

# **AQUASIM 2.0 – User Manual**

## **Computer Program for the Identification and Simulation of Aquatic Systems**

**Peter Reichert**

Swiss Federal Institute for Environmental  
Science and Technology (EAWAG)  
CH - 8600 Dübendorf  
Switzerland

September 1998

ISBN: 3-906484-16-5

# Preface

The ideas for the realization of the program AQUASIM described in this manual grew from the experiences made in a lot of interdisciplinary studies at the Swiss Federal Institute for Environmental Science and Technology (EAWAG), CH-8600 Dübendorf, Switzerland, in which I have been involved. It is not possible to mention all persons, who contributed with the discussion of their data interpretation and modelling problems to the concepts of this program.

By far the largest influence to the concepts of this program are due to Oskar Wanner and Jürg Ruchti. The large number of common data analysis and parameter estimation projects with Oskar Wanner let us recognise the usefulness of a more universal program than those available at that time. Jürg Ruchti raised my interest in object-oriented programming and for the programming language C++ that was used for the implementation. The discussions with him significantly improved the design of the program. The realization of the program BIOSIM specifically designed for biofilm modelling together with Oskar Wanner and Jürg Ruchti had also an important influence on this project. AQUASIM includes the functionality of BIOSIM as a special case.

Version 1.0 of the AQUASIM was documented in a technical report that contained information on modeling in general, on the selection of program tasks, on numerical algorithms, on object-oriented implementation concepts and on examples of program application (Reichert, 1994b). The user manual with a brief tutorial was given in the appendix of this report. Because of the addition of a new variable type (probe variable), of several new compartments (advective-diffusive reactor, saturated soil column, lake), significant extensions of the biofilm reactor compartment, and several new features for simulation and batch processing, this user manual got out of date. In addition, a new user interface for the most widely used platform (Microsoft Windows), made the use of the program more comfortable. Because most users are only interested in the use of the program and not in the implementation concepts, I decided to write a new user manual and, as a separate volume, a new, more attractive tutorial (Reichert, 1998). In this new user manual the equations solved by the program are given in the same chapter as the program use is described. This should facilitate the understanding of what the program does. For persons interested in numerical methods or in the implementation concepts, the technical report is still the most complete source of information. In addition, a brief description of the major program features (Reichert, 1994a) and a summary of the implementation concepts (Reichert, 1995) are also available.

I would like to thank Jürg Ruchti for the implementation of the formula variables and the plotting facilities, and Werner Simon for the realization of the saturated soil column compartment. Furthermore, Oskar Wanner contributed to the design of the extensions of the biofilm compartment, Claudia Fesch and Stefan Haderlein to the design of the

soil column compartment, and Gerrit Goudsmit and Johny Wüest to the design of the lake compartment. Gérard Mohler, Bouziane Outiti and Raoul Schaffner gave support in solving technical problems involving different hardware platforms, and Martin Omlin introduced me into the L<sup>A</sup>T<sub>E</sub>X system used for typesetting this manuscript. In addition, many program users gave hints on program errors and on possibilities for improvements.

Most parts of this manual have been newly written. I apologize for all errors that it may contain. If you detect errors or unclear paragraphs, please send a note to **reichert@eawag.ch**. I will improve this manual during the next years and try to eliminate as many errors as possible.

Peter Reichert, September 1998

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>File Handling</b>	<b>7</b>
<b>3</b>	<b>Model Formulation</b>	<b>9</b>
3.1	Variables . . . . .	12
3.1.1	State Variables . . . . .	14
3.1.2	Program Variables . . . . .	15
3.1.3	Constant Variables . . . . .	17
3.1.4	Real List Variables . . . . .	18
3.1.5	Variable List Variables . . . . .	21
3.1.6	Formula Variables . . . . .	23
3.1.7	Probe Variables . . . . .	26
3.2	Processes . . . . .	27
3.2.1	Dynamic Processes . . . . .	28
3.2.2	Equilibrium Processes . . . . .	30
3.3	Compartments . . . . .	32
3.3.1	Mixed Reactor Compartment . . . . .	34
3.3.2	Biofilm Reactor Compartment . . . . .	41
3.3.3	Advective-Diffusive Reactor Compartment . . . . .	59
3.3.4	Saturated Soil Column Compartment . . . . .	70
3.3.5	River Section Compartment . . . . .	85
3.3.6	Lake Compartment . . . . .	98
3.4	Links . . . . .	125
3.4.1	Advective Link . . . . .	126
3.4.2	Diffusive Link . . . . .	129
3.5	Numerical Parameters . . . . .	132
3.6	Deleting Calculated States . . . . .	135
<b>4</b>	<b>Simulation and Data Analysis</b>	<b>137</b>
4.1	Simulation . . . . .	138
4.2	Sensitivity Analysis . . . . .	144
4.3	Parameter Estimation . . . . .	151
<b>5</b>	<b>Visualization of Results</b>	<b>161</b>

<b>6</b>	<b>Appendix</b>	<b>171</b>
6.1	Character Interface Version . . . . .	171
6.2	Batch Version . . . . .	173
6.3	Troubleshooting . . . . .	176
6.3.1	Problem Self-Diagnosis . . . . .	177
6.3.2	Finding Help in the AQUASIM User Group . . . . .	187
6.3.3	Reporting Program Bugs and Suggestions for Improvements . . . . .	188

# Chapter 1

## Introduction

The program AQUASIM was designed for the identification and simulation of aquatic systems in the laboratory, in technical plants and in nature. This user manual describes the equations solved by this program, the methods of systems analysis that are available and program handling. An additional document contains a series of extensively documented tutorial examples (Reichert, 1998). A brief survey on the capabilities of version 1.0 of the program (Reichert, 1994a) as well as a description of program implementation techniques (Reichert, 1995) can be found in scientific journals. Finally, there exists an extensive technical report on all aspects of version 1.0 of the program (Reichert, 1994b). Examples of program applications can be found in the following publications:

- |                                      |  |
|--------------------------------------|--|
| Mixed reactor systems:               | (von Schulthess et al., 1994; Wild et al., 1994; Reichert et al., 1995; Siegrist et al., 1995; Wild et al., 1995; von Schulthess and Gujer, 1996; Uehlinger et al., 1996; Kuba et al., 1996; Murnleitner et al., 1997; Novack and Sigg, 1997; Glod et al., 1997a; Musvoto et al., 1997; Eisenbeis et al., 1997; Peeters et al., 1997; Glod et al., 1997b; Filipe and Daigger, 1997). |
| Biofilm systems:                     | (Wanner et al., 1994; Janning et al., 1995; Horn and Hempel, 1995; Wanner et al., 1995; Wanner, 1996; Wanner and Reichert, 1996; Reichert and Wanner, 1997; Vraný et al., 1997; Horn and Hempel, 1997; Mirpuri et al., 1997; Arcangeli and Arvin, 1997b; Arcangeli and Arvin, 1997a; Sanderson and Stewart, 1997; Suci et al., 1998; Beaudoin et al., 1997).                         |
| Advective-diffusive reactor systems: | (von Gunten et al., 1997).   |
| Saturated soil column systems:       | (Simon et al., 1997; Fesch et al., 1998b; Fesch et al., 1998a).  |

River systems:	(Londong et al., 1994; Albrecht et al., 1995; Jancarkova et al., 1997; Maryns and Bauwens, 1997).
Lake systems:	—.

An updated list of references of AQUASIM applications can be found at the EAWAG home page at <http://www.eawag.ch>.

## Program Design

Comparison of measurements with model calculations is the most important method of testing theories in the natural sciences. Most mathematical models of environmental systems consist of a set of nonlinear ordinary or partial differential equations. A computer program which solves these equations numerically is usually required for calculating model predictions. Most programs available for this purpose can be put into one of three categories: universal simulation software; environmental simulation programs; and system identification programs. Universal simulation software is very flexible with regard to model formulation, but it is difficult to use, especially for non-specialists. Environmental simulation programs are much easier to handle, but they usually implement a specific model selected by the designer of the program. This makes their use for the comparison of different models impossible. Finally, system identification programs provide important tools for model comparison and parameter estimation, but the class of models considered in these programs is in most cases restricted to linear or algebraic models, and models cannot be formulated in a way familiar to environmental scientists. Although the classification of simulation programs into these categories is not strict and there are also (a few) programs that cover tasks belonging to more than one of these categories, a universal identification and simulation program is not yet available. **The intention behind the design of the program AQUASIM was to provide a more universal identification and simulation tool for a class of aquatic systems important in the environmental sciences. An additional important program design criterion was user-friendliness, which was achieved not only by providing a graphical user interface, but also by utilizing a communication "language" familiar to environmental scientists. AQUASIM is extremely flexible in allowing the user to specify transformation processes, and, in addition to perform simulations for the user-specified model, it provides elementary methods for parameter identifiability analysis, for parameter estimation and for uncertainty analysis.** Version 1.0 of AQUASIM was developed in the years 1991-1994 in the Computer and Systems Sciences department of the Swiss Federal Institute for Environmental Science and Technology (EAWAG), CH-8600 Dübendorf, Switzerland, and it is maintained and extended since then. The program was designed mainly for internal use in research and teaching, but is also available to the public. Information on the newest developments of AQUASIM can be found at the EAWAG home page at <http://www.eawag.ch>.

## Program Tasks

AQUASIM is a program for the identification and simulation of aquatic systems. It performs the four tasks of

- simulation,

- identifiability analysis,
- parameter estimation,
- uncertainty analysis.

Due to the similarity of the mathematical techniques involved, identifiability and uncertainty analyses are combined to yield sensitivity analysis.

The first task of AQUASIM is to allow the user to perform model **simulations**. By comparing calculated results with measured data, such simulations reveal whether certain model assumptions are compatible with measured data. The existence of systematic deviations between calculations and measurements provides a hint that additional important processes may have to be considered, or corrections must be made in the way processes are formulated. AQUASIM allows the user to change model structure and parameter values easily.

AQUASIM's second task is to perform **sensitivity analyses** with respect to a set of selected variables. This feature allows the user to calculate linear sensitivity functions of arbitrary variables with respect to each of the parameters included in the analysis. These sensitivity functions help in assessing the identifiability of model parameters (identifiability analysis). Furthermore, the derivatives calculated in sensitivity analyses allow the user to estimate the uncertainty in any variable according to the linear error propagation formula. The calculation of the contribution of each parameter to the total uncertainty facilitates the detection of major sources of uncertainty (uncertainty analysis).

The third important task of AQUASIM is to perform **parameter estimations** automatically for a given model structure using measured data. This is not only important for obtaining neutral estimates of parameters, but is also a main prerequisite for efficiently comparing different models. Several calculations, each of them describing a single experiment with the possibility for several target variables, as well as universal and experiment-specific model parameters, can be combined to a single parameter estimation process. The quantitative measure of the deviation between model calculations and measurements, which is minimized by the parameter estimation algorithm, is useful for statistically assessing the adequacy of the model.

## User Interfaces

Three versions of the program AQUASIM with different user interfaces are provided. The **window interface version**, `aquasimw`, uses the machine's own graphical user interface. The use of this version is strongly recommended for editing models, for defining sensitivity analyses and parameter estimations, for specifying plot definitions, for performing short calculations and for viewing results. The second version is the **character interface version**, `aquasimc`. This version is intended for users having a simple terminal without graphical capabilities. It provides all features available in the window interface version except the capability of plotting results directly on the screen (listing results and preparing plots for printing, however, is also possible with this program version). The third version is the **batch version**, `aquasimb`, which is designed for submitting long calculations as batch jobs (it should be noted that simulations, and especially sensitivity analyses and parameter estimations, may require much computation time). This version allows the user to start a calculation for an AQUASIM system, defined with one of the interactive program versions, by specifying one simple command line. It is also possible to specify a series of AQUASIM jobs on a command file so that the consecutive execution of calculations together with



listing and plotting results can be combined to a single batch job. In the batch version of AQUASIM, models cannot be modified.

### Hardware Platforms and Operating Systems

AQUASIM is written in the standardized object oriented programming language C++. There is a strict separation between the core program and the different user interface layers. There exist two versions of the user interface layer for the window interface version of the program. The first uses a graphical user interface library, which is available for various hardware platforms and operating systems. This program design makes AQUASIM highly portable. A second implementation of the user interface version of the window interface version is specifically designed for the Microsoft Windows operating system. The character interface version and the batch version can be compiled on nearly any platform and operating system without the need of special libraries. Table 1.1 gives a survey on all currently supported computing platforms.

Hardware	Operating System	Window System	Program Version		
			w	c	b
Sun SparcStation	Solaris 2.x	OpenWindows	X	X	X
		Motif 1.2	X	X	X
IBM RS/6000	AIX 3.2.x	-	-	X	X
HP 700 series	HP-UX 9.x	-	-	X	X
DEC Alpha series	VMS 6.x	-	-	X	X
	DEC Unix	-	-	X	X
Intel 80486/Pentium	MS-DOS 5 or 6	Windows 3.1 (Win32s)	X	-	-
	Windows 95	native	X	X	X
	Windows NT	native	X	X	X
Apple Power Macintosh	MacOS 7.x	native	X	X	X

Table 1.1: Computing platforms supported by the current AQUASIM version.

### Organization of this Manual

Throughout this user manual it is assumed that the user is familiar with system specific handling of menus, windows, list boxes, buttons, etc.. The description mainly concentrates on the window interface version of the program. The functionality of the character interface version is the same with the exception that it is not possible to plot graphics to the screen with this program version. In section 6.1 a brief introduction to the character interface version is given. In section 6.2 it is shown how batch jobs can be submitted.

Figure 1.1 shows the main window of AQUASIM with the header, the menu bar and a button bar which facilitates the access to the most important menu commands (from left to right: **File**→New, **File**→Open, **File**→Save, **Edit**→Cut (inactive), **Edit**→Copy (inactive), **Edit**→Paste (inactive), **Edit**→System, **Edit**→Delete States, **Calc**→Simulation, **Calc**→Sensitivity Analysis, **Calc**→Parameter Estimation, **View**→Results, **View**→Close Dialogs). The commands in the four menus **File**, **Edit**, **Calc** and **View** are described in the chapters 2-5 of this user manual. The menu '**File**' (chapter 2) is used for saving, loading and printing the current AQUASIM system, which consists of

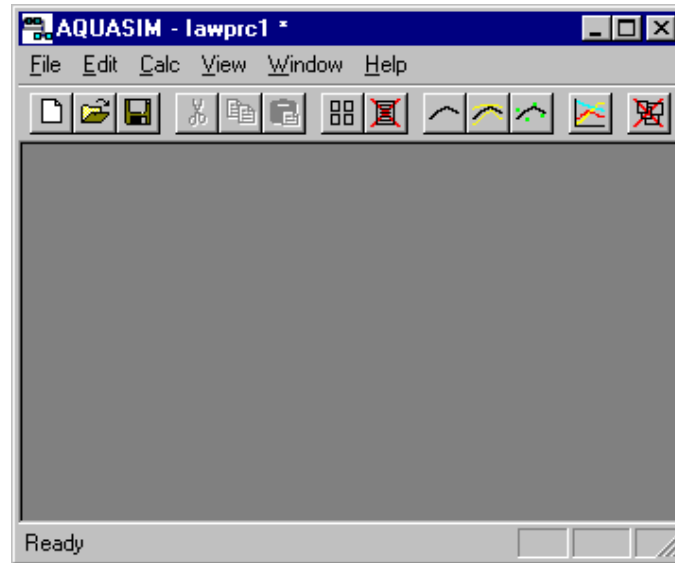


Figure 1.1: Main window of AQUASIM.

the mathematical model, measured data, definitions of sensitivity analyses and parameter estimations, plot definitions and calculated states. With the aid of the menu ‘**Edit**’ (chapter 3), the mathematical model and measured data can be entered and edited. The menu ‘**Calc**’ (chapter 4) is used to define and perform simulations, sensitivity analyses and parameter estimations. Finally, the menu ‘**View**’ (chapter 5) is used to specify plot definitions and to list and plot results. The appendix (chapter 6) contains additional information specific to the different program versions and some hints for troubleshooting.

### How to Proceed

The following procedure is recommended for learning to use the program AQUASIM:

1. Read the introduction to this manual (chapter 1) to obtain a general idea of program concepts and capabilities.
2. Skim over the chapters 2-5 to increase your knowledge of program concepts and to start learning to use the program interface. If you plan to work with the character interface version, read section 6.1 also.
3. Study the tutorial exercises described in a separate document (Reichert, 1998), and carefully read the corresponding sections of the chapters 2-5 of this user manual if you have problems in understanding the solutions.
4. Start using the program as a scientific and/or didactic tool. It may be helpful to implement the first model by modifying one of the example applications or one of the tutorial system files instead of starting from scratch. If problems arise, look at section 6.3 for hints on troubleshooting.



## Chapter 2

# File Handling

Fig. 2.1 shows the menu **File** of AQUASIM. The first 6 items of this menu are used to save



Figure 2.1: **File** menu.

and load AQUASIM systems, consisting of the user-specified mathematical model, measured data, definitions of sensitivity analyses and parameter estimations, plot definitions and calculated states. The item '**New**' is used to free memory from the current system, to allow the user to enter a new system. With the item '**Open**', a system previously stored with **Save** or **Save As** can be reloaded. Clicking the item '**Close**' results in deletion of the current system from memory, however, saved versions are not changed. The item '**Save**' is used to save a system, overwriting its old version on the disk. With the item '**Save As**' the current system can be saved under a new name. By clicking the item '**Revert to Saved**', the saved version of a system edited interactively can be reloaded. Note, that the items **Close**, **Save**, **Save As** and **Revert to Saved** are inactive, if no system has been opened or interactively entered. For a system interactively entered and not yet saved, the item **Save** results in the same operation as **Save As** and therefore allows to specify a file name. Before specification of a file name by loading or saving an AQUASIM system,

the item **Revert to Saved** is inactive. Furthermore, as long as a loaded system is not yet modified, the items **Save** and **Revert to Saved** are inactive. AQUASIM system files should not be edited with other programs, because such an attempt can result in inconsistent or unreadable files. AQUASIM system files can be transferred between all supported platforms using text (ASCII) data transfer. The file format is also compatible with electronic mail; mailed system files can directly be opened on any platform together with their mail headers. To keep a reasonable file size, it is recommended to delete calculated states before saving to an AQUASIM system file that is planned to be included in a mail message.

The menu item **Print Options** allows to select the print file format. As shown in Fig. 2.2, a long and a short format can be selected. In the long format, (nearly)

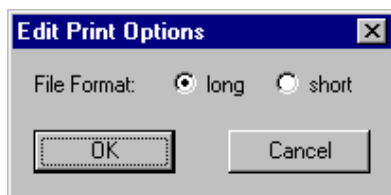


Figure 2.2: Dialog box for editing print options.

all user specifications are listed, whereas the short form allows to have a compact listing of the most essential model elements. To facilitate program portability, the menu item **Print to File** does not directly print the system definitions, but only writes them to a text file, the name of which can be specified by the user. Such text files have then to be submitted to a printer by the user either directly or after loading them into an editor or a text processing program. Printing an AQUASIM system is very useful, because the clear arrangement of all system definitions facilitates checking user input or understanding the meaning of objects loaded from an AQUASIM system file created by someone else. Note, however, that an AQUASIM system cannot be reloaded from a print file.

