



Oasis in the middle Heihe river basin

Real-time control of water abstraction from an aquifer in the Heihe river basin, China

Project Work, Environmental Engineering Studies ETH Zürich

November 2013

Author: Gianni Pedrazzini

Supervisor: Prof. Dr. Wolfgang Kinzelbach

ETH Zürich, Swiss Federal Institute of Technology Zürich
Department of Civil, Environmental and Geomatic Engineering
Institute of Environmental Engineering
Wolfgang-Pauli-Str-15
8093 Zürich, Switzerland



Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

I hereby declare that the written work I have submitted entitled:

Real-time control of water abstraction from an aquifer in the Heihe river basin, China
is original work which I alone have authored and which is written in my own words.

Author and date:

Gianni Pedrazzini

Supervisor and date:

Prof. Dr. Wolfgang Kinzelbach

With the signature I declare that I have been informed regarding normal academic citation rules and that I have read and understood the information on "Citation etiquette" (http://www.ethz.ch/students/exams/plagiarism_s_en.pdf). The citation conventions usual to the discipline in question here have been respected. The above written work may be tested electronically for plagiarism.

Abstract

The expansion of agriculture into arid and semi-arid areas, the increased water demand due to a double-crop practice and the convenience of groundwater as irrigation source have lead many farmers in China to switch from surface water irrigation to groundwater irrigation [Shah 2000]. Large areas in the north of China now suffer from groundwater overexploitation, and the necessity to develop sustainable management strategies has gained recognition in the Chinese Government. A project has recently been initiated by ETH Zürich to investigate the implementation of a real-time monitoring, modelling and controlling system in two areas of interest in North China [SDC Proposal 2013]. For this project, the Heihe River Basin located in the Gansu province is the site of interest.

This project work investigates the design of a control system that should serve as a decision tool to sustainably manage the aquifer as a resource of water. The idea is to make use of the aquifer as a water reservoir in times of need, but giving it time to recover during the periods in between. By doing so, in the long run, the groundwater level should be kept fluctuating around a certain level.

A simplified one-dimensional box-model operating in the discrete time domain is used to model the aquifer. The new groundwater level is computed recursively from the old value and the prediction of the external flows acting on it. The management strategy is implemented through a control unit, which regulates the amount of water the farmers are allowed to pump from the aquifer. The proposed control unit combines two single PID-controller signals to compute an optimal pumping rate. The objective of the controller is to provide sufficient water for irrigation while maintaining groundwater on an acceptable level. Stochastic time series input data for the model is generated synthetically by using a logAR1 model. A Kalman-Filter is used to find an optimal weight between predictions and measured values. As no real measured values are available for the controlled system, virtual measurements of a fictional true system state are simulated with the very same model.

The results show that the control scheme succeeds in controlling the system with respect to the defined objectives. The performance of the controller does however strongly depend on the nature of the input data. As the objectives of pumping "on-demand" and maintaining the groundwater level around a given threshold value are in some way of conflicting nature, each choice of pumping rate is always a trade-off between these two objectives. The model can now be used as tool to qualitatively and quantitatively assess the outcome of a certain decision we take regarding the relative importance each of these two objectives.

List of figures

| | | |
|-----------|---|----|
| Figure 1 | Monitored pumping well with IC-card [Kinzelbach 2013]. Using an IC-card, the user is allowed pump a specified quantity of water over a certain time. | 10 |
| Figure 2 | Map of the irrigation plain in the Heihe River Basin [Wikipedia, Google Maps] | 11 |
| Figure 3 | Schematic overview of the water flows interacting with the aquifer in the one-dimensional box model..... | 12 |
| Figure 4 | Simulink block diagram model setup. The boxes denote mathematical operations or in/outputs from and into MATLABs workspace. The arrows represent information flows. | 14 |
| Figure 5 | Scenario S1: | 24 |
| Figure 6 | Scenario S2: | 25 |
| Figure 7 | Scenario S3: | 26 |
| Figure 8 | Scenario S4: | 27 |
| Figure 9 | Scenario S5: | 28 |
| Figure 10 | Scenario S6: | 29 |
| Figure 11 | Plot of the pareto-front for simulations using the combined controller with different k-contr values (ranging from 0 to 1) over a duration of 1000 years. The initial 30 years of data was not considered due to the observed numerical oscillations related to initial conditions..... | 30 |

List of tables

| | | |
|---------|---------------------------|----|
| Table 1 | Simulation scenarios..... | 21 |
| Table 2 | Model parameters | 21 |

Table of contents

| | |
|--------------------------------------|----|
| Declaration of originality | 3 |
| Abstract | 5 |
| List of figures | 6 |
| List of tables..... | 6 |
| Table of contents..... | 7 |
| 1 Introduction | 9 |
| 1.1 General background..... | 9 |
| 1.2 ETH Project | 9 |
| 1.3 My project work..... | 10 |
| 2 Methodology..... | 12 |
| 2.1 General model description..... | 12 |
| 2.2 Software..... | 13 |
| 2.3 Optimization..... | 18 |
| 2.4 Control unit..... | 19 |
| 2.5 Initial conditions | 20 |
| 2.6 Kalman-Filter..... | 20 |
| 2.7 Simulations..... | 21 |
| 2.8 Summary of model parameters..... | 21 |
| 3 Results..... | 24 |
| 3.1 Scenario S1..... | 24 |
| 3.2 Scenario S2..... | 25 |
| 3.3 Scenario S3..... | 26 |
| 3.4 Scenario S4..... | 27 |
| 3.5 Scenario S5..... | 28 |
| 3.6 Scenario S6..... | 29 |
| 3.7 Scenario S7: Pareto front | 30 |
| 4 Conclusions..... | 32 |
| 5 Recommendations | 33 |
| 6 Acknowledgements..... | 34 |
| 7 References | 34 |
| 7.1 Literature | 34 |
| 7.2 Online sources..... | 35 |
| 7.3 Image sources | 35 |

| | | |
|-----|------------------------------------|----|
| 8 | Annex..... | 36 |
| 8.1 | Instructions to run the model..... | 36 |
| 8.2 | File "model_runner.m"..... | 36 |
| 8.3 | File "model_script.m"..... | 37 |
| 8.4 | File "logMAR1.m"..... | 40 |
| 8.5 | file "pareto_model_runner.m"..... | 41 |

1 Introduction

1.1 General background

The advantages of groundwater being accessible to many users, being relatively cheap and convenient, has made groundwater increasingly attractive as a source of irrigation water for farmers in China. In addition to that, in some parts of North China surface water has been drastically curtailed for industrial needs. Irrigation with groundwater brings along important advantages. As it is produced at the point of use it needs little transport. In addition to that, the perennial availability of groundwater makes it a source of irrigation water available on demand that not many surface water systems can offer [Shah 2000].

The rapid population growth and urbanization in China is widening the gap between water supply and water demand that is already severe [Hydrosolutions Ltd 2013]. The increasing demand for food production forces farmers to expand agriculture more and more into arid regions and rely on groundwater as main irrigation source. Further stress on water as irrigation source is induced by China's double-cropping system, where maize is being grown in summer and wheat in winter. Particularly in the North China Plain, where precipitation is small in winter, this strains heavily the groundwater supplies [Nature 2010].

As pumping rates are however highly exceeding natural recharge rates of the aquifer, overexploitation occurs. This resulted in severe groundwater depletion particularly in the arid and semi-arid regions of China [Kinzelbach 2013, Xiyang 2001].

A declining groundwater table brings along a number of related problems. Diminished draining rates to rivers can lead to depletion of groundwater fed wetlands and related aquatic ecosystems. In deltaic regions it can also lead to salt water intrusion and land subsidence. [Wada 2010]. For farmers the depleting groundwater table means increasing costs of lift, and therefore increased costs of production. At worst, when wells fall dry or pollution or salt water intrude the aquifer, groundwater has to be completely abandoned as a source of water.

In recent times there has been recognition by the Chinese government that the problem of declining groundwater level is an imminent threat to the China's food security and an acute problem, and that if no change in the current practice occurs, large agricultural areas will not be cultivable anymore in near future [SDC Proposal].

The big challenges now lie in developing sustainable aquifer management strategies. Of particular interest for this project are active management strategies, which include decision-making based on real time monitoring of groundwater levels and rates of abstraction.

1.2 ETH Project

ETH Zürich together with the Swiss Agency for Development and Cooperation SDC (Federal Department of Foreign Affairs) is currently initiating a project with the aim to investigate the possibility of a real-time optimization of water allocation and consumption on an aquifer scale. The overall goal of the project is *"to develop and apply an integrated real-time monitoring, modelling and controlling system to promote reasonable water resources allocation, prevent groundwater depletion, foster water table recovery, sustain groundwater dependent ecosystems and thus build up the local community's adaptation capacity to climate variability."* [SDC Proposal]

Real-time modelling implies that monitoring data of the pumping rates and levels in observation wells are automatically transferred to a regional groundwater model. This model is then used to simulate different scenarios. According to a certain strategy, an optimal allocation of water can then be calculated. A regulatory system consists of pumping wells locked by an IC-card system (see Figure 1), which allows the regulator to release a maximal quota of water that can be abstracted by a certain user (e.g. a farmer) over a given period of time. While automatic monitoring networks for groundwater level measurement are already well established in China, the monitoring of pumping volumes is not yet very much widespread [SDC Proposal 2013]. The ETH project aims to focus on two basins: the Heihe River Basin, located in the Zhangye region in Gansu and another basin in North China Plain, in the regions of Shijiazhuang and Handan.

Figure 1 Monitored pumping well with IC-card [Kinzelbach 2013]. Using an IC-card, the user is allowed pump a specified quantity of water over a certain time.



1.3 My project work

The model described in this project refers to the Heihe River Basin irrigation plain shown in Figure 2. After sourcing in the mountainous catchment area the Heihe flows across the large semi-arid irrigation district. The river leaves the irrigation plain in the north and finally dries up in two terminal lakes in Mongolia. Since the residual water leaving the irrigation plain is of high importance for downstream ecosystems, the Government has set a minimal threshold volume of water that has to be released for downstream ecosystems yearly. As this is limiting the maximal abstraction rates of river water and due to the poor reliability of surface irrigation water from canals the aquifer is now being overexploited by excessive groundwater pumping. The rapid drawdown of the groundwater table (up to 20m in some areas [Kinzelbach 2013]) has given rise to concerns and triggered studies and investigations to develop more advanced water management strategies [SDC Proposal 2013].

This project work investigates the implementation of a control system in a simplified one-dimensional box-model for the Heihe River Basin. The control system should serve as a decision tool to sustainably manage the aquifer as a resource of water by optimizing the pumping rate.

The objectives of the controller are:

- Ensure that groundwater level does not drop too far below a given target value.
- Ensure constant amount of irrigation water.

