed positions with keyword to precisely determine the desired data. If all the values in one layer are the same, a sentence “All values are” appears in the same line of the layer number. No table format appears for that layer. The history matching program can handle this scenario.

GEM restart option could not be used because it does not support the case that permeability field changes. All the cases have to be run from the beginning once the permeability values are changed. Fortunately, GEM runs very fast and this is not a big limitation.

9.9 SUMMARY

This chapter introduces the automatic history matching module developed through this research, including the structure, storage hierarchy, geostatistical

realizations, and special treatment for different reservoir simulators. The history matching module is embedded in the UT_IRSP platform and has the capability to process the input and output files of UTCHEM, ECLIPSE, and CMG reservoir simulators.

The input file for the automatic history matching module is quite general. Based on the Kalman filter methods, the module automatically implements the automatic history matching process: the folder generation, input preparation for each ensemble member, sequential or parallel computing, response data extraction from each ensemble member, the assimilation step and thereafter the updating of the permeability values in each ensemble member. In addition, the Tecplot format for each member and their average are created from initial to the final time.

Now the basic structure has already been developed, the important issue in future is how to apply to different cases using different reservoir simulators. For each application, it is strongly recommended to check the input and output formats for consistency with the program requirements.

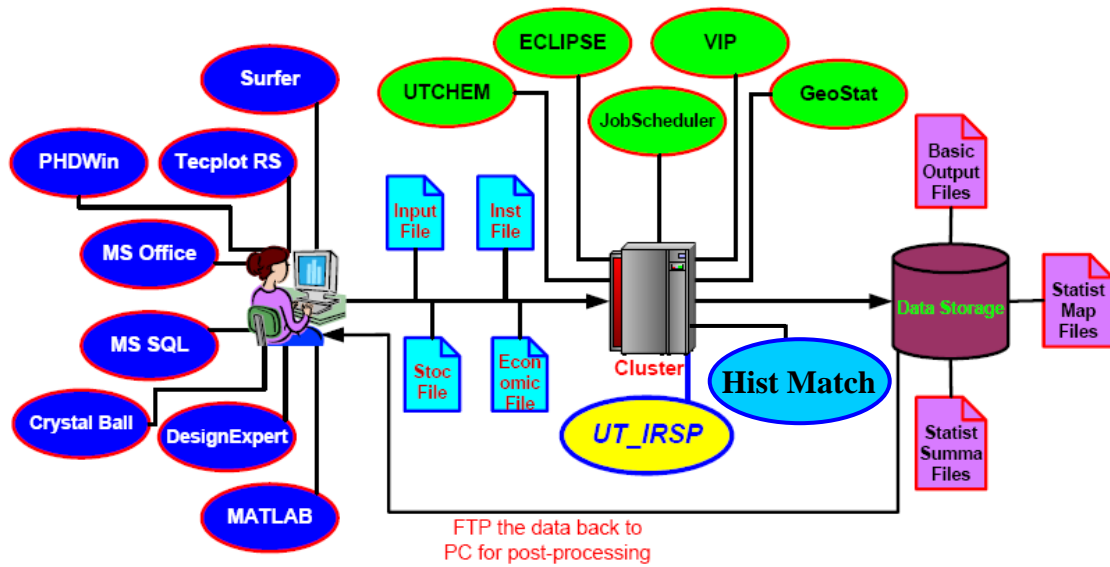


Figure 9-1: Environment of UT_IRSP with new “Hist Match” Module.

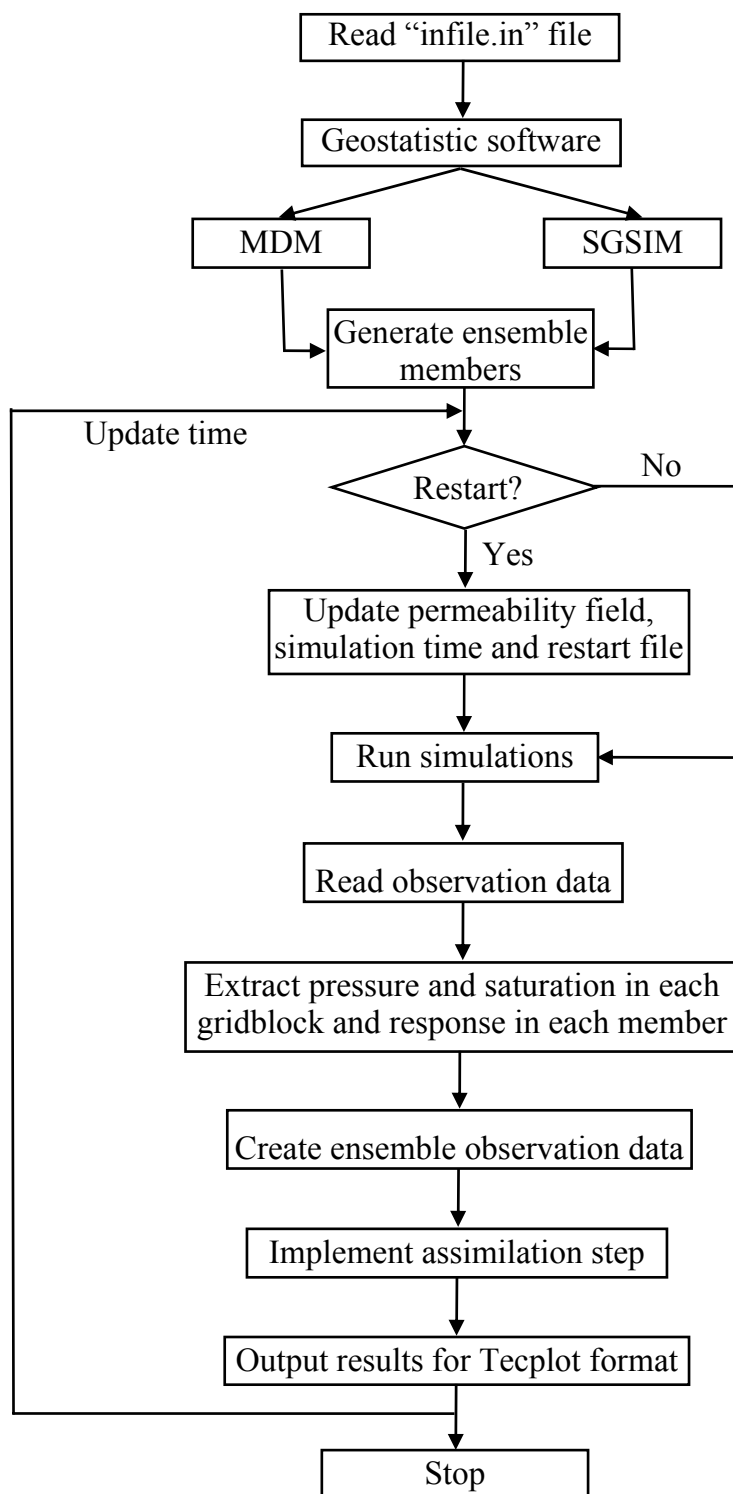


Figure 9-2: Automatic history matching module flow chart

Chapter 10: Summary, Conclusions, and Recommendations for Future Work

In this work, automatic history matching using the ensemble Kalman filter (EnKF) was studied. An efficient history matching module was developed and its application was demonstrated through several reservoir cases.

After a comprehensive literature review, stochastic history matching based on the EnKF was selected and investigated from the following aspects: Bayes' theorem, formula implementation with four calculation approaches, the initial sampling strategy improved by the singular value decomposition, a new weighted EnKF through weighing the importance of each ensemble member for the updated ensemble, and the impact of adding resampling procedure in the singular evolutive interpolated Kalman filter. The sampling interval and the uncertainty of geological information were also considered. Three different cases were studied: a two-dimensional waterflooding case, a seventeen-layer complex reservoir case study explored first by four-well natural depletion and later by seventeen-well waterflooding, and a twenty-layer case starting from the primary, secondary and then tertiary gas injection. A high-performance computing technique including the distributed and parallel simulation was implemented in the module and its application for history matching was demonstrated through a compositional reservoir case study.

10.1 CONCLUSIONS

The following conclusions are drawn from this study:

- A weighted EnKF, which weighs the contribution of each ensemble member, was proposed. Our case showed that both the weighted EnKF and the conventional EnKF give an excellent match of the field pressure and water

saturation at the end of each assimilation step. The weighted EnKF generates better production matches and does better recovery forecasting than those from the conventional EnKF. In addition, the weighted EnKF has faster convergence at the early time period during history matching. The superior performance of the weighted EnKF over the conventional EnKF was demonstrated through a seventeen-layer waterflooding reservoir case study.