The menu item **About** gives information on the installed program version. This information is also written to any output file written by AQUASIM.

Finally, clicking the menu item **Exit** results in program termination. If system definitions have been edited or if states have been calculated or deleted, the user is asked to save the changes.

For each interactive AQUASIM session, a log file with the name `aquasim.log` is written to the startup directory of the program (in the batch version, the name of the log file can be specified by the user). This log file contains information on the progress of calculations. In the case of normal program termination, the log file can be ignored (and deleted); in the case of problems during calculation, the information provided in the log file may help locating the problem. Restarting AQUASIM in the same directory as before, leads to overwriting the old version of the log file, so that only the log files of the most recent sessions (in different directories) are available (unless an old log file has been renamed).

## Chapter 3

# Model Formulation

In the program AQUASIM, a model consists of a system of ordinary and/or partial differential equations and algebraic equations, which deterministically describes the behaviour of a given set of important state variables of an aquatic system. The differential equations for water flow and substance transport can be selected by the choice of environmental or technical compartments, which can be connected by links. The source terms of these equations, which describe the effect of transformation processes, can be freely specified by the user. The definition of such transformation processes follows closely the notation of biochemical processes, as it is familiar to environmental scientists and engineers. The definition of processes, compartments and links is done with the aid of variables, which represent objects taking a possibly context-sensitive numerical value. Fig. 3.1 visualizes the mutual dependencies between the four subsystems of variables, processes, compartments and links. It is evident, that the variables form the basic subsystem required for

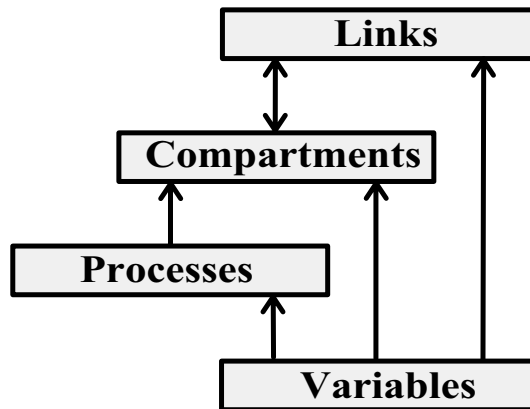


Figure 3.1: Main elements of model structure.

the formulation of processes, compartments and links. Processes must be defined before they can be activated in compartments. Finally, links can be used to connect compartments that are already defined. After a short overview of the four subsystems of the AQUASIM model structure, in this chapter, the definition of objects of these subsystems is explained in detail (sections 3.1 to 3.4). All these definitions together form the model

used by AQUASIM for simulation and data analysis.

The basic subsystem of the AQUASIM model structure is the system of **variables**. Variables are objects which are characterized by the property of taking a numerical value. This value may depend on the values of other variables. Six types of variables are distinguished: State variables are used to describe properties of water or of a surface in contact with water to be calculated by the model. Program variables make auxiliary quantities used in the program available to the system of variables. Constant variables and real list variables are used to provide measured quantities for use in the system of variables. In addition, constant variables are used as model parameters in sensitivity analyses and parameter estimations. Variable list variables and formula variables are used to build functional relations out of other variables. Finally, probe variables make the values of variables evaluated at a given location in a compartment globally available. The system of variables serves as a pool of variables for the formulation of the other subsystems.

The next subsystem of the AQUASIM model structure is the system of **processes**. Two types of processes are distinguished: Dynamic processes implement transformation or transfer processes, which are characterized by a common process rate and by individual stoichiometric coefficients describing the relative effect to different variables. Time evolution of variables affected by dynamic processes is determined by the solution of differential equations. The second type of processes are equilibrium processes, which determine the value of the corresponding variables by the solution of algebraic equations. Such processes are used to model processes which are so fast, that the corresponding variables can always be approximated to take their current equilibrium values. The variables of the system of variables may be used (and are needed) to formulate processes.

The next subsystem of the AQUASIM model structure is the system of **compartments**. This subsystem is designed to spatially divide the system under investigation. The following types of compartments are implemented in the current version of the program: Mixed reactor compartments are used to describe systems that can be approximated by an arrangement of well-mixed domains (e.g. stirred reactors, mixed lakes, etc.), biofilm reactor compartments are used to describe the growth and population dynamics of biofilms in which substrate gradients over depth are important, advective-diffusive reactor compartments can be used to describe systems with a longitudinal given water flow (e.g. plug flow reactors, rivers with given water flow, etc.), saturated soil column compartments are used to model advective-dispersive transport, exchange with stagnant pore volumes, adsorption and transformation of substances in saturated soil columns, river section compartments are used to describe hydraulics, transport and transformation processes in rivers, and lake compartments are used to model stratification, mixing, transport and transformation processes in horizontally well-mixed lakes.

The last subsystem of the AQUASIM model structure is the system of **links**. The objects of this subsystem are used to connect the compartments to the desired spatial configuration. To connect the compartments listed above, two types of links are distinguished: Advective links describe water flow and advective substance transport between compartments. These links can not only directly connect compartments, but also bifurcations and junctions can be built. Diffusive links model diffusive boundary layers or membranes between compartments. These elements can be diffusively penetrated by certain substances.

The menu `Edit` of AQUASIM shown in Fig. 3.2 is based on the model structure

described above. Each of the four menu items ‘**V**ariables’, ‘**P**rocesses’, ‘**C**ompart-

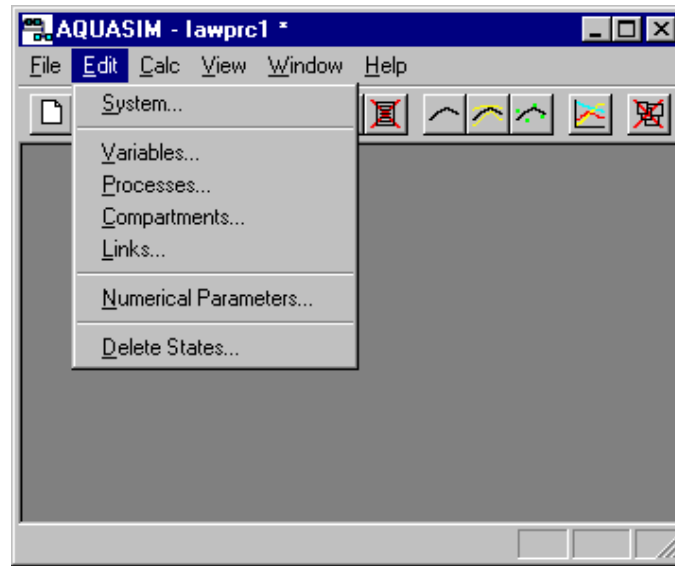


Figure 3.2: Edit menu.

**ments**’ and ‘**L**inks’ opens (or activates if it is already open) a modeless dialog box containing a list of objects already defined and control elements for defining new objects and for editing and deleting objects of the corresponding subsystem. If the screen is large enough, it is recommended to open all these dialog boxes together to accelerate editing the model (this can be done by selecting the ‘**S**ystem’ item in the **E**dit menu). The hierarchy of dialogs controlled by each of these dialog boxes is described in the following sections 3.1 to 3.4. As additional options, the **E**dit menu allows the user to change the values of ‘**N**umerical **P**arameters’ and to ‘**D**ele~~t~~e **S**tates’ calculated by the program. These options are described in the sections 3.5 and 3.6, respectively.



### 3.1 Variables

The basic objects for the formulation of models are variables. Variables are identified by their name. They are characterized by the property of taking a possibly context-sensitive numerical value. There are four main ranges of application of variables: Variables can be used for quantities to be determined by the model (e.g. by the solution of algebraic or differential equations) or which have a predefined meaning in a compartment (e.g. time and space coordinates), they can be used to provide data (e.g. model parameters or measured data series), they can be used to build functions depending on other variables (e.g. for the specification of process rates or stoichiometric coefficients) or they can be used as probes which make the values of other variables evaluated at a given location in a compartment globally available. According to these four categories, seven types of variables are distinguished:

- **State Variables** represent concentrations or other properties to be determined by a model according to user-selected transport and user-defined transformation processes.
- **Program Variables** make quantities such as time, space coordinates, discharge, etc. that are used for model formulation available as variables.
- **Constant Variables** describe single measured quantities that can also be used as parameters for sensitivity analyses or parameter estimations.
- **Real List Variables** are used to provide measured data or to formulate dependencies on other variables with the aid of interpolated data pairs.
- **Variable List Variables** are used to interpolate between other variables at given values of an arbitrary argument e.g. for multidimensional interpolation.
- **Formula Variables** allow the user to build new variables as algebraic expressions of other variables.
- **Probe Variables** make the values of other variables evaluated at a given location in a compartment globally available.

Figure 3.3 shows the dialog box used for editing variables. This dialog box is opened with the **Variables** command in the **Edit** menu shown in Figure 3.2. It is of modeless type in order to facilitate the editing process. The names of all variables already defined are listed alphabetically in the list box of this dialog box. The type of the currently selected variable is indicated at the bottom of the dialog box. The buttons of the dialog box allow the user to perform the following operations with variables: By clicking the button '**New**', new variables can be created from scratch. Alternatively, by clicking the button '**Duplicate**' the selected variable can be duplicated. With the button '**Edit**' or by double-clicking the variable name in the list box, a variable can be edited. The type of a variable can be changed by clicking the button '**Edit Type**'. During this procedure, type-specific data of the variable gets lost. Furthermore, by selecting two variables and clicking '**Exchange**', it is possible to exchange two variables in all other variables, processes, compartments, links and definitions of sensitivity analyses and parameter estimations, where they occur as arguments. This feature allows the user to quickly change models without losing data. Finally, the button '**Delete**' allows the program users to delete variables. Deletion of a variable is only possible, if the variable is not an argument of another variable, of a process, of a compartment, of a link or of a definition of a sensitivity analysis or of a parameter estimation. The buttons **Duplicate**, **Edit**, **Edit Type**, **Exchange** and **Delete** are inactive

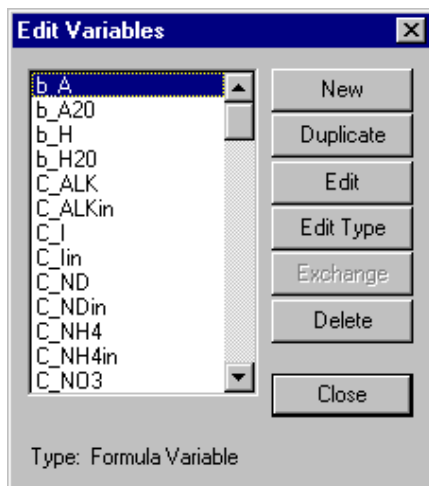


Figure 3.3: Dialog box for editing variables.

as long as no variable is selected. Clicking the '**Close**' button results in closing this dialog box. It can be reopened by clicking the **Variables** command in the **Edit** menu shown in Fig. 3.2.

The variables defined with the subdialogs to the dialog box shown in Fig. 3.3 serve as a pool of variables for use in other AQUASIM objects. A new variable may depend on any variables already defined (circular references are not allowed). It is important to define all necessary variables before starting to define an object of one of the other subsystems of processes, compartments and links.

After clicking one of the buttons **New** or **Edit Type** in the dialog box shown in Fig. 3.3 the variable type can be selected in the dialog box shown in Fig. 3.4. The seven types of

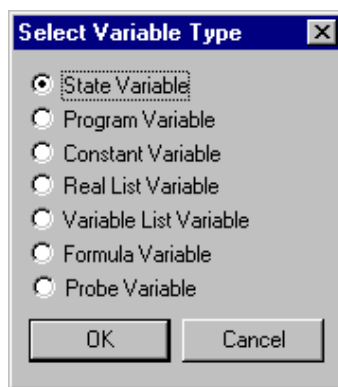


Figure 3.4: Dialog box for selecting the type of a variable.

variables shown in this selection box are described in more detail in the following seven subsections.

### 3.1.1 State Variables

State variables describe properties of water or of a surface in contact with water (e.g. temperature, masses or concentrations of dissolved or suspended substances or of substances attached to a surface). State variables obtain their meaning indirectly by the processes in which they are involved.

Figure 3.5 shows the dialog box used for defining or editing a state variable. As each

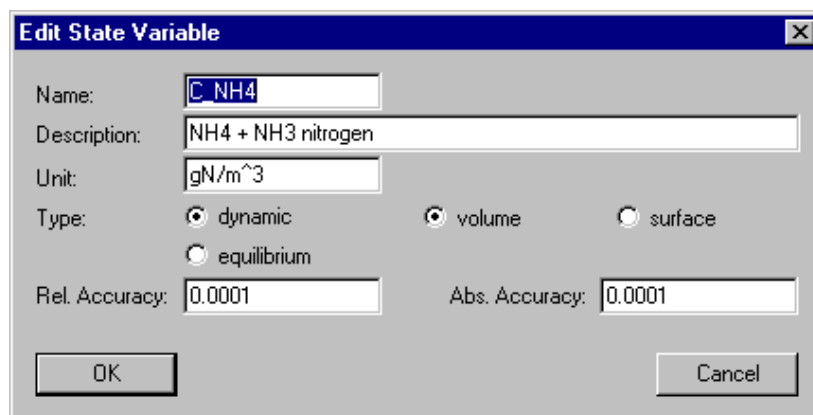


Figure 3.5: Dialog box for editing a state variable.

variable, a state variable needs a unique ‘**Name**’ as an identifier. A name of a variable consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. The following reserved names are not allowed as variable names: div, mod, and, or, not, if, then, else, endif, pi, sin, cos, tan, asin, acos, atan, sinh, cosh, tanh, deg, rad, exp, log, ln, log10, sign, abs, sqrt, min, max. To improve documentation of variables, a ‘**Description**’ and a ‘**Unit**’ can be specified. There are two main types of state variables: The values of ‘**dynamic**’ state variables are calculated as solutions to differential equations according to the transport processes determined by the choice of the compartment type and to the transformation rates specified by the user. ‘**equilibrium**’ state variables are used to describe quantities, the transformation processes of which are much faster than those of other variables, so that they can always be approximated to take the value corresponding to the current equilibrium state of their transformation processes. These equilibrium states depend on the values of the other variables and are given as the solution to algebraic equations provided by the user of the program. Dynamic state variables are further divided into dynamic ‘**volume**’ state variables and dynamic ‘**surface**’ state variables. Dynamic volume state variables are used to describe concentrations of substances transported with the water flow and quantified as mass per unit volume of water, whereas dynamic surface state variables are used to describe substances which are not transported with the water flow. Usually, this type of state variables is used to describe substances attached to a surface, which are quantified as total mass, as mass per unit length or as mass per unit of surface area (surface density). The distinction into volume and surface variables is not needed for equilibrium state variables. The edit fields ‘**Rel. Accuracy**’ and ‘**Abs. Accuracy**’ can be used to specify the precision of the numerical calculations. The integration algorithm uses the absolute accuracy plus the relative accuracy times the current value as an error criterion

to control the size of the time step. Therefore, not both of these accuracies are allowed to be zero, but pure absolute or pure relative error criteria are possible. It is important to specify reasonable values for these accuracies in order to obtain good behaviour of the integration algorithm. Good behaviour of the numerical algorithms is usually achieved, if the absolute accuracy and the product of a typical value of the state variable times the relative accuracy both are 4 to 6 orders of magnitude smaller than typical values of the state variable. See section 3.5 for more information on parameters of the numerical algorithms used in AQUASIM.

### 3.1.2 Program Variables

Program variables refer to predefined quantities of the modelled system. From a mathematical point of view, program variables can represent independent variables (time or space), parameters (calculation number, compartment index, etc.), or solutions to differential-algebraic systems of equations (discharge, reactor volume, etc.). The idea of program variables is to make the corresponding quantities, which anyway are present in model formulation, available for use in the system of variables. In some cases program variables can also be used to specify initial conditions within compartments (cf. section 3.3). Besides program variables for time and space coordinates and for compartment specific quantities, the set of program variables also includes a **Calculation Number**, which allows the user to distinguish different calculations.

Figure 3.6 shows the dialog box used for defining or editing a program variable. As

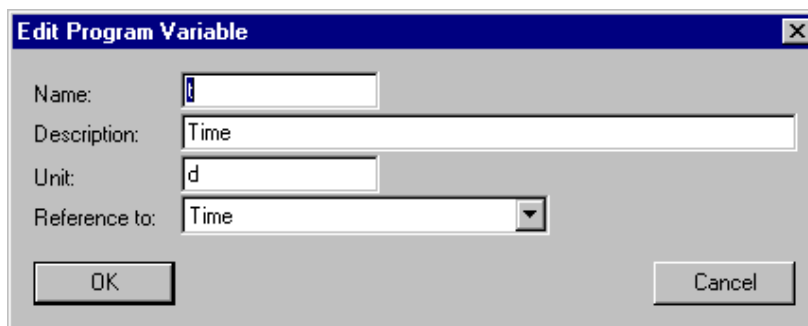


Figure 3.6: Dialog box for editing a program variable.

each variable, a program variable needs a unique **‘Name’** as an identifier. A name of a variable consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. The following reserved names are not allowed as variable names: div, mod, and, or, not, if, then, else, endif, pi, sin, cos, tan, asin, acos, atan, sinh, cosh, tanh, deg, rad, exp, log, ln, log10, sign, abs, sqrt, min, max. To improve documentation of variables, a **‘Description’** and a **‘Unit’** can be specified. With the aid of the list selection box **‘Reference to’** the user can select the quantity to which the program variable refers to. It is not possible to create more than one program variable referring to the same quantity. Table 3.1 lists the significance of all program variables considered in the current program version. Note that not all program variables are available in all compartments and links. In section 3.3 for each compartment a list of available program variables is given. The program variable **Calculation Number** is

a non-negative integer used for distinguishing different calculations (cf. section 4). The program variables **Compartment Index**, **Zone Index** and **Link Index** are used to make variables depend on compartments, zones within compartments and links. The values taken by the program variable **Zone Index** depends on the compartment, the values of the program variables **Compartment Index** and **Link Index** can be set in the dialog boxes for the definition of compartments and link (cf. sections 3.3 and 3.4). All other program variables have a physical meaning. Note that it depends on the compartment or link type, which program variables are defined. Program variables always return current values of the corresponding physical quantity as a function of simulation time and space coordinate within a compartment.

---

Calculation Number	Identifier for calculations (all compartments and links; value set in the dialog boxes shown in Figs. 4.5 and 4.17).
Time	Simulation time (all compartments and links).
Compartment Index	Identifier for compartments (all compartments; value set in the dialog boxes shown in Figs. 3.21, 3.29, 3.41, 3.53, 3.67 and 3.78).
Zone Index	Identifier for zones within compartments (all compartments; cf. section 3.3 for a description of possible values).
Link Index	Identifier for links (all links).
Discharge	Volumetric flow rate (all compartments, advective link).
Water Fraction	Volumetric fraction of water (all compartments; in some compartments always equal to 1).
Space Coordinate X	Space coordinate along the compartment (advective-diffusive reactor compartment, saturated soil column compartment, river section compartment).
Space Coordinate Z	Depth coordinate in the compartment (biofilm reactor compartment, lake compartment).
Reactor Volume	Total volume of the reactor (mixed reactor compartment, biofilm reactor compartment).
Bulk Volume	Volume of mixed water zone (mixed reactor compartment, biofilm reactor compartment).
Biofilm Thickness	Thickness of the biofilm (biofilm reactor compartment).
Growth Velocity of Biofilm	Advective velocity of biofilm solid matrix (biofilm reactor compartment).
Interface Velocity of Biofilm	Velocity of the interface between biofilm and bulk fluid (biofilm reactor compartment).
Detachment Velocity of Biofilm	Detachment velocity of particles from the biofilm surface (biofilm reactor compartment).
Attachment Velocity of Biofilm	Attachment velocity of particles onto the biofilm surface (biofilm reactor compartment).
Water Level Elevation	Elevation of water level above an absolute reference level (river section compartment).
Cross Sectional Area	Area of water body perpendicular to the flow direction (advective-diffusive compartment, saturated soil column compartment, river section compartment).
Perimeter Length	Length of the interface between water and the river bed perpendicular to the flow velocity (river section compartment).