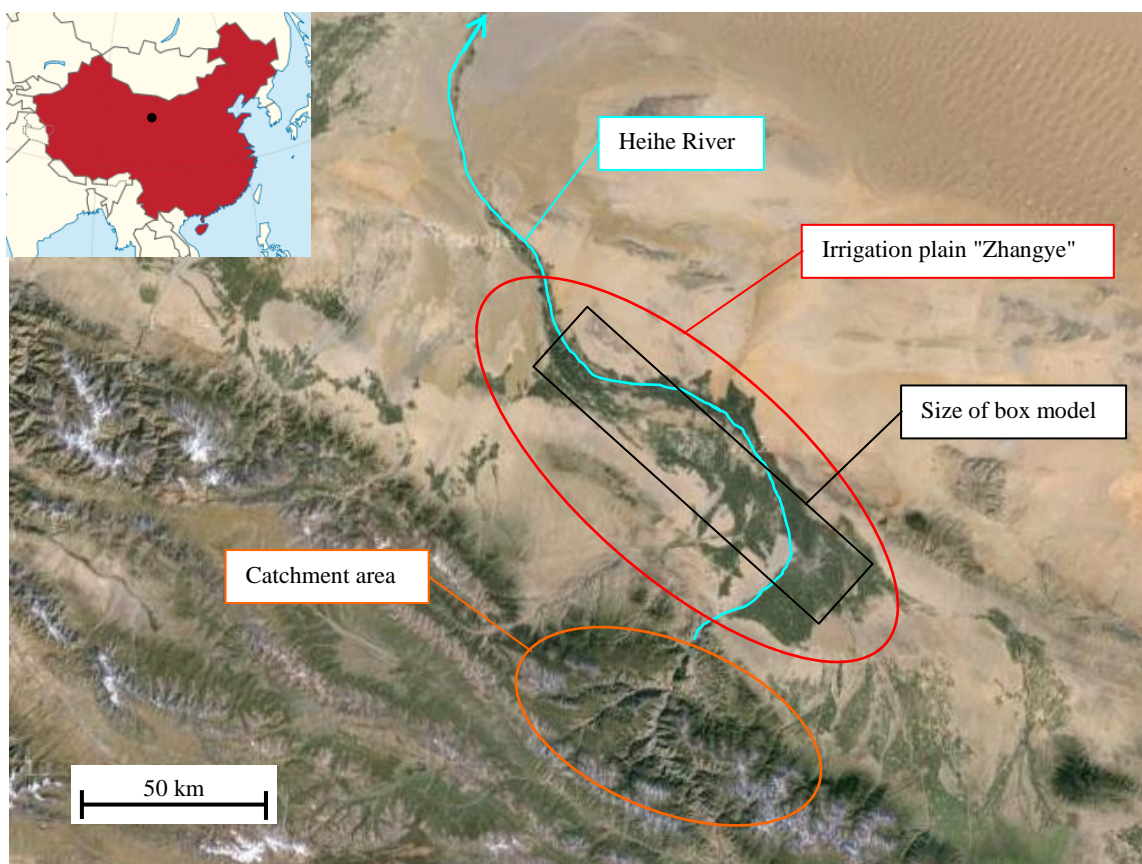
The underlying assumption is that there should be a given quantity of total irrigation water (denoted as $Q_{irr.target}$), which, for reasons of practicality, should fluctuate, as little as possible. This should allow the farmers to overcome periods of water scarcity and be able to rely on constant water at their disposal. At the same time we want to prevent further declining of the groundwater for reasons mentioned previously.

As one notes intuitively, these two objectives are to a large extent in conflicting nature to each other. If we pump purely "on-demand", we will not be able to control the groundwater level, whereas if we pump just in order to hold a certain groundwater level, the pumping irrigation water will not necessarily be covering the demand.

The idea is therefore to make use of the aquifer as a water reservoir in times of need, but giving it time to recover during the periods in between. By doing so, in the long run, the groundwater level should be kept fluctuating around a certain level.

A detailed model description follows in the subsequent section.

Figure 2 Map of the irrigation plain in the Heihe River Basin [Wikipedia, Google Maps]



2 Methodology

2.1 General model description

A conceptualized lumped one-dimensional box-model approach is used to describe the system dynamics in a discrete time domain. Figure 3 shows an overview of the system. The water content in the groundwater box represents the water content of the aquifer. The nature of the model itself is simple, it recursively computes the volume V at the new time as function of the volume at the old time and the predicted disturbance acting on it. The disturbance comprises the sum of all external flows of water affecting the aquifer during a certain time interval.

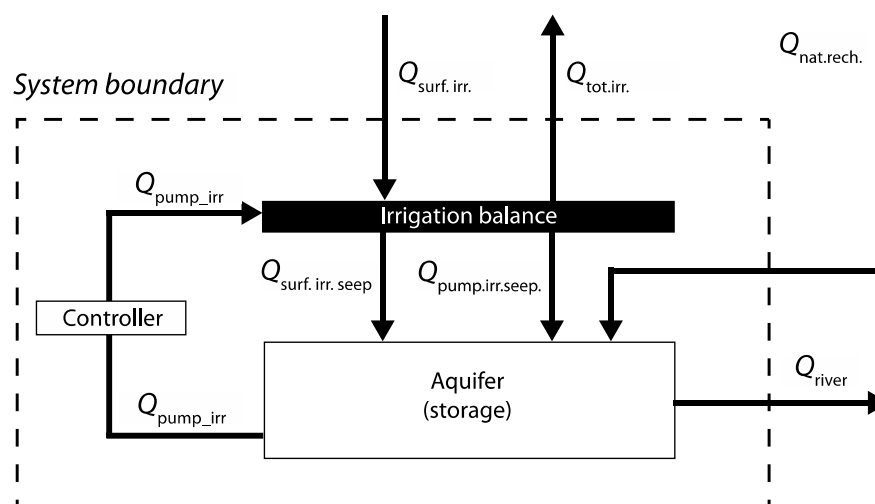
The idea behind this approach is that based on experience, measurements and meteorological and hydrological forecasting we could be able to provide some estimate of those flows. Quantification of snow volumes in the upper catchment could for instance be used as a hint for estimates of surface water in the upcoming spring and summer.

The physical mechanisms and interactions taking place in the groundwater may be way more complex than assumed in the model. On the temporal domain for instance one could think of time lags affecting the interaction of surface water with groundwater. Another aspect is certainly the distinct seasonality affecting the temporal distribution of the different flows. The simplicity of the model structure chosen allows however to keep an overview on the interaction between the different external forcing on the model and the focus primarily on the control mechanism

To simplify the interpretation of the results all flows were computed per unit area, therefore in m/year. This facilitates the work with the model as the dimension of the output numbers remains in the decimal range.

The box model has an areal extent of $2.4 \cdot 10^9 m^2$, which corresponds to a square of approximately size $25 km \cdot 100 km$. The bottom of the box is set at $1300 m.a.m.s.l.$, the natural elevation of the water table without irrigation activities is presumed to lie at $1355 m.a.m.s.l.$ The effective porosity of the aquifer is set to be 12% [Kinzelbach 2013].

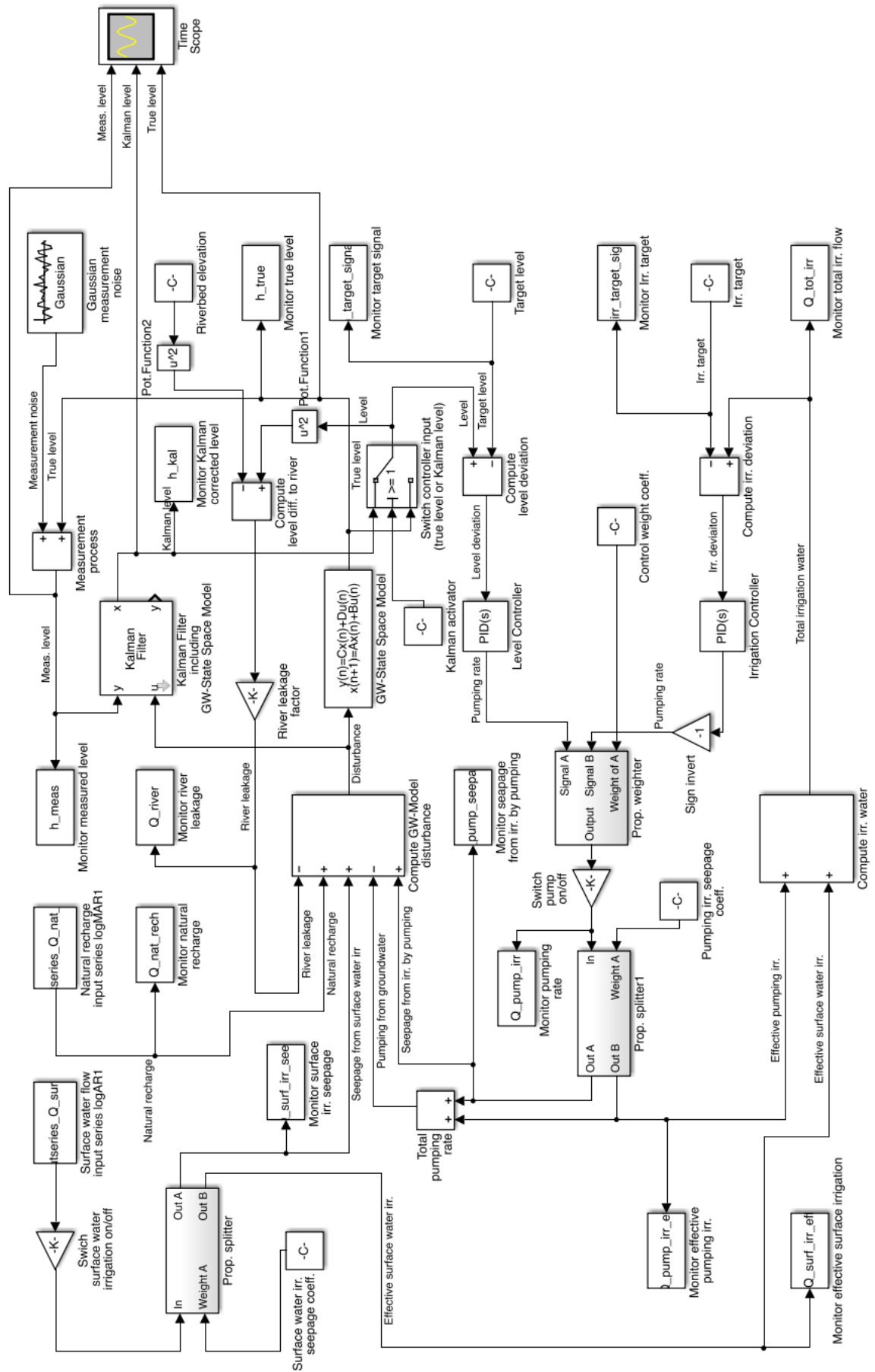
Figure 3 Schematic overview of the water flows interacting with the aquifer in the one-dimensional box model.



2.2 Software

The model implemented in MathWorks MATLAB R2013a using Simulink, the block diagram environment for multi-domain simulation. Simulink allows the simulation of dynamic systems, and allows exchange, import and export of MATLAB scripts and functions. The block diagram based graphical user interface consents keep a better overview over a complex system, and permits to easily add further elements into the system at a later point in time. Figure 4 shows an overview of the model design in Simulink.

Figure 4 Simulink block diagram model setup. The boxes denote mathematical operations or in/outputs from and into MATLABs workspace. The arrows represent signal flows.



