Scientific documentation of DHI.MPC's optimisation engine

Anne Katrine V. Falk, akf@dhigroup.com

September 20, 2019

Contents

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

Chapter 1

State-space model

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

1.1 One-step ahead prediction used to calculate k-step forecast

General state equation

$$x(i+1) = Ax(i) + B_u u(i) + B_d d(i)$$

The state equation takes the system one time step ahead, starting from the initial state $x(0)$. Using this equation recursively until time step k results in

$$\begin{aligned}
x(1) &= Ax(0) + B_u u(0) + B_d d(0) \\
\\
x(2) &= Ax(1) + B_u u(1) + B_d d(1) = A(Ax(0) + B_u u(0) + B_d d(0)) + B_u u(1) + B_d d(1) \\
&= A^2 x(0) + AB_u(0) + AB_d d(0) + B_u u(1) + B_d d(1) \\
\\
x(3) &= Ax(2) + B_u u(2) + B_d d(2) \\
&= A(A^2 x(0) + AB_u(0) + AB_d d(0) + B_u u(1) + B_d d(1)) + B_u u(2) + B_d d(2) \\
&= A^3 x(0) + A^2 B_u u(0) + A^2 B_d d(0) + AB_u u(1) + AB_d d(1) + B_u u(2) + B_d d(2) \\
\vdots &\quad \vdots \\
x(k) &= A^k x(0) + \\
&\quad A^{k-1} B_u u(0) + A^{k-2} B_u u(1) + \cdots + AB_u u(k-2) + B_u(k-1) + \\
&\quad A^{k-1} B_d d(0) + A^{k-2} B_d d(1) + \cdots + AB_d d(k-2) + B_d(k-1)
\end{aligned}$$

In matrix notation (valid for $k \geq 1$):

$$x(k) = A^k x(0) + \begin{bmatrix} A^{k-1}B_u & \cdots & AB_u & B_u \end{bmatrix} \begin{bmatrix} u(0) \\ \vdots \\ u(k-2) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} A^{k-1}B_d & \cdots & AB_d & B_d \end{bmatrix} \begin{bmatrix} d(0) \\ \vdots \\ d(k-2) \\ d(k-1) \end{bmatrix}$$

This is the state at time k. Now the mapping from state to output is carried out. The general output equation expresses the output $y(i)$ as a linear combination of the state $x(i)$ and the forcing $u(i)$.

$$y(i) = Cx(i) + Du(i)$$

In the present case, no dependency on $u(i)$ has been identified, and the output equation is reduced to

$$y(i) = Cx(i)$$

Inserting the state at time k in the output equation yields

$$y(k) = Cx(k) = CA^k x(0) + \begin{bmatrix} CA^{k-1}B_u & \cdots & CAB_u & CB_u \end{bmatrix} \begin{bmatrix} u(0) \\ \vdots \\ u(k-2) \\ u(k-1) \end{bmatrix} + \begin{bmatrix} CA^{k-1}B_d & \cdots & CAB_d & CB_d \end{bmatrix} \begin{bmatrix} d(0) \\ \vdots \\ d(k-2) \\ d(k-1) \end{bmatrix}$$

Finally all outputs from $i = 1$ to $i = k$ are collected. The vector of outputs at all time steps is termed $Y(k)$, $Y(k) = [y(1), \ldots, y(k-1), y(k)]^T$.

$$Y(k) = \underbrace{\begin{bmatrix} CA \\ \vdots \\ CA^k \end{bmatrix}}_{\mathcal{O}} x(0) + \underbrace{\begin{bmatrix} CB_u & 0 & \cdots & 0 \\ CAB_u & CB_u & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{k-1}B_u & CA^{k-2}B_u & \cdots & CB_u \end{bmatrix}}_{G_u} \underbrace{\begin{bmatrix} u(0) \\ \vdots \\ u(k-2) \\ u(k-1) \end{bmatrix}}_{U(k)} + \underbrace{\begin{bmatrix} CB_d & 0 & \cdots & 0 \\ CAB_d & CB_d & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{k-1}B_d & CA^{k-2}B_d & \cdots & CB_d \end{bmatrix}}_{G_d} \underbrace{\begin{bmatrix} d(0) \\ \vdots \\ d(k-2) \\ d(k-1) \end{bmatrix}}_{D(k)}$$

Note that $y(i)$, $u(i)$ and $d(i)$ are vectors themselves, i.e. $y(i) = [y_a(i), y_b(i), \cdots]^T$ contains all outputs from the system at time i, where subscript a, b referes to outputs from different model blocks.

> For a system of reaches and reservoirs, the outputs are levels or volumes in reservoirs and flows in reaches.The forcing/input/controllable variables are flows at control structures such as gates, pumps and valves. The disturbances are inflows to the system (from tributaries and runoff from rainfall), as well as extractions (irrigation diversions, leakage and evaporation). Even a small river network is likely to have a multitude of disturbances.

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

Chapter 2

Optimisation

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

2.1 Control introduction

The goal is to minimise the output's deviation from a specified reference during the forecast, by choosing "the best" values for the forcings. In a river network system the goal might be to stick to a level in a reservoir, or to obtain a specified flow in a reach. The means of influencing the reservoir levels and the flows, are the flows through regulators as gates and valves. The effect of choosing a specific flow for a regulator is quantified by the linear models. The future reference contains reference values for all time steps up to time k. Each of $r_1, \ldots, r_k$ are vectors themselves, i.e. r_i contains all reference values for the system at time step i, $r_i = [r_a(i), r_b(i), \ldots]^T$.

$$R = \begin{bmatrix} r_1 \\ \vdots \\ r_k \end{bmatrix}$$

The forecast of future outputs (from time $i = 1, \ldots, k$) is

$$Y(k) = \mathcal{O}x(0) + G_u U(k) + G_d D(k)$$

In short

$$Y = \mathcal{O}x_0 + G_u U + G_d D$$

The reference and the future outputs can be used to construct the objective function in various ways.

$$J = \|Y - R\|_Q^2 = \|\mathcal{O}x_0 + G_u U + G_d D - R\|_Q^2 \tag{2.1}$$

Eq. (2.1) is the weighted least squares. If Q is the identity matrix, this is just the ordinary least squares measure. The U that minimizes J in eq.2.1 is given by

$$U = -(G_u^T Q G_u)^{-1} G_u^T [\mathcal{O}x_0 + G_d D - R] \tag{2.2}$$

Eq. (2.1) is a simple example of how an objective function can be formulated from the reference and the future output. As no constraints have been formulated neither for the input (U) nor for the output (Y) then eq. (2.2) minimises J.

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

2.2 Quadratic Program

We start out by formulating the optimisation model for the system as a quadratic program. A quadratic program has the general form

$$\min\ J = \frac{1}{2}U^T HU + g^T U \tag{2.3}$$
$$FU \leq b \tag{2.4}$$

In the following sections we will treat development of the objective function and the constraints in separate sections. But, as we shall see later, it is not possible to keep them separated, because defining soft constraints ends up adding terms to the objective function.

2.2.1 Quadratic objective function

First the very basic objective function, which only tracks reference on the output is derived. In the following sections this basic objective function is augmented with terms for reducing the control activity and for tracking a reference on the input as well. We shall see that these additional terms are derived as augmentations to the expression for the basic objective function. Finally an expression for the general objective function, which can handle a combination of these augmentations, is presented. subsubsectionBasic objective function - track reference on output The basic expression for the quadratic objective function is

$$J = \|Y - R\|_Q^2 = (Y - R)^T Q(Y - R) \tag{2.5}$$

Inserting the predicted output, $Y = \mathcal{O}x_0 + G_u U + G_d D$, in eq. (2.5), we get

$$\begin{aligned}&(\mathcal{O}x_0 + G_u U + G_d D - R)^T Q(\mathcal{O}x_0 + G_u U + G_d D - R) = \\ &((\mathcal{O}x_0)^T + (G_u U)^T + (G_d D)^T - R^T)(Q\mathcal{O}x_0 + QG_u U + QG_d D - QR) =\end{aligned} \tag{2.6}$$

$$\begin{aligned}&(\mathcal{O}x_0)^T Q\mathcal{O}x_0 + (\mathcal{O}x_0)^T QG_u U + (\mathcal{O}x_0)^T QG_d D - (\mathcal{O}x_0)^T QR \\ &+(G_u U)^T Q\mathcal{O}x_0 + (G_u U)^T QG_u U + (G_u U)^T QG_d D - (G_u U)^T QR \\ &+(G_d D)^T Q\mathcal{O}x_0 + (G_d D)^T QG_u U + (G_d D)^T QG_d D - (G_d D)^T QR \\ &-R^T Q\mathcal{O}x_0 - R^T QG_u U - R^T QG_d D + R^T QR\end{aligned} \tag{2.7}$$

In eq. (2.7), the quadratic term is marked in red, the linear terms in green and the constant terms remain in black. QP solvers need the objective function specified as quadratic contribution and a linear contribution, typically in the form $J = \frac{1}{2}U^T HU + g^T U$. When minimising J, the constant terms do not matter. The quadratic term $(G_u U)^T QG_u U$ is manipulated to

$$U^T G_u^T QG_u U \tag{2.8}$$

Collecting the linear terms need a little more work.