Surface Width	Length of the interface between water and the atmosphere perpendicular to the flow velocity (river section compartment).
Friction Slope	Nondimensional friction force: Friction force divided by gravity force (river section compartment).
Density	Density of the water (lake compartment).
Area Gradient	Gradient of the cross-sectional area as a function of depth (lake compartment).
Brunt Vaisälä Frequency	Stability frequency of the water column (lake compartment).
Horizontal Velocity	Velocity of horizontal wind-driven flow (lake compartment).
Turbulent Kinitic Energy (TKE)	Turbulent kinetic energy (TKE) per unit mass of water in the lake (lake compartment).
Shear Production of TKE	Production of TKE due to shear of horizontal velocity (lake compartment).
Buoyancy Production of TKE	Production or loss of TKE due to density differences (lake compartment).
Dissipation	Dissipation of turbulent kinetic energy (lake compartment).
Energy of Seiche Oscillation	Total energy stored in Seiche motion (lake compartment).

---

Table 3.1: Significance of program variables.

### 3.1.3 Constant Variables

Constant variables can be used to describe single measured quantities consisting of a value and its accuracy characterized by the standard deviation. Alternatively, constant variables can be used as model parameters the values and standard deviations of which are estimated by the program. In this case the minimum and the maximum bound the legal range of values. For simulations, only the value of a constant variable is used. It remains constant during the simulation. The standard deviation together with the legal range is used during sensitivity analyses.

Figure 3.7 shows the dialog box used for defining or editing a constant variable. As each variable, a constant variable needs a unique **‘Name’** as an identifier. A name of a variable consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. The following reserved names are not allowed as variable names: div, mod, and, or, not, if, then, else, endif, pi, sin, cos, tan, asin, acos, atan, sinh, cosh, tanh, deg, rad, exp, log, ln, log10, sign, abs, sqrt, min, max. To improve documentation of variables, a **‘Description’** and a **‘Unit’** can be specified. The user has to specify the **‘Value’** of the variable which is used for simulations. In sensitivity analyses the **‘Standard Deviation’** is used to investigate the influence of uncertainty of model parameters to simulation results. The **‘Minimum’** and **‘Maximum’** bound the range of legal values. These bounds also hold for internal changes during sensitivity analyses and parameter estimations. For each constant variable, it can be decided, if it is **‘active for sensitivity analysis’** and if it is **‘active for parameter estimation’**.

These states can also be accessed with the aid of the dialog boxes shown in Figs. 4.11 and 4.16.

Figure 3.7: Dialog box for editing a constant variable.

### 3.1.4 Real List Variables

Quantities measured as a function of another variable, e.g. time series or spatial profiles, are represented by real list variables. For the definition of a real list variable a variable representing the argument, a list of argument-value data pairs, the standard deviations of the data and an interpolation method must be specified. The standard deviations can be given as global relative and absolute standard deviations or as individual standard deviations for all data values. Real list variables are usually evaluated as follows: In a first step, the variable given as the argument is evaluated. Then, the value of the variable is calculated by employing the selected interpolation method at the value of the argument as follows:

Linear interpolation:

If the value of the argument is smaller than the argument of the first list element, the value of the first list element is returned. If the value of the argument is larger than the value of the argument of the last list element, the value of the last list element is returned. If the value of the argument is within the range of the arguments of the list, the value on the connecting straight line between neighbouring data points corresponding to the argument is returned.

Cubic spline interpolation:

If the value of the argument is smaller than the argument of the first list element, the value of the first list element is returned. If the value of the argument is larger than the value of the argument of the last list element, the value of the last list element is returned. If the value of the argument is within the range of the arguments of the list, the interpolated value is given by cubic polynomials between neighbouring data points, which are determined by the conditions of continuous first and second derivatives at inner data points and by zero first derivative at

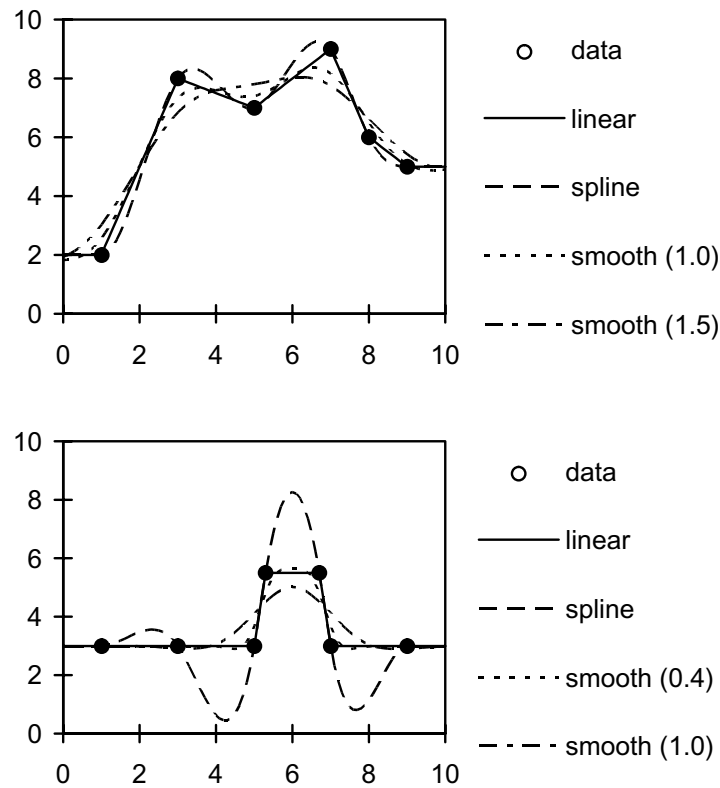


Figure 3.8: Comparison of interpolation and smoothing methods.

the end points.

Smoothing:

The values are defined by a curve smoothing the data points. This curve is given by the fit of a parabola through neighboring data points. For this fit, the data points are weighted with a normal distribution with a standard deviation chosen by the user (smoothing width) and centered at the actual value of the argument. The larger the width of this distribution, the smoother the behavior of the curve.

Fig. 3.8 shows interpolation of two real list variables with different interpolation and smoothing methods. Note that spline interpolation may lead to undesired oscillations in case of very abrupt changes in the data series.

An alternative use of real list variables is their use as target variables for parameter estimations. This is only possible, if the argument of the real list variable is either the program variable for time or the program variable describing the space coordinate of the compartment in which the comparison takes place. In this case, no interpolation is performed, but the variable to be compared with the real list variable is evaluated at the positions of the data pairs and the differences between the values of the two variables are



**Edit Real List Variable**

Name:

Description:

Unit:

Argument:

Std. Deviat.: ☒ global ☐ individual

Rel. Std. Dev.:  Abs. Std. Dev.:

Minimum:  Maximum:

Argument	Value	Std. Deviation
0	5.7	
1	5.7	
2.5	5.7	
5	5.6	
7.5	5.6	
10	5.6	
12.5	5.6	

Interpolation: ☒ linear ☐ spline ☐ smooth

☐ active for sensitivity analysis Smooth. Width:

OK Cancel

Figure 3.9: Dialog box for editing a real list variable.

summed up according to equation (4.13) using the standard deviations specified for the real list variable.

Figure 3.9 shows the dialog box used for defining or editing a real list variable. As each variable, a real list variable needs a unique ‘**Name**’ as an identifier. A name of a variable consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. The following reserved names are not allowed as variable names: div, mod, and, or, not, if, then, else, endif, pi, sin, cos, tan, asin, acos, atan, sinh, cosh, tanh, deg, rad, exp, log, ln, log10, sign, abs, sqrt, min, max. To improve documentation of variables, a ‘**Description**’ and a ‘**Unit**’ can be specified. The ‘**Argument**’ may be any other variable already defined. Its value is used to determine where to interpolate the list. Standard deviations can either be given as ‘**individual**’ standard deviations for all data values or as ‘**global**’ ‘**Relative Standard Deviations**’ and ‘**Absolute Standard Deviations**’. In the latter case, the standard deviation of a data value is calculated as the square root of the sum of the square of the absolute standard deviation plus the square of the product of the relative standard deviation times the current value of the variable. The absolute standard deviation may not be zero if some data elements of the list are zero. As for constant variables, the ‘**Minimum**’ and ‘**Maximum**’ bound the range of legal values. These bounds also hold for internal changes during sensitivity analyses. The list of data is built by elements consisting of a value of the argument, a value of the variable, and, in case of individual standard deviations, the

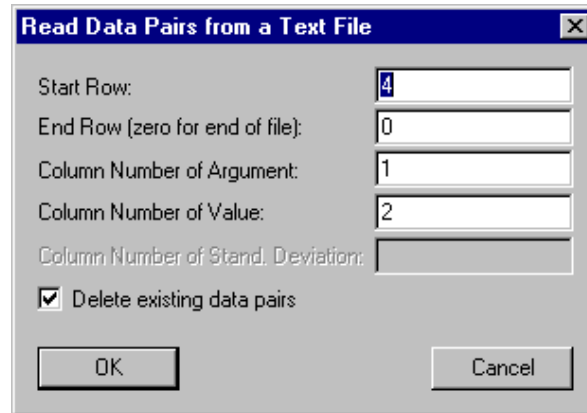


Figure 3.10: Dialog box for reading data pairs from a text file.

standard deviation of the value. The data pairs are sorted with increasing value of the argument. All data pairs have to differ in their arguments.

It is possible to **'Add'**, **'Replace'** and **'Delete'** data pairs, to **'Read'** them from text files (tab, space or comma delimited; missing lines and values as well as text columns allowed) and to **'Write'** them to text files.

The radio buttons **'linear'**, **'spline'** and **'smooth'** make it possible to select the interpolation technique.

For each real list variable, it can be decided, if it is **'active for sensitivity analysis'**

Fig. 3.10 shows the dialog box used for reading data pairs from a text file. This dialog box is opened by clicking the button **Read** in the dialog box shown in Fig. 3.9. The user can specify the data area by the **'Start Row'** and **'End Row'** and by the **'Column Number of Argument'**, the **'Column Number of Values'** and the **'Column Number of Standard Deviations'** (Standard deviation only if individual standard deviations are selected in the dialog box shown in Fig. 3.9). Furthermore, the user can choose to **'Delete existing data pairs'** or to add the data read from the file to the existing data.

### 3.1.5 Variable List Variables

Variable list variables are similar to real list variables, but instead of a value, another variable is given corresponding to each value of the argument. If these variables are variable list variables or real list variables, variable list variables can be used for multidimensional interpolation; if they are constant variables, parameter estimations of time series or of spatial profiles are possible.

Figure 3.11 shows the dialog box used for defining or editing a variable list variable. As each variable, a variable list variable needs a unique **'Name'** as an identifier. A name of a variable consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. The following reserved names are not allowed as variable names: **div**, **mod**, **and**, **or**, **not**, **if**, **then**, **else**, **endif**, **pi**, **sin**, **cos**, **tan**, **asin**, **acos**, **atan**, **sinh**, **cosh**, **tanh**, **deg**, **rad**, **exp**, **log**, **ln**, **log10**, **sign**, **abs**, **sqrt**, **min**, **max**. To improve documentation of variables, a **'Description'** and a **'Unit'** can be specified. Similarly to

**Edit Variable List Variable**

Name:

Description:

Unit:

Argument:

List of Data:

Argument	Value
33247	T_910109
33275	T_910206
33303	T_910306
33338	T_910410
33373	T_910515
33394	T_910605
33422	T_910703

Pairs: 12

Interpolation: ☒ linear ☐ spline ☐ smooth

Smooth. Width:

Figure 3.11: Dialog box for editing a variable list variable.

real list variables, an ‘**Argument**’ must be specified. It is possible to ‘**Add**’, ‘**Replace**’ and ‘**Delete**’ argument-variable pairs. The ‘**Interpolation Method**’ must also be selected. Look at the preceeding section on real list variables for a description of these interpolation techniques. Note that for variable list variables, spline interpolation and smoothing are not very efficient, because these methods need evaluation of all variables of the list for each evaluation of the variable list variable.

### 3.1.6 Formula Variables

Formula variables allow the user to build functional relations as algebraic expressions using previously defined variables (cyclic references are not allowed).

Figure 3.12 shows the dialog box used for defining or editing a formula variable. As

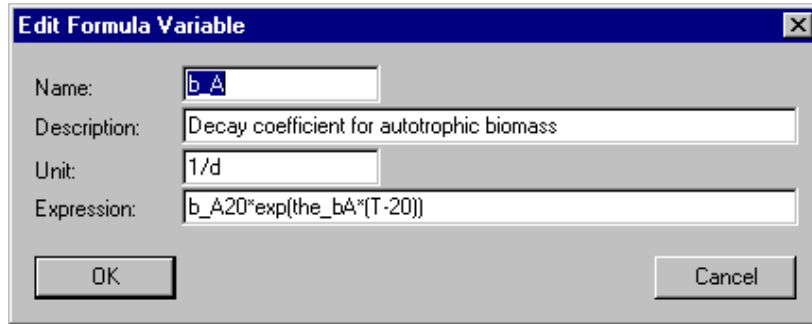


Figure 3.12: Dialog box for editing a formula variable.

each variable, a formula variable needs a unique ‘**Name**’ as an identifier. A name of a variable consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. The following reserved names are not allowed as variable names: div, mod, and, or, not, if, then, else, endif, pi, sin, cos, tan, asin, acos, atan, sinh, cosh, tanh, deg, rad, exp, log, ln, log10, sign, abs, sqrt, min, max. To improve documentation of variables, a ‘**Description**’ and a ‘**Unit**’ can be specified. An algebraic ‘**Expression**’ using the previously defined variables can be given to define the new variable. The formula syntax is given by an `<expression>` defined as given below, where `<varident>` must be the name of a variable already defined:

```

<varident>          = <letter> {<letter_or_digit>}
<letter>            = A | ... | Z | a | ... | z | _
<letter_or_digit>   = <letter> | <digit>
<digit>             = 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
<expression>        = <simple_expression> | <simple_expression> <relop>
                        <simple_expression>
<relop>             = = | # | <> | <= | < | >= | >
<simple_expression>  = <term> | <sign> <term> | <simple_expression> <addop>
                        <term>
<term>              = <factor> | <term> <mulop> <factor>
<sign>              = + | -
<addop>             = + | - | or
<factor>            = <varident> | <unsigned_constant> | (<expression>)
                        | <function> | not <factor>
<unsigned_constant> = <unsigned_number> | <predefined_constant>
<unsigned_number>   = <unsigned_integer> [.<unsigned_integer> [E [<sign>]
                        <unsigned_integer>]]

```

```

<predefined_constant> = pi
<unsigned_integer>    = <digit> { <digit> }
<mulop>               =  $\hat{}$  * | / | div | mod | and
<function>            = <arith_func_1_arg> | <arith_func_2_arg> | <if_function>
<arith_func_1_arg>    = <ident_func_1_arg> (<expression>)
<ident_func_1_arg>    = sin | cos | tan | asin | acos | atan | sinh |
                        cosh | tanh | deg | rad | exp | log | ln | log10
                        | sign | abs | sqrt
<arith_func_2_arg>    = <ident_func_2_arg> (<expression>, <expression>)
<ident_func_2_arg>    = min | max
<if_function>         = if <condition> then <expression> else <expression>
                        endif
<condition>           = <expression>

```

Note that this syntax makes it possible to specify algebraic expressions using variables, usual operations and elementary functions, and that even conditional branching with if-then-else-endif constructions is possible. The trigonometric functions use radians as the unit of the argument. Table 3.2 gives an overview on the functions and constants available in formula variables.

---

abs	Function with one argument returning the absolute value of the argument.
acos	Function with one argument returning the inverse of the cosinus function in units of radians evaluated at the argument.
asin	Function with one argument returning the inverse of the sinus function in units of radians evaluated at the argument.
atan	Function with one argument returning the inverse of the tangens function in units of radians evaluated at the argument.
cos	Function with one argument returning the cosinus function evaluated at the argument that must be given in radians.
cosh	Function with one argument returning the hyperbolic cosinus function evaluated at the argument that must be given in radians.
deg	Function with one argument returning the argument converted from radians to degrees.
exp	Function with one argument returning $e$ to the power of the argument.
ln	Function with one argument returning the natural logarithm of the argument.
log	Function with one argument returning the natural logarithm of the argument.
log10	Function with one argument returning the logarithm with base 10 of the argument.
max	Function with two arguments returning the maximum of the two arguments.
min	Function with two arguments returning the minimum of the two arguments.
pi	Constant returning the value of $\pi$ (=3.1415926...).
rad	Function with one argument returning the argument converted from degrees to radians.
sign	Function with one argument returning the sign of the argument (+1 for positive arguments, -1 for negative arguments, 0 for arguments equal to zero).
sin	Function with one argument returning the sinus function evaluated at the argument that must be given in radians.

<code>sinh</code>	Function with one argument returning the hyperbolic sinus function evaluated at the argument that must be given in radians.
<code>sqrt</code>	Function with one argument returning the square root of the argument.
<code>tan</code>	Function with one argument returning the tangens function evaluated at the argument that must be given in radians.
<code>tanh</code>	Function with one argument returning the hyperbolic tangens function evaluated at the argument that must be given in radians.

---

Table 3.2: Functions and constants that can be used in formula variables.

### 3.1.7 Probe Variables

Probe variables are used to make the value of another variable evaluated at a given location in a compartment globally available.

Figure 3.13: Dialog box for editing a probe variable.

Figure 3.13 shows the dialog box used for defining or editing a formula variable. As each variable, a formula variable needs a unique ‘**Name**’ as an identifier. A name of a variable consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. The following reserved names are not allowed as variable names: div, mod, and, or, not, if, then, else, endif, pi, sin, cos, tan, asin, acos, atan, sinh, cosh, tanh, deg, rad, exp, log, ln, log10, sign, abs, sqrt, min, max. To improve documentation of variables, a ‘**Description**’ and a ‘**Unit**’ can be specified. Then a ‘**Variable**’ must be selected and the ‘**Compartment**’, ‘**Zone**’ and the ‘**Location**’ in the compartment, where the variable has to be evaluated, must be specified. If the check box ‘**relative**’ is ticked, the location must be specified as relative coordinates between 0 and 1, otherwise the location must be given in absolute coordinates.