- An efficient and user-friendly automatic history matching module based on the EnKF has been developed. In this module, three reservoir simulators, UTCHEM, ECLIPSE and CMG are integrated. Three computational modes, sequential, distributed, distributed and parallel modes are provided in the module.
- The EnKF for sequential data assimilation is an effective method in reservoir history matching. As a Monte Carlo approach, this method uses the ensemble members to update the model parameters and avoids the tedious gradient calculation in traditional history matching methods. The application to several cases which include natural depletion, waterflooding and gas injection has demonstrated the efficiency and accuracy of this method.
- In application, after stochastic sampling of multiple initial members through geological information, a closed loop consisting of forecasting and assimilation steps follows. The forecasting step is implemented by running all the independent reservoir simulation models. Four different schemes for the assimilation step are studied: direct inverse, standard EnKF assimilation, and square root algorithms with and without measurement perturbations. For a small number of observations, direct inverse is convenient. For the

consideration of algorithm generality, the square root approaches are recommended.

- The initial random sampling strategy is improved by the concept of the singular value decomposition: we sample a large number of realizations and then use the singular value decomposition to resample a small group of realizations with less linear dependence. In our example, the sampling strategy with the singular value decomposition has a positive impact on the matched results. By using 100 realizations, we achieved the same level of accuracy as the results from the conventional approach using 200 realizations.
- The sampling frequency is important since we do not want to miss any valuable information, while we also do not want to over-sample data. The study results illustrate that as long as we have a long matching period, the 20-day sampling interval still gives us reasonable matching results, which are almost identical to those from the high-frequency sampling case (such as one-day interval) at the late time, though some structure fails to recover at the early beginning.
- Well production history matching curves demonstrate that despite some reported divergence, both the weighted EnKF and the conventional EnKF methods keep improving reservoir models even at late matching time. The well production is matched reasonably, though the permeability field maps are somehow different from the references. The non-uniqueness may cause this phenomenon. Another reason is that the inaccurate reservoir geological structure information leads to the visual drift away of the permeability field. The impact of the inaccurate reservoir geological information was further studied. We found that better matching reservoir permeability structures are

obtained if the semivariogram is known initially, than in cases in which the semivariogram is roughly estimated.

- The root mean square is not sufficient to calibrate the matching performance. Production history, such as oil and water rates, can be deemed the most effective approach for the matching quantification.
- A variant of the EnKF, the singular evolutive interpolated Kalman filter, is applied to a history-matching problem. This method operates with an enhanced sampling strategy embedded into its resampling step, which appears to improve the filter stability and help the filter to deliver rapid convergence both in model and data domains. Both this method and the EnKF are effective tools for forecasting uncertainty quantification.
- The EnKF based history matching has the straightforward advantage of distributed computing. A compositional reservoir case study demonstrated that the efficiency of distributed and parallel computing during history matching is attractive and can significantly reduce the execution time.

10.2 RECOMMENDATIONS FOR FUTURE WORK

The following are recommendations for future work:

- Reservoir production optimization and history matching should be integrated through a loop to fulfill the concept of real-time reservoir management. The EnKF is recommended for such production optimization.
- It is common that the permeability field after log transformation is not Gaussian distributed. In particular, the channels occur in the geologic facies. It is desirable to investigate the performance of the EnKF in the highly non-Gaussian reservoir field.

- Liquid-phase saturation is assumed as a Gaussian distribution and is updated together with the reservoir permeability field. For a highly-heterogeneous reservoir, the possible density function of water or oil saturation might be multi-modal. Such impact needs evaluation. In addition, physical meaning limits phase saturation between 0 and 1. During the process of the EnKF history matching, the possible occurrence of undershooting and overshooting should be investigated though this phenomenon did not occur in our studies.
- High-performance computing is attractive and suitable for EnKF history matching. We studied two-processor parallel and eight-node distributed performance. Simulations using nodes with more processors as well as more distributed nodes should be carried out.
- Real reservoir application by EnKF history matching is needed. In particular, it is interesting to investigate how to update the ensemble members if the initial reservoir structure does not include fractures or faults but later production confirms their existence.
- Measurements from well testing, well logging and seismic data should be included as the conditional information in EnKF history matching.

Appendix A: Linux Programming

A.1 INTRODUCTION OF LINUX

Linux is a freely distributed implementation of a UNIX-like kernel, the low-level core of a computer operating system. Linux, written by Linus Torvalds, was first released over the Internet in 1991. Since then, Linux has exploded in popularity, maturing with each new version and bug fix. Linux has the following characteristics shared by typical UNIX programs and systems: simplicity, focus, reusable components, filters, and open file formats. There are a number of popular Linux distributions, including the popular Red Hat Linux, Debian, and SuSE.

Linux applications are represented by two special types of files: executables and scripts. Executable files are programs that can be run directly by the computer; they correspond to Windows `.exe` files. Scripts are collections of instructions for another program to follow; these correspond to Windows `.bat` or `.cmd` files.

The `shell` is a program started after you log on to Linux; it provides a command-line interface between you and the Linux kernel. Typed commands are interpreted and sent to the `kernel`, which in turn opens, closes, reads, or writes files.

There are several types of shells in the Linux world. The two major types are the Bourne shell and the C shell. The Bourne shell uses a command syntax like the original shell on early UNIX systems. The name of the Bourne shell on most Linux systems is `/bin/sh`. The C shell, somewhat like the programming language C, is named `/bin/csh` on most Linux systems. In addition, several variations of these shells are available. The two most commonly used are the Bourne Again Shell, or “Bash” (`/bin/bash`), and “Tcsh” (`/bin/tcsh`). `bash` is a form of the Bourne shell that includes many of the advanced features found in the C shell. Because `bash` supports a

superset of the Bourne shell syntax, shell scripts written in the standard Bourne shell should work with `bash`. If you prefer to use the C shell syntax, Linux supports `tcsh`, which is an expanded version of the original C shell. Only when you start to write shell scripts or use advanced features of a shell do the differences between shell types begin to matter.

A shell script is used to group a series of commands into a single file. Note that for our cluster Petros, the Bourne shell is used. For a C shell script, the first line is typically `#!/bin/csh -f`. The `-f` option tells the shell not to read the user's `.cshrc` file on startup, which improves both speed and portability. However, it is a common mistake to do the same thing for Bourne shell scripts by using `#!/bin/sh -f`. The Bourne shell's `-f` option is completely different from C shell's `-f` option. It disables file name generation. For example, the following script:

```
#!/bin/sh
rm mdm*
```

will remove the names of all the files in the current directory whose names begin with `"mdm"`, but the following script:

```
#!/bin/sh -f
rm mdm*
```

will unconditionally print that file `"mdm"` does not exist. Therefore, we use `#!/bin/sh` in all our shell scripts instead of `#!/bin/sh -f`, to generalize the capability of our scripts.

A.2 FREQUENTLY USED LINUX COMMANDS

There are hundreds of commands in Linux. We select and list most commonly used commands during our work in Table A-1.

Table A-1 Commonly Used Linux Commands

Command	Meaning
awk	Scan and extract information from a line or string, based on the criteria.
cat	Display the contents of a file.
cd	Change directory.
chmod	Change access permissions.
clear	Clear terminal screen.
comm	Compare two sorted files line by line.
cp	Copy one or more files to another location.
df	Display free disk space.
diff	Display the differences between two files.
dir	Briefly list directory contents.
echo	Display message on screen
env	Setting information of environment variables.
exit	Exit the shell.
find	Search for files that meet a desired criterion.
grep	Search file(s) for lines that match a given pattern.
history	Command history.
kill	Stop a process from running.
ln	Make links between files.
locate	Find files.
ls	List information about file(s).
make	Recompile a group of programs.
man	Help manual.

mkdir	Create new folder.
more	Display output one screen at a time.
mv	Move or rename files or directories.
passwd	Modify a user password.
quota	Display disk usage and limits.
rm	Remove files.
rmdir	Remove folder(s).
sed	Stream editor to edit one or more files without user interaction.
set	Manipulate shell variables and functions.
ssh	Secure shell client (remote login program).
tar	Copy or restore files from an archive medium.
time	Measure program running time.
top	List processes running on the system.
vi	Screen-oriented text editor.
whereis	Report all known instances of a command.
who	Print all usernames currently logged in.

Appendix B: Portable Batch System

B.1 INTRODUCTION OF PORTABLE BATCH SYSTEM

The Portable Batch System, PBS, is a workload management system for Linux clusters. It supplies command to submit, monitor, and delete jobs. Workload management systems have three primary roles:

Queuing: the collecting together of work or tasks to be run on a computer. Users submit tasks or “jobs” to the resource management system where are then queued up until the system is ready to run them.

Scheduling: the process of selecting which jobs to run, when, and where, according to a predetermined policy. PBS tries to balance competing needs and goals on the system(s) to maximize efficient use of resources (both computer time and people time).

Monitoring: the act of tracking and reserving system resources and enforcing usage policy. This covers user- and system-level monitoring as well as monitoring of the scheduling policies to see how well they are meeting the stated goals.

Consequently, a PBS has the following three components:

Job Server: also called `pbs_server` provides the basic batch services such as receiving/creating a batch job, modifying the job, protecting the job against system crashes, and running the job.