2.2.1 State transition function and state-space representation

The state transition function for the computation of the new volume $V(t + \Delta t)$ is formulated as follows:

$$h(t + \Delta t) = h(t) + \frac{\Delta t}{n_{eff}} \sum Q_i(t)$$

where $\sum Q_i(t)$ is the sum of the predicted flows acting on the aquifer in the new time interval and Δt the duration of the time interval. As we consider the flows Q_i per unit area their dimension is m/year.

The model can be reformulated using a linear state-space representation in the discretized time domain [Gouxiang 2012]. This formulation is used to simplify implementation of the model into Simulink:

$$X(t_{i+1}) = A \cdot X(t_i) + b \cdot u(t_i)$$

$$y(t_i) = c \cdot X(t_i) + d \cdot u(t_i)$$

where X describes the system state at a point in time and u the disturbance acting on it in the current time interval. In this case the system state vector X contains only one variable, namely the level h of the water table in our aquifer:

$$X(t_i) = [h(t_i)]$$

The disturbance u comprises the sum of the predicted flows affecting the aquifer:

$$u(t_i) = \sum Q_i$$

A , b , c , and d are the system matrices, which can be defined as follows

$$A = [1]$$

$$b = \left[\frac{\Delta t}{n_{eff}} \right]$$

$$c = [1]$$

$$d = [0]$$

Where A is the state or system transition matrix, b input matrix, c the observation matrix also known as output matrix and d feed-through matrix. Δt is the time interval and n_{eff} the porosity. As we only have one system state variable in the system state vector X , our observation y of the system corresponds to the system state itself and therefore $c = [1]$.

The state-space representation can be seen as the general representation used for linear systems. It is useful since it can easily be implemented into the predefined Simulink-blocks that MATLAB uses to solve differential equations (for instance with MATLAB's ode45 differential solver). As the recursive state transition equation can simply be algebraically solved at the moment, this is not particularly important. It leaves however the possibility of adding complexity to the system at later point in time. For instance one could reformulate the system equation into a differential equation, to model more complex system dynamics.

2.2.2 Disturbance

The disturbance u was computed as the sum of all flows affecting the aquifer. The selected parameter values and respective sources are listed in Table 2. The individual flows are described in the following section.

$$u(t_i) = [Q_{nat.rech.}(t_i) + Q_{surf.irr.seep.}(t_i) - Q_{pump.irr.}(t_i) + Q_{pump.irr.seep.}(t_i) - Q_{ri\ddot{a}ser}(t_i)]$$

2.2.3 Natural aquifer recharge $Q_{nat.rech.}$

The natural recharge of the aquifer $Q_{nat.rech.}$ describes the natural aquifer recharge as a consequence of deep percolation of precipitation water. It stands for the natural amount of recharge that takes place regardless of the irrigation activities. As the model at this stage is purely fictional, the recharge data was synthetically generated. To account for the natural fluctuations occurring as consequence climate dynamics a stochastic logarithmic AR1 time series model was used.

The autoregressive component ensures a certain degree of memory in the process, so that the new recharge is not a purely random guess, but correlates to some extent to the previous one. The log-character ensures that the generated values are always positive since we want to avoid negative recharge values while preserving the statistical moments of the distribution. Due to the limited amount of time and because the emphasis of the project lied on the control mechanism of the pumping rate, no time series analysis of any input data was performed. The generated logAR1 series serves therefore a purely hypothetical basis for computations, and is by no means truly representative of the real data.

The logAR1 model was computed as follows [Burlando 2013, Kottegoda 1997]

$$\tilde{x}(t + \Delta t) = \theta_1 \tilde{x}(t) + \sqrt{1 - r_1^2} \varepsilon$$

$$\tilde{x}(t) = \frac{x(t) - m_x}{s_x}$$

$$x(t) = \log(y(t))$$

$$m_{\log(y)} = \log\left(\frac{m_y^2}{\sqrt{m_y^2 + s_y^2}}\right)$$

$$s_{\log(y(t))} = \sqrt{\log\left(1 + \frac{s_y^2}{m_y^2}\right)}$$

where \tilde{x} is the standardization of the normal distributed realization x with mean m_x and standard deviation s_x . y is the log-normal distributed variable, in our case the recharge, with mean m_y and standard deviation s_y . To ensure ergodicity, meaning that the mean of the generated ensemble corresponds to the true mean, a constant of was added to the generated data series in order to shift the series by the required amount.

2.2.4 Surface water $Q_{surf.irr.}$

The surface water represents the water coming from upstream streams, sources and reservoirs that is routed to the irrigation district. It is also presumed to be subject to fluctuations, as it is partly related to precipitation and snowfall. The same logAR1 model described previously was therefore used.

Since the both natural recharge and surface irrigation are somehow related to precipitation, they are considered to be cross-correlated to some degree. The correlation was implemented by correlating the two random generated numbers ε of the two series¹.

2.2.5 Surface irrigation seepage water $Q_{surf.irr.seep.}$

This component reflects the seepage to the aquifer occurring as a consequence of irrigation by surface water. It is computed as a constant fraction ($k_{surf.irr.seep.}$) of the surface irrigation water. The underlying simplifying assumption is that always the same proportion of irrigation water inevitably percolates and recharges the aquifer.

2.2.1 Surface irrigation water $Q_{surf.irr.eff.}$

The effective surface irrigation water $Q_{surf.irr.eff.}$ is the remaining surface irrigation water after deduction of the seepage component $Q_{surf.irr.seep.}$.

2.2.2 Pumping irrigation water $Q_{pump.irr.}$

The water that is abstracted from the aquifer by through pumps is denoted as $Q_{pump.irr.}$. This flow is regulated by the controller unit described later on.

2.2.3 Pumping irrigation seepage water $Q_{pump.irr.seep.}$

This represents the fraction of pumping irrigation water that percolates into the aquifer. Same as for the surface irrigation water the assumption is that this fraction (defined by $k_{pump,seepage}$) remains constant.

2.2.4 Pumping irrigation water $Q_{pump.irr.eff.}$

After deduction of the seepage component $Q_{pump.irr.seep.}$, the remaining pumping irrigation water $Q_{pump.irr.eff.}$ is the net amount of water left for the plants.

2.2.5 River leakage Q_{river}

The river leakage is computed using an approach for one-dimensional stationary flow [Kinzelbach 2005:Vorlesungsskript Grundwasser I] by which the river leakage Q_{river} appears to be proportional to the difference of the squares of the head h and the river elevation h_{river}

$$Q_{river} = \alpha_{river} \cdot (h^2 - h_{river}^2)$$

¹ Done as described by MathWorks, Article on generation of correlated random numbers, <http://www.mathworks.com/support/solutions/en/data/1-CYHA89/index.html>, 4.11.2013

The parameter α_{river} was calibrated so that $Q_{river}(h = h_{natural}) = Q_{river.nat}$, where $Q_{river.nat}$ is the natural river leakage balancing out with the natural rate of recharge. The river acts as a natural controller on the groundwater level by discharging water proportionally to the head difference to the river. If no groundwater is being pumped, it is the only sink in the box model.

2.2.6 Total irrigation water $Q_{tot.irr.}$

The total irrigation water is computed as the sum of the irrigation flow by groundwater pumping and by surface water irrigation minus the seepage components. This is the water quantity that we presume is effectively available for the plants. Evaporation was not taken into account.

$$Q_{tot.irr.} = Q_{pump.irr.} + Q_{surf.irr.} - Q_{pump.irr.seep} - Q_{surf.irr.seep}$$

2.3 Optimization

As we are dealing with a multi-objective problem, the control unit is designed to minimize the two following variables:

- Level deviation:

$$h_{target} - h$$

- Irrigation deficit/surplus:

$$Q_{target} - Q_{tot.irr.}$$

where h_{target} is our target groundwater level, h our actual water table level, $Q_{tot.irr.}$ the actual computed total irrigation water and Q_{target} the irrigation target. The targets values for the level is set to be

$$h_{target} = 1350m. a. m. s. l.$$

This level corresponds to a 5m drawdown with respect to the natural level of $h_{nat} = 1355m. a. m. s. l.$. We consider this value an acceptable drawdown since a minimal drawdown always takes place as we pump water from the aquifer.

The irrigation target Q_{target} was set to be equal to the sum of the inflows into the aquifer ($Q_{mean,net,GW-inflow}$) and the surface irrigation water supply ($Q_{surf.irr.} \cdot (1 - k_{surf.irr.seep})$) under steady state conditions, whereby the stochastic inputs are represented by their mean values.

$$Q_{mean,net,GW-inflow} = Q_{surf.irr.mean} * k_{surf.irr.seep} + Q_{nat.rech.mean} - Q_{river,expected}$$

$$Q_{target} = Q_{mean,net,GW-inflow} + Q_{surf.irr.mean} \cdot (1 - k_{surf.irr.seep}) + Q_{overpump}$$

$$Q_{river,expected} = \alpha_{river} \cdot (h_{target}^2 - h_{river}^2)$$

The additional term $Q_{overpump}$ allows implementing an overpumping factor, which can bring the sum of flows out of balance. $Q_{river,expected}$ represents the expected river leakage under the assumption $h = h_{target}$.

By doing so we set the irrigation target Q_{target} to a value that would theoretically balance out all flows and lead to a stable system if none of the inputs were stochastic but deterministically defined (using the mean values instead of the stochastic generated series). This is considered to be the basic assumption if we want to bring stability into the system. The mean of all inflows should equal the mean of all outflows in the long run.

If the system was purely deterministic this assumption would be sufficient to stabilize it. The necessity for a control unit is however justified by the fluctuations of the inputs. If we want to prevent the groundwater level from fluctuating too much we must use some kind of control mechanism. This is shown in the result section further on in the report. We also note that

2.4 Control unit

The core of the model is the control unit that regulates Q_{pump} . The controller is designed to fulfil two targets: maintaining the groundwater level on the long term on an acceptable level while ensuring sufficient irrigation water in times of need. For this purpose a combination of two PID-controllers (proportional, integral, differential) was used.

One serves to maintain the level (further on denoted as level controller) and one to ensure sufficient irrigation water supply (denoted as irrigation controller). The output $Q_p^{contr.}$ of each controller can be written as follows [as described in Gujer 2010]:

$$Q_p^{contr.} = K_p \cdot e(t_i) + K_i \cdot \int_0^{t_i} e(t_i) dt + K_d \cdot \frac{de(t_i)}{dt}$$

whereby e is the deviation from the target, K_p , K_i and K_d the transfer coefficients (or gains) for the proportional, integral and differential members. The deviation e is computed from the difference between the actual signal and the target value.

The level controller uses the computed groundwater level deviation from target as input whereas the irrigation controller the irrigation water deviation from target. Each of the controllers suggests an optimal pumping rate as output, which presumably optimally regulates the system according to the setup of the controller. The effective pumping rate Q_p is then computed by taking a weighted sum of both pumping rates.

$$Q_p = (1 - k_{contr.}) \cdot Q_p^{irr.contr.} + k_{contr.} \cdot Q_p^{level.contr.}$$

where $Q_p^{irr.contr.}$ is the output of the irrigation controller, $Q_p^{level.contr.}$ the output of the level controller and $k_{contr.}$ the weighting coefficient. The idea of this combined system is to avoid an instantaneous transition between the two controllers, which could imply hard jumps in the effective pumping rate Q_p signal, possibly leading to oscillations or instabilities in the model. By considering a weighted sum we ensure the effective pumping rate Q_p follows a much more continuous and smooth trajectory. If the deviation of one of the signals becomes very large, the corresponding controller gradually takes over and as soon as that deviation is reduced, the original proportions of control signals is re-established. The controller is implemented in the form of a feedback loop. The gains of the controller were adapted based on simple try-and-error manual calibration procedure. No formal loop-shaping procedure was used, as this is not considered to be of fundamental importance at this stage of the project.