$$\begin{aligned}&(\mathcal{O}x_0)^T QG_u U + (G_u U)^T Q\mathcal{O}x_0 + (G_u U)^T QG_d D - (G_u U)^T QR + (G_d D)^T QG_u U - R^T QG_u U \\ &= ((\mathcal{O}x_0)^T + (G_d D)^T Q - R^T)QG_u U + (G_u U)^T Q(\mathcal{O}x_0 + G_d D - R)\end{aligned} \tag{2.9}$$

Due to the symmetry of Q, the second term $(G_u U)^T Q(\mathcal{O}x_0 + G_d D - R)$ is equal to $(\mathcal{O}x_0)^T + (G_d D)^T - R^T)QG_u U$. (Is this the correct argument?), and the linear term is reduced to $2((\mathcal{O}x_0)^T + (G_d D)^T Q - R^T)QG_u U$. Finally the objective function reads

$$J = U^T G_u^T QG_u U + 2(\mathcal{O}x_0 + G_d D - R)^T QG_u U \tag{2.10}$$

When minimising the objective function, the factor 2 does not influence the result, i.e.

$$\min\ J = \min\ (\frac{1}{2}U^T \underbrace{G_u^T QG_u}_{H} U + \underbrace{(\mathcal{O}x_0 + G_d D - R)^T QG_u}_{g^T} U) \tag{2.11}$$

The quadratic and linear contributions to the objective function in the form $J = \frac{1}{2}U^T HU + g^T U$ can therefore be calculated as

$$H = G_u^T QG_u \tag{2.12}$$
$$g = ((\mathcal{O}x_0 + G_d D - R)^T QG_u)^T = G_u^T Q(\mathcal{O}x_0 + G_d D - R) \tag{2.13}$$

Observe that if Q is not time dependent, then H does not depend on time either. This means that H needs not be re-calculated each time a new optimisation starts. On the other hand, g contains time-dependent information: starting from the initial state x_0, over the disturbances D, and probably also the reference vector R.

Penalise the control inputs

One way to reduce the control activity is to put a penalty on the input. The objective function (J) is then expressed as

$$J = \|Y - R\|_Q^2 + \|U\|_\Sigma^2 \tag{2.14}$$

The first part of the objective function has already been calculated in eq. (2.10), so we just insert this in eq. (2.14) and collect quadratic and linear terms

$$\begin{aligned}J &= U^T G_u^T QG_u U + 2(\mathcal{O}x_0 + G_d D - R)^T QG_u U + U^T \Sigma U \\ &= U^T (G_u^T QG_u + \Sigma)U + 2(\mathcal{O}x_0 + G_d D - R)^T QG_u U\end{aligned} \tag{2.15}$$

which results in

$$H = G_u^T QG_u + \Sigma \tag{2.16}$$
$$g = G_u^T Q(\mathcal{O}x_0 + G_d D - R) \tag{2.17}$$

In other words: penalising the input adds an extra term to the quadratic part of the objective (namely Σ), whereas the linear part stays the same as in the basic objective function.

This formulation implicitely sets "zero" as reference for the control inputs, $J = \|Y - R\|_Q^2 + \|U\|_\Sigma^2 = \|Y - R\|_Q^2 + \|U - 0\|_\Sigma^2$

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

Track a reference for both outputs and inputs

The objective function for a system with reference tracking on both output and input. The input U is penalised for deviating from R_u, where R_u is a reference for the control action, for instance a desired flow through a turbine.

$$J = \|Y - R\|_Q^2 + \|U - R_u\|_\Sigma^2 \tag{2.18}$$
$$Y = \mathcal{O}x_0 + G_u U + G_d D \tag{2.19}$$

Expanding the term $\|U - R_u\|_\Sigma^2$ yields

$$\|U - R_u\|_\Sigma^2 = (U - R_u)^T \Sigma (U - R_u) = U^T \Sigma U - U^T \Sigma R_u - R_u^T \Sigma U + R_u^T \Sigma R_u \tag{2.20}$$

Adding quadratic and linear terms to the expressions for H and g results in

$$H = G_u^T Q G_u + \Sigma \tag{2.21}$$
$$g = G_u^T Q(\mathcal{O}x_0 + G_d D - R) - \Sigma R_u \tag{2.22}$$

Penalise the change in the control inputs

$$J = \|Y - R\|_Q^2 + \|\Delta U\|_{\Sigma_\Delta}^2 \tag{2.23}$$

We expand the second term using

$$U_k = \begin{bmatrix} \begin{bmatrix} u_a \\ u_b \\ \vdots \end{bmatrix}_0 \\ \begin{bmatrix} u_a \\ u_b \\ \vdots \end{bmatrix} \\ \begin{bmatrix} u_a \\ u_b \\ \vdots \end{bmatrix}_{k-1} \end{bmatrix} \quad U_{k-1} = \begin{bmatrix} \begin{bmatrix} u_a \\ u_b \\ \vdots \end{bmatrix}_{-1} \\ \begin{bmatrix} u_a \\ u_b \\ \vdots \end{bmatrix} \\ \begin{bmatrix} u_a \\ u_b \\ \vdots \end{bmatrix}_{k-2} \end{bmatrix} \tag{2.24}$$

$$\Delta U = U_k - U_{k-1} = \underbrace{\begin{bmatrix} \begin{bmatrix} -u_a \\ -u_b \\ \vdots \end{bmatrix}_{-1} \\ \begin{bmatrix} \vdots \\ 0 \\ \vdots \end{bmatrix} \\ \begin{bmatrix} \vdots \\ 0 \\ \vdots \end{bmatrix} \end{bmatrix}}_{ini} + \underbrace{\begin{bmatrix} 1 & \cdots & 0 & & & & & & \\ \vdots & \ddots & \vdots & & & & & & \\ 0 & \cdots & 1 & & & & & & \\ -1 & \cdots & 0 & 1 & \cdots & 0 & & & \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & & & \\ 0 & \cdots & -1 & 0 & \cdots & 1 & & & \\ & & & -1 & \cdots & 0 & 1 & \cdots & 0 \\ & & & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ & & & 0 & \cdots & -1 & 0 & \cdots & 1 \end{bmatrix}}_{G_{\Delta U}} \underbrace{\begin{bmatrix} \begin{bmatrix} u_a \\ u_b \\ \vdots \end{bmatrix}_0 \\ \begin{bmatrix} u_a \\ u_b \\ \vdots \end{bmatrix} \\ \begin{bmatrix} u_a \\ u_b \\ \vdots \end{bmatrix}_{k-1} \end{bmatrix}}_{U_k} \tag{2.25}$$

Inserting yields

$$\begin{aligned} &\|\Delta U\|_{\Sigma_\Delta}^2 = (\Delta U)^T \Sigma_\Delta \Delta U = (ini + G_{\Delta U} U_k)^T \Sigma_\Delta (ini + G_{\Delta U} U_k) \\ &= ini^T \Sigma_\Delta ini + ini^T \Sigma_\Delta G_{\Delta U} U_k + U_k^T G_{\Delta U}^T \Sigma_\Delta ini + (G_{\Delta U} U_k)^T \Sigma_\Delta (G_{\Delta U} U_k) \\ &= ini^T \Sigma_\Delta ini + ini^T \Sigma_\Delta G_{\Delta U} U_k + U_k^T G_{\Delta U}^T \Sigma_\Delta ini + U_k^T G_{\Delta U} \Sigma_\Delta G_{\Delta U}^T U_k \end{aligned} \tag{2.26}$$

Adding the quadratic and linear terms to the expressions for H and g results in

$$H = G_u^T Q G_u + G_{\Delta U}^T \Sigma_\Delta G_{\Delta U} \tag{2.27}$$
$$g = G_u^T Q(\mathcal{O}x_0 + G_d D - R) + G_{\Delta U}^T \Sigma_\Delta ini \tag{2.28}$$

General objective function

The objective functions in the previous sections are based on the assumption that a reference is specified for *all* outputs, and when augmenting with penalties on input or on input change, that a penalty is specified for *all* inputs. If no reference is specified for an output, we have to work around this by specifying zero weights in the Q matrix and by specifying dummy values in the parts of the reference vector that corresponds to this output. Likewise if we want to specify a penalty on input change for only a selection of the inputs, we have to specify zero weights for the remaining inputs, in order to make these give a zero contribution to the objective function.

For practical applications we would like to be able to specify penalties for deviation from output reference for selected outputs in combination with penalties for change in input as well as penalties for reference deviation on input for selected inputs. As we have seen in the previous sections, the different penalties produce well-defined augmentations to the basic objective function. We only need one more thing to construct a general objective function: Indicator matrices, which point out the selected outputs and inputs for each type of penalty.