Job Executor: a daemon (`pbs_mom`) that actually places the job into execution when it receives a copy of the job from the Job Server. This daemon is informally called MOM (a reverse-engineered acronym that stands for Machine Oriented Mini-server) as it is the mother of all executing jobs. MOM creates a new session as close to a user login session as is possible and returns the job’s output to the user.

Job Scheduler: a daemon that contains the site's policy controlling which job is run and where and when it is run. PBS allows each site to create its own Scheduler. The Scheduler communicates with various MOMs to learn about the state of a system's resources and with the Server to learn about the availability of jobs to execute.

B.2 TERM DEFINITIONS OF PORTABLE BATCH SYSTEM

The following defines important terms and concepts of PBS.

Node: A node to PBS is a computer system with a single operating system image, a unified virtual memory space, one or more CPUs and one or more IP addresses. Frequently, the term execution host is used for node. A computer such as the SGI Origin 3000, which contains multiple CPUs running under a single operating system, is one node. Systems like Linux clusters, which contain separate computational units each with their own operating system, are collections of nodes. Nodes can be defined as either cluster nodes or timeshared nodes.

Virtual Processors: A node may be declared to consist of one or more virtual processors. The term virtual is used because the number of virtual processors declared does not have to equal the number of real processors (CPUs) on the physical node. The default number of virtual processors on a node is the number of currently functioning physical processors; the PBS manager can change the number of virtual processors as required by local policy.

Cluster Node: A node whose purpose is geared toward running parallel jobs is called a cluster node. If a cluster node has more than one virtual processor, the virtual processors may be assigned to different jobs (job shared) or used to satisfy the requirements of a single job (exclusive). This ability to temporally allocate the entire node to the exclusive use of a single job is important for some multimode parallel applications. Note that PBS enforces a one-to-one allocation scheme of cluster node

virtual processors ensuring that the virtual processors are not over-allocated or over-subscribed between multiple jobs.

Cluster This is any collection of nodes controlled by a single instance of PBS (i.e., by one PBS server). Our Dell cluster Petros has eight nodes with dual shared memory processors.

Queue: A queue is a named container for jobs within a Server. There are two types of queues defined by PBS: routing and execution. A routing queue is a queue used to move jobs to other queues including those that exist on different PBS servers. A job must reside in an execution queue to be eligible to run and will remain in an execution queue during the time it is running. In spite of the name, jobs in a queue need not be processed in queue order (first-come first-served).

B.3 ENVIRONMENT VARIABLES OF PORTABLE BATCH SYSTEM

In order to make sure that the system environment interacts seamlessly with PBS, there are mainly three items that need to be checked: (1) User must have access to the resources/hosts that the site has configured for PBS; (2) User must have a valid account on the execution hosts; (3) User must be able to transfer files between hosts. In many cases, the system administrator will have already set up the user environment to work with PBS. For Petros, we can use directly PBS without any user environment setting.

While running, PBS could provide useful information such as job name, host node, and directory, through a number of environment variables. Such environment variables are taken from the user's environment or created by PBS. All PBS-provided variable names start with the characters "PBS_". Some are then followed by a capital O ("PBS_O") indicating that the variable is from the job's originating environment. Table A.1 gives a full listing of environment variables and their meanings. PBS resources are installed in the folder of `/opt/torque` in our cluster Petros.

Table B-1 PBS Environment Variables

Variable	Meaning
ENVIRONMENT	Indicates if job is a batch job, or a PBS interactive job.
NCPUS	Number of threads (or cpus per process (cpp)) on the node.
PBS_JOBCOOKIE	Unique identifier for inter-MOM job-based communication.
PBS_JOBID	The job identifier assigned to the job by the batch system.
PBS_JOBNAME	The job name supplied by the user.
PBS_MOMPORT	Port number on which this job's MOMs will communicate.
PBS_NODEFILE	The filename containing a list of nodes assigned to the job.
PBS_NODENUM	Logical node number of this node allocated to the job.
PBS_O_HOME	Value of HOME from submission environment.
PBS_O_HOST	The host name on which the qsub command was executed.
PBS_O_LANG	Value of LANG from submission environment.
PBS_O_LOGNAME	Value of LOGNAME from submission environment.
PBS_O_MAIL	Value of MAIL from Submission environment.
PBS_O_PATH	Value of PATH from submission environment.
PBS_O_QUEUE	The original queue name to which the job was submitted.
PBS_O_SHELL	Value of SHELL from submission environment.
PBS_O_SYSTEM	The operating system name where qsub was executed.
PBS_O_WORKDIR	The absolute path of directory where qsub was executed.
PBS_QUEUE	The name of the queue from which the job is executed.
PBS_TASKNUM	The task (process) number for the job on this node.
TMPDIR	The job-specific temporary directory for this job.

B.4 SCRIPT SAMPLE FOR PORTABLE BATCH SYSTEM

Below are the steps needed to run a history matching framework with distributed style:

1. Create a job script containing the following PBS options:
 - request the resources that will be needed (i.e. number of processors, wall-clock time, etc.) and
 - use commands to prepare for execution of the executable (i.e. enter the working directory, etc.).
2. Submit the job script file to PBS.
3. Monitor the job.

We take a simple job as an example: run MDM stochastic field generation software by using PBS. Under the folder of “/home/liang/PBS-test”, we first prepare two input files needed by MDM software: “head.dat2” and “input-1.dat1”. The content of “head.dat2” is:

```
12
1 1 1 80.
1 1 2 80.
1 1 3 80.
1 1 4 80.
1 1 5 80.
1 1 6 80.
11 11 1 80.
11 11 2 80.
11 11 3 80.
11 11 4 80.
11 11 5 80.
11 11 6 80.
```

The “input-1.dat1” file is:

```
mdm-1.dat1          Fri Jan 12 21:36:32 2007
NX, NUMBER OF X DIVISIONS          ?
30
```

NY, NUMBER OF Y DIVISIONS	?
30	
NZ, NUMBER OF Z DIVISIONS	?
6	
NV, NUMBER OF VARIOGRAMS (IN THE NESTED MODEL)	?
1.0000	
NR, NUMBER OF REALIZATIONS, <1 AS 1 EXCEPT O-FORM.	?
1.0000	
NS, NUMBER OF FOR STARTING RANDOM NUMBER (SEED)	?
1804289383	
MO, OUTPUT (INPUT) OPTIONS, SUM OF ALL OPTIONS	?
13322	
DX, GRID SIZE IN X DIRECTION, CONSISTENT UNIT	?
60	
DY, GRID SIZE IN Y DIRECTION, SAME UNIT AS DX	?
60	
DZ, GRID SIZE IN Z DIRECTION, SAME UNIT AS DX	?
5	
#1: P OF P-NORMAL FOR THIS TERM OF VARIOGRAM, 0=LOG-N.	?
0.	
#1: MEAN VALUE FOR THIS TERM, CONSISTENT UNIT AS S.D.	?
40	
#1: CORRELATION LENGTH, X MAJOR AXIS, SAME UNIT AS DX	?
130	
#1: CO.MODEL, -1=EXP, -2=DEXP, -3=SPH, -5=INPUTACF, +=POWER	?
-1	
#1: RATIO OF CORRELATION LENGTHS IN MAJOR AXES, LX/LY	?
1	
#1: RATIO OF CORRELATION LENGTHS IN MAJOR AXES, LX/LZ	?
10	

To run the job in PBS, we have a PBS script `pbs.job` and a MDM job pre- and post-processing script `test.job`, respectively. For running PBS, we only need to type “`qsub pbs.job`” in the folder of “PBS-test”. The “`pbs.job`” file is as follows:

```
#!/bin/sh
#Note: PBS directives begin with "#PBS".
#### Output files
#PBS -e test.err
#PBS -o test.log

#Select the number of nodes to use:
#PBS -l nodes=1
```

```
#If the job incomplete by specified time, it is terminated.
#PBS -l walltime=10:00
```

```
#Output some useful job information:
```

```
echo
echo -----
echo 'Job is running on node': cat $PBS_NODEFILE
echo -----
echo PBS: qsub is running on $PBS_O_HOST
echo PBS: originating queue is $PBS_O_QUEUE
echo PBS: executing queue is $PBS_QUEUE
echo PBS: working directory is $PBS_O_WORKDIR
echo PBS: execution mode is $PBS_ENVIRONMENT
echo PBS: job identifier is $PBS_JOBID
echo PBS: job name is $PBS_JOBNAME
echo PBS: current home directory is $PBS_O_HOME
echo PBS: PATH = $PBS_O_PATH
echo PBS: Directory is `pwd`
echo PBS: Running on host `hostname`
echo -----
echo
```

```
#Execute the run:
```

```
bash /home/liang/PBS-test/test.job >& pbs.log
```