2.5 Initial conditions

An initial head condition of $h_{initial}$ was set so that we can observe the initial recovery response. In the case of Heihe this offset represents the degradation that has already taken place in the last decades due to over pumping.

2.6 Kalman-Filter²

For the purpose of using the model as a real-time simulation tool a Kalman-Filter was added to the model. The Kalman-Filter is a linear filter algorithm developed by Rudolf Kalman in 1960 [Wikipedia]. The filter enables to find a new best estimate based the model prediction, the measured value and a measure of confidence of the model prediction. The principle of least squares estimation is hereby adopted in a recursive manner.

2.6.1 Prediction

The system state at a point in time is completely described by a state vector. For the "true" system state, the state vector is denoted by the variable T , whereas the "modelled" system state is described by the variable X . This had to be done since there are no real measurements available. The measurements of the true system state T are assumed to be biased by random noise eta representing inaccuracies that may occur through the measurement process. The matrices A, b, c, d are described in the previous section.

$$T(t_i) = A \cdot T(t_{i-1}) + b \cdot u(t_i)$$

$$y(t_i) = c \cdot T(t_i) + d \cdot u(t_i) + eta(t_i)$$

$$X(t_i) = A \cdot X(t_{i-1}) + b \cdot u(t_i)$$

The predicted system state at time t_i is a linear function of the old system state at time t_{i-1} , and a disturbance component $u(t_i)$

The propagation of the model error covariance matrix P is given by the following equation

$$P(t_i) = A \cdot P(t_{i-1}) \cdot A^T + Q$$

whereby Q represents the covariance matrix of the observation process of the model.

Since at current stage the model is not being tested in the real environment no measurements are available. To use the Kalman-Filter, measurements are however necessary.

2.6.2 Observation/Measurements

The observation (or measurement) y of the true system state T happens through a projection by the observation matrix c and the addition of an observation error eta . The observation error eta is considered to follow a Gauss-distribution with mean 0 (no bias) and variance $eta_{meas.var}$.

$$y(t_i) = c \cdot T(t_i) + eta(t_i)$$

² Information on the Kalman-Filter was summarized from the following sources: [Perreault 2012], [Wikipedia],

2.6.3 Correction

The Kalman-corrected prediction X_k of the model prediction X is calculated by the following equation

$$X_k(t_i) = X(t_i) + K(t_i) \cdot (y(t_i) - c \cdot X(t_i))$$

The Kalman-corrected error covariance matrix P_k is computed as follows

$$P_k(t_i) = (I - K(t_i) \cdot c) \cdot P(t_i)$$

Whereby K describes the Kalman matrix (also known as Kalman-gain)

$$K(t_i) = P(t_i) \cdot c^T \cdot S^{-1}(t_i)$$

and S the residual covariance

$$S(t_i) = c \cdot P(t_i) \cdot c^T + R$$

2.7 Simulations

Table 1 shows an overview of the scenarios that were simulated. The idea behind the scenarios is to show how the different elements of the model interact by adding them one by one. The Kalman-Filter is just activated for scenario S6; reason is that it facilitates the visualization of the plots for the reader, as there would be too many lines on the plot otherwise. It can theoretically however be activated on each simulation. The last scenario evaluates a Pareto-front using computed statistics of 20 simulations with different k_{contr} values.

Table 1 Simulation scenarios

| Scenario short name | Description |
|---------------------|---|
| S1 | No surface water irrigation, no irrigation by pumping |
| S2 | Only surface water irrigation |
| S3 | Surface water irrigation + pumping irrigation + irrigation controller |
| S4 | Surface water irrigation + pumping irrigation + level controller |
| S5 | Surface water irrigation + pumping irrigation + combined controller |
| S6 | Surface water irrigation + pumping irrigation + combined controller + Kalman-Filter |
| S7 | Pareto-front evaluation for 20 different k_{contr} -values (Surface water irrigation + pumping irrigation + combined controller) |

2.8 Summary of model parameters

The model parameters are listed in Table 2. The nomenclature corresponds to the one in the "model_script.m" file.

Table 2 Model parameters

| Parameter or variable | Description | Unit | Default value | Source |
|--------------------------------------|-------------|------|---------------|--------|
| General simulation parameters | | | | |

| | | | | |
|------------------------------------|--|-----------|----------------|-----|
| duration | Simulation duration | years | 300 | (4) |
| dt | simulation time increment, must be coherent to input data. Yearly values are used here | years | 1 | (4) |
| sim_model | reference to simulink model file | - | simulink_model | (4) |
| fix_seed | 0: seed not fixed, always generate new random numbers. 1: seed fixed, generate same random number series repeatedly | - | 1 | (4) |
| Natural recharge Q_nat_rech | | | | |
| Q_nat_rech_mean | mean | m/year | 0.083 | (1) |
| Q_nat_rech_stddev | std. dev. | m/year | 0.0083 | (2) |
| sigma_Q_nat_rech | autoregressive weigh parameter | - | 0.6 | (2) |
| Surface water Q_surf_irr | | | | |
| Q_surf_irr_mean | mean | m/year | 0.5 | (1) |
| Q_surf_irr_stddev | std. dev. | m/year | 0.05 | (2) |
| sigma_Q_surf_irr | autoregressive weigh parameter | - | 0.6 | (2) |
| cross_corr | cross-correlation of random component of Q_nat_rech and Q_surf_irr | - | 0.8 | (2) |
| k_surf_irr_seep | fraction of surface irrigation water that percolates to the aquifer and is therefore not available to the plants. | - | 0.2 | (1) |
| River leakage Q_river | | | | |
| h_nat | natural water level (without any irrigation) | m.a.m.s.l | 1355 | (1) |
| h_river | elevation of riverbed | m.a.m.s.l | 1341 | (1) |
| Q_river_nat | natural river flow at h=h_nat. Must equal to Q_nat_rech_mean in order balance out the equation. | m/year | 2 | (1) |
| Controller settings | | | | |
| pump_irr_switch | 1: switch on irrigation by pumping, 0: switch off irrigation by pumping | - | 1 | (4) |
| k_pump_irr_seep | fraction of pumping irrigation water that percolates to the aquifer and is therefore not available to the plants. | - | 0.1 | (1) |
| surf_irr_switch | 1: switch on surface water irrigation, 0: switch off surface water irrigation | - | 1 | (4) |
| k_contr | real number [0 1], 0: only irr. controller 1: only level controller, in between mixed controller | - | 0.5 | (4) |
| kalman_switch | integer [0 1], 0: deactivate Kalman-Filter, 1: activate Kalman-Filter | - | 0 | - |
| h_target | target water level which that should be aimed for by the | m.a.m.s.l | 1350 | (2) |

| | | | | |
|---|--|----------------|------|-----|
| | controller | | | |
| h_delta | confidence interval for level, only relevant for plot | m | 5 | - |
| f_overpump | over pumping factor, increase irrigation target by adding f_overpump | m/year | 0 | - |
| Q_irr_delta | confidence interval for irrigation water, only relevant for plot | m/year | 3 | - |
| PID-irrigation-controller parameters | | | | |
| kP_irr | proportional gain | - | 5 | (3) |
| kl_irr | integral gain | - | 0.8 | (3) |
| kD_irr | differential gain | - | 0 | (3) |
| kN_irr | filter coefficient | - | 0 | (3) |
| PID-level-controller parameters | | | | |
| kP_irr | proportional gain | - | 1.5 | (3) |
| kl_irr | integral gain | - | 0.5 | (3) |
| kD_irr | differential gain | - | 0 | (3) |
| kN_irr | filter coefficient | - | 0 | (3) |
| State-space groundwater model parameters | | | | |
| A | state transition matrix | - | [1] | (4) |
| b | input matrix | - | [dt] | (4) |
| c | observation matrix | - | [1] | (4) |
| d | feed through matrix | - | [0] | (4) |
| Kalman filter parameters | | | | |
| Q_kalman | process covariance | m ² | 0.01 | (2) |
| R_kalman | measurement noise variance | m ² | 1 | (2) |
| eta_meas_var | measurement noise variance (same as above) | m ² | 1 | (2) |
| P_initial | initial | m ² | 1 | (2) |
| Initial conditions | | | | |
| h_initial | initial head condition | m.a.m.s.l | 1340 | (2) |

(1): data sheet provided by Prof. Dr. Wolfgang Kinzelbach

(2): assumption

(3): roughly calibrated by trial and error

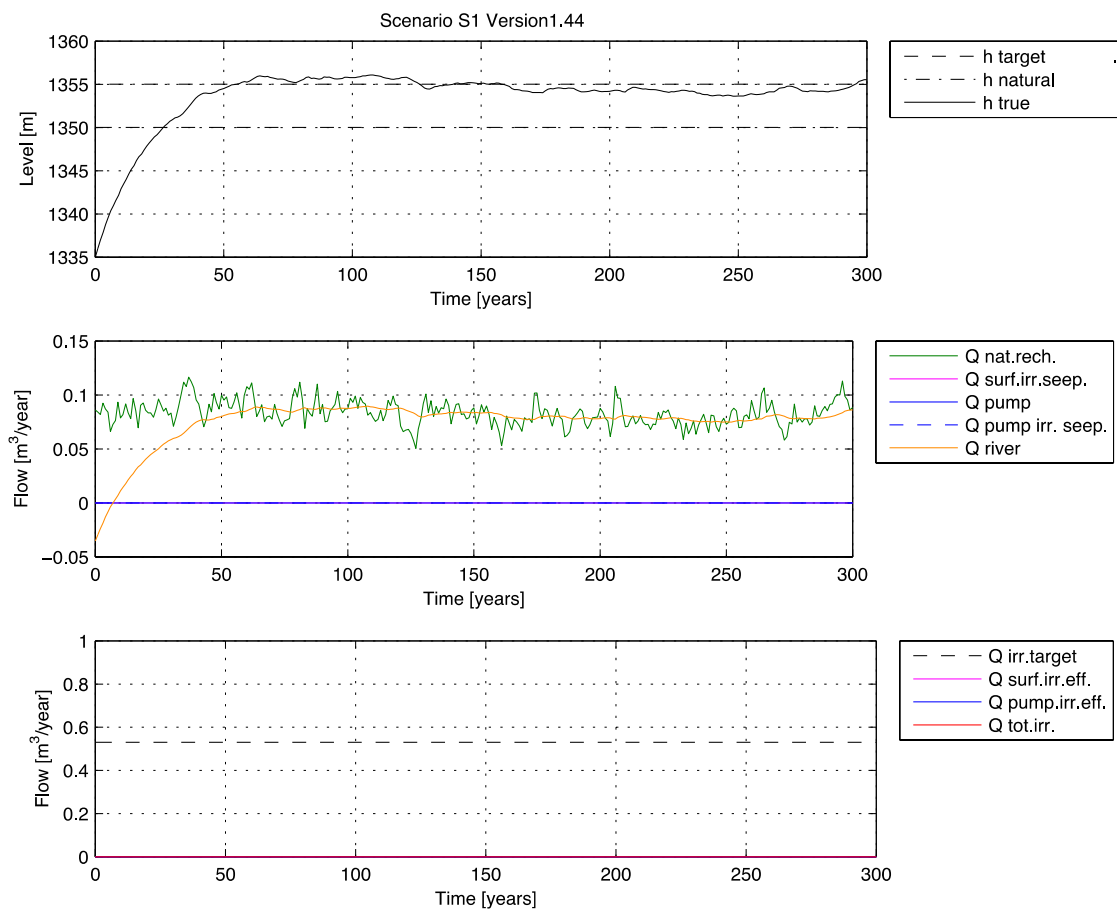
(4): model setting

3 Results

3.1 Scenario S1

The results of scenario S1 are shown in Figure 1. No irrigation at all takes place in this scenario. We note that the level fluctuates around the natural water level, as this is defined to be the level at which Q_{river} balances out with the average natural recharge. Hereby we observe how the river acts as a natural control unit and regulates the leakage according to the level.