2.2.2 Unconstrained optimisation

If no constraint are present (neither on input nor on output), the min J can be found by differentiating J with respect to U and setting $J' = 0$.

$$J = \frac{1}{2} U^T H U + g^T U \implies J' = \frac{1}{2} U^T (H^T + H) + g^T \implies J' = U^T H + g^T \text{as } H \text{ is symmetric} \tag{2.29}$$

Setting $J' = 0$ yields

$$J' = 0 \implies U^T H + g^T = 0 \implies U^T = -g^T H^{-1} \implies \tag{2.30}$$
$$U^T = -g^T H^{-1} \implies U = (-g^T H^{-1})^T \tag{2.31}$$

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

The U that minimises J finally reads

$$U = -H^{-1}g \tag{2.32}$$

Below is listed how this expression looks with H and g inserted for the different variations of the objective function:

$$J = \|Y - R\|_Q^2 \qquad : U = -(G_u^T Q G_u)^{-1} G_u^T Q(\mathcal{O}x_0 + G_d D - R) \tag{2.33}$$

$$J = \|Y - R\|_Q^2 + \|U\|_\Sigma^2 \qquad : U = -(G_u^T Q G_u + \Sigma)^{-1} G_u^T Q(\mathcal{O}x_0 + G_d D - R) \tag{2.34}$$

$$J = \|Y - R\|_Q^2 + \|U - R_u\|_\Sigma^2 \qquad : U = -(G_u^T Q G_u + \Sigma)^{-1}(G_u^T Q(\mathcal{O}x_0 + G_d D - R) - \Sigma R_u) \tag{2.35}$$

$$J = \|Y - R\|_Q^2 + \|\Delta U\|_{\Sigma_\Delta}^2 \qquad : U = -(G_u^T Q G_u + G_{\Delta U}^T \Sigma_\Delta G_{\Delta U})^{-1}(G_u^T Q(\mathcal{O}x_0 + G_d D - R) + G_{\Delta U}^T \Sigma_\Delta ini) \tag{2.36}$$

2.2.3 Constraints

The constraints equation $FU \leq b$ defines all constraints in the system, both on the input and the output

$$U_{min} \leq U \leq U_{max} \tag{2.37}$$

$$Y_{min} \leq Y \leq Y_{max} \tag{2.38}$$

Many QP solvers treat box constraints on the input separately, so in the following referring to "the constraint inequalities" means the inequalities stemming from constraints on the output. Inserting Y in the constraint inequalities gives

$$Y_{min} \leq \mathcal{O}x_0 + G_u U + G_d D \leq Y_{max} \iff \tag{2.39}$$

$$-\mathcal{O}x_0 - G_d D + Y_{min} \leq G_u U \leq -\mathcal{O}x_0 - G_d D + Y_{max} \iff \tag{2.40}$$

$$lhs \leq G_u U \leq rhs \tag{2.41}$$

where the form $lhs \leq G_u U \leq rhs$ is ready to be entered in the solver as box constraints. These constraints are *hard* and cannot be violated. *Soft* constraints allow a violation of the bound. This is formulated by adding slack variables, s, to the bound. The slack variables enter the models as decision variables, and if the optimiser chooses to violate a bound, this is penalised in the objective function.

$$FU \leq b + s \tag{2.42}$$

For *soft* constraints the box formulation can no longer be used, as the slack variables break the symmetry of the upper and lower bounds. Reformulating to inequalities that are all "less than or equal to" gives

$$\begin{bmatrix} G_u \\ -G_u \end{bmatrix} U \leq \begin{bmatrix} -\mathcal{O}x_0 - G_d D + Y_{max} \\ \mathcal{O}x_0 + G_d D - Y_{min} \end{bmatrix} \tag{2.43}$$

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

2.3 Ensemble Quadratic Program

This section describes how the quadratic program is augmented to model an ensemble approach where the disturbance forecasts are described by ensembles. This is a likely situation, as e.g. rainfall forecast increasingly are calculated as ensembles.

The augmented matrices are presented for the case where the ensemble contains three members. This is for the sake of easing the overview of the idea wihtout too many dots in the matrices.

The output equation patches up the outputs related to each member of the disturbance ensemble, i.e. the first "row" corresponds to the output obtained by using disturbance number 1 and so on. Note that the forcing U is not an ensemble - the final goal is to find the U which best balances the advantages and drawbacks of all realisations.

$$\tilde{Y} = \tilde{\mathcal{O}}x_0 + \tilde{G_u}U + \tilde{G_d}\tilde{d} \iff \tag{2.44}$$

$$\begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \begin{bmatrix} \mathcal{O} \\ \mathcal{O} \\ \mathcal{O} \end{bmatrix} x_0 + \begin{bmatrix} G_u \\ G_u \\ G_u \end{bmatrix} U + \begin{bmatrix} G_d & & \\ & G_d & \\ & & G_d \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} \tag{2.45}$$

The augmented reference $\tilde{R}$ and the augmented weight $\tilde{Q}$. The reference is just duplicated, whereas the weight (the penalty for deviating from the reference) is scaled by the probability of occurence of the respective ensemble member. The p's must sum up to one, $\sum p_i = 1$

$$\tilde{R} = \begin{bmatrix} R \\ R \\ R \end{bmatrix} \tag{2.46}$$

$$\tilde{Q} = \begin{bmatrix} p_1 Q & & \\ & p_2 Q & \\ & & p_3 Q \end{bmatrix} \tag{2.47}$$

The quadratic program to solve becomes

$$min \|\tilde{Y} - \tilde{R}\|^2_{\tilde{Q}} + \|U - R_u\|^2_{\Sigma} s.t. \tag{2.48}$$

$$\begin{bmatrix} Y_{min} \\ Y_{min} \\ Y_{min} \end{bmatrix} \leq \tilde{Y} \leq \begin{bmatrix} Y_{max} \\ Y_{max} \\ Y_{max} \end{bmatrix} \tag{2.49}$$

The quadratic objective $\tilde{H}$ becomes

$$\tilde{H} = 2(\tilde{G_u}^T \tilde{Q} \tilde{G_u} + \Sigma) \tag{2.50}$$

The term Σ relates to the penalty on the forcing deviating from its reference. This term is identical to the non-ensemble case. The term $\tilde{G_u}^T \tilde{Q} \tilde{G_u}$ relates to the penalty on the output deviating from the output reference, and this term is ensemble-augmented. Expanding this term yields

$$\tilde{G_u}^T \tilde{Q} \tilde{G_u} = \begin{bmatrix} G_u^T & G_u^T & G_u^T \end{bmatrix} \begin{bmatrix} p_1 Q & & \\ & p_2 Q & \\ & & p_3 Q \end{bmatrix} \begin{bmatrix} G_u \\ G_u \\ G_u \end{bmatrix} = G_u^T p_1 Q G_u + G_u^T p_2 Q G_u + G_u^T p_3 Q G_u = (p_1 + p_2 + p_3) G_u^T Q G_u = G_u^T Q G_u \tag{2.51}$$

This means that the contribution form $\tilde{G_u}^T \tilde{Q} \tilde{G_u}$ to the ensemble QP is identical to that of the ordinary QP, and thus in total that the quadratice objective is identical to that of the ordinary QP

$$\tilde{H} = H \tag{2.52}$$

2.3.1 Soft-constrained ensemble QP

Quadratic objective

The quadratic objective $\tilde{H}_{soft}$ for the soft-constrained ensemble QP becomes

$$\tilde{H}_{soft} = \left[\begin{array}{ccc|ccc} \tilde{H} & p_1 P_u & & & & \\ & & p_2 P_u & & & \\ & & & p_3 P_u & & \\ \hline & & & & p_1 P_l & \\ & & & & & p_2 P_l \\ & & & & & & p_3 P_l \end{array}\right] \tag{2.53}$$

where P_u is the matrix which contains the penalties for exceeding upper constraints, and P_l is the matrix which contains penalties for going below lower constraints.

The constraints matrix for the ordinary soft-constrained QP reads

$$Z_{h_{soft}} = \begin{bmatrix} Z_h & -I_u & \\ -Z_h & & -I_l \end{bmatrix} \tag{2.54}$$

where Z_h is the constraints matrix of the hard-constrained QP. For soft constraints Z_h is augmented by I_u and I_l which are index matrices that points out the upper and lower constraints which are allowed to be violated. Each column in the index matrices corresponds to a slack variable, which measures the exceedances. The slack variables enter the QP as extra decision variables .

In the ensemble QP, this augmentation pattern is followed, but now each ensemble member requires its own set of slack variables.

$$\tilde{Z}_{h_{soft}} = \left[\begin{array}{ccc|ccc} Z_h & -I_u & & & & \\ Z_h & & -I_u & & & \\ Z_h & & & -I_u & & \\ \hline -Z_h & & & & -I_l & \\ -Z_h & & & & & -I_l \\ -Z_h & & & & & & -I_l \end{array}\right] \tag{2.55}$$

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

Linear objective

The linear contribution to the objective function reads

$$\tilde{g} = 2(\tilde{G}_u^T \tilde{Q}(\tilde{\mathcal{O}} x_0 + \tilde{G}_d \tilde{d} - \tilde{R}) - \Sigma R_u) \tag{2.56}$$

The first term, $\tilde{G}_u^T \tilde{Q}(\tilde{\mathcal{O}} x_0 + \tilde{G}_d \tilde{d} - \tilde{R}$ relates to the output's deviation from the reference, and all matrices and vectors are "tilded" because they are the ensemble-augmented versions. The second term, ΣR_u relates to the forcing's deviation from the specified reference R_u this term is identical to the non-ensemble case.