## 3.2 Processes

Transformation processes can be defined by a set of process rates, each of which describes the contribution of the process to the temporal change of the concentration of a given substance. Characteristic times of such processes may vary over several orders of magnitude. The consequence of the existence of widely varying time scales within a system is that the concentrations determined by fast processes converge so fast to their current equilibrium values (which themselves depend on slower processes) that the transient phase is not important for the behavior of the system on the slower time scale. In such situations, a separation of time scales, which solves concentrations determined by fast processes directly for their equilibrium solution, can be advantageous. This leads to a replacement of differential equations of fast processes by algebraic equations. Therefore, the following two types of processes are introduced:

- **Dynamic Processes** describe substance transformations the dynamics of which is important on the time scale of the simulation.
- **Equilibrium Processes** describe the effect of very fast processes which lead to permanent equilibrium values of the corresponding state variables.

Figure 3.14 shows the dialog box for editing processes. This dialog box is opened with

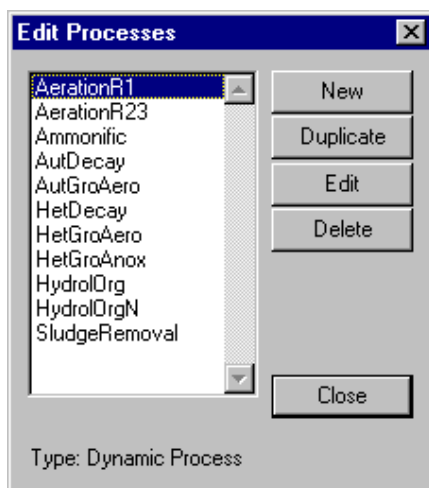


Figure 3.14: Dialog box for editing processes.

the **Processes** command in the **Edit** menu shown in Figure 3.2. It is of modeless type in order to facilitate the editing process. The names of all processes already defined are listed alphabetically in the list box of this dialog box. The type of the currently selected process is indicated at the bottom of the dialog box. The buttons of this dialog box allow the user to perform the following operations with processes: By clicking the button '**New**', new processes may be created from scratch. Alternatively, by clicking the button '**Duplicate**', the selected process can be duplicated. With the button '**Edit**' or by double-clicking the process name in the list box, a process can be edited. Finally, the button '**Delete**' allows the program users to delete processes. Deletion of a process is only possible, if it is not active within a compartment. The buttons **Duplicate**, **Edit** and **Delete** are inactive as long as no process is selected. Clicking the '**Close**' button results in closing of this dialog



box. It can be reopened by clicking the **Processes** command in the **Edit** menu shown in Fig. 3.2.

For the definition of transformation processes with the subdialogs assigned to the dialog box shown in Fig. 3.14, all variables listed in the dialog box shown in Fig. 3.3 may be used. The processes serve as a pool for use in compartments: Each process can be activated or inactivated in each compartment (if it does not contain illegal dependencies; cf. section 3.3).

After clicking the button **New** in the dialog box shown in Fig. 3.14, the process type can be selected in the dialog box shown in Fig. 3.15. The two types of processes shown in

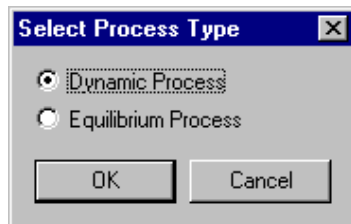


Figure 3.15: Dialog box for selecting the type of a process.

this selection box are described in more detail in the following two sections.

### 3.2.1 Dynamic Processes

A clear presentation of dynamic biochemical processes is very important to facilitate users of the program to obtain a survey of the interactions between the components of the system. The method of presentation used in AQUASIM was made popular for technical biochemical systems by the report of the IAWQ task group on mathematical modelling for design and operation of biological wastewater treatment (Henze et al., 1986). It is based on work on chemical reaction engineering (Petersen, 1965). Dynamic processes describe transformations by their contribution to the temporal rate of change of (dynamic) state variables. Usually, a biological or chemical process transforms several substances in fixed stoichiometric proportions. Therefore, it is advantageous to separate a common factor as a process rate, and to describe a process by this rate and by stoichiometric coefficients for all substances involved in the process. The contribution of a process to the temporal change of the concentration of a substance is then given as the product of the common process rate and the substance-specific stoichiometric coefficient. This decomposition of process rates into a common process rate and individual stoichiometric coefficients is not unique; to make it unique, one of the stoichiometric coefficients is usually set to unity. Physical processes or transfer processes which due to spatial averaging also have the mathematical form of a source term of the differential equation can be integrated easily into this general scheme. The notion of a stoichiometric coefficient has then a more general meaning and can include geometric factors as well. With this concept, the total transformation rate of a substance  $s_j$  is given by

$$r_j = \sum_i \nu_{i,j} r_{p_i} \quad (3.1)$$

where  $r_j$  ( $ML^{-3}T^{-1}$ ) is the total transformation rate of the substance  $s_j$ ,  $\nu_{i,j}$  (–) is the stoichiometric coefficient of the substance  $s_j$  for the process  $p_i$  and  $r_{p_i}$  ( $ML^{-3}T^{-1}$ ) is the rate of the process  $p_i$ . A clear presentation of a process model is given by writing the stoichiometric matrix ( $\nu_{i,j}$ ), supplemented by the process rates  $p_i$  in an additional column. This results in a process matrix as shown in Table 3.3. The nonzero elements of a row of

Process	Substances				Rate
	$s_1$	$s_2$	$s_3$	...	
$p_1$	$\nu_{1,1}$	$\nu_{1,2}$	$\nu_{1,3}$	...	$r_{p_1}$
$p_2$	$\nu_{2,1}$	$\nu_{2,2}$	$\nu_{2,3}$	...	$r_{p_2}$
$p_3$	$\nu_{3,1}$	$\nu_{3,2}$	$\nu_{3,3}$	...	$r_{p_3}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$

Table 3.3: Representation of a process model with the aid of a process matrix.

such a matrix show, which substances are affected by a given process, whereas the nonzero elements of a column indicate, which processes have an influence to a given substance. It is a useful convention to use positive process rates. In this case the signs of the stoichiometric coefficients indicate consumption (–) or production (+) of the corresponding substance.

Figure 3.16 shows the dialog box used for defining or editing a dynamic process. As

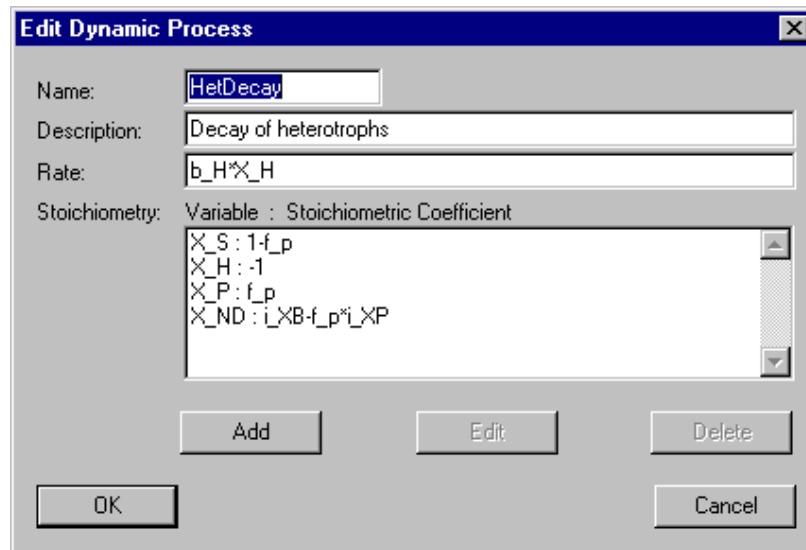


Figure 3.16: Dialog box for editing a dynamic process.

each process a dynamic process needs a unique ‘**Name**’ as an identifier. A name of a process consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. To improve documentation of processes, a ‘**Description**’ can be given optionally. The ‘**Rate**’ contains the common factor of the transformation rates of all variables involved. As shown in Fig. 3.17, for each ‘**Variable**’ involved in the process, an individual ‘**Stoichiometric Coefficient**’, given

as an algebraic expression according to the syntax of formula variables (cf. section 3.1.6) has to be specified. The list of stoichiometric coefficients can be edited using the buttons

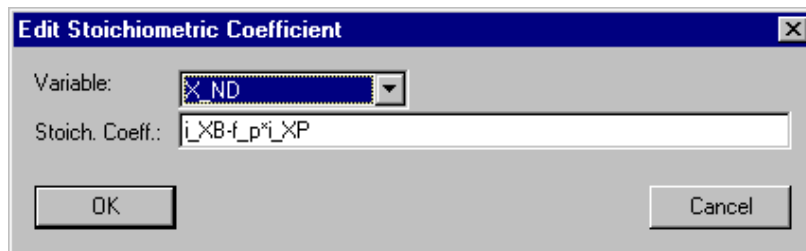


Figure 3.17: Dialog box for editing a stoichiometric coefficient of a dynamic process.

‘Add’, ‘Edit’ and ‘Delete’. The contribution of the process to the transformation rate of a variable is given as the product of the common rate with the individual stoichiometric coefficient. During simulations, a dynamic process has only an effect to variables, which are of the type of dynamic state variables. The fact, that in the list of stoichiometric coefficients, any type of variables is allowed, makes it easier to switch between different models (e.g. if variables are changed from calculated dynamic state variables to measured real list variables, the processes have not to be changed).

### 3.2.2 Equilibrium Processes

Equilibrium processes are used for processes, which are much faster than the processes which determine the typical time scale of the simulation. A variable determined by such a process can be treated as taking always the value corresponding to its equilibrium state. Therefore, its value is given as the solution of an algebraic equation:

$$r_{eq} = 0 \quad (3.2)$$

where  $r_{eq}$  depends on the variable involved and on other variables influencing the equilibrium value.

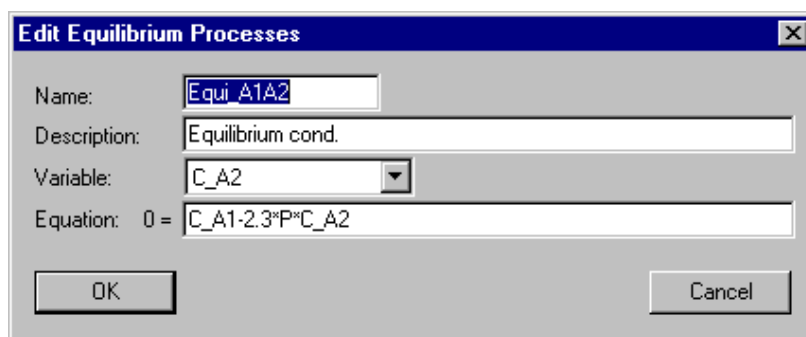


Figure 3.18: Dialog box for editing an equilibrium process.

Figure 3.18 shows the dialog box used for defining or editing an equilibrium process. As each process an equilibrium process needs a unique ‘Name’ as an identifier. A name

of a process consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. To improve documentation of processes, a **‘Description’** can be given optionally. For the selected **‘Variable’**, the **‘Equation’** to be solved can be given as an algebraic expression, which is set equal to zero. The syntax of this algebraic expression is the same as that of formula variables described in section 3.1.6. The variable itself has to be an argument of this expression. In a similar way as in the case of dynamic processes, an equilibrium process has only an effect, if the selected variable is of the type of an equilibrium state variable.

Although equilibrium state variables and equilibrium processes can be defined in AQUASIM, the difficulty for the numerical algorithm to find the solution to the system of nonlinear algebraic equations and limitations of the description of transport processes make the implementation of fast processes with dynamic state variables and fast dynamic processes often more advantageous as the implementation with equilibrium state variables.

### 3.3 Compartments

The geometrical configuration of an AQUASIM system consists of a set of compartments of given types. In order to be flexible enough to describe the desired system, six types of compartments are distinguished:

- **Mixed Reactor Compartments** are used to describe well-mixed domains as e.g. stirred reactors, mixed lakes, etc..
- **Biofilm Reactor Compartments** are used to describe growth and population dynamics of biofilms in which substrate gradients over the depth are important.
- **Advective-Diffusive Reactor Compartments** are used to describe systems with a longitudinal given water flow such as plug flow reactors.
- **Saturated Soil Column Compartments** are used to model transport, adsorption and transformation of substances in saturated soil columns including exchange with dead zones or immobile pore volume.
- **River Section Compartments** are used to describe the hydraulics, transport and transformation processes in rivers.
- **Lake Compartments** are used to model stratification, mixing, transport and transformation processes in horizontally well-mixed lakes.

Figure 3.19 shows the dialog box for editing compartments. This dialog box is opened



Figure 3.19: Dialog box for editing compartments.

with the **Compartments** command in the **Edit** menu shown in Figure 3.2. It is of modeless type in order to facilitate the editing process. The names of all compartments already defined are listed alphabetically in the list box of this dialog box. The type of the currently selected compartment is indicated at the bottom of this dialog box. The buttons of this dialog box allow the user to perform the following operations with compartments: By clicking the button '**New**', new compartments may be created from scratch. Alternatively, by clicking the button '**Duplicate**', the selected compartment can be duplicated. With the button '**Edit**' or by double-clicking the compartment name in the list box, a compartment can be edited. Finally, the button '**Delete**' allows the program users to delete

compartments. Deletion of a compartment is only possible, if the compartment is not an argument of a link or of a definition of a sensitivity analysis or of a parameter estimation. The buttons **Duplicate**, **Edit** and **Delete** are inactive as long as no compartment is selected. The buttons '**Activate**' and **Inactivate** can be used to activate and inactivate compartments from the calculation. The names of inactive compartments are indented in the list box of the dialog box shown in Fig. 3.19. Clicking the '**Close**' button results in closing this dialog box. It can be reopened by choosing the **Compartments** command in the **Edit** menu shown in Fig. 3.2.

After clicking one of the buttons **New** in the dialog box shown in Fig. 3.19 the compartment type can be selected in the dialog box shown in Fig. 3.20. The six types of

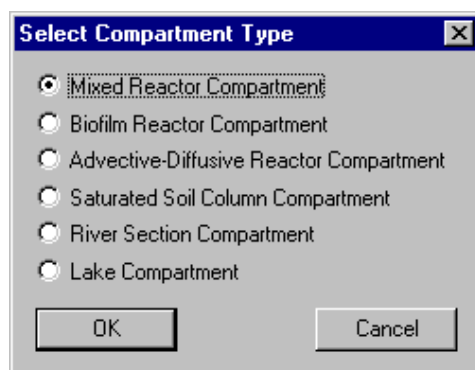


Figure 3.20: Dialog box for selecting the type of a compartment.

compartments shown in this selection box are described in more detail in the following six subsections.

### 3.3.1 Mixed Reactor Compartment

#### Overview

This simplest compartment of AQUASIM describes inflow, outflow and transformation processes of substances in a completely stirred reactor with constant or variable volume.

#### Equations Solved by AQUASIM

If the volume is selected to be variable, the current volume of the compartment is calculated as the solution to the differential equation

$$\frac{dV_R}{dt} = Q_{in} - Q_{out} \quad (3.3)$$

where  $t$  is time,  $V_R$  is the reactor volume,  $Q_{in}$  the volumetric inflow, and  $Q_{out}$  the volumetric outflow. Otherwise, the volume remains constant (the outflow is then equal to the inflow). The temporal change of the concentration of substances dissolved or suspended in the water is given as

$$\frac{dC_i}{dt} = \frac{I_{in,C_i}}{V_R} - \frac{Q_{in}}{V_R} C_i + r_{C_i} \quad , \quad (3.4)$$

where  $C_i$  is the substance concentration represented by a dynamic volume state variable,  $I_{in,C_i}$  is the loading of the substance described by the concentration  $C_i$  into the reactor (mass per unit of time), and  $r_{C_i}$  is the transformation rate of the substance described by the concentration  $C_i$ . This concentration rate is given as the sum of the products of the process rates times the stoichiometric coefficients of the substance described by the concentration  $C_i$  of all processes active in the compartment. The temporal change of substances attached to a surface is given by

$$\frac{dS_i}{dt} = r_{S_i} \quad (3.5)$$

where  $S_i$  is the concentration, surface density or mass of the attached substance represented by a dynamic surface state variable, and  $r_{S_i}$  is the transformation rate of the substance described by  $S_i$ . This transformation rate is calculated analogously to the transformation rate  $r_{C_i}$  described above. The dimension of  $S_i$  can be chosen by the program user who is responsible to make consistent process definitions: If a substance is converted from dissolved ( $C_i$ ) to attached ( $S_i$ ; e.g. by an adsorption process) the stoichiometric coefficients must be chosen in order to convert the units of  $C_i$  correctly to those of  $S_i$ . For equilibrium state variables algebraic equations specified as equilibrium processes are solved in the compartment.

#### User Definitions

Figure 3.21 shows the dialog box used for defining or editing a mixed reactor compartment. The edit field '**Name**' is used to specify the name of the compartment. Each compartment needs a unique name as an identifier. A name of a compartment consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. In the edit field '**Comp. Index**' a nonnegative integer number can be specified as a compartment index. This value can be accessed with the aid of the program variable `Compartment Index` to make variables or process rates dependent on the compartment.

Figure 3.21: Dialog box for editing a mixed reactor compartment.

To improve documentation of compartments, the edit field '**Description**' can optionally be used to store comments on specific implementation features of a compartment.

The buttons '**Variables**', '**Processes**', '**Init. Cond.**' and '**Input**' are used to activate and inactivate state variables, to activate and inactivate processes, to specify initial conditions and to define inputs to a compartment, respectively. These options are described later in this subsection.

The next option allows the user to select the reactor type. Selection of the radio button '**constant volume**' leads to the use of a reactor the volume of which does not change; selection of the radio button '**variable volume**' leads to a reactor, the volume of which is calculated dynamically according to the equation given above. The edit field '**Volume**' is used to specify the volume of the reactor. For a reactor with variable volume, the number given here is used as the initial volume at the start time of the simulation unless an initial condition for the program variable `Reactor Volume` is defined. In the latter case, this initial condition is used. For a reactor with variable volume, in the edit field '**Outflow**' the outflow of the reactor must be specified (the inflow is given in the dialog box opened by clicking the button **Input** as described below). In this case, the change in reactor volume is given by the difference in in- and outflow as given by equation (3.3)

The check box '**active for calculation**' can be used to activate or inactivate the compartment from the calculations. This check box has the same functionality as the buttons **Activate** and **Inactivate** in the dialog box shown in Fig. 3.19.

The button '**Acc.**' is used to specify the numerical accuracy of program variables as described later in this subsection.