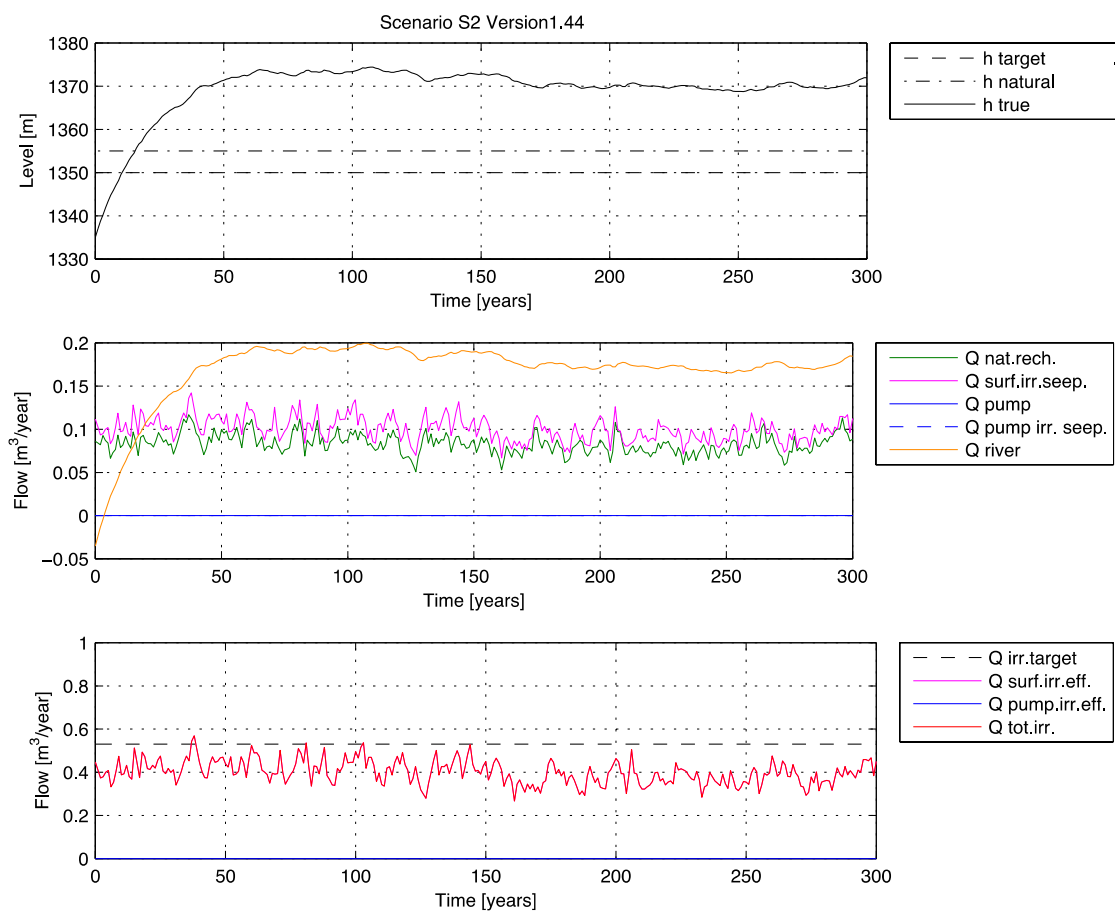
Figure 5 Scenario S1:



3.2 Scenario S2

Scenario S2 shows the effect of pure surface water irrigation on the aquifer. Since there is no pumping in the aquifer yet the level rises to the point where Q_{river} balances out again with the sum of the natural recharge and the additional seepage occurring as a consequence of the surface irrigation ($Q_{surf.irr.seep}$). The aquifer is hereby subject to an artificial recharge occurring from surface water irrigation. We also see how the generated time series of $Q_{surf.irr.seep}$ and $Q_{nat.rech.}$ do correlate to some extent. As far as the irrigation is concerned, we note that the total irrigation water $Q_{tot.irr.}$ corresponds to the the water provided by surface irrigation, which is by far not able to satisfy the demand $Q_{irr.target}$.

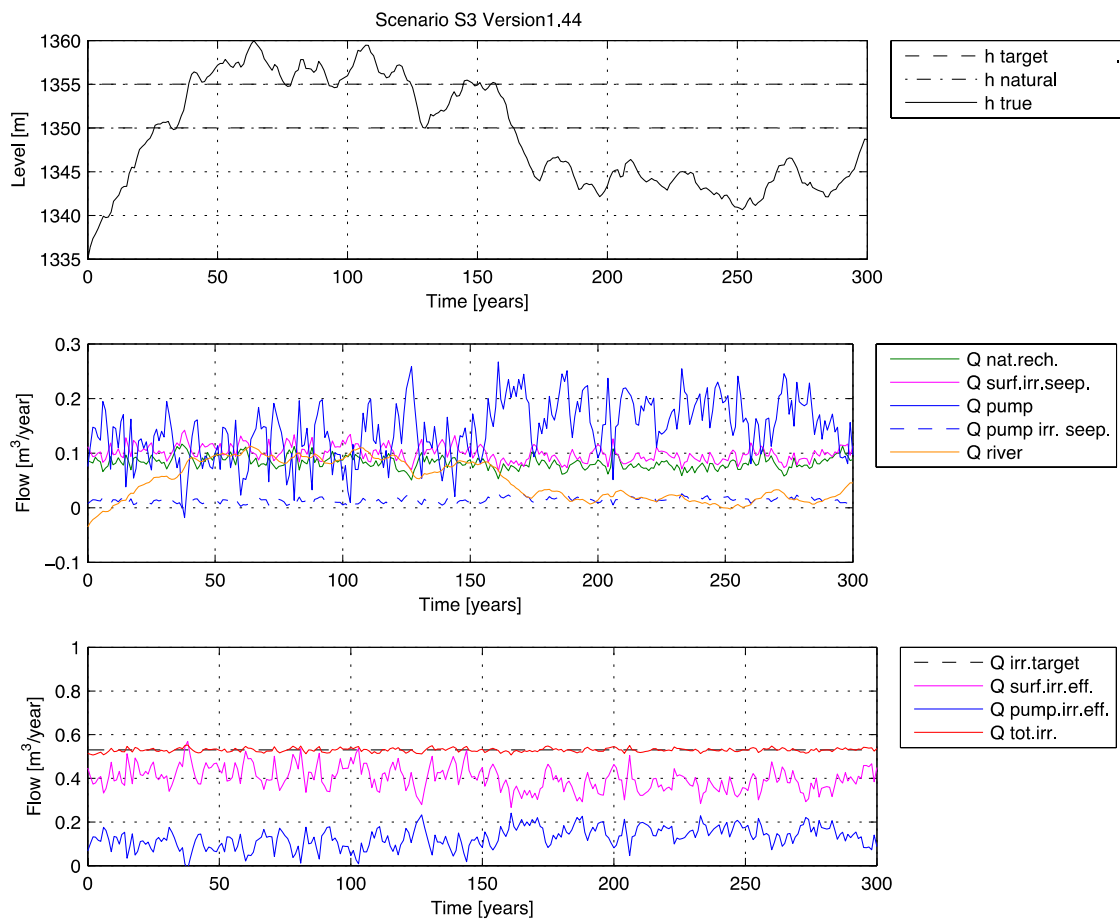
Figure 6 Scenario S2:



3.3 Scenario S3

The current scenario S3 simulates the combination of surface water irrigation and irrigation by pumping. We hereby set the controller to "irrigation controller"-mode, so it only controls the pumping rate to match the irrigation deficit/surplus. As we see the controller can successfully even out the fluctuations in surface water irrigation and provide a total irrigation water quantity very close to $Q_{irr.target}$. The cost of this "demand-oriented" irrigation practice is a strongly fluctuating groundwater level. The natural control mechanism occurring from the river leakage is not able anymore to maintain a constant water level as the magnitude in the rates of abstraction is too big. We observe that a few times the $Q_{surf.irr.eff.}$ peaks higher than the irrigation target, and the controller then sets the pumping rate to a negative value. This would mean that for short time the surplus of surface irrigation water could be explicitly used to artificially recharge the aquifer.

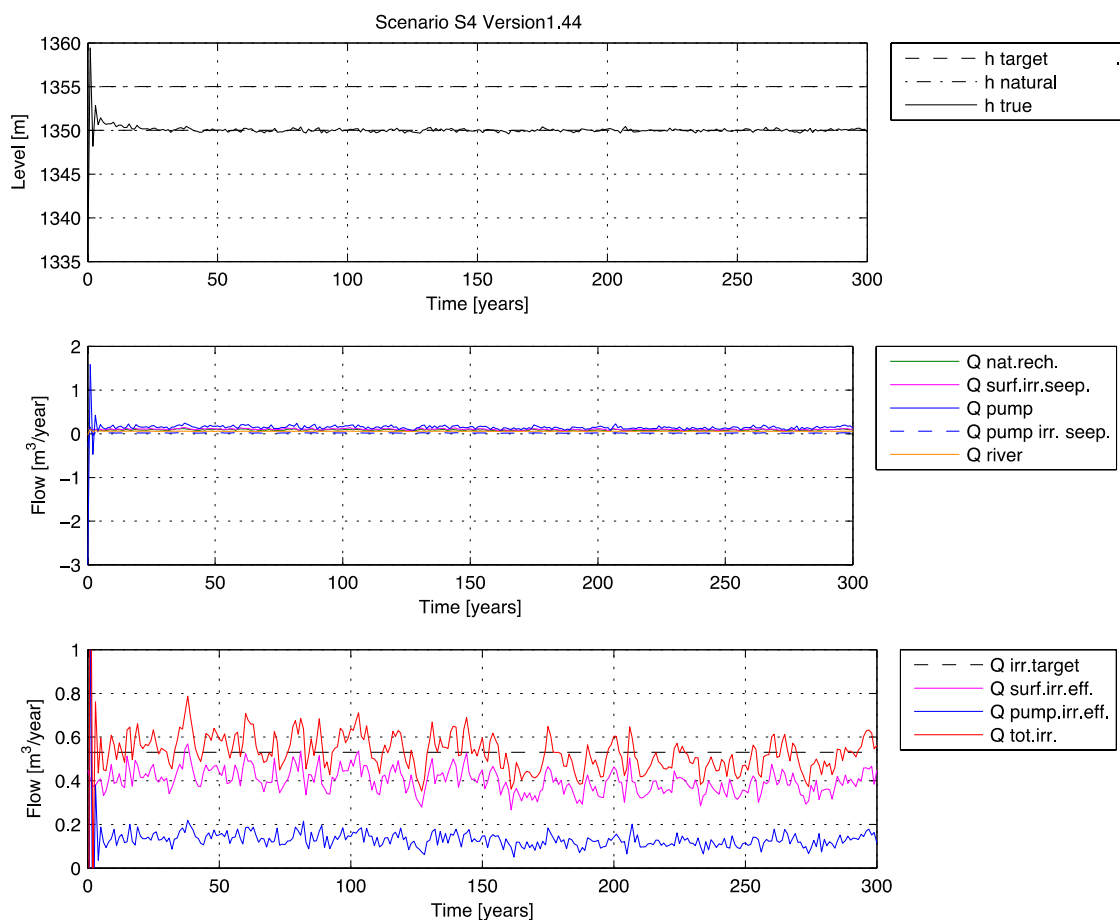
Figure 7 Scenario S3:



3.4 Scenario S4

Scenario S4 shows the situation if the controller would be set to "level controller"-mode. We see how it successfully maintains the groundwater level close to its target level. As a consequence however the irrigation suffers from a highly unsteady water supply. The purpose of this scenario is barely to show the performance of the level controller before using the combined controller. We note some oscillations in the pumping rate in the initial phase, these might be due to the parameter set of gains used for the level controller. Particularly the integral gain tends to produce fluctuations when the level must adjust from a high offset. The overshoots can also be related to a problem of too large modelling time step. Choosing a smaller modelling time step, for instance monthly or daily steps, could reduce this problem.

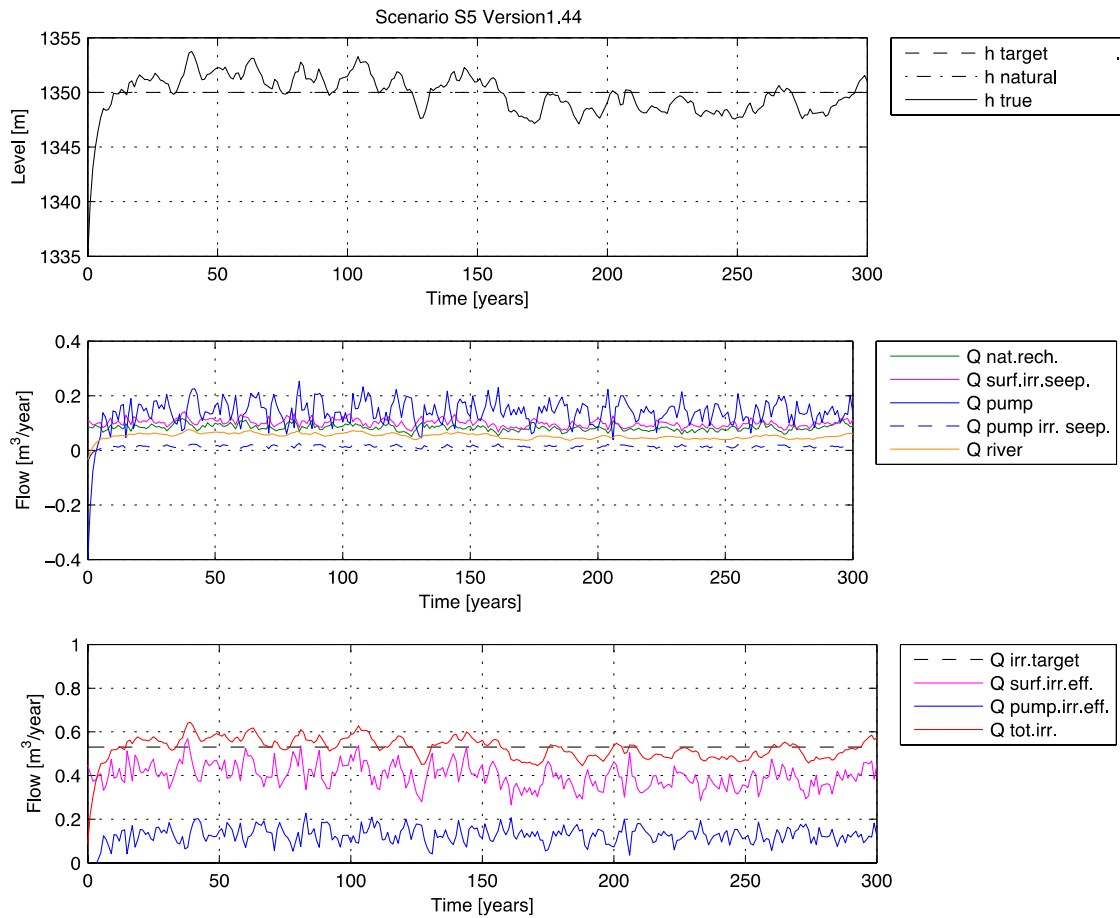
Figure 8 Scenario S4:



3.5 Scenario S5

The present scenario S5 shows the result of a combined level- and irrigation-control system. The combined control system manages to provide a more or less constant irrigation water supply while maintaining the groundwater level fluctuating around its target value. The controlling unit takes advantage of the water stored in the aquifer in order to minimize deficits during periods of lower surface irrigation water supply.

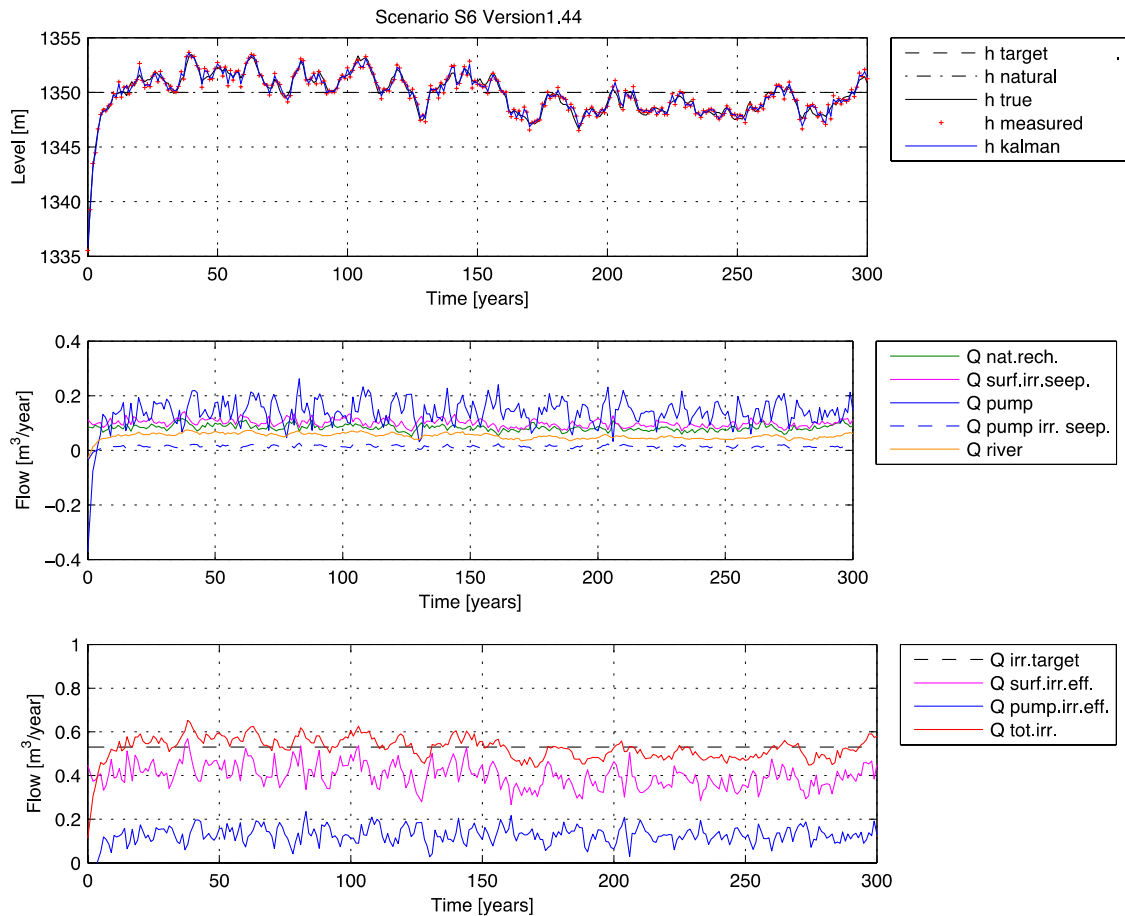
Figure 9 Scenario S5:



3.6 Scenario S6

Scenario S6 shows the model in combination with the Kalman-Filter. The model now uses a combination of the predicted value and the measured value under consideration of the respective uncertainties. The results themselves do not differ significantly from the previous simulation. It is in fact more useful when real measurement data are available. At present the real measurements are simulated by adding a noise component to a previously "true" system state that was generated with the very same model that is used for the predictions.

Figure 10 Scenario S6:



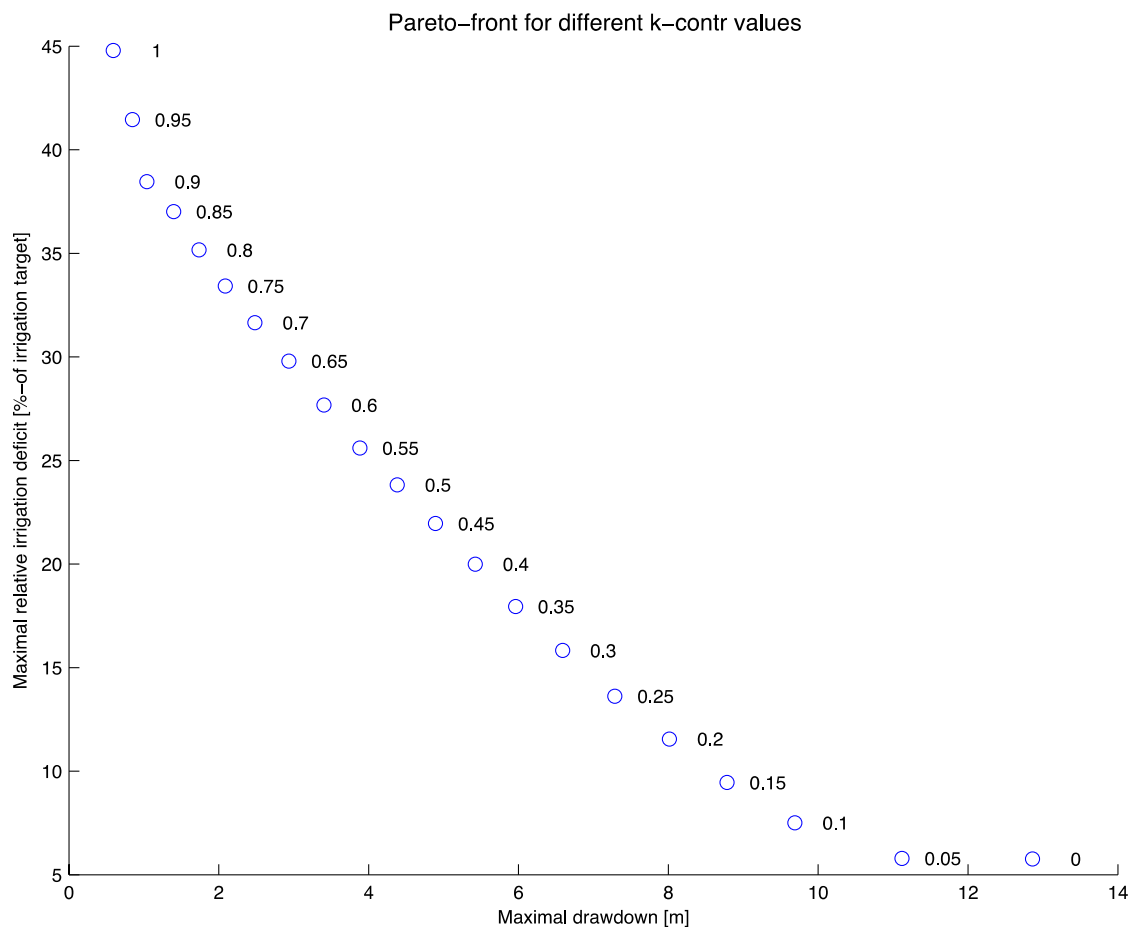
3.7 Scenario S7: Pareto front

Figure 11 shows a plot of the model performance with respect to the maximal drawdown measured and the maximal per cent irrigation deficit encountered during a simulation. We can possibly judge the performance of a simulation by considering these two indicators. A k_{contr} value of 0.5 for instance means if we accept a maximal irrigation deficit of about 24% we may encounter a maximal drawdown of 4.3 (always statistically speaking).

The simulation was here fore run over a period of 10'000 years, in order to have a statistically speaking more representative outcome. In order to avoid the effect of initial oscillations, the first 30 years were not considered, as they sometimes contain overshoots and peaks due to influence of the initial conditions. The simulation was run for 20 different control weight values k_{contr} , ranging from 0 (purely irrigation control) to 1 (purely level control). The respective values of k_{contr} are plotted next to each point.

The scatter plot shows what happens if we assign more importance to the level control or to the irrigation water control. We clearly see that the two objectives are conflicting each other and that under the present model setup we always must accept a trade-off between the two.

Figure 11 Plot of the pareto-front for simulations using the combined controller with different k-contr values (ranging from 0 to 1) over a duration of 1000 years. The initial 30 years of data was not considered due to the observed numerical oscillations related to initial conditions.



4 Conclusions

Personally speaking the project helped me a great deal to understand different elements of a real-time controlling scheme: the state-space representation of a linear system, the implementation and functionality of a Kalman-Filter and the embedding of models in MATLABs Simulink environment.

The results show that the combined controller is one possibility to successfully control the pumping rate in the simplified 1-D aquifer system. The different scenarios help to understand the mechanism and interaction of the individual components.

The application of the model to reality is however not crucial. As the model requires the different flow components as inputs, those have to be estimated in advance. The estimation of these flows may turn out to be challenging. Predictions could be based on meteorological and climatological models. Also the computation of certain model parameters may turn out to be difficult due to difficulties in measuring and possible lack of long-term data.

As the objectives of pumping "on-demand" and maintaining the groundwater level around a given threshold value are in some way of conflicting nature, each choice of pumping rate is always a trade-off that one makes. The model can now be used as tool to qualitatively and quantitatively assess the outcome of a certain decision we take regarding the relative importance of these two objectives.

5 Recommendations

In prospective of further investigations the following aspects may be of particular interest:

- Performing a time series analysis of real data in order to make the synthetic generated data series more realistic.
- Loop shaping to properly calibrate PID-controller.
- Performing a sensitivity analysis to assess the sensitivity of the outcome with respect to certain parameters.
- The calibration and validation of the model using real data series would be certainly interesting. As far as the combined controller is concerned no data can be measured as long as it is not being used in reality. The measured level data is the one to be most likely available. Other data such as total irrigation water would probably have to be estimated either from proxy-indicators (e.g. crop production) or from meteorological data.
- Simulations under scenarios of unsteady state input data, to simulate changing climate.
- Change time step sizes of simulation in order to improve model and controller performance. Hereby one could use different time steps for the modelling as for the pumping management decisions. As the monitoring data may be transmitted instantly over satellite, one could use a monthly or even daily time step for the model. The decision regarding the new pumping water quantity could then still be performed on a yearly basis.
- Consider patterns of seasonality in the input data.
- Considering uncertainties in the input data. Use the Ensemble Kalman-Filter instead of the normal Kalman-Filter. A number of simulations is run considering the statistical distribution of an unknown parameter. The new error covariance matrix is then computed out of the statistics of the simulation results, rather than through a linear mapping of the old covariance matrix.
- A further step would be to perform a spatial refinement of the model into different zonal subdivision. The ultimate challenge would be to combine the model with a discretized spatial domain groundwater model, such as the finite differences model MODFLOW.

6 Acknowledgements

I would like to express particular gratitude to Prof. Dr. W. Kinzelbach for his precious support and advice.

7 References

7.1 Literature

- Burlando P. (2013), Lecture notes of lecture "Water Resources Management", Lecture 4 "Stochastic Modelling"
- Feng, W., M. Zhong, J.-M. Lemoine, R. Biancale, H.-T. Hsu, and J. Xia (2013), Evaluation of groundwater depletion in North China using the Gravity Recovery and Climate Experiment (GRACE) data and ground-based measurements, *Water Resour. Res.*, 49, 2110–2118, doi:10.1002/wrcr.20192.
- Guzzella L. (2009), *Analysis and Synthesis of Single-Input Single-Output Control Systems*, second revised edition. vdf Hochschulverlag AG, Zürich
- Gujer W. (2010), *System Analysis for Water Technology*, Springer Verlag, Berlin ISBN: 978-3-642-09596-2
- Guoxiang G. (2012), *Discrete-Time Linear Systems - Theory and Design with Applications*, Springer New York Dordrecht Heidelberg London, ISBN 978-1-4614-2280-8
- Hydrosolutions Ltd, (2013?) *Analysis of Household Adaptation Mechanisms in China*, Final Report prepared for the World Bank, 3rd Draft, Zürich, Switzerland
- Kinzelbach W. (2013), PPT Presentation: Heihe_Control_Beijing_May_2013_final
- Kinzelbach W. (2013), Data-sheet with numbers quantifying flows affecting the aquifer
- Kinzelbach W. Stauffer F. (2005), *Vorlesungsskript Grundwasser I*, Institut für Hydromechanik und Wasserwirtschaft ETH Zürich
- Kottegoda, N. T. , and Rosso, R. (1997). *Statistics, probability and reliability for civil and environmental engineers* , McGraw–Hill, New York
- Perreault B. (2012), *Introduction to the Kalman Filter and its Derivation*, Concordia College, Moorhead Minnesota
- SDC Proposal (2013), Kinzelbach W. *Climate Change Adaptation in China: A Real-Time Monitoring, Modelling and Controlling system to prevent Groundwater Depletion*, Joint Project Proposal to Swiss Agency for Development and Cooperation SDC (Federal Department of Foreign Affairs).
- Shah, T.; D. Molden, R. Sakthivadivel, D. Seckler. 2000. *The global groundwater situation: Overview of opportunities and challenges*. Colombo, Sri Lanka: International Water Management Institute.
- Wada, Y., L. P. H. van Beek, C. M. van Kempen, J. W. T. M. Reckman, S. Vasak, and M. F. P. Bierkens (2010), Global depletion of ground- water resources, *Geophys. Res. Lett.*, 37, L20402, doi:10.1029/ 2010GL044571.

Xiying Zhang, Dong Pei, Chunsbeng Hu (2001) Conserving Groundwater for Irrigation in the North China Plain, *Irrig Sci* (2003) 21: 159–166 DOI 10.1007/s00271-002-0059-x

7.2 Online sources

Mathworks, Article on generation of correlated random numbers,
<http://www.mathworks.com/support/solutions/en/data/1-CYHA89/index.html>,
4.11.2013

Nature, International Weekly Journal of Science, China faces up to groundwater crisis,
<http://www.nature.com/news/2010/100713/full/466308a.html>, Published online 13
July 2010 *Nature* 466, 308 (2010), 7.11.2013

Mathworks, Simulink Kalman-Filter block by Yi Cao (2008), Cranfield University,
Copyright (c) 2010, All rights reserved. [http://www.mathworks.ch/matlabcentral/
fileexchange/18465-learning-the-kalman-filter-in-simulink-v2-
1/content/html/runkalmanfilter.html](http://www.mathworks.ch/matlabcentral/fileexchange/18465-learning-the-kalman-filter-in-simulink-v2-1/content/html/runkalmanfilter.html), 7.11.2013

Wikipedia, Article on Kalman-Filter, <http://de.wikipedia.org/wiki/Kalman-Filter>, 8.11.2013

7.3 Image sources

Cover sheet image: College of Engineering Peking University, Oasis in the Heihe River Basin, <http://en.coe.pku.edu.cn/news/397.htm>

Figure 2: large map: Google Maps, www.maps.google.com 7.Nov.2013,
small map: Wikipedia,
[http://de.wikipedia.org/wiki/Datei:China_on_the_globe_\(claimed_hatched\)_\(Asia_centered\).svg](http://de.wikipedia.org/wiki/Datei:China_on_the_globe_(claimed_hatched)_(Asia_centered).svg), 7.Nov.2013

Figure 1: Kinzelbach W., Presentation: "Heihe_Control_Beijing_May_2013_final", Oct.2013

8 Annex

8.1 Instructions to run the model

1. Copy the file folder "Matlab Files Final" to your hard drive.
2. Run MathWorks MATLAB r2013a and open the file folder. The following files are required to be in the folder: model_runner.m, model_skript.m, logAR1corr.m, pareto_model_runner.m, simulink_model.slx.
3. Customize your simulation settings in the files "model_runner.m", "model_skript.m", "logAR1corr.m". For a description of the parameters see Table 2.
4. Run "model_runner.m" by entering "model_runner.m" into the command window. The model will now run through the scenarios and save the generated plots (one figure for each scenario named "Scenario_S#.eps") into the current folder. To run the pareto-front evaluation run the file "pareto_model_runner.m". Pleaser read the instructions in the file header.