Expanding the first term yields

$$\tilde{G}_u^T \tilde{Q}(\tilde{\mathcal{O}} x_0 + \tilde{G}_d \tilde{d} - \tilde{R} = \begin{bmatrix} G_u^T & G_u^T & G_u^T \end{bmatrix} \begin{bmatrix} p_1 Q & & \\ & p_2 Q & \\ & & p_3 Q \end{bmatrix} \begin{bmatrix} \mathcal{O} x_0 + G_d d_1 - R \\ \mathcal{O} x_0 + G_d d_2 - R \\ \mathcal{O} x_0 + G_d d_3 - R \end{bmatrix} = \begin{bmatrix} G_u^T & G_u^T & G_u^T \end{bmatrix} \begin{bmatrix} p_1 Q(\mathcal{O} x_0 + G_d d_1 - R) \\ p_2 Q(\mathcal{O} x_0 + G_d d_2 - R) \\ p_3 Q(\mathcal{O} x_0 + G_d d_3 - R) \end{bmatrix} \tag{2.57}$$

which finally is reduced to

$$G_u^T (\underbrace{p_1 Q(\mathcal{O} x_0 + G_d d_1 - R)}_{\text{1. realisation contribution}} + \underbrace{p_2 Q(\mathcal{O} x_0 + G_d d_2 - R)}_{\text{2. realisation contribution}} + \underbrace{p_3 Q(\mathcal{O} x_0 + G_d d_3 - R))}_{\text{3. realisation contribution}} \tag{2.58}$$

In short the contribution to the linear objective is obtained by summing the contributions from each ensemble member, each of them scaled by the probability. Finally the forcing deviation contribution is added. For ensemble size n the linear objective reads.

$$\tilde{g} = 2 \left(\sum_{i=1}^{i=n} p_i Q(\mathcal{O} x_0 + G_d d_i - R) - \Sigma R_u \right) \tag{2.59}$$

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

Chapter 3

Surrogate model blocks

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

3.1 Reach

3.1.1 Reach with no lateral flows

General state equation

$$x(i+1) = Ax(i) + B_u u(i) + B_d d(i)$$

Output equation

$$y(i) = Cx(i)$$

The outflow from the reach is modelled as a delay on the inflow. The delay is k_d, and is counted in time steps. At time i, the relation is:

$$Q_{out}(i) = Q_{in}(i - k_d)$$

At time $i+1$, the relation is

$$Q_{out}(i+1) = Q_{in}(i - k_d + 1) \tag{3.1}$$

The state vector must be augmented by the previous releases in order to model the delay.

$$x(i) = \begin{bmatrix} Q_{in}(i-k_d) \\ \text{previous} \\ \vdots \\ \vdots \\ \text{flows} \end{bmatrix} = \begin{bmatrix} Q_{in}(i-k_d) \\ Q_{in}(i-k_d+1) \\ \vdots \\ \vdots \\ Q_{in}(i-1) \end{bmatrix}$$

If the upstream inflow is a forcing (i.e. can be subject to control), then the system has no disturbances, and the full state space equation reads

$$x(i+1) = \begin{bmatrix} Q_{out}(i+1) \\ Q_{in}(i-k_d+2) \\ \vdots \\ \vdots \\ Q_{in}(i) \end{bmatrix} = \underbrace{\left[\begin{array}{c|cccc} 0 & 1 & \cdots & \cdots & 0 \\ \hline 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & 1 \\ 0 & 0 & \cdots & \cdots & 0 \end{array}\right]}_{A \text{ is } k_d * k_d} \underbrace{\begin{bmatrix} Q_{out}(i) \\ Q_{in}(i-k_d+1) \\ Q_{in}(i-k_d+2) \\ \vdots \\ Q_{in}(i-1) \end{bmatrix}}_{x(i)} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}}_{B_u} \underbrace{\begin{bmatrix} Q_{in}(i) \end{bmatrix}}_{u(i)} + \begin{bmatrix} \text{no disturbance contributions} \end{bmatrix} \tag{3.2}$$

The augmented parts of A and B_u match the extra elements in the state vector x. The augmented parts are visualised by separating lines in the matrices. The first equation in this matrix system corresponds to eq. (3.1).

The following equations are identities, $Q_{in}(i-k_d+2) = Q_{in}(i-k_d+2)$ to $Q_{in}(i) = Q_{in}(i)$. Note that all but the last of the identity equations stem from multiplying the previous flows in $x(k)$ by the augmented part of A. The last identity equation stems from multiplying the forcing ($u(i)$) by the "1" at the last position of the augmented B_u. In total, this is the mechanism that models the time delay of the flow.

The output from the reach model is the outflow, $Q_{out}(i+1)$. This is the first element in the state vector, and the task of the output equation is to point out this element.

$$y(i) = \overbrace{\begin{bmatrix} 1 & \underbrace{|0 \quad \dots \quad 0}_{k_d - 1} \end{bmatrix}}^{C \text{ is } 1 * k_d} x(i)$$

In eq. (3.2) we see that in general, the augmented part of A is a $(k_d - 1) \times (k_d - 1)$ matrix of all zeros with ones in the upper bi-diagonal. This general formulation works for $k_d \geq 3$, because in these cases the augmented part of A will be at least 2×2, and the bi-diagonal exists.

Below, the special cases are written out. $k_d = 3$ is the lowest "ordinary" creation of A's augmented part.

$$x(i+1) = \begin{bmatrix} Q_{out}(i+1) \\ Q_{in}(i-1) \\ Q_{in}(i) \end{bmatrix} = \underbrace{\left[\begin{array}{c|cc} 0 & 1 & \\ \hline & 0 & 1 \\ & & 0 \end{array}\right]}_{A} \underbrace{\begin{bmatrix} Q_{out}(i) \\ Q_{in}(i-2) \\ Q_{in}(i-1) \end{bmatrix}}_{x(i)} + \underbrace{\left[\begin{array}{c} 0 \\ \hline 0 \\ 1 \end{array}\right]}_{B_u} \underbrace{\begin{bmatrix} Q_{in}(i) \end{bmatrix}}_{u(i)}$$

$k_d = 2$. Remember not to create bi-diagonal "ones" in A's augmentation - the bi-diagonals do not exist!

$$x(i+1) = \begin{bmatrix} Q_{out}(i+1) \\ Q_{in}(i) \end{bmatrix} = \underbrace{\left[\begin{array}{c|c} 0 & 1 \\ \hline 0 & 0 \end{array}\right]}_{A} \underbrace{\begin{bmatrix} Q_{out}(i) \\ Q_{in}(i-1) \end{bmatrix}}_{x(i)} + \underbrace{\left[\begin{array}{c} 0 \\ \hline 1 \end{array}\right]}_{B_u} \underbrace{\begin{bmatrix} Q_{in}(i) \end{bmatrix}}_{u(i)}$$

$k_d = 1$ needs special treatment, because the state vector is not augmented by the previous flows. The one step time delay is "inherent nature" of the system where the outflow one time step ahead (i.e. at time $i+1$) is a function of the forcing at time i.

$$x(i+1) = \begin{bmatrix} Q_{out}(i+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 0 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} Q_{out}(i) \end{bmatrix}}_{x(i)} + \underbrace{\begin{bmatrix} 1 \end{bmatrix}}_{B_u} \underbrace{\begin{bmatrix} Q_{in}(i) \end{bmatrix}}_{u(i)}$$

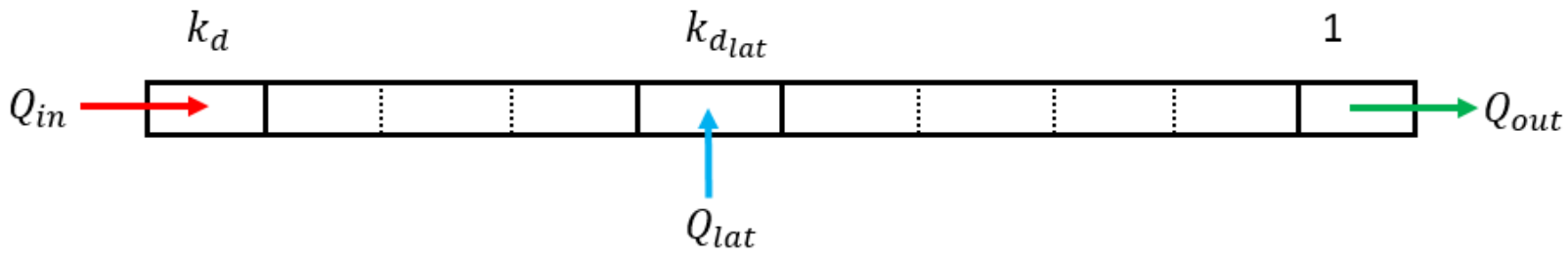
DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

3.1.2 Reach with one lateral flow (point source)

In this section the state space model for a reach with a controlled upstream inflow and a lateral flow along the reach is described.

The outflow at the end of the reach is the sum of the upstream inflow contribution and the lateral flow contribution, which is delayed less than the upstream inflow.



At time i, the relation is:

$$Q_{out}(i) = Q_{in}(i - k_d) + Q_{lat}(i - k_{d_{lat}})$$

At time $i + 1$, the relation is

$$Q_{out}(i+1) = Q_{in}(i - k_d + 1) + Q_{lat}(i - k_{d_{lat}} + 1)$$

General case - point inflow in any cell but the last

In the case of a reach with no lateral flows (described in sec. 3.1.1), the state vector was augmented by the previous upstream inflows. After the same template, the state vector is now augmented by the previous lateral flows as well. The full state space equation reads

$$x(i+1) = \begin{bmatrix} Q_{out}(i+1) \\ \hline Q_{in}(i-k_d+2) \\ \vdots \\ \vdots \\ Q_{in}(i) \\ \hline Q_{lat}(i-k_{d_{lat}}+2) \\ \vdots \\ \vdots \\ Q_{lat}(i) \end{bmatrix} = \underbrace{\left[\begin{array}{c|ccc|ccc} 0 & 1 & \cdots & 0 & 1 & \cdots & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & 1 & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \hline 0 & 0 & \cdots & 0 & 0 & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & 1 & \vdots & \ddots & 1 \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{array}\right]}_{A} \underbrace{\begin{bmatrix} Q_{out}(i) \\ \hline Q_{in}(i-k_d+1) \\ \vdots \\ \vdots \\ Q_{in}(i-1) \\ \hline Q_{lat}(i-k_{d_{lat}}+1) \\ \vdots \\ \vdots \\ Q_{lat}(i-1) \end{bmatrix}}_{x(i)} + \underbrace{\begin{bmatrix} 0 \\ \hline 0 \\ \vdots \\ \vdots \\ 1 \\ \hline 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}}_{B_u} \underbrace{[Q_{in}(i)]}_{u(i)} + \underbrace{\begin{bmatrix} 0 \\ \hline 0 \\ \vdots \\ \vdots \\ 0 \\ \hline 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}}_{B_d} \underbrace{[Q_{lat}(i)]}_{d(i)}$$

The A-matrix now contains two augmentation parts that occur as sub-matrices in the block diagonal: the first for delaying the upstream inflow and the second for delaying the lateral inflow.