The following is “test.job” file:

```
#!/bin/bash
cd /home/liang/PBS-test
mkdir liang
cd liang
ln -s ../input-1.dat1 INPUT
ln -s ../head.dat2 INDAT
time mdm.exe
mv ECHO liang.eco
mv OUTXYZ liang.dat
rm OUTPT
rm POUTPT
rm VOUTPT
rm fort.*
chmod 755 *.*
cd ..
chmod 755 liang
```

B.5 USEFUL COMMANDS FOR PORTABLE BATCH SYSTEM

qsub: Once a PBS job script is created, it is submitted to PBS via the `qsub` command. In its simplest form, `qsub` takes a single parameter, the name of the script file that you wish to submit.

qstat: This command allows you to view the contents of the PBS queue.

showq: Displays information about active, eligible, blocked, and/or recently completed jobs.

qdel: The `qdel` command takes a single argument, a job number. You can use `qdel` to abort execution of your job.

qalter: The `qalter` command is helpful for altering the parameters of a job **after** it's submitted. `qalter` takes two arguments: the PBS directive that you wish to change, and the job number that you want to change. For example, if you forgot to set the walltime that your job requires, you can change it after it has been submitted:

pbsnodes: The `pbsnodes` command, while a useful PBS administration command, can also be informative to the PBS user. `pbsnodes -a` will list all PBS nodes, their attributes, and job status. This is a useful way to get a list of valid machine properties for use in a `#PBS -l` directive.

qpeek: This command can show you any output your running PBS job is generating without having to wait for completion and for PBS to deliver the standard output and standard error. The only argument to the program is the PBS jobid.

showscript: Will return the contents of the PBS script that you have submitted. The only argument is the job's PBS jobid.

checkjob: Gives detailed information about your job. This is very useful if your job is remaining in the queued state, and you would like to see why PBS has not executed it.

Appendix C: Example of Observation Data File

As an example, the following is the file “CHAPT601.dat” discussed in Chapter 9. Note that the sequence of observation data should be consistent with the data types and wells given in the input file “infile.in”.

```
98.67391
9.18E-05
98.67396
4.22E-05
481.3386
482.7922
483.8275
484.6595
485.4850
486.5020
487.5168
488.3432
489.1720
489.7944
490.2107
490.6273
491.0442
491.4644
491.8933
492.5287
493.9996
480.8779
482.3409
483.3974
484.2502
485.1318
486.2222
487.2974
488.1570
489.0262
489.6819
490.0985
490.5135
490.9252
491.3373
491.7532
492.3753
493.8315
```

Bibliography

- Abacioglu, Y., Oliver, D.S., and Reynolds, A.C.: "Efficient Reservoir History Matching Using Subspace Vectors," *Computational Geosciences*, 5(2), 151-172, 2001.
- Al-Matar, B.S., Pathak, A., Mandal, D., Killough, J., Fleming, G., Engle, C., Brock, N., and Varma, S.: "Next-Generation Modeling of a Middle Eastern Multireservoir Complex," paper SPE 106069 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 26-28 February, 2007.
- Alpak, F.O.: *Algorithms for Numerical Modeling and Inversion of Multi-Phase Fluid-Flow and Electromagnetic Measurements*, PhD Dissertation, The University of Texas at Austin, May 2005.
- April, J., Glover, F., Kelly, J., Laguna, M., Erdogen, M., Mudford, B., and Stegemeier, D.: "Advanced Optimization Methodology in the Oil and Gas Industry: The Theory of Scatter Search Techniques with Simple Examples," paper SPE 82009 presented at the SPE Hydrocarbon Economics and Evaluation Symposium, Dallas, TX, 5-8 April, 2003a.
- April, J., Glover, F., Kelly, J., Laguna, M.: "A New Optimization Methodology for Portfolio Management," paper SPE 84332 presented at the SPE Annual Technical Conference and Exhibition in Denver, CO, 5-8 October, 2003b.
- Arroyo-Negrete, E.R.: "Continuous Reservoir Model Updating Using Streamline Assisted Ensemble Kalman Filter," paper SPE 106518 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, 24-27 September, 2006.
- Beckner, B.L., Hutfilz, J.M., Ray, M.B., and Tomich, J.F.: "EM^{power}: New Reservoir Simulation System," paper SPE 68116 presented at the 2001 SPE Middle East Oil Show, Bahrain, 17-20 March 2001.
- Bi, Z., Oliver, D.S. and Reynolds, A.C.: "Conditioning 3D Stochastic Channels to Pressure Data," *Soc. Petrol. Eng. J.*, 5(4), 474-484, 2000.
- Bissell, R.C., Dubrule, O., Lamy, P., Swaby, P., and Lepine, O.: "Combing Geostatistical Modelling with Gradient Information for History Matching: The Pilot Point Method," paper SPE 38730 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, 5-8 October 1989.

Bissell, R.C., Sharma, Y., and Killough, J.E.: "History Matching Using the Method of Gradients: Two Case Studies," paper SPE 28590 presented at the SPE Annual Technical Conference and Exhibition, New Orleans, LA, 25-28 September, 1994.

BLAS website: <http://www.netlib.org/blas/>

Bloemen, H.H.J., Belfroid, S.P.C., Sturm, W.L., and Verhelst, J.P.C.M.G.: "Soft Sensing for Gas Lift Wells," *Soc. Petrol. Eng. J.*, 11(4), 454-463, 2006.

Brown, C.L.: *Simulation of Surfactant Enhanced Remediation of Aquifers Contaminated with Dense Non-Aqueous Phase Liquids*, MS Thesis, The University of Texas at Austin, 1993.

Brouwer, D.R., Jansen, J.D., van der Starre, S., Van Kruijsdijk, C.P.J.W., and Berentsen, C.W.J.: "Recovery Increase through Water Flooding with Smart Well Technology," paper SPE 68979 presented at the SPE European Formation Damage Conference, The Hague, The Netherlands, 21-22 May, 2001.

Brouwer, D.R. and Jansen, J.D.: "Dynamic Optimization of Water Flooding with Smart Wells Using Optimal Control Theory," paper SPE 78278 presented at the SPE 13th European Petroleum Conference, Aberdeen, Scotland, U.K., 29-31 October, 2002.

Brouwer, D.R., Naevdal, G., Jansen, J.D., Vefring, E.H., and Van Kruijsdijk, C.P.J.W.: "Improved Reservoir Management through Optimal Control and Continuous Model Updating," paper SPE 90149 presented at the SPE Annual Technical Conference and Exhibition, Houston, TX, 26-29 September, 2004.

Burgers, G., Leerwen, P.J.V., and Evensen, G.: "Analysis Scheme in the Ensemble Kalman Filter," *Monthly Weather Review*, 126, 1719-1724, 1998.

Caers, J.: "Efficient Gradual Deformation Using a Streamline-Based Proxy Method," *Journal of Petroleum Science and Engineering*, 39, 57-83, 2003.

Carter, R.D., Kemp, L.F., Pierce, A.C., and Williams, D.L.: "Performance Matching with Constraints," *Soc. Petrol. Eng. J.*, 14(4), 187-196, 1974.

Chavent, G.M., Dupuy, M., and Lemonnier, P.: "History Matching by Use of Optimal Control Theory," *Soc. Petrol. Eng. J.*, 15(1), 74-86, 1975.

Chen, W.H., Gavalas, G.R., Seinfeld, J.H., and Wasserman, M.L.: "A New Algorithm for Automatic History Matching," *Soc. Petrol. Eng. J.*, 14(12), 593-608, 1974.

Chien M.C.H and Northrup, E.J.: "Vectorization and Parallel processing of Local Refinement and Adaptive Schemes in a General Purpose Reservoir Simulator",

- paper SPE 25258, presented at the 12th SPE Symposium on Reservoir Simulation, New Orleans, LA, February 1993.
- Chien, M.C.H., Tchelepi, H.A., Yardumian, H.E, Chen, W.H.: “A Scalable Parallel Multi-Purpose Reservoir Simulator,” paper SPE 37976 presented at the SPE Symposium on Reservoir Simulation, Dallas, TX, June 8-11, 1997.
- Christie, M., MacBeth, C., and Subbey S.: “Multiple History-Matched Models for Teal South,” *The Leading Edge*, 21(3), 286-289, March 2002.
- Christie, M., Demyanov, V., and Erbas D.: “Uncertainty Quantification for Porous Media Flows,” *Journal of Computational Physics*, 217, 143-158, 2006.
- Cao, H., Tchelepi, H.A., Wallis, J., and Yardumian, H.: “Parallel Scalable Unstructured CPR-Type Linear Solver for Reservoir Simulation,” paper SPE 96809 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 31 January-2 February, 2005.
- Collins, D.A., Grabenstetter, J.E., and Sammon, P.H.: “A Shared-Memory Parallel Black-Oil Simulator with a Parallel ILU Linear Solver,” paper SPE 79713 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 3-5 February, 2003.
- Corser, G.P., Harmse, J.E., Corse, B.A., Weiss, M.W., and Whitflow, G.L.: “Field Test Results for a Real-Time Intelligent Drilling Monitor,” paper SPE 59227 presented at the IADC/SPE Drilling Conference, New Orleans, LA, 23-25 February, 2000.
- Cullick, A.S., Heath, D., Narayanan, K., April, J., and Kelly, J.: “Optimizing Multiple-Field Scheduling and Production Strategy with Reduced Risk,” paper SPE 84239 presented at the SPE Annual Technical Conference and Exhibition, Denver, CO, 5-8 October, 2003.
- Cullick, A.S., Johnson, D., and Shi, G.: “Improved and More Rapid History Matching with a Nonlinear Proxy and Global Optimization,” paper SPE 101933 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, 24-27 September, 2006.
- DeBaun, D., Byer, T. , Childs, P., Chen, J., Saaf, F., Wells, M., Liu, J., Cao, H., Pianelo, L., Tilakraj, V., Crumpton, P., Walsh, D., Yardumian, H., Zorzynski, R., Lim, K.-T., Schrader, M., Zapata, V., Nolen, J., Tchelepi, H.: “An Extensible Architecture for Next Generation Scalable Parallel Reservoir Simulation,” paper SPE 93274 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 31 January-2 February, 2005.
- Decisioneering, Inc.: *Crystal Ball User Manual*, 2004
- Decisioneering, Inc.: *OptQuest® for Crystal Ball 2000*, 2001.