8.2 File "model_runner.m"

```
clear
```

```
model='model_script'  
version= 'Version1.44'
```

```
%% S1 Nothing
```

```
scenario='S1'  
pump_irr_switch=0  
surf_irr_switch=0  
k_contr=0  
kalman_switch=0  
run(model)
```

```
%% S2 Surface water irrigation
```

```
scenario='S2'  
pump_irr_switch=0  
surf_irr_switch=1  
k_contr=0  
kalman_switch=0  
run(model)
```

```
%% S3 Surface water irrigation + pumping irrigation + irr. controller
```

```
scenario='S3'  
pump_irr_switch=1  
surf_irr_switch=1  
k_contr=0  
kalman_switch=0  
run(model)
```

```
%% S4 Surface water irrigation + pumping irrigation + level controller
```

```
scenario='S4'  
pump_irr_switch=1  
surf_irr_switch=1  
k_contr=1
```

```
kalman_switch=0
run(model)

%% S5 Surface water irrigation + pumping irrigation + combined controller
scenario='S5'
pump_irr_switch=1
surf_irr_switch=1
k_contr=0.5
kalman_switch=0
run(model)

%% S6 Surface water irrigation + pumping irrigation + combined controller + Kalman filter
scenario='S6'
pump_irr_switch=1
surf_irr_switch=1
k_contr=0.5
kalman_switch=1
run(model)
```

8.3 File "model_script.m"

```
% Written by Gianni Pedrazzini (pedrazzini.gianni@gmail.com) october 2013. This script runs a si-
% mulink model which
% simulates pumping activities on an aquifer described by a 1-D box-model.
% it includes two PID-control units and a Kalman-Filter.

% pump_switch=0;
% surf_irr_switch=0;
% k_contr=0;
% kalman_switch=0;
% scenario='S1';
% version='Version v1.1';

%% General simulation parameters

duration=300;
dt=1;
sim_model='simulink_model';
fix_seed=0;
Area=2.4e9;

%% Input data settings

n_eff=0.12

% Natural recharge Q_nat_rech
Q_nat_rech_mean=0.083;
Q_nat_rech_stdev=0.0083; % stdev of recharge
sigma_Q_nat_rech=0.6;% autoregressive component parameter

% Surface water Q_surf_irr
Q_surf_irr_mean=0.5;
Q_surf_irr_stdev=0.05;
sigma_Q_surf_irr=0.6;
cross_corr=0.8;
k_surf_irr_seep=0.2;
```

```
% River leakage Q_river
h_nat=1355;
h_river=1341;
Q_river_nat=0.083;

%% Controller settings

%pump_irr_switch=1
k_pump_irr_seep=0.10;
%surf_irr_switch=1
%k_contr=0.5 % Real number [0 1], 0: only irr. controller 1: only level controller, in between mixed
controller
%kalman_switch=0 % Integer [0 1], 0: deactivate Kalman, 1: activate Kalman filter
h_target=1350;
h_delta=5;
Q_overpump=0;
Q_irr_delta=3;

%% PID-irrigation controller parameters

kP_irr=5; % proportional
kI_irr=0.8; % integral
kD_irr=0; % differential
kN_irr=1;

%% PID-level controller parameters

kP_lev=0.2%1.5%2; % proportional
kI_lev=0.02%0.5; % integral
kD_lev=0; % differential
kN_lev=0;

%% State-space groundwater model parameters

A=[1];
b=[dt/n_eff];
c=[1];
d=[0];

%% Kalman filter parameters

Q_kalman=0.5; % process covariance w
R_kalman=0.5; % v
P_initial=1;
eta_meas_var=R_kalman; % measurement noise

%% Initial conditions
h_initial=1335;

%% Compute relevant numbers

alpha_river=Q_river_nat/(h_nat^2-h_river^2);
Q_river_expected=alpha_river*(h_target^2-h_river^2);
mean_net_gw_inflow=Q_surf_irr_mean*k_surf_irr_seep+Q_nat_rech_mean-
Q_river_expected;
Q_irr_target=mean_net_gw_inflow+Q_surf_irr_mean*(1-k_surf_irr_seep)+Q_overpump;

%% Generate time series
```

```

time=[1:1:duration]';
Inputseries_Q_nat_rech.time=time;
Inputseries_Q_surf_irr.time=time;
[Inputseries_Q_nat_rech.signals.values,Inputseries_Q_surf_irr.signals.values]=logAR1corr(duration,[Q_nat_rech_mean Q_surf_irr_mean],[Q_nat_rech_stddev Q_surf_irr_stddev],[sigma_Q_nat_rech sigma_Q_surf_irr],cross_corr,fix_seed);

h_nat_series(1:duration)=h_nat;

%% Run model
sim(sim_model);

%% Plot results
% Plot properties
font_size=8;
font_name='Helvetica';
legend_location='NorthEastOutside';
figure_title=['Scenario ' scenario ' ' version];

figure(1)
clf

% Plot level
subplot(3,1,1)

hold on
plot(h_target_signal.time,h_target_signal.signals.values,'-k')
plot(time,h_nat_series,'-k')
plot(h_true.time,h_true.signals.values,'-k')

if kalman_switch==1
    plot(h_meas.time,h_meas.signals.values,'r+', 'MarkerSize',2)
    plot(h_kal.time,h_kal.signals.values,'b')
    legend('h target      .','h natural','h true','h measured','h kalman','Location',legend_location)
else
    legend('h target      .','h natural','h true','Location',legend_location)
end

%plot(h_target_signal.time,h_target_signal.signals.values+h_delta,':r')
%plot(h_target_signal.time,h_target_signal.signals.values-h_delta,':r')

%ylim([1340 1380])
title(figure_title,'Fontname',font_name,'Fontsize',font_size)
xlabel('Time [years]','Fontname',font_name,'Fontsize',font_size)
ylabel('Level [m]','Fontname',font_name,'Fontsize',font_size)
set(gca,'Fontname',font_name,'Fontsize',font_size)
box on
grid
hold off

% Plot GW flows
subplot(3,1,2)
hold on
plot(Q_nat_rech.time,Q_nat_rech.signals.values,'Color',[0,0.5,0])
plot(Q_surf_irr_seep.time,Q_surf_irr_seep.signals.values,'m')

```

```

plot(Q_pump_irr.time,Q_pump_irr.signals.values,'b')
plot(Q_pump_seepage.time,Q_pump_seepage.signals.values,'-b')
plot(Q_river.time,Q_river.signals.values,'Color',[255/255,140/255,0])%'c')

xlabel('Time [years]','Fontname',font_name,'FontSize',font_size)
ylabel('Flow [m^3/year]','Fontname',font_name,'FontSize',font_size)
legend('Q nat.rech.','Q surf.irr.seep.','Q pump','Q pump irr. seep.','Q river','Location',legend_location)
set(gca,'Fontname',font_name,'FontSize',font_size)
box on
grid
hold off

% Plot irrigation flows
subplot(3,1,3)
hold on
plot(Q_irr_target.time,Q_irr_target.signals.values,'-k')
plot(Q_surf_irr_eff.time,Q_surf_irr_eff.signals.values,'m')
plot(Q_pump_irr_eff.time,Q_pump_irr_eff.signals.values,'b')
plot(Q_tot_irr.time,Q_tot_irr.signals.values,'r')
%plot(Q_irr_target.time,Q_irr_target.signals.values+Q_irr_delta,':k')
%plot(Q_irr_target.time,Q_irr_target.signals.values-Q_irr_delta,':k')

xlabel('Time [years]','Fontname',font_name,'FontSize',font_size)
ylabel('Flow [m^3/year]','Fontname',font_name,'FontSize',font_size)
ylim([0 1])
legend('Q irr.target','Q surf.irr.eff.','Q pump.irr.eff.','Q tot.irr.','Location',legend_location)
set(gca,'Fontname',font_name,'FontSize',font_size)
grid
box on
hold off

name=['Scenario_' num2str(scenario)];

saveas(gcf,name,'eps2c');

```

8.4 File "logMAR1.m"

```
function [X,Y]=logAR1corr(n_values,XY_mean,XY_stdev,sigma_XY,c,fix_seed)
```

```

%% Generate
if fix_seed==1
    s=rng;
end

eta = normrnd(0,1,n_values, 2);
R = [1 c; c 1]; %% // Our correlation matrix, taken from the article
eta = eta * chol(R);

for n=1:2

    %% Disturbance generator logAR1 Model

    %% Parameters

```



```

base=10;
n_values;

m=XY_mean(n);
v=XY_stdev(n)^2;

eta_Z=eta(:,n);
Z_mean=XY_mean(n);
Z_stdev=XY_stdev(n);
sigma_Z=sigma_XY(n);

Z_log_mean=log10((m^2)/sqrt(v+m^2));
Z_log_stdev=sqrt(log10(1+v/(m^2)));

%% Initialize
Z=Z_mean;

%% Generate crosscorrelated noise

for j=1:n_values-1

    Z(j+1,:)=base^(Z_log_mean+sigma_Z*(log10(Z(j,:))-Z_log_mean)+Z_log_stdev*sqrt(1-
sigma_Z^2)*eta_Z(j,:));
end

if fix_seed==1
    rng(s);
end

%% Correction to preserve mean exactly

Z_mean_eff=mean(Z);
d_mean=Z_mean_eff-Z_mean;
Z_corrected=Z-d_mean;
series(:,n)=Z_corrected;

end

X=series(:,1);
Y=series(:,2);

```

8.5 file "pareto_model_runner.m"

```

clear

% Written by Gianni Pedrazzini (pedrazzini.gianni@gmail.com) october 2013. This script evaluates
% the pareto-front around a series of different k-contr values

% Notes: make following parameter changes in the "model_sricpt.m" file
% before running this

% h_initial=1355;
% duration = 10000

```

```
% Reason is that we want to sample a longer period to get more accurate
% statistics and that we dont want to consider the oscillations due to
% adjustment to an initial offset.

model='model_script'
version= 'Version1.44'

scenario='S5'
pump_irr_switch=1
surf_irr_switch=1
kalman_switch=0

n=1

stepsize=0.05

for i=0:stepsize:1

k_conr=i
save_k_conr(n)=k_conr;
run(model)

h_true_extracted=h_true.signals.values;
h_true_extracted(1:30)=[];

Q_tot_irr_extracted=Q_tot_irr.signals.values;
Q_tot_irr_extracted(1:30)=[];

delta_h_max(n)=h_target-min(h_true_extracted);
delta_Q_tot_irr_max(n)=Q_irr_target-min(Q_tot_irr_extracted);
delta_Q_tot_irr_max_rel(n)=100*delta_Q_tot_irr_max(n)/Q_irr_target;

n=n+1

end

%% Plot

dx=0.3
dy=0
k_conr_label=num2str(save_k_conr)
k_conr_label_text=cellstr(k_conr_label)

font_size=8;
font_name='Helvetica';
legend_location='NorthEastOutside';
figure_title=['Scenario ' scenario ' ' version];

figure(2)
clf
hold on
scatter(delta_h_max,delta_Q_tot_irr_max_rel)
text(delta_h_max+dx,delta_Q_tot_irr_max_rel+dy,k_conr_label_text,'Fontname',font_name,
'Fontsize',font_size)
hold off

xlabel('Maximal drawdown [m]','Fontname',font_name,'Fontsize',font_size)
```

```
ylabel('Maximal relative irrigation deficit [%-of irrigation target]', 'Fontname', font_name, 'Fontsize', font_size)  
title('Pareto-front for different k-contr values')  
set(gca, 'Fontname', font_name, 'Fontsize', font_size)  
  
saveas(gcf, 'Pareto_plot', 'eps2c');
```