As for each compartment, the user has to select which state variables are active. This is done by clicking the button '**Variables**' of the dialog box shown in Fig. 3.21. This action opens the dialog box shown in Fig. 3.22. The two list boxes of this dialog box show the active variables and the available variables, respectively. The button '**Activate**' is used to activate available variables selected in the right list box, and the button '**Inactivate**' is used to inactivate active variables selected in the left list box. If an active variable is selected while activating another variable, the new active variable is inserted in the list of active variables immediately before the selected variable, otherwise it is appended to the



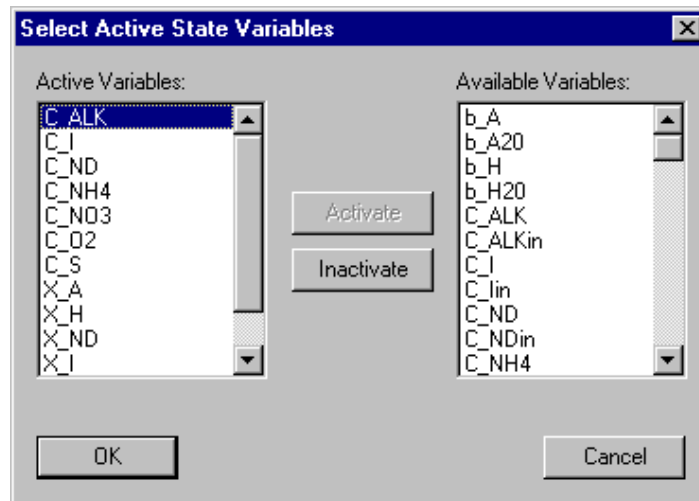


Figure 3.22: Dialog box for activating and inactivating state variables in a compartment.

end of the list of active variables. This gives the user the possibility to influence the order of active variables (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The list of active variables may contain variables of any type, but activation has only an effect to state variables. Inactive state variables return a value of zero. The reason for allowing other types of variables in the list of active state variables is to facilitate switching between different models that do not contain the same state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of active variables of the compartments.

Similarly to the activation of state variables, the user has to select which processes are active in a compartment. This is done by clicking the button '**Processes**' of the dialog box shown in Fig. 3.21. This action opens the dialog box shown in Fig. 3.23. The two list boxes in this dialog box show the active processes and the available processes, respectively. The button '**Activate**' is used to activate available processes selected in the right list box, and the button '**Inactivate**' is used to inactivate an active processes selected in the left list box. If an active process is selected while activating another process, the new active process is inserted in the list of active processes immediately before the selected process, otherwise it is appended to the end of the list of active processes. This gives the user the possibility to influence the order of active processes (the order is irrelevant for the program, but it may be convenient for the user to have a certain order).

Initial conditions for a mixed reactor compartment can be specified by clicking the button '**Init. Cond.**' of the dialog box shown in Fig. 3.21. This action opens the dialog box shown in Fig. 3.24. This dialog box shows a list of all initial conditions already specified. Each row of the list box contains the name of a variable, followed by the zone of the compartment for which the initial condition is specified (in brackets) and by the algebraic expression specifying the initial value. For each combination of a variable with a zone only one unique initial condition can be specified. The list of initial conditions can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If an initial condition is selected while adding a new initial condition, the new initial condition is inserted in the list immediately before the selected initial condition, otherwise it is appended to the end

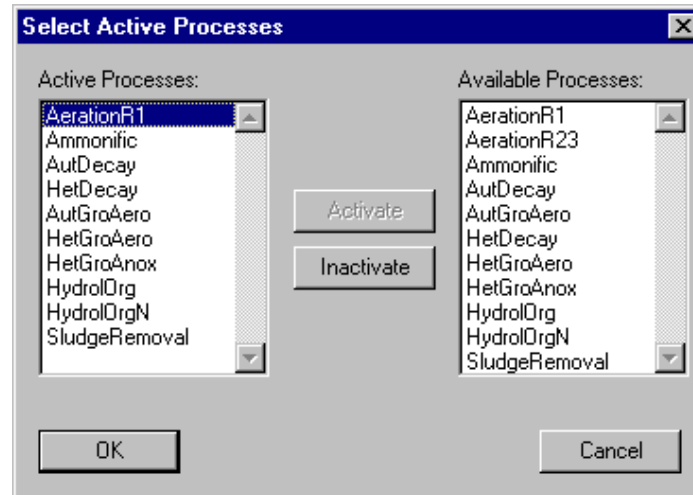


Figure 3.23: Dialog box for activating and inactivating processes in a compartment.

of the list of initial conditions. This gives the user the possibility to influence the order of the initial conditions (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The dialog box used for editing a single initial condition is shown in Fig. 3.25. In this dialog box the fields '**Variable**', '**Zone**' and '**Init. Cond.**' allow the user to select a variable and a zone and to specify an initial condition. In a mixed reactor compartment there is only one zone **Bulk Volume**. Initial conditions for any type of variables can be specified, but only initial conditions for state variables and, in the case of a reactor with variable volume, initial conditions for the program variable **Reactor Volume**, are used by the program. The reason for allowing to define initial conditions for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of initial conditions of the compartments. Initial conditions of a mixed reactor compartment may not depend on state variables and they can only depend on the program variables **Calculation Number** and **Time**.

Input to a mixed reactor compartment can be specified by clicking the button '**Input**' of the dialog box shown in Fig. 3.21. This action opens the dialog box shown in Fig. 3.26. The edit field '**Water Inflow**' of this dialog box is used to specify the discharge of water into the bulk volume of the reactor,  $Q_{in}$ , and the list box contains substance loadings,  $I_{in,C_i}$ , into the reactor. For each variable only one unique loading can be specified. The list of loadings can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If a loading is selected while adding a new loading, the new loading is inserted in the list immediately before the selected loading, otherwise it is appended to the end of the list of loadings. This gives the user the possibility to influence the order of the loadings (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.27 shows the dialog box used to specify a single input loading of a substance. In this dialog box the fields '**Variable**' and '**Loading**' allow the user to select a variable and to specify a loading,  $I_{in,C_i}$ . Note that all loadings must be specified as mass per unit of time. A loading of a mixed reactor compartment may depend on the program variable **Discharge** and on dynamic volume state variables. These variables return the discharge

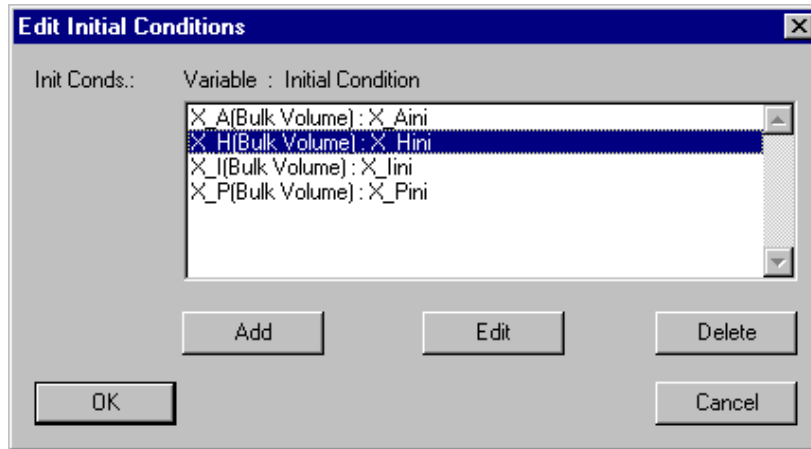


Figure 3.24: Dialog box for editing initial conditions for a mixed reactor compartment.

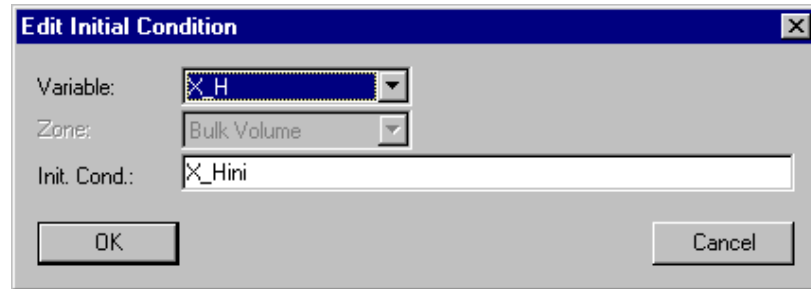


Figure 3.25: Dialog box for editing a single initial condition for a mixed reactor compartment.

and the concentrations of dynamic volume state variables resulting from all advective links connected to the inlet of the reactor. Loadings for any type of variables can be specified, but only loadings for dynamic volume state variables are used by the program. The reason for allowing to define loadings for variables of other types is to facilitate the user switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of loadings of the compartments.

The button '**Acc.**' of the dialog box shown in Fig. 3.21 is used to specify the numerical accuracies of program variables. Figure 3.28 shows the dialog box used for this purpose. It allows the user to specify relative and absolute accuracies of the program variables '**Discharge**',  $Q$ , and reactor '**Volume**',  $V_R$ , in the compartment. Good behaviour of the numerical algorithms is usually achieved if the absolute accuracy and the product of the relative accuracy times a typical value of the variable both are 4 to 6 orders of magnitude smaller than typical values of the variable.

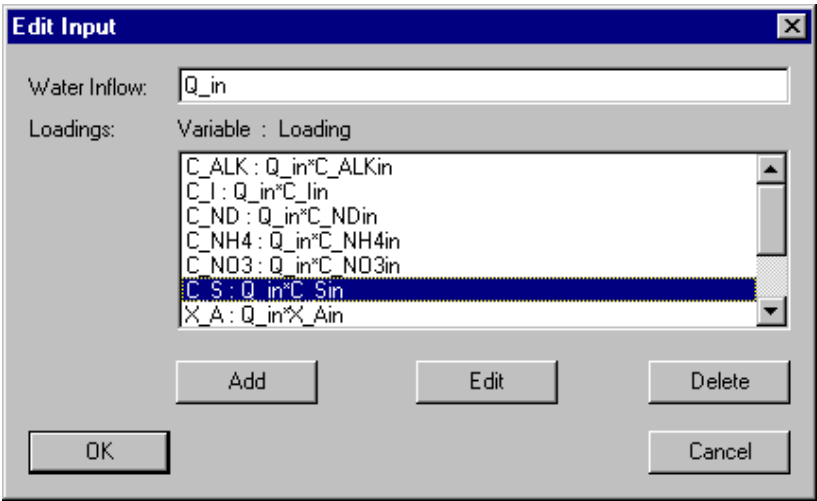


Figure 3.26: Dialog box for editing inputs to a mixed reactor compartment.

In Table 3.4 the program variables available in a mixed reactor compartment are summarized (for a complete overview of all program variables see Table 3.1 on page 17).

Calculation Number	Identifier for calculations (value set in the dialog boxes shown in Figs. 4.5 and 4.17).
Time	Simulation time.
Compartment Index	Identifier for compartments (value set in the dialog box shown in Fig. 3.21).
Zone Index	Identifier for zones within compartments (returns a value of 0).
Discharge	Volumetric flow rate.
Water Fraction	Volumetric fraction of water (returns a value of 1).
Reactor Volume	Total volume of the reactor.
Bulk Volume	Volume of mixed water zone (returns the value of the reactor volume).

Table 3.4: Program variables available in the mixed reactor compartment.

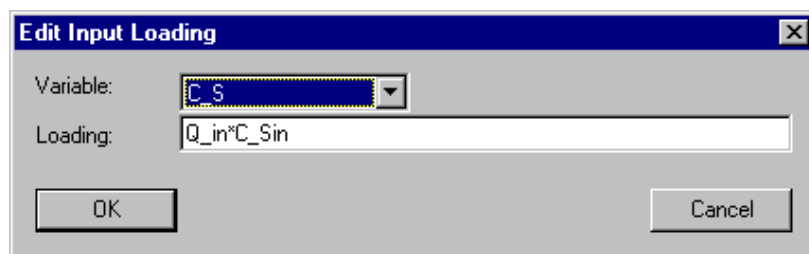


Figure 3.27: Dialog box for editing a single input to a mixed reactor compartment.

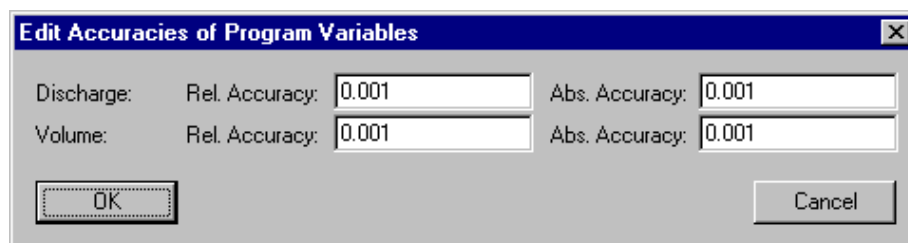


Figure 3.28: Dialog box for editing the accuracies of program variables of a mixed reactor compartment.

### 3.3.2 Biofilm Reactor Compartment

#### Overview

A biofilm consists of a solid matrix with pore water that can contain dissolved substances and suspended solids. The growth or decay of organisms forming the solid matrix leads to expansion or shrinking of the biofilm. The consumption of substrate of the organisms present at a high concentration in the biofilm solid matrix can lead to a growth limitation by the diffusive mass transfer into the depth of the biofilm. The biofilm reactor compartment of AQUASIM describes a reactor with a completely mixed bulk water volume and with a biofilm growing on a substratum surface in the reactor. Solids can attach or detach at the surface of the biofilm and in its interior. Any transformation processes can be defined. The description of the biofilm in AQUASIM is one-dimensional. Only the direction perpendicular to the substratum, which has the largest concentration gradients, is resolved. All variables are averaged over areas parallel to the substratum. Biofilm reactor compartments can be linked advectively or diffusively to other AQUASIM compartments.

#### Equations Solved by AQUASIM

The equations solved in the biofilm reactor compartment are based on the development of a one-dimensional mixed-culture biofilm model since 1984 (Wanner and Gujer, 1984; Wanner and Gujer, 1986; Gujer and Wanner, 1990; Wanner, 1994; Wanner, 1995; Wanner and Reichert, 1996; Reichert and Wanner, 1997). The equations as they are solved by AQUASIM are described in the last two papers of this list. In the following, a summary of these equations is given.

In order to formulate the one-dimensional conservation laws

$$\frac{\partial \hat{\rho}}{\partial t} + \frac{\partial \hat{j}}{\partial z} = \hat{r} \quad (3.6)$$

compartment-specific expressions for the one-dimensional density,  $\hat{\rho}$  (amount of conserved quantity per unit compartment length), for the one-dimensional flux,  $\hat{j}$  (amount of the conserved quantity transported per unit time), and for the one-dimensional source term,  $\hat{r}$  (amount produced per unit compartment length and per unit time), must be derived.

Three zones are distinguished in the biofilm reactor compartment. Variables present in different zones are distinguished by the index ‘M’ for the solid matrix, ‘P’ for the pore volume, and ‘B’ for the bulk volume. In addition, the index ‘F’ is used for biofilm, when the distinction into solid matrix and pore volume is not relevant (e.g. for the biofilm thickness,  $L_F$ ). The spatial dimension perpendicular to the substratum is resolved by the space coordinate  $z$ . This coordinate takes a value of zero at the substratum and it increases with increasing distance from the substratum up to the biofilm thickness,  $L_F$ .

In order to formulate the biofilm equations, 4 types of components of a conservation law must be distinguished: The array of one-dimensional densities of these types of components is given as follows:

$$\hat{\rho} = \begin{pmatrix} AX_{M,i} \\ AX_{P,i} \\ A\epsilon_{l,F}C_{P,i} \\ A\theta \end{pmatrix} \quad (3.7)$$

The first component of equation 3.7 describes particulate species in the biofilm matrix. The one-dimensional density of these species is given as the product of the area parallel to the substratum,  $A$ , and the concentration of particulate species in the biofilm matrix averaged over this area,  $X_{M,i}$ . The second component of equation 3.7 describes the particulate species in the biofilm pore water. Similarly to the first component, the one-dimensional density of these species is given as the product the area parallel to the substratum,  $A$ , and the concentration of these species averaged over this area,  $X_{P,i}$ . Note that this implies a definition of a concentration of particulate species as mass per unit of total volume and not only per unit of pore water volume. This definition makes the concentrations  $X_{M,i}$  and  $X_{P,i}$  comparable with each other. Note, however, that this leads to a step in the concentration  $X_{P,i}$  at the biofilm surface, because there is no solid matrix in the bulk volume and the concentration of suspended solids is continuous if defined as mass per unit of water phase volume. The third component of equation 3.7 describes substances dissolved in the pore water of the biofilm. The concentration of dissolved substances is defined as mass per unit of water phase volume. Therefore, their one-dimensional density is given as the product of the area parallel to the substratum,  $A$ , the liquid phase volume fraction,  $\epsilon_{l,F}$ , and the concentration of the dissolved species averaged over the area  $A$ ,  $C_{P,i}$ . All concentrations,  $X_i$  and  $C_i$ , are represented in AQUASIM as dynamic volume state variables. The liquid phase volume fraction is given as

$$\epsilon_{l,F} = \theta - \sum_{i=1}^{n_X} \frac{X_{P,i}}{\rho_{X_i}} \quad (3.8)$$

where  $\rho_{X_i}$  is the density of the particulate species described by the concentration  $X_i$ ,  $n_X$  is the number of particulate species, and  $\theta$  is the porosity of the biofilm given as unity minus the volume fraction of the solid matrix

$$\theta = 1 - \sum_{i=1}^{n_X} \frac{X_{M,i}}{\rho_{X_i}} \quad (3.9)$$

The last component of equation 3.7 describes the porosity of the biofilm. The one-dimensional density for this variable (free volume per unit biofilm depth) is given as the product of the porosity,  $\theta$ , and the area parallel to the substratum,  $A$ .

The one-dimensional fluxes of the substances with one-dimensional densities as de-

scribed by equation 3.7 are given as follows:

$$\hat{\mathbf{j}} = \begin{pmatrix} Au_F X_{M,i} - AD_{M,X_i} \frac{\partial X_{M,i}}{\partial z} \\ -(1-\theta)Au_F \frac{X_{P,i}}{\theta} - \theta AD_{P,X_i} \frac{\partial \left(\frac{X_{P,i}}{\theta}\right)}{\partial z} \\ + A \sum_{k=1}^{n_X} \frac{D_{M,X_k}}{\rho_{X_k}} \frac{\partial X_{M,k}}{\partial z} \frac{X_{P,i}}{\theta} + A \sum_{k=1}^{n_X} \frac{\theta D_{P,X_k}}{\rho_{X_k}} \frac{\partial \left(\frac{X_{P,k}}{\theta}\right)}{\partial z} \frac{X_{P,i}}{\theta} \\ -(1-\epsilon_{l,F})Au_F C_{P,i} - \epsilon_{l,F}AD_{P,C_i} \frac{\partial C_{P,i}}{\partial z} \\ + \frac{\epsilon_{l,F}}{\theta} A \sum_{k=1}^{n_X} \frac{D_{M,X_k}}{\rho_{X_k}} \frac{\partial X_{M,k}}{\partial z} C_{P,i} + \frac{\epsilon_{l,F}}{\theta} A \sum_{k=1}^{n_X} \frac{\theta D_{P,X_k}}{\rho_{X_k}} \frac{\partial \left(\frac{X_{P,k}}{\theta}\right)}{\partial z} C_{P,i} \\ \theta Au_F + A \sum_{k=1}^{n_X} \frac{D_{M,X_k}}{\rho_{X_k}} \frac{\partial X_{M,k}}{\partial z} \end{pmatrix} \quad (3.10)$$

In this equation,  $u_F$  is the advective velocity of the biofilm matrix given by

$$u_F = \frac{1}{A} \int_0^z \left( \frac{1}{1-\theta} \sum_{k=1}^{n_X} \frac{r_{M,X_k}}{\rho_{X_k}} + r'_\theta \right) A dz' \quad (3.11)$$

where  $r'_\theta$  is the excess growth of porosity defined by

$$r'_\theta = r_\theta - \frac{\theta}{1-\theta} \sum_{k=1}^{n_X} \frac{r_{M,X_k}}{\rho_{X_k}} \quad (3.12)$$

and  $r_\theta$  is the growth rate of free volume between the solid matrix. With this notation, the porosity of a biofilm remains constant if  $r'_\theta = 0$ , because the same fraction of free volume is produced as is of solid matrix volume. The first component of equation (3.10) describes the advective and diffusive flow of solids within the biofilm matrix. The second component of equation (3.10) describes advection and diffusion of solids suspended in the pore volume of the biofilm. The first term of this component describes advection with the water flow produced by growth and decay of organisms in the biofilm matrix. Because the organisms consist mainly of water, in a growing biofilm, the water contained in the organisms moving outward at the velocity  $u_F$  in the solid matrix must be compensated by water flowing through the pore water into the biofilm. The first term describes advection of suspended solids with this water flow. The second term describes diffusion or Brownian motion of the solids in the pore water. The third and fourth terms of the second component describe advection with the water flow induced by diffusion of solids in the solid matrix and in the pore water, respectively. The third component of equation (3.10) describes advection and diffusion of dissolved substances in the pore volume of the biofilm. The terms have the same meaning as those in the second component for suspended solids. The last component of equation (3.10) represents the flow of free volume in the biofilm. Free volume is transported advectively with velocity  $u_F$  analogously to the solid matrix, but diffusive transport in the solid matrix has a reverse effect on free volume.