- Delshad, M., Pope, G.A., and Sepehrnoori, K.: "A Compositional Simulator for Modeling Surfactant Enhanced Aquifer Remediation, 1 Formulation," *J. Contamin. Hydrol.*, 23(4), 303-327, 1996.
- Deutsch, C.V. and Journel, A.G.: *GSLIB, Geostatistical Software Library and User's Guide*, Second Edition, Oxford University Press, New York, 1998.
- Dogru, A.H., Li, K.G., Sunaidi, H.A., Habiballah, W.A., Fung, L., Al-Zamil, N., and Shin, D.: "A Massively Parallel Reservoir Simulator for Large Scale Reservoir Simulation," paper SPE 51886 presented at the 1999 SPE Reservoir Simulation Symposium, Houston, TX 14-17 February, 1999.
- Dong, Y., Gu, Y. and Oliver, D.S.: "Sequential Assimilation of 4D Seismic Data for Reservoir Description Using the Ensemble Kalman Filter," *Journal of Petroleum Science and Engineering*, 53(1-2), 83-99, August 2006.
- Doucet A., de Freitas N., and Gordon N. (Eds.): *Sequential Monte Carlo Methods in Practice*, Springer, New York, 2001.
- EISPACK website: <http://www.netlib.org/eispack/>
- Eisenmann, P., Gounot, M.-T., Juchereau, B., and Whittaker, S.J.: "Improved Rxo Measurements through Semi-Active Focusing," paper SPE 28437 presented at the SPE Annual Technical Conference and Exhibition, New Orleans, LA, 25-28 September, 1994.
- Erbas, D. and Christie, M.A.: "Effect of Sampling Strategies on Prediction Uncertainty Estimation," paper SPE 106229 presented at the 2007 SPE Reservoir Simulation Symposium, Houston, TX, 26-28 February, 2007.
- Evensen, G.: "Using the Extended Kalman Filter with a Multi-Layer Quasi-Geostrophic Ocean Model," *J. Geophys. Res.*, 97(C11), 17905-17924, 1992.
- Evensen, G.: "Sequential Data Assimilation with a Non-Linear Quasi-Geostrophic Model Using Monte-Carlo Methods to Forecast Error Statistics," *J. Geophys. Res.*, 99(C5), 10143-10162, 1994.
- Evensen, G.: "The Ensemble Kalman Filter: Theoretical Formulation and Practical Implementation," *Ocean Dynamics*, 53(4), 343-367, 2003.
- Evensen, G.: "Sampling Strategies and Square Root Analysis Schemes for the EnKF," *Ocean Dynamics*, 54(6), 539-560, 2004.
- Evensen, G.: *Data Assimilation: The Ensemble Kalman Filter*, Springer, 2006.

Evensen, G., Hove, J., Meisingset, H.C., Reiso, E., Seim, K.S., Espelid, Ø.: “Using the EnKF for Assisted History Matching of a North Sea Reservoir Model,” paper SPE 106184 presented at the 2007 SPE Reservoir Simulation Symposium, Houston, TX, 26-28 February, 2007.

FFTW website: <http://www.fftw.org/>

Fjerstad, P.A., Sikandar, A.S., Cao, H., Liu, J., and Da Sie, W.: “Next Generation Parallel Computing for Large-Scale Reservoir Simulation,” paper SPE 97358 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 31 January-2 February, 2005.

Fletcher, R.: *Practical Methods of Optimization*, 2nd Edition, John Wiley & Sons, New York, 1987.

Forsythe, G.E., Malcolm, M.A., and Moler, C.B.: *Computer Methods for Mathematical Computations*, Prentice-Hall Inc., Englewood Cliffs, NJ, 1977.

Gai, H.: “Downhole Flow Control Optimization in the World’s 1st Extended Reach Multilateral Well at Wytch Farm,” paper SPE 67728 presented at the IADC/SPE Drilling Conference, Amsterdam, The Netherlands, 27 February-1 March, 2001.

Gao, G. and Reynolds, A.C.: “An Improved Implementation of the LBFGS Algorithm for Automatic History Matching,” *Soc. Petrol. Eng. J.*, 11(1), 5-17, 2006.

Gao, G., Zafari, M. and Reynolds, A.C.: “Quantifying Uncertainty for the PUNQ-S3 Problem in a Bayesian Setting with RML and EnKF,” paper SPE 93324 presented at the SPE 2005 SPE Reservoir Simulation Symposium, Houston, TX, 31 January-2 February, 2005.

Gavalas, G.R., Shah, P.C., and Seinfeld, J.H.: “Reservoir History Matching by Bayesian Estimation,” *Soc. Petrol. Eng. J.*, 16(6), 337-350, 1976.

Glover, F.: “Heuristics for Integer Programming Using Surrogate Constraints,” *Decision Sciences*, 8, 156-166, 1977.

Glover, F.: “Genetic Algorithms and Scatter Search: Unsuspected Potentials,” *Statistics and Computing*, 4, 131-140, 1994.

Glover, F. and Laguna, M.: *Tabu Search*, Kluwer Academic Publishers, Boston, 1997.

Golden Software, Inc.: *Surfer® 7 User’s Guide*, 1999.

Golub, G.H. and Van Loan, C.F.: *Matrix Computations*, 2nd edition, The Johns Hopkins University Press, Baltimore and London, 1989.

- Gómez-Hernández, J.J., Sahuquillo, A., and Capilla J.E.: “Stochastic Simulation of Transmissivity Fields Conditional to Both Transmissivity and Piezometric Data— I. Theory,” *Journal of Hydrology*, 203, 162-174, 1997.
- Gomez, S., Gosselin, O., and Barker, J.W.: “Gradient-Based History Matching With a Global Optimization Method,” *Soc. Petrol. Eng. J.*, 6(2), 200-208, 2001.
- Gu, Y. and Oliver, D.S.: “History Matching of the PUNQ-S3 Reservoir Model Using the Ensemble Kalman Filter,” *Soc. Petrol. Eng. J.*, 10(2), 217-224, 2005.
- Gu, Y. and Oliver, D.S.: “The Ensemble Kalman Filter for Continuous Updating of Reservoir Simulation Models,” *Journal of Energy Resources Technology*, 128(1), 79-87, 2006.
- Guyaguler, B.: *Optimization of Well Placement and Assessment of Uncertainty*, Ph.D. dissertation, Stanford University, 2002.
- Guyanguler, B. and Horne, R.: “Uncertainty Assessment of Well placement Optimization,” paper SPE 71625 presented at the SPE Annual Technical Conference and Exhibition, New Orleans, LA, September 30-October 3, 2001.
- Han, C., Delshad, M., Sepehrnoori, K., and Pope, G.A.: “A Fully Implicit, Parallel, Compositional Chemical Flooding Simulator,” paper SPE 97217 presented at the SPE Annual Technical Conference and Exhibition, Dallas, TX, 9-12 October 2005.
- Haugen, V., Natvik, L.-J., Evensen, G., Berg, A., Flornes, K., and Nævdal, G.: “History Matching Using the Ensemble Kalman Filter on a North Sea Field Case,” paper SPE 102430 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, 24-27 September, 2006.
- He, N., Reynolds, A.C., and Oliver, D.S.: “Three-Dimensional Reservoir Description from Multiwell Pressure Data and Prior Information,” *Soc. Petrol. Eng. J.*, 2(3), 312-327, September 1997.
- He, Z., Yoon, S., and Datta-Gupta, A.: “Streamline-Based Production Data Integration With Gravity and Changing Field Conditions,” *Soc. Petrol. Eng. J.*, 7(4), 423-436, December 2002.
- Hoteit, I., Pham, D.T., and Blum, J.: “A Simplified Reduced Order Kalman Filtering and Application to Altimetric Data Assimilation in Tropical Pacific,” *J. Mar. Syst.*, 36, 101-127, 2002.
- Holland, J.H.: *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, 1975