The B_u-vector's contribution for the lateral flow is the augmentation by a zero vector. The B_d-matrix contains zeros in the first augmentation part (this augmentation relates to the delay of the controlled inflow, which is not a disturbance); the second augmentation part contains a "1" at the bottom element.

The output equation has two augmented parts as well.

$$y(i) = \underbrace{\left[\begin{array}{c|ccc|ccc} 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{array}\right]}_{C} x(i)$$

Special case - point inflow in last cell

If the point inflow is located in the last cell, the point inflow causes no state augmentation. The contribution from the point inflow occurs in B_d (or in B_u in case the inflow is manipulated by MPC).

$$x(i+1) = \begin{bmatrix} Q_{out}(i+1) \\ \hline Q_{in}(i-k_d+2) \\ \vdots \\ \vdots \\ Q_{in}(i) \end{bmatrix} = \underbrace{\left[\begin{array}{c|ccc} 0 & 1 & \cdots & 0 \\ \hline 0 & 0 & 1 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & 1 \\ 0 & 0 & \cdots & 0 \end{array}\right]}_{A} \underbrace{\begin{bmatrix} Q_{out}(i) \\ \hline Q_{in}(i-k_d+1) \\ \vdots \\ \vdots \\ Q_{in}(i-1) \end{bmatrix}}_{x(i)} + \underbrace{\begin{bmatrix} 0 \\ \hline 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}}_{B_u} \underbrace{[Q_{in}(i)]}_{u(i)} + \underbrace{\begin{bmatrix} 1 \\ \hline 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}}_{B_d} \underbrace{[Q_{lat}(i)]}_{d(i)}$$

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

3.1.3 Reach with one lateral flow (distributed)

In this section the state space model for a reach with a controlled upstream inflow and a lateral flow along the reach is described.
The outflow at the end of the reach is the sum of the upstream inflow contribution and the lateral flow contribution.

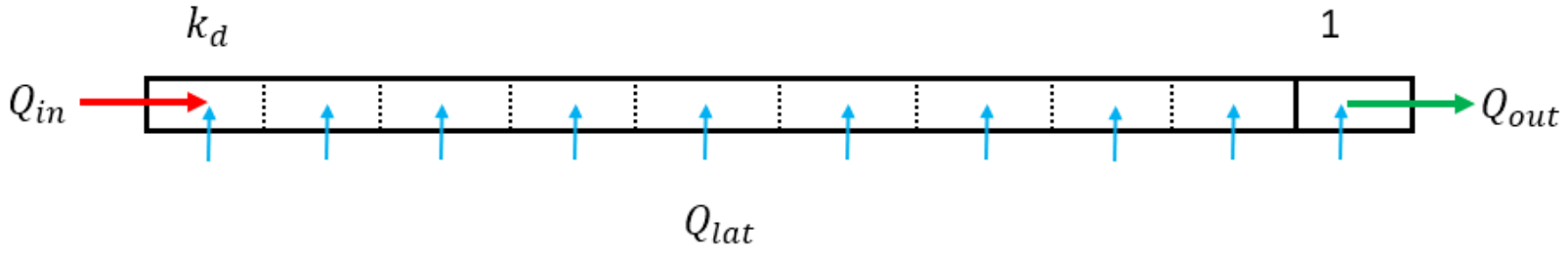


Figure 3.1: A reach with distributed lateral inflow.

At time i, the relation is:

$$Q_{out}(i) = Q_{in}(i-k_d) + \frac{1}{k_d}Q_{lat}(i-k_d) + \cdots + \frac{1}{k_d}Q_{lat}(i-2) + \frac{1}{k_d}Q_{lat}(i-1)$$

At time $i+1$, the relation is

$$Q_{out}(i+1) = Q_{in}(i-k_d+1) + \frac{1}{k_d}Q_{lat}(i-k_d+1) + \cdots + \frac{1}{k_d}Q_{lat}(i-1) + \frac{1}{k_d}Q_{lat}(i)$$

In this case, the state vector must be augmented by the previous lateral flows as well, after same template as for the upstream inflow. Fig. 3.1 shows how the delay time determines the discretisation of the reach: if the delay is k_d steps, this corresponds to k_d cells in the reach. Distributing the inflow along the entire reach corresponds to dividing the flow value equally between the cells in the reach, so each reach receives $1/k_d$. Inflows to all cells but the last one give a contribution in the first row of the A-matrix at the positions that corresponds to the augmented states from the lateral flow. Inflows to the last cell (delay = 1) is in effect a "non-delayed" disturbance (a delay of one time step is inherent in the model) and must occur in the first row of B_d (or of B_u in case of a controlled flow).

The full state space equation reads

$$x(i+1) = \left[\begin{array}{c} Q_{out}(i+1) \\ \hline Q_{in}(i-k_d+2) \\ \vdots \\ \vdots \\ Q_{in}(i) \\ \hline Q_{lat}(i-k_{d_{lat}}+2) \\ \vdots \\ \vdots \\ Q_{lat}(i) \end{array}\right] = \underbrace{\left[\begin{array}{c|cccc|ccccc} 0 & 1 & \cdots & \cdots & 0 & 1/k_d & 1/k_d & \cdots & 1/k_d \\ \hline 0 & 0 & 1 & \cdots & 0 & 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & & \vdots \\ \vdots & \vdots & & \ddots & 1 & \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & \cdots & 0 & 0 & \cdots & \cdots & 0 \\ \hline 0 & 0 & \cdots & \cdots & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & & \ddots & \vdots & \vdots & & \ddots & 1 \\ 0 & 0 & \cdots & \cdots & 0 & 0 & \cdots & \cdots & 0 \end{array}\right]}_{A} \underbrace{\left[\begin{array}{c} Q_{out}(i) \\ \hline Q_{in}(i-k_d+1) \\ \vdots \\ \vdots \\ Q_{in}(i-1) \\ \hline Q_{lat}(i-k_d+1) \\ \vdots \\ \vdots \\ Q_{lat}(i-1) \end{array}\right]}_{x(i)} + \underbrace{\left[\begin{array}{c} 0 \\ \hline 0 \\ \vdots \\ \vdots \\ 1 \\ \hline 0 \\ \vdots \\ \vdots \\ 0 \end{array}\right]}_{B_u} \underbrace{[Q_{in}(i)]}_{u(i)} + \underbrace{\left[\begin{array}{c} 1/k_d \\ \hline 0 \\ \vdots \\ \vdots \\ 0 \\ \hline 0 \\ \vdots \\ \vdots \\ 1 \end{array}\right]}_{B_d} \underbrace{[Q_{lat}(i)]}_{d(i)}$$

The output equation has two augmented parts as well.

$$y(i) = \underbrace{\left[\begin{array}{c|cccc|cccc} 1 & 0 & \cdots & \cdots & 0 & 0 & \cdots & \cdots & 0 \end{array}\right]}_{C} x(i)$$

Examples

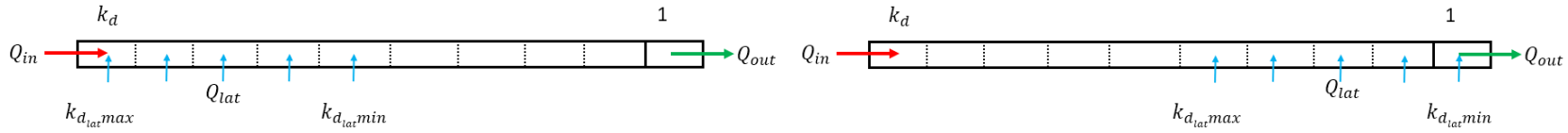


Figure 3.2: A reach with lateral inflow distributed to the upper half (left), and to the lower half (right).