The following one-dimensional source terms are required to complete the set of biofilm equations.

$$\hat{\mathbf{r}} = \begin{pmatrix} Ar_{M,X_i} - Ak_{de,vol,X_i}X_{M,i} + Ak_{at,vol,X_i}X_{P,i} \\ Ar_{P,X_i} + Ak_{de,vol,X_i}X_{M,i} - Ak_{at,vol,X_i}X_{P,i} \\ Ar_{C_i} \\ Ar_{\theta} \end{pmatrix} \quad (3.13)$$

In these equations  $r$  are transformation rates and  $k_{de,vol,X_i}$  and  $k_{at,vol,X_i}$  are substance-dependent volume detachment and attachment coefficients, respectively. The terms containing the detachment and attachment coefficients describe exchange of solids between the pore water and the solid matrix of the biofilm.

Application of the general expression for differential conservation laws (3.6) to the definitions given by the equations (3.7) to (3.13) leads to the following set of 4 differential equations.

The first equation describes the behaviour of the constituents of the biofilm solid matrix:

$$\begin{aligned} \frac{\partial X_{M,i}}{\partial t} = & -u_F \frac{\partial X_{M,i}}{\partial z} + \frac{1}{A} \frac{\partial}{\partial z} \left( AD_{M,X_i} \frac{\partial X_{M,i}}{\partial z} \right) - r'_{\theta} X_{M,i} \\ & + \left( r_{M,X_i} - \frac{X_{M,i}}{1-\theta} \sum_{k=1}^{n_X} \frac{r_{M,X_k}}{\rho_{X_k}} \right) - k_{de,vol} X_{M,i} + k_{at,vol} X_{P,i} \end{aligned} \quad (3.14)$$

The second equation describes the behaviour of solids suspended in the pore water of the biofilm:

$$\begin{aligned} \frac{\partial X_{P,i}}{\partial t} = & (1-\theta)u_F \frac{\partial \left( \frac{X_{P,i}}{\theta} \right)}{\partial z} - u_F \frac{\partial \theta}{\partial z} \frac{X_{P,i}}{\theta} + \sum_{k=1}^{n_X} \frac{r_{M,X_k}}{\rho_{X_k}} \frac{X_{P,i}}{\theta} \\ & + (1-\theta) \frac{X_{P,i}}{\theta} r'_{\theta} + \frac{1}{A} \frac{\partial}{\partial z} \left( \theta AD_{P,X_i} \frac{\partial \left( \frac{X_{P,i}}{\theta} \right)}{\partial z} \right) \\ & - \frac{1}{A} \frac{\partial}{\partial z} \left( A \sum_{k=1}^{n_X} \frac{D_{M,X_k}}{\rho_{X_k}} \frac{\partial X_{M,k}}{\partial z} \frac{X_{P,i}}{\theta} \right) \\ & - \frac{1}{A} \frac{\partial}{\partial z} \left( A \sum_{k=1}^{n_X} \frac{\theta D_{P,X_k}}{\rho_{X_k}} \frac{\partial \left( \frac{X_{P,k}}{\theta} \right)}{\partial z} \frac{X_{P,i}}{\theta} \right) \\ & + r_{P,X_i} + k_{de,vol,X_i} X_{M,i} - k_{at,vol,X_i} X_{P,i} \end{aligned} \quad (3.15)$$

The third equation describes the behaviour of substances dissolved in the pore water

of the biofilm:

$$\begin{aligned}
\frac{\partial \epsilon_{l,F}}{\partial t} C_{P,i} + \epsilon_{l,F} \frac{\partial C_{P,i}}{\partial t} = & (1 - \epsilon_{l,F}) u_F \frac{\partial C_{P,i}}{\partial z} + \sum_{k=1}^{n_X} \frac{r_{M,X_k} \epsilon_{l,F}}{\rho_{X_k} \theta} C_{P,i} \\
& + (1 - \theta) \frac{\epsilon_{l,F}}{\theta} r'_{theta} C_{P,i} + \frac{1}{A} \frac{\partial}{\partial z} \left( \epsilon_{l,F} A D_{P,C_i} \frac{\partial C_{P,i}}{\partial z} \right) \frac{\partial C_{P,i}}{\partial z} \\
& - \frac{1}{A} \frac{\partial}{\partial z} \left( \frac{\epsilon_{l,F}}{\theta} A \sum_{k=1}^{n_X} \frac{D_{M,X_k}}{\rho_{X_k}} \frac{\partial X_{M,k}}{\partial z} C_{P,i} \right) \\
& - \frac{1}{A} \frac{\partial}{\partial z} \left( \epsilon_{l,F} A \sum_{k=1}^{n_X} \frac{D_{P,X_k}}{\rho_{X_k}} \frac{\partial \left( \frac{X_{P,k}}{\theta} \right)}{\partial z} C_{P,i} \right) + r_{C_i}
\end{aligned} \tag{3.16}$$

The fourth equation describes the changes in biofilm porosity:

$$\frac{\partial \theta}{\partial t} = -u_F \frac{\partial \theta}{\partial z} - \frac{1}{A} \frac{\partial}{\partial z} \left( A \sum_{k=1}^{n_X} \frac{D_{M,X_k}}{\rho_{X_k}} \frac{\partial X_{M,k}}{\partial z} \right) + (1 - \theta) r'_\theta \tag{3.17}$$

In equation (3.16)  $\partial \epsilon_{l,F} / \partial t$  can be calculated with the aid of the temporal derivative of equation (3.8):

$$\frac{\partial \epsilon_{l,F}}{\partial t} = \frac{\partial \theta}{\partial t} - \sum_{i=1}^{n_X} \frac{1}{\rho_{X_i}} \frac{\partial X_{P,i}}{\partial t} \tag{3.18}$$

The above equations must be combined with the equation

$$\frac{dL_F}{dt} = u_L \tag{3.19}$$

describing the temporal change of the biofilm thickness,  $L_F$ . In this equation,  $u_L$  is the velocity of the interface layer between biofilm and bulk volume. The velocity  $u_L$  is given by

$$u_L = u_F(L_F) - u_{de} + u_{at} \tag{3.20}$$

where  $u_{de}$  and  $u_{at}$  are the detachment and the attachment velocities of solids at the biofilm surface, respectively. The detachment velocity,  $u_{de}$ , can either be given globally as a function of the biofilm thickness,  $L_F$ , the advective velocity at the biofilm surface,  $u_F(L_F)$ , time,  $t$ , surface shear stress,  $\tau_{surf}$ , etc.

$$u_{de} = u_{de}(L_F, u_F(L_F), t, \tau_{surf}, \dots) \tag{3.21a}$$

or, in the case of a diffusive biofilm matrix (all  $D_{M,X_i} > 0$ ), individual surface detachment rate coefficients,  $k_{de,surf,X_i}$ , can be given alternatively. The latter case results in a detachment velocity of

$$u_{de} = \frac{1}{1 - \theta(L_F)} \sum_{k=1}^{n_X} \frac{k_{de,surf,X_k} X_{M,k}}{\rho_{X_k}} \tag{3.21b}$$

The attachment velocity is calculated similarly to the equation (3.21b) as

$$u_{at} = \frac{1}{1 - \theta(L_F)} \sum_{k=1}^{n_X} \frac{k_{at,surf,X_k} X_{L,k}}{\rho_{X_k}} \quad (3.22)$$

where  $k_{at,surf,X_i}$  is the surface attachment coefficient for the solids described by the concentrations  $X_i$ . In all the above equations,  $X_{L,i}$  is the concentration of solids of type  $i$  in the liquid boundary layer above the biofilm immediately adjacent to the biofilm surface.

In order to make the solution of the above system of differential equations unique, boundary conditions must be specified.

For the case of a rigid biofilm matrix (all  $D_{M,X_i} = 0$ ) no boundary condition is required for equation (3.14) at the substratum-biofilm interface. For the case of a diffusive biofilm matrix (all  $D_{M,X_i} > 0$ ) equation (3.14) is solved with the following boundary condition at the substratum-biofilm interface:

$$\frac{\partial X_{M,i}}{\partial z}(z = 0) = 0 \quad (3.23a)$$

The boundary condition for equation (3.14) at the biofilm surface is determined by attachment and detachment processes. For a rigid biofilm matrix (all  $D_{M,X_i} = 0$ ) the boundary condition is given by

$$(u_F - u_L)X_{M,i}(L_F) = \begin{cases} (u_{de} - u_{at})X_{M,i}(L_F) & \text{for } u_{de} > u_{at} \\ (u_{de} - u_{at})\frac{k_{at,surf,X_i}}{u_{at}}X_{L,i} & \text{for } u_{de} < u_{at} \end{cases} \quad (3.23b)$$

For a diffusive biofilm matrix the boundary condition for equation (3.14) at the biofilm surface is given by

$$(u_F - u_L)X_{M,i}(L_F) - D_{M,X_i} \frac{\partial X_{M,i}}{\partial z}(L_F) = k_{de,surf,X_i}X_{M,i}(L_F) - k_{at,surf,X_i}X_{L,i} \quad (3.23c)$$

The boundary conditions for equation (3.15) that describes the behaviour of suspended solids in the biofilm pore water are given by a no flux condition at the substratum-biofilm interface:

$$\frac{\partial X_{P,i}}{\partial z}(z = 0) = 0 \quad (3.24a)$$

and by a continuity equation of the concentrations in the water at the biofilm surface

$$X_{P,i}(L_F) = \theta(L_F)X_{L,i} \quad (3.24b)$$

Note that due to the definition of  $X_{P,i}$  as mass per total of biofilm volume (including the solid matrix), this continuity condition for concentrations in the water phase leads to a discontinuity of the values of  $X_{P,i}$  at the biofilm surface. The advantage of the definition of  $X_{P,i}$  as mass per unit of total biofilm volume is that the concentrations  $X_{P,i}$  can directly be compared with the concentrations  $X_{M,i}$  of solids in the biofilm matrix.

The boundary conditions of equation (3.16) that describes the behaviour of substances dissolved in the pore water of the biofilm are as follows. At the substratum-biofilm interface, the boundary condition is given as a continuity equation of the flow through the substratum

$$-A\epsilon_{l,F}D_{P,C_i}\frac{\partial C_{P,i}}{\partial z}(z=0) = I_{substr,C_i} \quad (3.25a)$$

The total flow through the substratum,  $I_{substr,C_i}$ , is zero (impermeable substratum) unless a diffusive link is connected to the bottom of the biofilm in order to model a permeable membrane. In the latter case the flow through the substratum is proportional to the difference between the substance concentration in the pore water and the concentration in the other compartment connected to the diffusive link (the substance-specific constants of proportionality are defined in the link definition). The boundary condition at the biofilm surface is a continuity condition for the concentration in the water

$$C_{P,i}(L_F) = C_{L,i} \quad (3.25b)$$

The equation (3.17) does not require any boundary conditions.

The total flows of solids out of the biofilm (negative values for flows into the biofilm) are given as

$$\begin{aligned} I_{L,X_i} = & Au_F X_{M,i} - AD_{F,X_i} \frac{\partial X_{M,i}}{\partial z}(L_F) \\ & - (1-\theta)Au_F \frac{X_{P,i}}{\theta} - \theta AD_{P,X_i} \frac{\partial \left(\frac{X_{P,i}}{\theta}\right)}{\partial z}(L_F) \\ & + A \sum_{k=1}^{n_X} \frac{D_{M,X_k}}{\rho_{X_k}} \frac{\partial X_{M,k}}{\partial z} \frac{X_{P,i}}{\theta}(L_F) + A \sum_{k=1}^{n_X} \frac{\theta D_{P,X_k}}{\rho_{X_k}} \frac{\partial \left(\frac{X_{P,k}}{\theta}\right)}{\partial z} \frac{X_{P,i}}{\theta}(L_F) \\ & - Au_L X_{M,i}(L_F) \end{aligned} \quad (3.26)$$

This expression is the sum of the flow from the solid matrix and the pore volume at  $z = L_F$  plus the last term correcting for the movement of the biofilm surface.

The total flows of dissolved substances out of the biofilm (negative values for flows into the biofilm) are given as

$$\begin{aligned} I_{L,C_i} = & - (1-\theta) \frac{\epsilon_{l,F}}{\theta} Au_F C_{P,i}(L_F) - \epsilon_{l,F} AD_{P,C_i} \frac{\partial C_{P,i}}{\partial z}(L_F) \\ & + \frac{\epsilon_{l,F}}{\theta} A \sum_{k=1}^{n_X} \frac{D_{M,X_k}}{\rho_{X_k}} \frac{\partial X_{M,k}}{\partial z} \frac{X_{P,i}}{\theta}(L_F) \\ & + \frac{\epsilon_{l,F}}{\theta} A \sum_{k=1}^{n_X} \frac{\theta D_{P,X_k}}{\rho_{X_k}} \frac{\partial \left(\frac{X_{P,k}}{\theta}\right)}{\partial z} \frac{X_{P,i}}{\theta}(L_F) \\ & - Au_L \epsilon_{l,F} C_{P,i}(L_F) \end{aligned} \quad (3.27)$$

This expression is the flow at  $z = L_F$  plus a correction that considers the movement of the interface.

The equations for the biofilm described so far are connected to the bulk volume of the biofilm reactor through a liquid boundary layer. In AQUASIM the liquid boundary layer

is only roughly described with the aid of a mass transfer resistance (Wanner and Reichert, 1996). The equation for particulate components is given as

$$X_{L,i} - X_{B,i} = \frac{\kappa_{L,X_i}}{A(L_F)} I_{L,X_i} \quad (3.28)$$

A similar expression describes the behaviour of the dissolved components

$$C_{L,i} - C_{B,i} = \frac{\kappa_{L,C_i}}{A(L_F)} I_{L,C_i} \quad (3.29)$$

In these equations,  $\kappa_{L,X_i}$  and  $\kappa_{L,C_i}$  are the diffusive resistances for the substances  $X_i$  and  $C_i$ , and  $X_{B,i}$  and  $C_{B,i}$  are the concentrations in the bulk volume. Note that for the case of a resistance of zero, a continuity equation for the concentrations in the liquid layer adjacent to the biofilm and in the bulk volume results.

As a last series of equations, the equations for the bulk volume must be given.

The mass balance for particulate substances in the bulk volume is given as

$$\frac{d}{dt}(V_B X_{B,i}) = I_{in,X_i} - X_{B,i} Q_{ef} + I_{L,X_i} + V_B r_{X_i} \quad (3.30)$$

where  $V_B$  is the bulk volume,  $I_{in,X_i}$  is the total input of the substance described by the concentration  $X_i$  into the bulk volume through advective and diffusive links and as specified as input in the compartment definition.

The mass balance equation for dissolved substances in the bulk volume is given as

$$\frac{d}{dt}(V_B \epsilon_{l,B} C_{B,i}) = I_{in,C_i} - \epsilon_{l,B} C_{B,i} Q_{ef} + I_{L,C_i} + V_B r_{C_i} \quad (3.31)$$

where  $\epsilon_{l,B}$  is the liquid phase volume fraction in the bulk volume (the porosity is equal to unity in the bulk volume).

In AQUASIM, two types of reactors are distinguished: A confined reactor has a constant total reactor volume for the biofilm and the bulk water. For this reactor type the bulk volume is calculated as

$$V_B = V_R - \int_0^{L_F} A(z') dz' \quad , \quad V_R = \text{const} \quad (3.32a)$$

where  $V_R$  is the constant total reactor volume. The unconfined reactor type describes a reactor with a constant bulk volume. For this reactor type, the total reactor volume,  $V_R$ , is not constant but is given as the sum of the (variable) biofilm volume and the bulk volume as

$$V_R = \int_0^{L_F} A(z') dz' + V_B \quad , \quad V_B = \text{const} \quad (3.32b)$$

The effluent discharge,  $Q_{ef}$ , depends on the influent and on the varying reactor volume. In the case of the confined reactor

$$Q_{ef} = Q_{in} \quad (3.33a)$$

whereas due to the varying total volume for the unconfined reactor

$$Q_{ef} = Q_{in} - A u_L \quad (3.33b)$$

### User Definitions

Within a biofilm reactor compartment three zones, the **Bulk Volume**, the **Biofilm Matrix** and the **Pore Water** are distinguished. Variables and process rates can be made dependent on the zone by using the program variable **Zone Index** which takes the value 0 in the bulk volume zone, 1 in the biofilm matrix zone and 2 in the pore water zone.

Figure 3.29 shows the dialog box used for defining or editing a biofilm reactor compartment. The edit field '**Name**' is used to specify the name of the compartment. Each

Figure 3.29: Dialog box for editing a biofilm reactor compartment.

compartment needs a unique name as an identifier. A name of a compartment consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

The edit field '**Comp. Index**' can be used to specify a nonnegative integer number as a compartment index. This value can be accessed with the aid of the program variable **Compartment Index** to make variables or process rates dependent on the compartment.

To improve documentation of compartments, the edit field '**Description**' can optionally be used to store comments on specific implementation features of a compartment.

The buttons '**Variables**', '**Processes**', '**Init. Cond.**' and '**Input**' are used to activate and inactivate state variables, to activate and inactivate processes, to specify initial conditions and to define inputs to a compartment, respectively. These options are described later in this subsection.

The buttons '**Particulate Variables**' and '**Dissolved Variables**' are used to define properties of dissolved or particulate substances for dynamic volume state variables. These options are also described later in this subsection.

The next option allows the user to select the reactor type. Selection of the radio button ‘**confined**’ leads to the use of a reactor with a constant total volume for biofilm and bulk fluid. The value for this total reactor volume can be specified in the edit field labelled ‘**Reactor Vol.**’. In such a reactor, growth of the biofilm leads to a decrease in bulk volume as described by equation (3.32a). Selection of the radio button ‘**unconfined**’ leads to the use of a reactor with a constant bulk volume. In this case, a growing biofilm leads to a growth in total reactor volume. The value of the bulk volume can be specified in the edit field ‘**Bulk Volume**’. This option can be used to model free-surface water flow over a biofilm.