- Hu, L.: "Gradual Deformation and Iterative Calibration of Gaussian-Related Stochastic Models," *Mathematical Geology*, 32(1) 87-108, 2000.
- Hu, L. and Jenni, S.: "History Matching of Object-Based Stochastic Reservoir Models," *Soc. Petrol. Eng. J.*, 10(3), 312-323, September 2005.
- Jackson, D.D.: "The Use of a Priori Data to Resolve Non-Uniqueness in Linear Inversion," *Geophys. J. R. Astr. Soc.*, 57, 137-157, 1979.
- Jacquard, P., and Jain, C.: "Permeability Distribution from Field Pressure Data," *Soc. Petrol. Eng. J.*, 5(4), 281-294, 1965.
- Jahns, H.O.: "A Rapid Method for Obtaining a Two-Dimensional Reservoir Description from Well Pressure Response Data," *Soc. Petrol. Eng. J.*, 6(12), 315-327, 1966.
- Jiang, Yuanlin: "A Unified Framework for General-Purpose Simulation of Unstructured Reservoirs, Advanced Wells and Surface Facilities," to be presented at The Universities Forum on Reservoir Description and Simulation, Scarborough, England, 2-6 September, 2007.
- Jutila, H.A. and Goodwin, N.H.: "Schedule Optimization to Complement Assisted History Matching and Prediction under Uncertainty," paper SPE 100253 presented at the SPE Europec/EAGE Annual Conference and Exhibition, Vienna, Austria, 12-15 June, 2006.
- Kaarstad T., Frauen, J., Bjorstad, P., and Espedal, M.: "A Massively Parallel Reservoir Simulator," paper SPE 29139 presented at the 13th Symposium on Reservoir Simulation, San Antonio, TX, 1995.
- Kahaner, D., Moler, C., and Nash, S.: *Numerical Methods and Software*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1989.
- Kalman, R.: "A New Approach to Linear Filtering and Prediction Problems," *Trans. ASME, J. Basic Eng.*, 82D, 35-45, 1960.
- Khargoria, A., Zhang, F., Li, R., and Jalali, Y.: "Application of Distributed Electrical Measurements and Inflow Control in Horizontal Wells Under Bottom-Water Drive," paper SPE 78275 presented at the SPE 13th European Petroleum Conference, Aberdeen, Scotland, U.K., 29-31 October, 2002.
- Killough, J.E. and Bhogeswara, R.: "Simulation of Compositional Reservoir Phenomena on a Distributed Memory Parallel Computer," *Journal of Petroleum Technology*, November 1991.

- Kilough J.E., Camilleri, D., and Harlow B.: "A Parallel Simulator on Local Grid Refinement," paper SPE 37978 presented at the SPE Symposium on Reservoir Simulation, Dallas, TX, June 8-11, 1997.
- Kitanidis, P.K.: "Quasi-Linear Geostatistical Theory for Inversing," *Water Resour. Res.*, 31(10), 2411-2419, 1995.
- Kromah, M.J., Liou, J., and MacDonald, D.G.: "Step Change in Reservoir Simulation Breathes Life into a Mature Oil Field," paper SPE 94940 presented at the SPE Latin American and Caribbean Petroleum Engineering Conference, Rio de Janeiro, Brazil, 20-23 June, 2005.
- Lach, J., McMillen, K., Archer, R., Holland, J., DePauw, R., and Ludvigsen, B.E.: "Integration of Geologic and Dynamic Models for History Matching, Medusa Field," paper SPE 95930 presented at the SPE Annual Technical Conference and Exhibition, Dallas, TX, 9-12 October, 2005.
- Lacroix, S., Vassilevski, M., Wheeler, M.F., and Wheeler, J.: "Iterative Solvers of the Implicit Parallel Accurate Reservoir Simulator (IPARS)," *Numerical Linear Algebra with Applications*, 4, 537-549, 2001.
- Landa, J.L., Kalia, R.K., Nakano, A., Nomura, K., and Vashishta, P.: "History Matching and Associated Forecast Uncertainty Analysis – Practical Approaches Using Cluster Computing," paper SPE 10751 presented at the International Petroleum Technology Conference in Doha, Qatar, 21-23 November, 2005.
- LAPACK website: <http://www.netlib.org/lapack/>
- Lawson, W.G. and Hansen, J.A.: "Implications of Stochastic and Deterministic Filters as Ensemble-Based Data Assimilation Methods in Varying Regimes of Error Growth," *Monthly Weather Review*, 132(8), 1966-1981, August 2004.
- Leeuwenburgh, O., Evensen, G., and Bertino, L.: "The Impact of Ensemble Filter Definition on the Assimilation of Temperature Profiles in the Tropical Pacific," *Quarterly Journal-Royal Meteorological Society*, 131(613), 3291-3300, October 2005.
- Leitao, H.C. and Schiozer, D.J.: "A New Automated History Matching Algorithm Improved by Parallel Computing," paper SPE 53977 presented at the 1999 SPE Latin American and Caribbean Petroleum Engineering Conference, Caracas, Venezuela, 21-2 April, 1999.
- Li, K.G., Dogru, A.H., McDonald, A.E., Merchant, A.R., Al-Mulhem, A.A., Al-Ruwaili, S.B., Sobh, N.A., Al-Sunaidi, H.A.: "Improving the Performance of Mars Simulator on Cray-2 Supercomputer," paper SPE 29856 presented at the Middle East Oil Show in Bahrain, 11-14 March 1995.

- Li, R., Reynolds, A.C., and Oliver, D.S.: "History Matching of Three-Phase Flow Production Data," paper SPE 66351 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 11-14 February, 2001.
- Li, R., Reynolds, A.C., and Oliver, D.S.: "Sensitivity Coefficients for Three-Phase Flow History Matching," *Journal of Canadian Petroleum Technology*, 42(4), 70-77, 2003.
- Liang, B., Alpak, F.O., Sepehrnoori, K., and Delshad, M.: "A Singular Evolutive Interpolated Kalman Filter for Rapid Uncertainty Quantification," paper SPE 106170 presented at the 2007 SPE Reservoir Simulation Symposium, Houston, TX, 26-28 February, 2007.
- Litvak, M., Christie, M., Johnson, D., Colbert, J., and Sambridge, M.: "Uncertainty Estimation in Prediction Predictions Constrained by Production History and Time-Lapse Seismic in a GOM Oil Field," paper SPE 93146 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 31 January-2 February, 2005.
- Liu, N., Betancourt, S., and Oliver, D.S.: "Assessment of Uncertainty Assessment Methods," paper SPE 71624 presented at the 2001 SPE Annual Technical Conference and Exhibition, New Orleans, LA, 30 September-3 October, 2001.
- Liu, N. and Oliver, D.S.: "Evaluation of Monte Carlo Methods for Assessing Uncertainty," *Soc. Petrol. Eng. J.*, 8(2), 1-15, 2003.
- Liu, N. and Oliver, D.S.: "Experimental Assessment of Gradual Deformation Method," *Mathematical Geology*, 36(1), 65-77, 2004.
- Liu, N. and Oliver, D. S.: "Critical Evaluation of the Ensemble Kalman Filter on History Matching of Geological Facies," paper SPE 92867 presented at the SPE 2005 SPE Reservoir Simulation Symposium, Houston, TX, 31 January-2 February, 2005.
- Lorentzen, R.J., Fjelde, K.K., Frøyen, J., Lage, A.C.V.M., Nævdal, G., and Vefring, E.H.: "Underbalanced and Lowhead Drilling Operations: Real Time Interpretation of Measured Data and Operational Support," paper SPE 71384 presented at the SPE Annual Technical Conference and Exhibition, New Orleans, LA, 30 September-3 October, 2001.
- Lorentzen, R.J., Nævdal, G., Vallès, B., Berg, A.M., and Grimstad, A.-A.: "Analysis of the Ensemble Kalman Filter for Estimation of Permeability and Porosity in Reservoir Models," paper SPE 96375 presented at the SPE Annual Technical Conference and Exhibition, Dallas, TX, 9-12 October, 2005.
- Lorentzen, R.J., Berg, A.M., Nævdal, G., Vefring, E.H.: "A New Approach for Dynamic Optimization of Waterflooding Problems," paper SPE 99690 presented