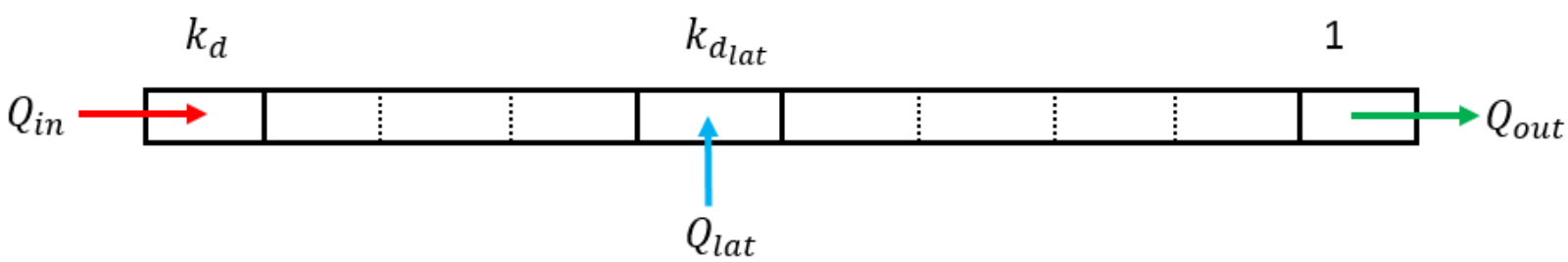
Imagine a reach with a delay of 10 time steps, $k_d = 10$. If a flow is distributed to the first half of the reach (the most upstream part), then A is augmented by a $(k_d-1)\times(k_d-1) = 9\times 9$ matrix. In the first row of A, the flow is distributed to the first 5 cells with equal weight (1/5).

If the flow is distributed to the second half of the reach (the most downstream part), the augmentation is a bit different. As the maximum delay of the distributed inflow is 5 (half of the total reach delay, which is $k_d = 10$, the augmented contribution to A will only be a $(5-1)\times(5-1) = 4\times 4$ matrix. Each of the 4 elements in the augmentations's first row is set to 1/5, and the contribution from distributing to the last cell (also 1/5) enters in the first row of B_d.

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

3.1.4 Reach - IDEA - total outflow is aggregated by the output equation



At time i, the relation is:

$$Q_{out}(i) = Q_{in}(i - k_d) + Q_{lat}(i - k_{d_{lat}})$$

At time $i + 1$, the relation is

$$Q_{out}(i + 1) = Q_{in}(i - k_d + 1) + Q_{lat}(i - k_{d_{lat}} + 1)$$

In the reach model presented in the previous sections, the outflow $Q_{out}(i)$ is modelled as a state variable of the reach block. Another model would be to include *only* the delayed flows in the state and make the output equation do the aggregation. The state space equation then reads:

$$x(i+1) = \begin{bmatrix} Q_{in}(i-k_d+1) \\ \vdots \\ \vdots \\ Q_{in}(i) \\ \hline Q_{lat}(i-k_{d_{lat}}+1) \\ \vdots \\ \vdots \\ Q_{lat}(i) \end{bmatrix} = \underbrace{\left[\begin{array}{cccc|cccc} 0 & 1 & \cdots & 0 & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & & \vdots \\ \vdots & & \ddots & 1 & \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & 0 & 0 & \cdots & \cdots & 0 \\ \hline 0 & \cdots & \cdots & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \ddots & & \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \vdots & \vdots & & \ddots & 1 \\ 0 & \cdots & \cdots & 0 & 0 & \cdots & \cdots & 0 \end{array}\right]}_{A} \underbrace{\begin{bmatrix} Q_{in}(i-k_d) \\ \vdots \\ \vdots \\ Q_{in}(i-1) \\ Q_{lat}(i-k_{d_{lat}}) \\ \vdots \\ \vdots \\ Q_{lat}(i-1) \end{bmatrix}}_{x(i)} + \underbrace{\begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 1 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}}_{B_u} \underbrace{\left[Q_{in}(i)\right]}_{u(i)} + \underbrace{\begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}}_{B_d} \underbrace{\left[Q_{lat}(i)\right]}_{d(i)}$$

And the output equation becomes

$$y(i) = \left[\begin{array}{cccc|cccc} 1 & \cdots & \cdots & 0 & 1 & \cdots & \cdots & 0 \end{array}\right] x(i) = Q_{in}(i - k_d) + Q_{lat}(i - k_{d_{lat}}) = Q_{out}(i)$$

Now, the question is: how to link the outflow from this reach to the inflow of another model block? The globally linked state space model inserts the links in the state space matrices, but now the outflow is no longer represented as a state. Can equality constraints be used?

$$Q_{b_{in}}(i) = Q_{a_{out}}(i) = y_a(i)$$

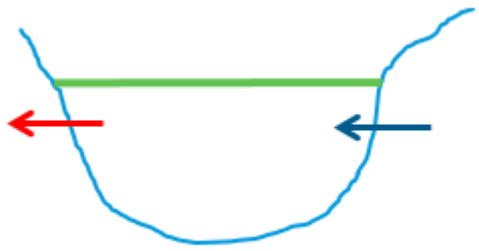
One advantage of this model is that A is now a block-diagonal matrix, where all subblokcs have similar structure, namely ones in the upper bi-diagonal. Potentially, this can be exploited to reduce the memory consumption and the computation time used for calculating the powers of A.

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

3.2 Physical reservoir

3.2.1 Reservoir - non-linear level-volume relation



The state space model for the reservoir calculates the volume difference over a time step. In case of a reservoir with one uncontrolled inflow (Q_{in}) and one controlled outflow (Q_{out}), the mass balance for a time step reads:

$$x(i+1) = [V(i+1)] = \underbrace{[1]}_{A}\underbrace{[V(i)]}_{x(i)} + \underbrace{[-\Delta t]}_{B_u}\underbrace{[Q_{out}(i)]}_{u(i)} + \underbrace{[\Delta t]}_{B_d}\underbrace{Q_{in}(i)}_{d(i)}$$

The output equation maps the only state variable (the volume) in a one-to-one relation.

$$y(i) = \underbrace{[1]}_{C} x(i)$$

Using the volume as state variable gives a linear relation between the forcing (controlled flow) and the state (volume), but the reservoir level also needs to be modelled, as initial values and target values will probably relate to the level rather than to the volume. In the general case, the reservoir level is a non-linear function of the volume. We deal with the non-linearity by mapping "initial level" to "initial volume" prior to the simulation; after the simulation, resulting volumes are mapped back to levels. Also, a target level specified for the MPC is mapped to a target volume in advance.

3.2.2 Tank - linear level-volume relation

In the special case where the level-volume relation is linear, the level can be used as state variable instead. This corresponds to viewing the reservoir as a tank with vertical sides; then a given volume difference will result in the same level difference, no matter what the initial level was.

$$x(i+1) = [h(i+1)] = \underbrace{[1]}_{A}\underbrace{[h(i)]}_{x(i)} + \underbrace{\left[-\frac{\Delta t}{area}\right]}_{B_u}\underbrace{[Q_{out}(i)]}_{u(i)} + \underbrace{\left[\frac{\Delta t}{area}\right]}_{B_d}\underbrace{Q_{in}(i)}_{d(i)} \tag{3.3}$$

The output equation maps the only state variable (the level) in a one-to-one relation.

$$y(i) = \underbrace{[1]}_{C} x(i)$$

Note that the parameter "*area*" entered in eq. 3.3. This parameter replaces the specification of the level-volume curve which is needed when using volume as state variable. Physically *area* can be viewed as the surface area of the tank; and this area will be constant as the tank has vertical sides. However, the vertical-side assumption will be an approximation in many cases, and the *area* will more act as a calibration parameter than a physically measurable quantity.

It has been considered to change the state variable of a tank to volume, because then there is no need to distinguish between the two reservoir approaches when linking to other blocks. However, this idea was discarded for two main reasons. First, an unnecessary conversion between level and volume is imposed in the simple tank model. Second, to do this level-volume conversion, then we have to choose an arbitary value for the initial volume if the bottom level of the reservoir is not known.

Both inflow and outflow controlled

What happens if both inflow and outflow is controlled? There will be no disturbance contribution, and the state space equation reads:

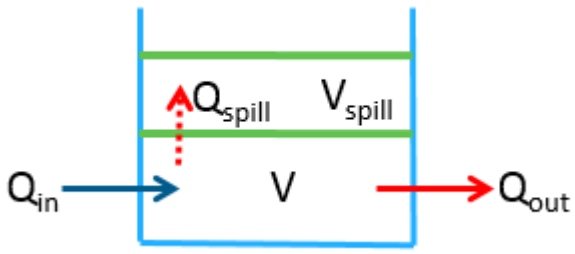
$$x(i+1) = [V(i+1)] = \underbrace{[1]}_{A}\underbrace{[V(i)]}_{x(i)} + \underbrace{[-\Delta t \quad +\Delta t]}_{B_u}\underbrace{\begin{bmatrix} Q_{out}(i) \\ Q_{in}(i) \end{bmatrix}}_{u(i)} + [0]$$

In the control problem, any pair of (Q_{out}, Q_{in}) that results in the desired volume difference would then be an optimal solution. However, if there is a penalty on the forcing, the optimal solution will be the one which minimises the control action, i.e. aims at as small flows as possible.

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

3.2.3 Reservoir with overflow



In this reservoir, the volume will never exceed the maximum volume. Instead all excess water is removed when the threshold is reached. This is a simple model of a reservoir with an overflow structure. Note that all excess water leaves the reservoir instantly. This makes this model block appropriate for modelling spill structures with occasional spills, but it cannot be used to model a outflow over a structure which depends on the water level above the weir crest.

In sewer applications this block can be used to model the sewer overflow from a "virtual basin", which represents the total volume of water in a specified area of the pipe network.