The next option is used to select the properties of the pore volume. Selection of the radio button ‘**liquid phase only**’ leads to a model without suspended solids in the pore water. Such a model is adequate to describe a dense biofilm with very small pores in which there is no relevant motion of suspended solids. Selection of the radio button ‘**with suspended solids**’ leads to a model in which solids can penetrate a biofilm within the pore water. Such a model is suitable to describe biofilms with a high porosity.

The radio buttons ‘**rigid**’ and ‘**diffusive**’ allow the user to select how the biofilm matrix should be modelled. In a rigid biofilm matrix there is no diffusive mass transport of solids, whereas in a diffusive biofilm matrix a small effective diffusion is also allowed for solids in the biofilm matrix. This option requires the diffusivities  $D_{M,X_i}$  to be specified as properties of particulate variables as described later in this subsection.

The treatment of surface detachment of solids from the biofilm can be selected with the aid of the radio buttons ‘**individual rate**’ and ‘**global velocity**’. If **individual rate** is selected, detachment is calculated with the aid of the substance-specific detachment rate coefficient,  $k_{de,surf,X_i}$ . This option can only be selected for a diffusive biofilm matrix. If this option is selected, the detachment velocity,  $u_{de}$  is calculated according to the equation (3.21b). If a global detachment velocity,  $u_{de}$ , is specified, solids are detached from the biofilm according to their relative occurrence at the biofilm surface.

The edit field ‘**Biofilm Area**’ is used to specify the area of the biofilm,  $A$ , as a function of the distance from the substratum,  $z$ , which is accessible by the program variable **Space Coordinate Z**. It is not allowed to use a time dependence or a dependence on state variables for the specification of the biofilm area. Some possibilities for this function are: A constant surface area

$$A = \text{const} \quad (3.34a)$$

describes a planar biofilm, a surface area of the form

$$A = 2\pi L_{cy}(r_{cy} + z) \quad (3.34b)$$

describes a biofilm growing on the outside of a cylinder with radius  $r_{cy}$  and length  $L_{cy}$ , a surface area of the form

$$A = 2\pi L_{cy}(r_{cy} - z) \quad (3.34c)$$

describes a biofilm growing on the inside of a cylinder with radius  $r_{cy}$  and length  $L_{cy}$ , and a surface area of the form

$$A = 4\pi n_{sp}(r_{sp} + z)^2 \quad (3.34d)$$

describes biofilms growing on  $n_{sp}$  spherical particles of radius  $r_{sp}$ .

The edit field '**Rate Porosity**' is used to specify the excess growth rate of porosity,  $r'_\theta$ , defined in equation (3.12). If this rate is set to zero and if no volume attachment or detachment is active, the porosity of the biofilm remains constant.

As last options, '**Num. Grid Pts.**', '**Resolution**' and '**Acc.**', in the dialog box used for the definition of a biofilm reactor compartment (Fig. 3.29) the user can select the number of grid points, the discretization order of the numerical algorithm and the accuracies of program variables. The number of grid points is used to specify by how many discrete points the continuous  $z$ -axis is approximated. If the number of grid points is set to  $n_{gp}$ , the depth of the biofilm is resolved into 2 boundary points and  $n_{gp} - 2$  grid points located in the middle of  $n_{gp} - 2$  cells of equal thickness. Two additional grid points are used to describe the boundary layer and the bulk volume. For the division of the  $z$ -axis described above the '**low resolution**' method applies a simple first-order discretization scheme that is very robust but can have significant numerical diffusion. The '**high resolution**' method uses a second-order discretization scheme that applies flux-limiters to avoid oscillations of the numerical solutions (Reichert, 1994b, chapter 6). The high resolution method is not yet implemented in the biofilm compartment of the current version of AQUASIM. The button '**Acc.**' is used to specify the numerical accuracies of program variables as described later in this subsection.

The check box '**active for calculation**' can be used to activate or inactivate the compartment from the calculations. This check box has the same functionality as the buttons **Activate** and **Inactivate** in the dialog box shown in Fig. 3.19.

As for each compartment, the user has to select which state variables are active. This is done by clicking the button '**Variables**' of the dialog box shown in Fig. 3.29. This action opens the dialog box shown in Fig. 3.30. The two list boxes of this dialog box show the

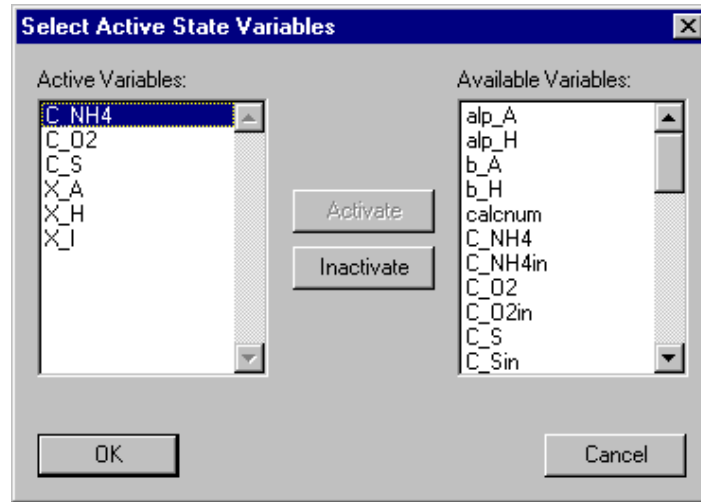


Figure 3.30: Dialog box for activating and inactivating state variables in a compartment.

active variables and the available variables, respectively. The button '**Activate**' is used to activate available variables selected in the right list box, and the button '**Inactivate**' is used to inactivate active variables selected in the left list box. If an active variable is selected while activating another variable, the new active variable is inserted in the list of active variables immediately before the selected variable, otherwise it is appended to the



end of the list of active variables. This gives the user the possibility to influence the order of active variables (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The list of active variables may contain variables of any type, but activation has only an effect to state variables. Inactive state variables return a value of zero. The reason for allowing other types of variables in the list of active state variables is to facilitate switching between different models that do not contain the same state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of active variables of the compartments.

Similarly to the activation of state variables, the user has to select which processes are active in a compartment. This is done by clicking the button '**Processes**' of the dialog box shown in Fig. 3.29. This action opens the dialog box shown in Fig. 3.31. The two list

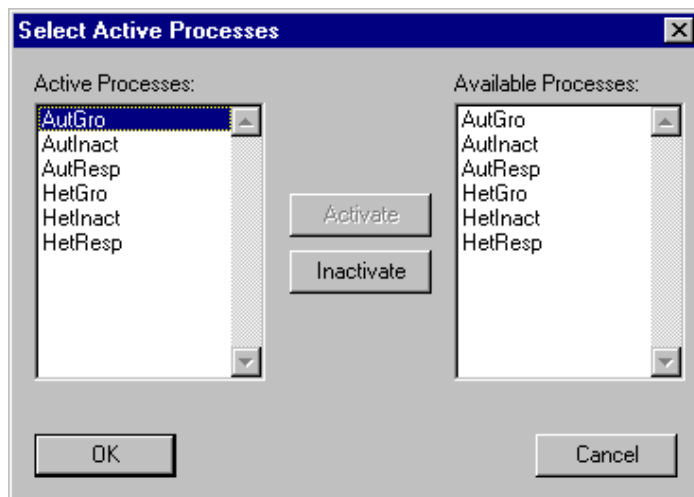


Figure 3.31: Dialog box for activating and inactivating processes in a compartment.

boxes in this dialog box show the active processes and the available processes, respectively. The button '**Activate**' is used to activate available processes selected in the right list box, and the button '**Inactivate**' is used to inactivate an active processes selected in the left list box. If an active process is selected while activating another process, the new active process is inserted in the list of active processes immediately before the selected process, otherwise it is appended to the end of the list of active processes. This gives the user the possibility to influence the order of active processes (the order is irrelevant for the program, but it may be convenient for the user to have a certain order).

Initial conditions for a biofilm reactor compartment can be specified by clicking the button '**Init. Cond.**' of the dialog box shown in Fig. 3.29. This action opens the dialog box shown in Fig. 3.32. This dialog box shows a list of all initial conditions already specified. Each row of the list box contains the name of a variable, followed by the zone of the compartment for which the initial condition is specified (in brackets) and by the algebraic expression specifying the initial value. For each combination of a variable with a zone only one unique initial condition can be specified. The list of initial conditions can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If an initial condition is selected while adding a new initial condition, the new initial condition is inserted in the list immediately before the selected initial condition, otherwise it is appended to the end

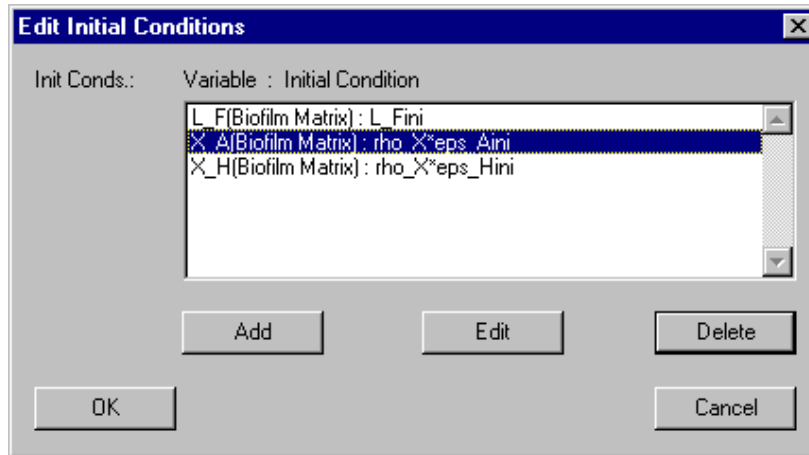


Figure 3.32: Dialog box for editing initial conditions for a biofilm reactor compartment.

of the list of initial conditions. This gives the user the possibility to influence the order of the initial conditions (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The dialog box used for editing a single initial condition is shown in Fig. 3.33. In this dialog box the fields '**Variable**', '**Zone**' and '**Init.**

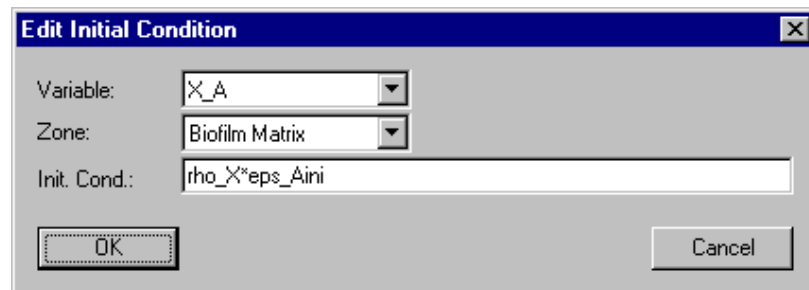


Figure 3.33: Dialog box for editing a single initial condition for a biofilm reactor compartment.

**Cond.'** allow the user to select a variable and a zone and to specify an initial condition. In a biofilm compartment there are the zones **Bulk Volume**, **Biofilm Matrix** and **Pore Volume**. Initial conditions for any type of variables can be specified, but only initial conditions for state variables and for the program variable **Biofilm Thickness** are used by the program. The reason for allowing to define initial conditions for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of initial conditions of the compartments. Initial conditions of a biofilm reactor compartment may not depend on state variables and they can only depend on the program variables **Calculation Number**, **Time** and **Space Coordinate Z**.

Input to a biofilm reactor compartment can be specified by clicking the button '**Input**' of the dialog box shown in Fig. 3.29. This action opens the dialog box shown in Fig. 3.34.

The edit field ‘**Water Inflow**’ of this dialog box is used to specify the discharge of water

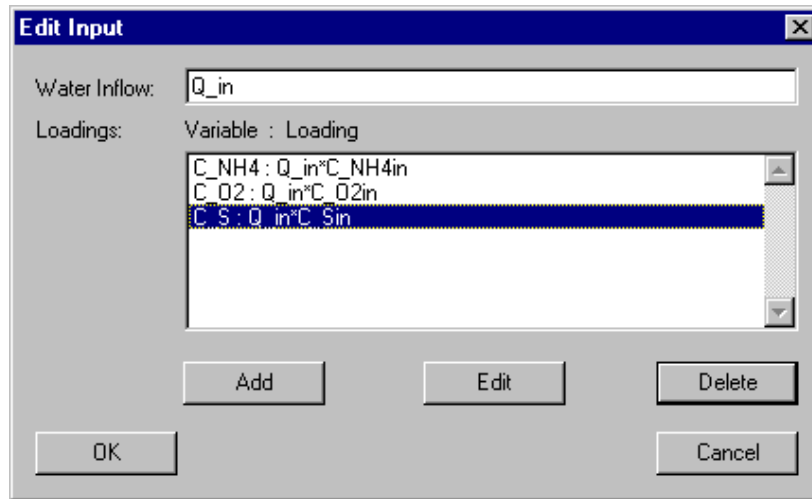


Figure 3.34: Dialog box for editing inputs to a biofilm reactor compartment.

into the bulk volume of the reactor,  $Q_{in}$ , and the list box contains substance loadings,  $I_{in,C_i}$  and  $I_{in,X_i}$ , into the bulk volume. For each variable only one unique loading can be specified. The list of loadings can be edited using the buttons ‘**Add**’, ‘**Edit**’ and ‘**Delete**’. If a loading is selected while adding a new loading, the new loading is inserted in the list immediately before the selected loading, otherwise it is appended to the end of the list of loadings. This gives the user the possibility to influence the order of the loadings (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.35 shows the dialog box used to specify a single input loading of a substance. In this dialog box the fields ‘**Variable**’ and ‘**Loading**’ allow the user to select

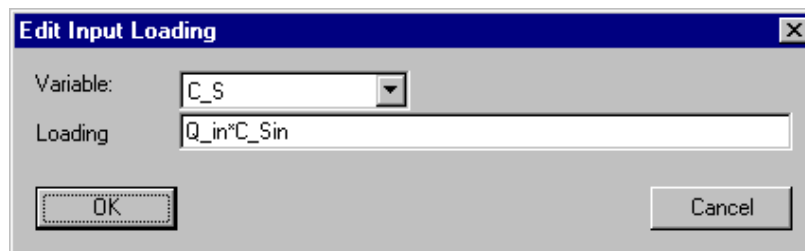


Figure 3.35: Dialog box for editing a single input loading to a biofilm reactor compartment.

a variable and to specify a loading,  $I_{in,C_i}$  or  $I_{in,X_i}$ . Note that this loading represents a mass per unit of time. A loading to a biofilm reactor compartment may depend on the program variables **Discharge** and on dynamic volume state variables. These variables return the discharge and the concentrations of dynamic volume state variables resulting from all advective links connected to the inlet of the reactor. Loadings for any type of variables can be specified, but only loadings for dynamic volume state variables are used by the program. The reason for allowing to define loadings for variables of other types is

to facilitate the user switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of input loadings of the compartments.

The button '**Particulate Variables**' of the dialog box shown in Fig. 3.29 allows the user to specify properties of particulate variables. Figure 3.36 shows the dialog box used for this purpose. It contains a list box with the names of all defined particulate variables.

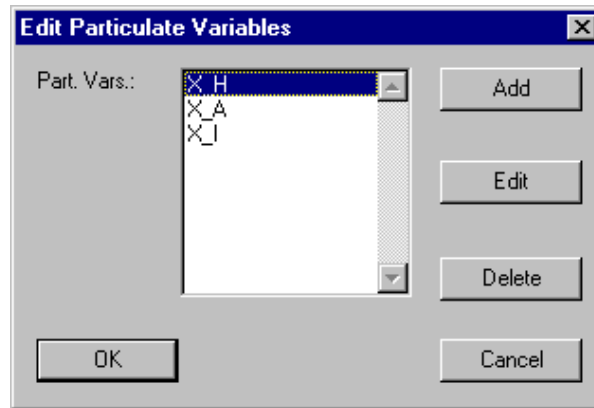


Figure 3.36: Dialog box for editing properties of particulate variables in a biofilm reactor compartment.

Note that a variable can only be in one of the lists of particulate and dissolved variables. The buttons '**Add**', '**Edit**' and '**Delete**' allow the user to edit this list. Figure 3.37 shows the dialog box used to specify the properties of a particulate variable in a biofilm reactor compartment. The field '**Variable**' allows the user to select a variable for which

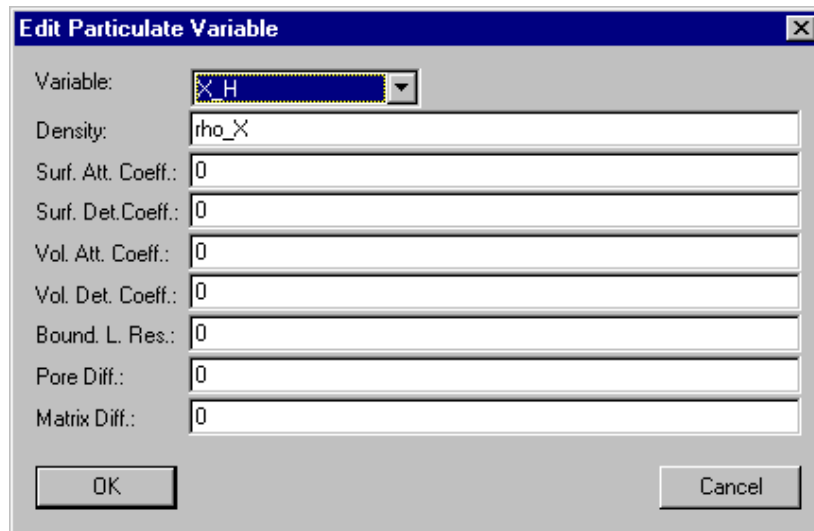


Figure 3.37: Dialog box for editing properties of a single particulate variable in a biofilm reactor compartment.

particulate properties are to be defined. The edit field '**Density**' is used to specify the density,  $\rho_{X_i}$ , of the variable. The expression for the density may not depend on time. The edit fields '**Surf. Att. Coeff.**', '**Surf. Det. Coeff.**', '**Vol. Att. Coeff.**' and '**Vol. Det. Coeff.**' are used to specify expressions for the surface and volume attachment and detachment coefficients,  $k_{at,surf}$ ,  $k_{de,surf}$ ,  $k_{at,vol}$  and  $k_{de,vol}$ , respectively. Note that the values entered for the surface detachment coefficient only has an effect if the radio button **individual rate** of the option **Surf. Detach.** of the dialog box shown in Fig. 3.29 is selected. Similarly the expressions entered for volume attachment and detachment coefficients are only used if the radio button **with free particles** is selected for the option **Pore Volume** of the dialog box shown in Fig. 3.29. The edit field '**Bound. L. Res.**' is used to specify the boundary layer resistance,  $\kappa_{L,X_i}$ , for the particulate species described by the concentration  $X_i$ . The last two edit fields, '**Pore Diff.**' and '**Matrix Diff.**' are used to specify the coefficients for pore volume diffusion,  $D_{P,X_i}$  and  $D_{M,X_i}$ , respectively. Note that the first of these values has only an effect if the radio button **with free particles** is selected for the option 'Pore Volume' of the dialog box shown in Fig. 3.29 and that the second value only has an effect if the radio button **diffusive** for the option **Biofilm Matrix** is selected. Otherwise the values entered here are ignored.