- at the Intelligent Energy Conference and Exhibition, Amsterdam, The Netherlands, 11-13 April, 2006.
- Makhlouf, E.M., Chen, W.H., Wasserman, M.L., and Seinfeld, J.H.: "A General History Matching Algorithm for Three-Phase, Three-Dimensional Petroleum Reservoirs," *SPE Advanced Technology Series*, 1(2), 83-92, 1993.
- Marsily, G., Lavedan, G., Boucher, M., Fasanino, G.: "Interpretation of Interference Tests in a Well Field Using Geostatistical Techniques to Fit the Permeability Distribution in a Reservoir Model," in the edited book *Geostatistics for Natural Resources Characterization, Part 2*, 831-849, Kluwer Academic Publishers, 1984.
- McKinney, D.C., and Lin, M.D.: "Genetic Algorithm Solution of Groundwater Management Models," *Water Resources Research*, 30(6), 1897-1906, June 1994.
- Nævdal, G., Brouwer, D. R. and Jansen, J.-D.: "Waterflooding Using Closed-Loop Control," *Computational Geosciences*, 10(1), 37-60, March 2006.
- Nævdal, G., Johnsen, L. M., Aanonsen, S.I., and Vefring, E.H.: "Reservoir Monitoring and Continuous Model Updating using Ensemble Kalman Filter," *Soc. Petrol. Eng. J.*, 10(1), 66-74, 2005.
- Nævdal, G., Mannseth, T., and Vefring, E.H.: "Near-Well Reservoir Monitoring through Ensemble Kalman Filter," paper SPE 75235 presented at the SPE/DOE Improved Oil Recovery Symposium, Tulsa, OK, 13-17 April, 2002.
- Nerger, L., Danilov, S., Kivman, G., Hiller, W., and Schröter, J.: "Data Assimilation with the Ensemble Kalman Filter and the SEIK Filter Applied to a Finite Element Model of the North Atlantic," *J. Mar. Syst.*, 65(1-4), 288-298, 2007.
- Nicotra, G., Godi, A., Cominelli, A., Christie, M.: "Production Data and Uncertainty Quantification: A Real Case Study," paper SPE 93280 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 31 January-2 February, 2005.
- Oliver, D.S.: "Multiple Realizations of the Permeability Field from Well-Test Data," *Soc. Petrol. Eng. J.*, 1(2), 145-154, 1996.
- Oliver, D.S., Cunha, L.B., and Reynolds, A.C.: "Markov Chain Monte Carlo Methods for Conditioning a Permeability Field to Pressure Data," *Mathematical Geology*, 29(1), 61-91, 1997.
- Oliver, D.S., He, Nanqun, and Reynolds, A.C.: "Conditioning Permeability Fields to Pressure Data," paper presented at the 5th European Conference on the Mathematics of Oil Recovery, Leoben, Austria, 3-6 September, 1996.

- Oliver, D.S., Reynolds, A.C., Bi, Z., and Abacioglu, Y.: "Integration of Production Data into Reservoir Models," *Petroleum Geoscience*, 7, S65-S73, 2001.
- Ouenes, A., Brefort, B., Meunier, G., and Dupere, S.: "A New Algorithm for Automatic History Matching: Application of Simulated Annealing Method (SAM) to Reservoir Inverse Modeling," paper SPE 26297, 1993.
- Ouenes, A., William, W., Sultan, A.J., and Anwar, J.: "Parallel Reservoir Automatic History Matching Using a Network of Workstations and PVM," paper SPE 29107 presented at the SPE Reservoir Simulation Symposium, San Antonio, TX, 12-15 February, 1995.
- Panda, M.N. and Lake, L.W.: "Parallel Simulated Annealing for Stochastic Reservoir Modeling," paper SPE 26418 presented at the SPE Annual Technical Conference and Exhibition, Houston, TX, 3-6 October, 1993.
- Parashar M., Wheeler, J. A., Pope, G., Wang, K., Wang, P., "A New Compositional EOS Reservoir Simulator: Part II - Framework and Multiprocessing," paper SPE 37977, presented at the SPE Symposium on Reservoir Simulation, Dallas, TX, June 8-11, 1997.
- Park, K. And Choe, J.: "Use of Ensemble Kalman Filter with 3-Dimensional Reservoir Characterization," paper SPE 100178 presented at the SPE Europec/EAGE Annual Conference and Exhibition, Vienna, Austria, 12-15 June, 2006.
- Pham, D.T., Verron, J., and Gourdeau, L.: "Singular Evolutive Kalman Filters for Data Assimilation in Oceanography," *C. R. Acad. Sci.*, 326(4), 255-260, 1998a.
- Pham, D.T., Verron, J., and Roubaud, M.C.: "A Singular Evolutive Extended Kalman Filter for Data Assimilation in Oceanography," *J. Mar. Syst.*, 16, 323-340, 1998b.
- Portellaand, R.C.M. and Prais, F.: "Use of Automatic History Matching and Geostatistical Simulation to Improve Production Forecast," paper SPE 53976 presented at the Latin American and Caribbean Petroleum Engineering Conference, Caracas, Venezuela, 21-23 April, 1999.
- RamaRao, B.G., LaVenue, A.M., Marsily, G., Marietta, M.G.: "Pilot Point Methodology for Automated Calibration of an Ensemble of Conditionally Simulated Transmissivity Fields, 1, Theory and Computational Experiments," *Water Resour. Res.*, 31(3), 475-494, 1995.
- Rame, M., Delshad, M.: "A Compositional Reservoir Simulator on Distributed Memory Parallel Computers," paper SPE 29103 presented at the 13th Symposium on Reservoir Simulation, San Antonio, TX, 1995.

- Reinlie, S.T.: *Analysis of Continuous Monitoring Data and Rapid, Stochastic Updating of Reservoir Models*, PhD Dissertation, The University of Texas at Austin, May 2006.
- Reynolds, A.C., He, N., Chu, L., and Oliver, D.S.: "Reparameterization Techniques for Generating Reservoir Descriptions Conditioned to Variograms and Well-Test Pressure Data," *Soc. Petrol. Eng. J.*, 1(4), 413-426, 1996.
- Rodrigues, J.R.P.: "Calculating Derivatives for History Matching in Reservoir Simulators," paper SPE 93445 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 31 January-2 February, 2005.
- Roggero, F. and Guerillot, D.: "Gradient Method and Bayesian Formalism Application to Petrophysical Parameter Characterization," paper presented at the 5th European Conference on the Mathematics of Oil Recovery, Leoben, Austria, 3-6 September, 1996.
- Roggero, F. and Hu, L.Y.: "Gradual Deformation of Continuous Geostatistical Models for History Matching," paper SPE 49004 presented at the SPE Annual Technical Conference and Exhibition, New Orleans, LA, 27-30, September 1998.
- Romero, C.E., Carter, J.N., Gringarten, A.C., and Zimmerman, R.W.: "A Modified Genetic Algorithm for Reservoir Characterisation," paper SPE 64765 presented at the SPE International Oil and Gas Conference and Exhibition in China, Beijing, China, 7-10 November, 2000.
- Rotondi, M., Nicotra, G., Godi, A., Contento F.M., Blunt M.J., and Christie, M.A.: "Hydrocarbon Production Forecast and Uncertainty Quantification: A Field Application," paper SPE 102135 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, 24-27 September, 2006.
- Sambridge M.: "Geophysical Inversion with a Neighborhood Algorithm-I: Searching a Parameter Space," *Geophys. J. Int.*, 138, 479-494, 1999a.
- Sambridge M.: "Geophysical Inversion with a Neighborhood Algorithm-I: Appraising the Ensemble," *Geophys. J. Int.*, 138, 727-746, 1999b.
- Sarma, P., Durlofsky, L.J., and Aziz, K.: "Efficient Closed-Loop Production Optimization under Uncertainty," paper SPE 94241 presented at the SPE/EAGE Annual Conference, Madrid, Spain, 13-16 June, 2005.
- Sarma, P., Chen, W.H., Durlofsky, L.J., and Aziz, K.: "Production Optimization with Adjoint Models under Nonlinear Control-State Path Inequality Constraints," paper SPE 99959 presented at the Intelligent Energy Conference and Exhibition, Amsterdam, The Netherlands, 11-13 April, 2006.