The challenge is to express the threshold (the volume at which overflow happens) in a linear state space model. The following formulation expresses the overflow discharge as a slack variable, which enters the optimisation model as an optimisation variable just as the controllable inflow/outflow to the reservoir. However, there are further restriction on the choice of slack variable values, because the overflow discharge is required to be zero as long as the reservoir volume is below maximum. Penalising the slack variable in the ordinary way (put deviation from zero into the objective function) is prone to activate the slack variable at a time where the volume threshold is not reached, if this will benefit the overall objective (i.e. if some larger penalties at other locations in the network can be avoided). This behaviour is circumvented by penalising the accumulated overflow volume instead. An overflow at an earlier time will weigh higher than an overflow at a later time, thus the optimisation has an incentive to postpone the overflow to a later time. We have no formal proof that this formulation will work in all cases, though.

The state space model for the reservoir calculates the volume difference over a time step. In case of a reservoir with one uncontrolled inflow (Q_{in}) and one controlled outflow (Q_{out}), the mass balance for a time step reads:

$$x(i+1) = \begin{bmatrix} V_{total}(i+1) \\ V_{spill}(i+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} V_{total}(i) \\ V_{spill}(i) \end{bmatrix}}_{x(i)} + \underbrace{\begin{bmatrix} -\Delta t & 0 \\ 0 & \Delta t \end{bmatrix}}_{B_u} \underbrace{\begin{bmatrix} Q_{out}(i) \\ Q_{spill}(i) \end{bmatrix}}_{u(i)} + \underbrace{\begin{bmatrix} \Delta t \\ 0 \end{bmatrix}}_{B_d} \underbrace{Q_{in}(i)}_{d(i)}$$

V_{total} is the sum of the reservoir volume (V) and the overflow volume (V_{spill}), $V_{total} = V + V_{spill}$. Note that the reservoir volume is not explicitly expressed in the above equation, but it is calculated as an output in the output equation, which reads

$$y(i) = \underbrace{\begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} V_{total}(i) \\ V_{spill}(i) \end{bmatrix}}_{x(i)} = \begin{bmatrix} V(i) \\ V_{spill}(i) \end{bmatrix}$$

The slack variable (Q_{spill}) cannot be negative, as the overflow discharge cannot "go back into the reservoir".

$$Q_{spill}(i) \geq 0, i = 1, \ldots, k$$

This automatically implies that the overflow volume (V_{spill}) is positive.

In order to make the reservoir overflow when a given threshold volume is reached, the volume (V) must be restricted by a hard output constraint

$$V(i) = V_{total}(i) - V_{spill}(i) \leq V_{max}, i = 1, \ldots, k$$

This output constraint corresponds to constraining the first row in the output equation. The second row in the output equation is the overflow volume (V_{spill}). As already mentioned, this is implicitly constrained to be non-negative by requiring $Q_{spill} \geq 0$.

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

3.3 Linear reservoir

The linear reservoir is not a physical reservoir, but represents a smoothing of the flow (which inherently also results in a delay). It is thought of as a very basic building block, which can be used for eaxampel in series with a reach block, so as to model the combination of transport time and storage in a channel. The characteristic of a linear reservoir is that the outflow is in ratio to the volume; and thus outflow and volume are interchangeable as state variable.

The volume in the next time step is a linear function of the volume in the current time step:

$$V(i+1) = V(i)k \text{ ,where } k < 1 \text{ describes a decay}$$

The solution to this difference equation is

$$V(i) = V(0)k^i = V(0)e^{\ln(k)i}, k < 1 \Rightarrow \ln(k) < 0$$

The volume leaving the reservoir in time step i is

$$\Delta V(i) = V(i+1) - V(i) = V(i) * k - V(i) = V(i)(k-1)$$

The half-life time,$t_{1/2}$, is the time it takes to reduce to the half volume. In the discrete case this must be counted in number of steps, i.e. $i_{1/2} = t_{1/2}/\Delta t$. If the half-life time ($t_{1/2}$) is specified, then k can be calculated:

$$V(i_{1/2}) = \frac{1}{2}V(0) \Leftrightarrow V(0)k^{i_{1/2}} = \frac{1}{2}V(0) \Leftrightarrow k^{i_{1/2}} = \frac{1}{2} \Leftrightarrow$$

$$k = e^{-\frac{\ln(2)}{i_{1/2}}} = e^{-\frac{\ln(2)}{t_{1/2}/\Delta t}}$$

The mean life time (or mean residence time), τ, is "the average time that the water spends in the reservoir". τ is related to the half-life time $t_{1/2}$ through

$$t_{1/2} = \tau \ln(2) \Leftrightarrow \tau = \frac{t_{1/2}}{\ln(2)}$$

The outflow in time step i relates to the volume by

$$Q(i) = -\frac{\Delta V(i)}{\Delta t} = -\frac{V(i)(k-1)}{\Delta t} = \frac{V(i)(1-k)}{\Delta t} = V(i)\beta$$

The above equation states that there is a linear relation between the outflow and the volume, and the flow-volume conversion coefficient β relates to k via

$$\beta = \frac{1-k}{\Delta t} = \frac{1 - e^{-\ln(2)/(t_{1/2}/\Delta t)}}{\Delta t}$$

In the general case, where inflows to the linear reservoir occurs, the volume balance is

$$\begin{aligned} V(i+1) &= V(i) + \Delta t(-Q_{out}(i) + Q_{in}(i)) \Leftrightarrow \\ V(i+1) &= V(i) + \Delta t(-V(i)\beta + Q_{in}(i)) \Leftrightarrow \\ V(i+1) &= V(i)\underbrace{(1-\beta\Delta t)}_{k} + \Delta t Q_{in}(i) \end{aligned}$$

Reformulated in terms of the outflow Q_{out} we get

$$\begin{aligned} \frac{1}{\beta}Q_{out}(i+1) &= \frac{1}{\beta}Q_{out}(i)(1-\beta\Delta t) + \Delta t Q_{in}(i) \Leftrightarrow \\ Q_{out}(i+1) &= Q_{out}(i)(1-\beta\Delta t) + \beta\Delta t Q_{in}(i) \\ Q_{out}(i+1) &= Q_{out}(i)k + Q_{in}(i)(1-k) \end{aligned}$$

When using a linear reservoir *model block* in an MPC setup, it is important to remember that the *outflow* cannot be a used as a decision variable, but inflows to the reservoir can be both controlled (i.e. decision variables) and uncontrolled (disturbances). For the state space model we split the inflows into controlled and uncontrolled inflows. Furthermore, either outflow *or* volume should be chosen as state variable (not both, if we know one, we know the other). We choose to use the outflow as state variable of the state model , and the state space matrices reads

$$x(i+1) = [Q_{out}(i+1)] = \underbrace{[k]}_{A}\underbrace{[Q_{out}(i)]}_{x(i)} + \underbrace{[1-k]}_{B_u}\underbrace{Q_{in}^{controlled}(i)}_{u(i)} + \underbrace{[1-k]}_{B_d}\underbrace{Q_{in}^{uncontrolled}(i)}_{d(i)}$$

The output equation just points out the outflow as the output

$$y(i) = \underbrace{[1]}_{C} x(i)$$

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

Chapter 4

Connecting model blocks

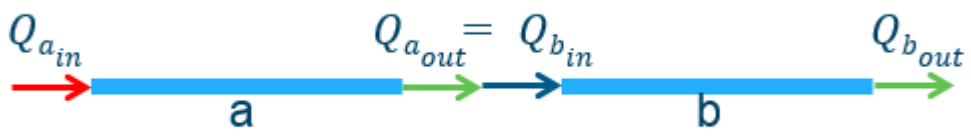
DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

4.1 Connecting flows in the state space model

4.1.1 Reach-reach (uncontrolled link)

Two reaches with no lateral inflows are connected. The outflow from the upstream reach (model block a) is piped into the inflow of the downstream reach (model block b).



The states of the two subsystems are collected in a single state vector, and A, B_u and B_d matrices are patched together of the matrices from the subsystems. The state equation reads:

$$x(i+1) = \begin{bmatrix} Q_{a_{out}}(i+1) \\ Q_{a_{in}}(i-k_{d_a}+2) \\ \vdots \\ \vdots \\ Q_{a_{in}}(i) \\ \hline Q_{b_{out}}(i+1) \\ Q_{b_{in}}(i-k_{d_b}+2) \\ \vdots \\ \vdots \\ Q_{b_{in}}(i) \end{bmatrix} = \underbrace{\left[\begin{array}{ccccc|ccccc} 0 & 1 & \cdots & \cdots & 0 & & & & & \\ 0 & 0 & 1 & \cdots & 0 & & & & & \\ \vdots & \vdots & \ddots & \ddots & \vdots & & & & & \\ \vdots & \vdots & & \ddots & 1 & & & & & \\ 0 & 0 & \cdots & \cdots & 0 & & & & & \\ \hline & 0 & \cdots & \cdots & 0 & 0 & 1 & \cdots & \cdots & 0 \\ & & & & & 0 & 0 & 1 & \cdots & 0 \\ & & & & & \vdots & \vdots & \ddots & \ddots & \vdots \\ & & & & & \vdots & \vdots & & \ddots & 1 \\ 1 & 0 & & & 0 & 0 & 0 & \cdots & \cdots & 0 \end{array}\right]}_{A} \underbrace{\begin{bmatrix} Q_{a_{out}}(i) \\ Q_{a_{in}}(i-k_{d_a}+1) \\ \vdots \\ \vdots \\ Q_{a_{in}}(i-1) \\ \hline Q_{b_{out}}(i) \\ Q_{b_{in}}(i-k_{d_b}+1) \\ \vdots \\ \vdots \\ Q_{b_{in}}(i-1) \end{bmatrix}}_{x(i)} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 1 \\ \hline 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{bmatrix}}_{B_u} \underbrace{[Q_{a_{in}}(i)]}_{u(i)}$$

Observe the red "1" in the lower left corner of the A-matrix. This is the link that pipes the outflow from a to the inflow of b; writing out the equation for the last row yields $Q_{b_{in}}(i) = Q_{a_{out}}(i)$. Not that the coupled system is a a system *without disturbances*. The inflow to model block b, which would induce a disturbance matrix B_d in the uncoupled case is eliminated, and $Q_{b_{in}}(i)$ instead receives its flow via the link.

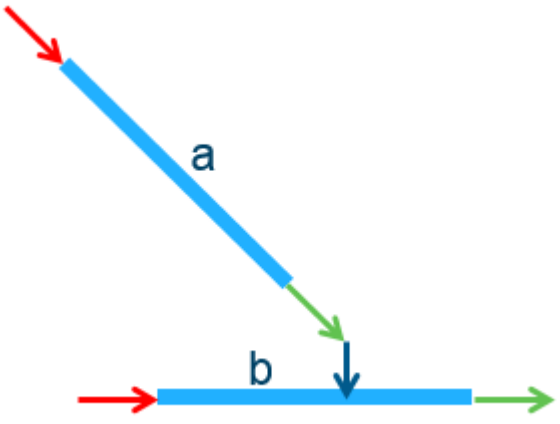
The *value* of the link is "1", because this is a link that sends one flow into another; thus there is a one-to-one correspondence.

This is an example of linking an uncontrolled outflow to an inflow. The resulting connection is uncontrolled. This has to be the case when passing water on from the outflow of a reach - at least when the reach is modelled as a delay model, which does not allow for storage of the water.

It central to place the linking "1" at the correct position in the patched-up A-matrix. The row number is equal to the position of $Q_{b_{in}}(i)$ in the the state vector, $x(i+1)$. The column number is equal to the position of $Q_{a_{out}}(i+1)$ in the state vector. This is still the case if the system contains lateral flows. Below is shown $x(i+1)$ and A for a system with two submodels (a and b), where the latter has a lateral inflow. The link element in A and the aforementioned positions in $x(i+1)$ are coloured red.

$$x(i+1) = \begin{bmatrix} \color{red}{Q_{a_{out}}(i+1)} \\ Q_{a_{in}}(i-k_{d_a}+2) \\ \vdots \\ \vdots \\ Q_{a_{in}}(i) \\ \hline Q_{b_{out}}(i+1) \\ Q_{b_{in}}(i-k_{d_b}+2) \\ \vdots \\ \vdots \\ \color{red}{Q_{b_{in}}(i)} \\ \hline Q_{b_{lat}}(i-k_{d_{b,lat}}+2) \\ \vdots \\ \vdots \\ Q_{b_{lat}}(i) \end{bmatrix}, A = \left[\begin{array}{ccccc|ccccccccc} 0 & 1 & \cdots & \cdots & 0 & & & & & & & & & \\ 0 & 0 & 1 & \cdots & 0 & & & & & & & & & \\ \vdots & \vdots & \ddots & \ddots & \vdots & & & & & & & & & \\ \vdots & \vdots & & \ddots & 1 & & & & & & & & & \\ 0 & 0 & \cdots & \cdots & 0 & & & & & & & & & \\ \hline & 0 & \cdots & \cdots & 0 & 0 & 1 & \cdots & \cdots & 0 & 1 & \cdots & \cdots & 0 \\ & & & & & 0 & 0 & 1 & \cdots & 0 & 0 & \cdots & \cdots & 0 \\ & & & & & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & & \vdots \\ & & & & & \vdots & \vdots & & \ddots & 1 & \vdots & & \ddots & \vdots \\ \color{red}{1} & 0 & & & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & \cdots & \cdots & 0 \\ \hline & & & & & & & & & & 0 & 1 & \cdots & 0 \\ & & & & & & & & & & \vdots & \ddots & \ddots & \vdots \\ & & & & & & & & & & \vdots & & \ddots & 1 \\ & & & & & & & & & & 0 & \cdots & \cdots & 0 \end{array}\right]$$

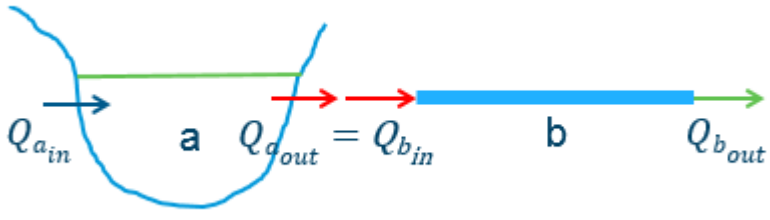
The outflow from reach a could be connected to the lateral inflow of reach b as well: in that case the row number of the link element must be shifted to match the row number of $Q_{b_{lat}}(i)$.



DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

4.1.2 Reservoir-reach (controlled link)



The controlled flow of a reservoir (model block a) is connected to the inflow of the downstream reach (model block b).

$$x(i+1) = \begin{bmatrix} V_a(i+1) \\ Q_{b_{out}}(i+1) \\ Q_{b_{in}}(i-k_{d_b}+2) \\ \vdots \\ \vdots \\ Q_{b_{in}}(i) \end{bmatrix} = \underbrace{\left[\begin{array}{c|cccc} 1 & & & & \\ \hline & 0 & 1 & \cdots & 0 \\ & 0 & 0 & 1 & \cdots & 0 \\ & \vdots & \vdots & \ddots & \ddots & \vdots \\ & \vdots & \vdots & & \ddots & 1 \\ & 0 & 0 & \cdots & \cdots & 0 \end{array}\right]}_{A} \underbrace{\begin{bmatrix} V_a(i) \\ Q_{b_{out}}(i) \\ Q_{b_{in}}(i-k_{d_b}+1) \\ \vdots \\ \vdots \\ Q_{b_{in}}(i-1) \end{bmatrix}}_{x(i)} + \underbrace{\begin{bmatrix} -\Delta t \\ 0 \\ 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}}_{B_u} \underbrace{\left[Q_{b_{in}}(i)\right]}_{u(i)} + \underbrace{\left[\begin{array}{c} \Delta t \\ \hline 0 \\ \vdots \\ \vdots \\ \vdots \\ 0 \end{array}\right]}_{B_d} \underbrace{\left[Q_{a_{in}}(i)\right]}_{d(i)}$$

Because the link is controlled, it occurs in the B_u-matrix (first element, coloured red). The value of the link, $-\Delta t$, reflects the conversion from flow to volume.

The patched-up output equation reads

$$y(i) = \begin{bmatrix} V(i) \\ Q_{b_{out}}(i) \end{bmatrix} = \left[\begin{array}{c|cccc} 1 & 0 & \cdots & \cdots & 0 \\ \hline 0 & 1 & \cdots & \cdots & 0 \end{array}\right] x(i)$$

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com

4.1.3 Linear reservoir-linear reservoir (uncontrolled link)

The upstream linear reservoir a pipes its outflow to the downstream linear reservoir b. Volume considerations

$$\begin{bmatrix} V_a(i+1) \\ V_b(i+1) \end{bmatrix} = \begin{bmatrix} k_a & 0 \\ 1-k_a & k_b \end{bmatrix} \begin{bmatrix} V_a(i) \\ V_b(i) \end{bmatrix} + \begin{bmatrix} \Delta t \\ 0 \end{bmatrix} Q_{a_{in}}(i) \tag{4.1}$$

Observe the red "$1-k_a$". This is the link that takes the volume leaving reservoir a, and adds it to the volume of reservoir b. Now, rewriting in terms of reservoir outflows, the last of the two equations in eq. 4.1 becomes

$$\begin{aligned}
V_b(i+1) &= (1-k_a)V_a(i) + k_b Vb(i) \Leftrightarrow \\
\frac{1}{\beta_b} Q_{b_{out}}(i+1) &= (1-k_a)\frac{1}{\beta_a} Q_{a_{out}}(i) + k_b \frac{1}{\beta_b} V_b(i) \Leftrightarrow \\
Q_{b_{out}}(i) &= \frac{1-k_a}{\beta_a} \beta_b Q_{a_{out}}(i) + k_b \frac{1}{\beta_b} Q_{b_{out}}(i) \Leftrightarrow \\
Q_{b_{out}}(i) &= \underbrace{\frac{1-k_a}{\beta_a}}_{\Delta t} \beta_b Q_{a_{out}}(i) + k_b Q_{b_{out}}(i) \Leftrightarrow \\
Q_{b_{out}}(i) &= \Delta t \beta_b Q_{a_{out}}(i) + k_b Q_{b_{out}}(i) \Leftrightarrow \\
Q_{b_{out}}(i) &= (1-k_b) Q_{a_{out}}(i) + k_b Q_{b_{out}}(i) \Leftrightarrow
\end{aligned}$$

When using outflow as state variable, the coupled system reads

$$\begin{bmatrix} Q_{a_{out}}(i+1) \\ Q_{b_{out}}(i+1) \end{bmatrix} = \begin{bmatrix} k_a & 0 \\ 1-k_b & k_b \end{bmatrix} \begin{bmatrix} Q_{a_{out}}(i) \\ Q_{b_{out}}(i) \end{bmatrix} + \begin{bmatrix} 1-k_a \\ 0 \end{bmatrix} Q_{a_{in}}(i)$$

DHI A/S, Agern Allé 5
2970 Hørsholm, Denmark

www.dhigroup.com