- Sarma, P., Durlofsky, L.J., Aziz, K., and Chen, W.H.: "A New Approach to Automatic History Matching Using Kernel PCA," paper 106176 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 26-28 February, 2007.
- Schiozer, D.J. and Sousa, S.H.G.: "Use of External Parallelization to Improve History Matching," paper SPE 39062 presented at the Fifth Latin American and Caribbean Petroleum Engineering Conference and Exhibition, Rio de Janeiro, Brazil, 30 August-3 September, 1997.
- Schulze-Riegert, R.W., Axmann, J.K., Haase, O., Rian, D.T., and You, Y.-L.: "Evolutionary Algorithms Applied to History Matching of Complex Reservoirs," *SPE Reservoir Evaluation & Engineering*, 5(2), 163-173, April 2002.
- Scott, S.L., Wainwright, R.L., Raghavan, R., and Demuth, H.: "Application of Parallel (MIMD) Computers to Reservoir Simulation", paper SPE 16020 presented at the ninth SPE Symposium on Reservoir Simulation, San Antonio, TX, 1987.
- Shewuck, J.R.: "An Introduction to the Conjugate Gradient Method without the Agonizing Pain," Internal Report, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, August, 1994 (available at <http://www-2.cs.cmu.edu/~jrs/jrspapers.html>).
- Shiralkar, G.S., Stephenson, R.E., Joubert, W., Lubeck, O., van Bloemen, B.: "Falcon: A Production Quality Distributed Memory Reservoir Simulator," paper SPE 37975 presented at the SPE Symposium on Reservoir Simulation, June 8-11, Dallas, TX, 1997.
- Skjervheim, J.-A., Evensen, G., Aanonsen, S.I., Ruud, B.O., and Johansen, T.A.: "Incorporating 4D Seismic Data in Reservoir Simulation Models Using Ensemble Kalman Filter," paper SPE 95789 presented at the SPE Annual Technical Conference and Exhibition, Dallas, TX, 9-12 October, 2005.
- Sousa, S.H.G., Maschio, C., and Schiozer D.J.: "Scatter Search Metaheuristic Applied to the History-Matching Problem," paper SPE 102975 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, 24-27 September, 2006.
- Srinivasan, S., and Bryant, S.: "Integrating Dynamic Data in Reservoir Models Using a Parallel Computational Approach," paper SPE 89444 presented at the 2004 SPE/DOE Thirteenth Symposium on Improved Oil Recovery, Tulsa, OK, 17-21 April, 2004.
- Stat-Ease, Inc.: *Design-Expert® 6 User's Guide*, 2003.
- Subbey, S., Christie, M., and Sambridge, M.: "A Strategy for Rapid Quantification of Uncertainty in Reservoir Performance Prediction," paper SPE 79678 presented at the SPE Reservoir Simulation Symposium, Houston, TX, 3-5 February, 2003.

- Subbey, S., Christie, M., and Sambridge, M.: "Prediction under Uncertainty in Reservoir Modeling," *Journal of Petroleum Science and Engineering*, 44, 143-153, 2004.
- Sudaryanto, B. and Yortsos, Y.C.: "Optimization of Fluid Front Dynamics in Porous Media Using Rate Control: I. Equal Mobility Fluids," *Physics of Fluids*, 12(6), 1656-1670, 2000.
- Suzuki, S. and Caers, J.: "History Matching with an Uncertain Geological Scenario," paper SPE 102154 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, 24-27 September, 2006
- Tarantola, A. and Valette, B.: "Inverse Problems = Quest for Information," *J. Geophys.*, 50, 159-170, 1982.
- Tavassoli, Z., Carter, J.N., and King, P.R.: "Errors in History Matching," *Soc. Petrol. Eng. J.*, 9(3), 352-361, September, 2004.
- Tecplot, Inc.: *Tecplot 10 User's Manual*, 2003.
- Tecplot, Inc.: *Tecplot RS 4.0 User's Manual*, 2004.
- Triantafyllou, G., Hoteit, I., and Petihakis, G.: "A Singular Evolutive Interpolated Kalman Filter for Efficient Data Assimilation in a 3-D Complex Physical-Biogeochemical Model of the Cretan Sea," *J. Mar. Syst.*, 40-41, 213-231, 2003.
- Türeyen, Ö.I.: *A Parallel Modelling Approach to Reservoir Characterization*, PhD Dissertation, Stanford University, March 2005.
- Vasco, D.W., Yoon, S., and Datta-Gupta, A.: "Integrating Dynamic Data Into High-Resolution Reservoir Models Using Streamline-Based Analytic Sensitivity Coefficients," *Soc. Petrol. Eng. J.*, 389-399, December 1999.
- van Leeuwen, P.J.: "A Variance-Minimizing Filter for Large-Scale Applications," *Monthly Weather Review*, 131, 2071-2084, 2003.
- Wang, P., Balay, S., Sepehrnoori, K., Wheeler, J., Abate, J., Smith, B., Pope, G.A.: "A Massively Parallel Reservoir Simulator for Large Scale Reservoir Simulation," paper SPE 51886 presented at the 1999 SPE Reservoir Simulation Symposium, Houston, TX, 14-17 February, 1999.
- Wen, X. and Chen, W.H.: "Real-Time Reservoir Model Updating Using Ensemble Kalman Filter," paper SPE 92991 presented at the SPE 2005 SPE Reservoir Simulation Symposium, Houston, TX, 31 January-2 February, 2005.

- Wen, X., Deutsch, C.V., Cullick, A.S.: "High Resolution Reservoir Models Integrating Multiple-Well Production Data," paper SPE 38728 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, 5-8 October, 1997.
- Wheeler, J.A, and Smith, R.A., "Reservoir Simulation on a Hypercube," paper SPE 19804 presented at the 64th Annual SPE Conference and Exhibition, San Antonio, TX, October 1989.
- Williams, G.J.J., Mansfield, M., MacDonald, D.G., and Bush, M.D.: "Top-Down Reservoir Modelling," paper SPE 89974 presented at the SPE Annual Technical Conference and Exhibition, Houston, TX, 26-29 September, 2004.
- Wu, Z., Reynolds, A.C., and Oliver, D.S.: "Conditioning Geostatistical Models to Two-Phase Production Data," *Soc. Petrol. Eng. J.*, 3(2), 142-155, 1999.
- Xiu, D., Karniadakis, G.E.: "Modeling Uncertainty in Flow Simulations via Generalized Polynomial Chaos," *Journal of Computational Physics*, 187, 137-167, 2003.
- Xue, G. and Datta-Gupta: "Structure Preserving Inversion: An Efficient Approach to Conditioning Stochastic Reservoir Models to Dynamic Data," paper SPE 38727 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, 5-8 October, 1997.
- Yadav, S.: "History Matching Using Face-Recognition Technique Based on Principal Component Analysis," paper SPE 102418 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, 24-27 September, 2006.
- Yang, Anping: *Stochastic Heterogeneity and Dispersion*, PhD Dissertation, The University of Texas at Austin, 1990.
- Yarus, J.M. and Chambers, R.L.: "Practical Geostatistics – An Armchair Overview for Petroleum Reservoir Engineers," *JPT*, 78-87, November, 2006.
- Yeten, B., Durlofsky, L., and Aziz, K.: "Optimization of Well Type, Location and Trajectory," paper SPE 77565 presented at the SPE Annual Technical Conference and Exhibition, San Antonio, TX, September 29-October 2, 2002a.
- Yeten, B., Durlofsky, L.J., and Aziz, K.: "Optimization of Smart Well Control," paper SPE 79031 presented at the SPE International Thermal Operation and Heavy Oil Symposium and International Horizontal Well Technology Conference, Calgary, Alberta, Canada, 4-7 November, 2002b.
- Yeten, B., Brouwer, D.R., Durlofsky, L.J., and Aziz, K.: "Decision Analysis under Uncertainty for Smart Well Deployment," *Journal of Petroleum Science and Engineering*, 44, 175-191, 2004.

- Zafari, M. and Reynolds, A.C.: "Assessing the Uncertainty in Reservoir Description and Performance Predictions with the Ensemble Kalman Filter," paper SPE 95750 presented at the SPE 2005 SPE Annual Technical Conference and Exhibition, Dallas, TX, 9-12 October, 2005.
- Zhang, D., Lu, Z., and Chen, Y.: "Dynamic Reservoir Data Assimilation with an Efficient, Dimension-Reduced Kalman Filter," paper SPE 95277 presented at the SPE Annual Technical Conference and Exhibition, Dallas, TX, 9-12 October, 2005.
- Zhang, F., Reynolds, A.C.: "Optimization Algorithms for Automatic History Matching of Production Data," paper presented at the 8th European Conference on the Mathematics of Oil Recovery, Freiberg, Germany, 3-6 September, 2002.
- Zhang, F., Skiervheim, J.A., Reynolds, A.C., and Oliver, D.S.: "Automatic History Matching in a Bayesian Framework, Example Applications," paper SPE 84461 presented at the SPE Annual Technical Conference and Exhibition, Denver, CO, 5-8 October, 2003.
- Zhang, Jiang: *IRSS: An Integrated Reservoir Simulation System*, PhD Dissertation, The University of Texas at Austin, August 2005.

Vita

Baosheng Liang was born in Chengwu County, Shandong Province, China on February 14, 1979, the son of Dunfa Liang and Meilian Xie. After completing his education at No. 7 High School of Chengwu County, Shandong Province in July 1996, he entered Southwest Petroleum University, Sichuan Province in August, the same year. He was continuously ranked number one in the following four-year study among 160 students in the Petroleum Engineering major. In July 2000, he earned the Bachelor of Science degree in Petroleum Engineering and was admitted to the graduate program without examination to continue his study at the same university. In July 2003, he received the Master of Science degree in Petroleum Engineering. One month later, he enrolled the Ph.D. program in the Department of Petroleum and Geosystems Engineering, The University of Texas at Austin.

Permanent address: Daliuzhuang, Dizhuang Cun, Sunsi Xiang, Chengwu County
Heze City, Shandong Province, 274200
China

This dissertation was typed by the author.