The button '**Dissolved Variables**' of the dialog box shown in Fig. 3.29 allows the user to specify properties of dissolved variables. Figure 3.38 shows the dialog box used for this purpose. It contains a list box with the names of all defined dissolved variables. Note

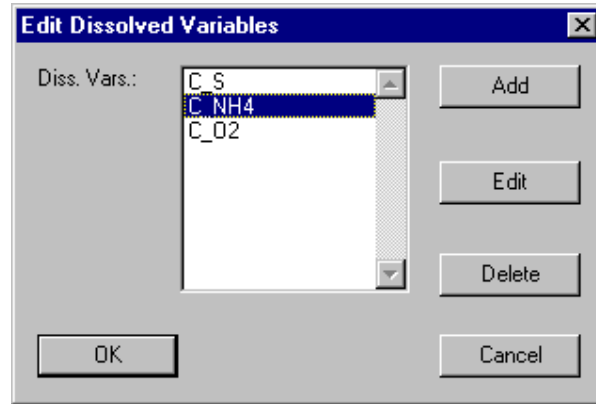


Figure 3.38: Dialog box for editing properties of dissolved variables in a biofilm reactor compartment.

that a variable can only be in one of the lists of particulate and dissolved variables. The buttons '**Add**', '**Edit**' and '**Delete**' allow the user to edit this list. Figure 3.39 shows the dialog box used to specify the properties of a dissolved variable in a biofilm reactor compartment. The field '**Variable**' allows the user to select a variable for which dissolved properties are to be defined. The edit field '**Bound. L. Res.**' is used to specify the boundary layer resistance,  $\kappa_{L,C_i}$ , for the dissolved species described by the concentration  $C_i$ . The last edit field, '**Pore Diff.**' is used to specify the coefficient for pore volume diffusion,  $D_{P,C_i}$  of the dissolved species described by the concentration  $C_i$ .

The button '**Acc.**' of the dialog box shown in Fig. 3.29 is used to specify the numerical accuracies of program variables. Fig. 3.40 shows the dialog box used for this purpose.

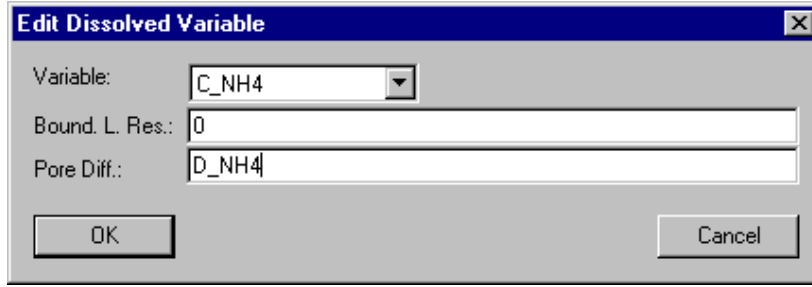


Figure 3.39: Dialog box for editing properties of a single dissolved variable in a biofilm reactor compartment.

It allows the user to specify relative and absolute accuracies of the program variables **‘Discharge’**,  $Q$ , reactor or bulk **‘Volume’**,  $V_R$  or  $V_B$ , **‘Biofilm Thickness’**,  $L_F$ , and **‘Water Fraction’**,  $\epsilon_{l,F}$  (the last value is also used as the accuracy of the volume fractions of particulate species in the biofilm). Good behaviour of the numerical algorithms is usually achieved if the absolute accuracy and the product of the relative accuracy times a typical value of the variable both are 4 to 6 orders of magnitude smaller than typical values of the variable.

In Table 3.5 the program variables available in a biofilm reactor compartment are summarized (for a complete overview of all program variables see Table 3.1 on page 17).

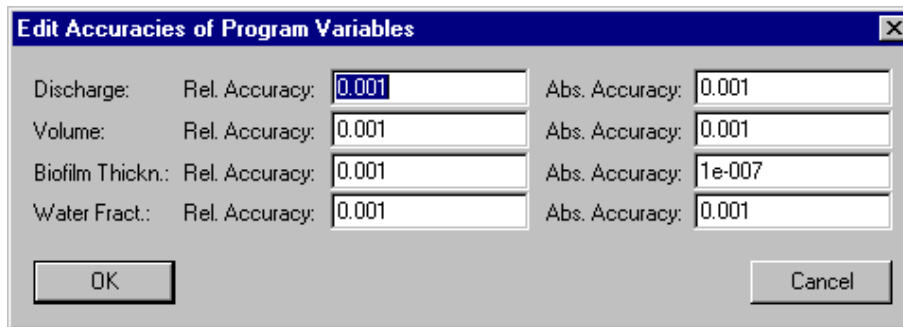


Figure 3.40: Dialog box for editing the accuracies of program variables of a biofilm reactor compartment.

---

Calculation Number	Identifier for calculations (value set in the dialog box shown in Figs. 4.5 and 4.17).
Time	Simulation time.
Compartment Index	Identifier for compartments (value set in the dialog box shown in Fig. 3.29).
Zone Index	Identifier for zones within compartments (returns a value of 0 in the bulk volume zone, a value of 1 in the biofilm matrix, and a value of 2 in the biofilm pore volume).
Discharge	Volumetric flow rate.
Water Fraction	Volumetric fraction of water.
Space Coordinate Z	Depth coordinate in the biofilm (zero at the substratum-biofilm interface).
Reactor Volume	Total volume of the reactor.
Bulk Volume	Volume of mixed water zone outside of the biofilm.
Biofilm Thickness	Thickness of the biofilm.
Growth Velocity of Biofilm	Advective velocity of biofilm solid matrix.
Interface Velocity of Biofilm	Velocity of the interface between biofilm and bulk fluid.
Detachment Velocity of Biofilm	Detachment velocity of particles from the biofilm surface.
Attachment Velocity of Biofilm	Attachment velocity of particles onto the biofilm surface.

---

Table 3.5: Program variables available in the biofilm reactor compartment.

### 3.3.3 Advective-Diffusive Reactor Compartment

#### Overview

The advective-diffusive reactor compartment of AQUASIM can be used to describe one-dimensional advective-diffusive transport of substances in a flow-through reactor, exchange with settled, sorbed or sessile substances or organisms, and substance transformations. The inlet and outlet of the advective-diffusive reactor compartment can be linked advectively to other AQUASIM compartments.

#### Equations Solved by AQUASIM

In order to formulate the one-dimensional conservation laws

$$\frac{\partial \hat{\rho}}{\partial t} + \frac{\partial \hat{j}}{\partial x} = \hat{r} \quad (3.35)$$

compartment-specific expressions for the one-dimensional density,  $\hat{\rho}$  (amount of conserved quantity per unit compartment length), for the one-dimensional flux,  $\hat{j}$  (amount of the conserved quantity transported per unit time), and for the one-dimensional source term,  $\hat{r}$  (amount produced per unit compartment length and per unit time), must be derived.

For the case of the advective-diffusive reactor compartment, 3 types of components of a conservation law must be distinguished. The array of one-dimensional densities of these types of components is given as follows:

$$\hat{\rho} = \begin{pmatrix} A \\ AC_i \\ S_i \end{pmatrix} \quad (3.36)$$

The first component of equation (3.36) describes the conservation of water volume within the compartment (water is approximated to be incompressible). The one-dimensional density of water volume in the compartment (volume per unit length) is given by the cross-sectional area,  $A$ , of the compartment. The second component of equation (3.36) describes substances transported dissolved or suspended with the water flow. Their one-dimensional densities are given as the product of the cross-sectional area,  $rA$ , and the laterally averaged concentration,  $C_i$ . The last component of equation (3.36) describes substances settled to the bottom or sorbed to surfaces within the compartment.

The value of the variable  $A$  must be specified as a function of the distance along the compartment,  $x$ . Its current value is then generally accessible by the program variable **Cross Sectional Area**. The variables  $C_i$  are represented by dynamic volume state variables, and the variables  $S_i$  are represented by dynamic surface state variables. For equilibrium state variables algebraic equations specified as equilibrium processes are solved everywhere along the compartment.

The one-dimensional fluxes of the quantities with one-dimensional densities as described by equation (3.36) are given as follows:

$$\hat{j} = \begin{pmatrix} Q \\ QC_i - AD \frac{\partial C_i}{\partial x} \\ 0 \end{pmatrix} \quad (3.37)$$



In this equation  $Q$  refers to the volumetric discharge through the compartment, and  $D$  is the coefficient of longitudinal diffusion or dispersion.

The following one-dimensional source terms are required to complete the set of equations for the advective-diffusive reactor compartment:

$$\hat{\mathbf{r}} = \begin{pmatrix} q \\ Ar_{C_i} + \left(\frac{1+\text{sign}(q)}{2}\right) q C_{lat,i} + \left(\frac{1-\text{sign}(q)}{2}\right) q C_i \\ r_{S_i} \end{pmatrix} \quad (3.38)$$

The first component of equation (3.38) describes the lateral water inflow as volume per unit length of the compartment,  $q$ . In many cases, this longitudinal inflow is zero and the only inflow is that at the compartment inlet. This inflow is specified as a boundary condition below (3.42a). The second component describes the effect of transformation processes in the compartment and the effect of lateral inflows or outflows ( $C_{lat,i}$  is the concentration in the lateral inflow  $q$ ). The last component describes the effect of transformation processes on settled or sorbed substances or of growing organisms.

Application of the general law for differential conservation laws (3.35) to the definitions given by the equations (3.36) to (3.38) leads to the following set of 3 differential equations.

The first equation describes water flow through the compartment:

$$\frac{\partial Q}{\partial x} = q \quad (3.39)$$

The spatial gradient of the discharge,  $Q$ , is determined by the lateral inflow,  $q$ . Positive values of  $q$  (inflow) increase the downstream discharge, negative values (outflow) decrease the downstream discharge.

The second equation describes the behaviour of substances transported in the compartment:

$$\begin{aligned} \frac{\partial C_i}{\partial t} = & -\frac{1}{A} \frac{\partial}{\partial x} (QC_i) + \frac{1}{A} \frac{\partial}{\partial x} \left( AD \frac{\partial C_i}{\partial x} \right) + r_{C_i} \\ & + \left( \frac{1+\text{sign}(q)}{2} \right) \frac{q}{A} C_{in,i} + \left( \frac{1-\text{sign}(q)}{2} \right) \frac{q}{A} C_i \end{aligned} \quad (3.40)$$

The concentration is affected by advection with the water flow (first term), dispersion or diffusion (second term), transformation processes (third term) and by lateral inflow or outflow (fourth and fifth terms).

The third equation describes the behaviour of settled or sorbed substances or of sessile organisms:

$$\frac{\partial S_i}{\partial t} = r_{S_i} \quad (3.41)$$

The concentration of such substances is only affected by transformation processes. Note that settling or sorption must also be formulated as a transformation process transforming dissolved species,  $C_i$ , to settled or sorbed species,  $S_i$ .

In order to make the solution to the above system of differential equations unique, one boundary condition for the ordinary differential equation (3.39) and two boundary conditions for the partial differential equation (3.40) are required. The ordinary differential equation (3.41) that does not contain spatial derivatives does not require boundary conditions.

The boundary condition for equation (3.39) that describes discharge through the compartment is given by

$$Q(x_s) = Q_{in} \quad (3.42a)$$

at the start point,  $x_s$ , of the column. According to equation (3.39), due to the lateral inflow,  $q$ , this results in a discharge of

$$Q(x_e) = Q(x_s) + \int_{x_s}^{x_e} q \, dx \quad (3.42b)$$

at the end of the compartment.

The boundary conditions for equation (3.40) are given by the continuity of the substance loadings entering the compartment and by a ‘transmission boundary condition’ (Shamir and Harleman, 1967) at the end of the compartment:

$$QC_i(x_s) - AD \frac{\partial C_i}{\partial x}(x_s) = I_{in,C_i} \quad (3.43a)$$

$$\frac{\partial^2 C_i}{\partial x^2} = 0 \quad (3.43b)$$

where  $I_{in,C_i}$  is the total (given) mass input of the substance described by the concentration  $C_i$  per unit of time. The second of these boundary conditions (3.43b) is omitted for dispersion-free transport.

### User Definitions

Figure 3.41 shows the dialog box used for defining or editing an advective-diffusive reactor compartment. The edit field ‘**Name**’ is used to specify the name of the compartment.

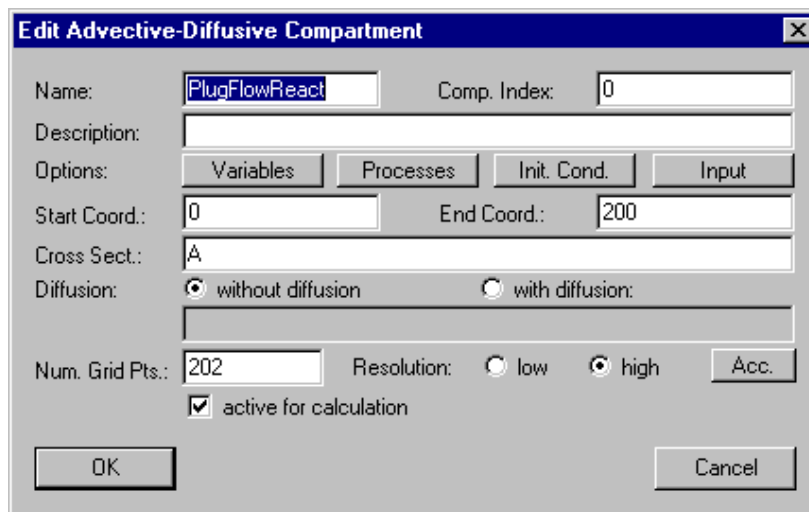


Figure 3.41: Dialog box for editing an advective-diffusive reactor compartment.

Each compartment needs a unique name as an identifier. A name of a compartment consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

The edit field '**Comp. Index**' can be used to specify a nonnegative inter number as a compartment index. This value can be accessed with the aid of the program variable **Compartment Index** to make variables or process rates dependent on the compartment.

To improve documentation of compartments, the edit field '**Description**' can optionally be used to store comments on specific implementation features of a compartment.

The buttons '**Variables**', '**Processes**', '**Init. Cond.**' and '**Input**' are used to activate and inactivate state variables, to activate and inactivate processes, to specify initial conditions and to define inputs to a compartment. These options are described later in this subsection.

The edit fields '**Start Coord.**' and '**End Coord.**' are used to define the location of the compartment inlet,  $x_s$ , and of the compartment outlet,  $x_e$ , respectively. The values of variables are resolved continuously with the space coordinate  $x$  between these two locations. Although all equations in this subsection are formulated with the  $x$ -axis pointing in flow direction ( $x_s < x_e$ ), the program works also correctly with an  $x$ -axis defined in the reverse direction ( $x_s > x_e$ ).

The edit field '**Cross Sect.**' is used to specify the wetted cross-sectional area of the compartment,  $A$ , as a function of the longitudinal spatial coordinate  $x$ , which is accessible by the program variable **Space Coordinate X**. Note that when the advective-diffusive reactor compartment is used to model a porous medium,  $A$  is given as the product of the porosity and the total cross-sectional area. It is not allowed to use a time dependence or a dependence on state variables for the specification of the cross-sectional area.

As a next option, '**Diffusion**', in the dialog box shown in Fig. 3.41 the user can select how to model diffusion or dispersion. If the user selects the radio button '**without diffusion**', a purely advective equation with  $D = 0$  is solved (note, however, that due to the spatial discretization numerical diffusion can occur). If the user selects the radio button '**with diffusion**' a positive diffusion or dispersion coefficient  $D$  must be specified. This coefficient can be made dependent on time, discharge or cross-sectional area by using the program variables **Time**, **Discharge** or **Cross-Sectional Area**.

As last options, '**Num. Grid Pts.**', '**Resolution**' and '**Acc.**', in the dialog box used for the definition of an advective-diffusive reactor compartment (Fig. 3.41) the user can select the number of grid points, the discretization order of the numerical algorithm and the accuracies of program variables. The number of grid points is used to specify by how many discrete points the continuous  $x$ -axis is approximated. If the number of grid points is set to  $n_{gp}$ , the compartment is resolved longitudinally into 2 boundary points and  $n_{gp} - 2$  grid points located in the middle of  $n_{gp} - 2$  cells of equal thickness. For this division of the  $x$ -axis the '**low resolution**' method applies a simple first-order discretization scheme that is very robust but can have significant numerical diffusion. The '**high resolution**' method uses a second-order discretization scheme that applies flux-limiters to avoid oscillations of the numerical solutions (Reichert, 1994b, chapter 6). The button '**Acc.**' is used to specify the numerical accuracies of program variables as described later in this subsection.

The check box '**active for calculation**' can be used to activate or inactivate the compartment from the calculations. This check box has the same functionality as the buttons **Activate** and **Inactivate** in the dialog box shown in Fig. 3.19.

As for each compartment, the user has to select which state variables are active. This is done by clicking the button '**Variables**' of the dialog box shown in Fig. 3.41. This action

opens the dialog box shown in Fig. 3.42. The two list boxes of this dialog box show the

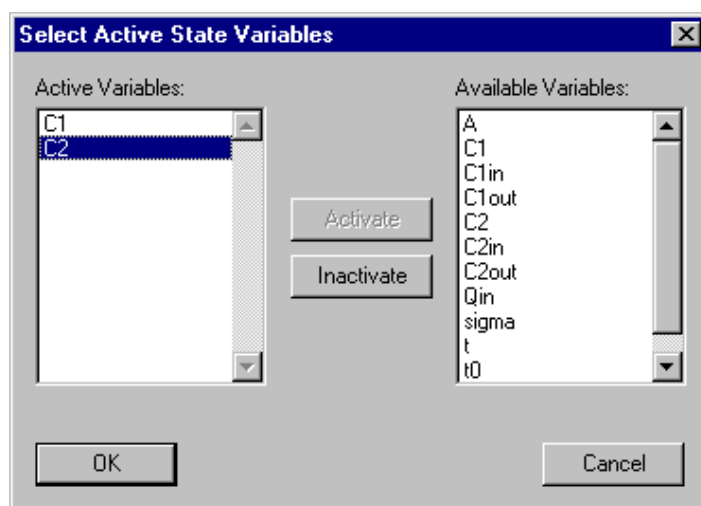


Figure 3.42: Dialog box for activating and inactivating state variables in a compartment.

active variables and the available variables, respectively. The button '**Activate**' is used to activate available variables selected in the right list box, and the button '**Inactivate**' is used to inactivate active variables selected in the left list box. If an active variable is selected while activating another variable, the new active variable is inserted in the list of active variables immediately before the selected variable, otherwise it is appended to the end of the list of active variables. This gives the user the possibility to influence the order of active variables (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The list of active variables may contain variables of any type, but activation has only an effect to state variables. Inactive state variables return a value of zero. The reason for allowing other types of variables in the list of active state variables is to facilitate switching between different models that do not contain the same state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of active variables of the compartments.

Similarly to the activation of state variables, the user has to select which processes are active in a compartment. This is done by clicking the button '**Processes**' of the dialog box shown in Fig. 3.41. This action opens the dialog box shown in Fig. 3.43. The two list boxes in this dialog box show the active processes and the available processes, respectively. The button '**Activate**' is used to activate available processes selected in the right list box, and the button '**Inactivate**' is used to inactivate a active processes selected in the left list box. If an active process is selected while activating another process, the new active process is inserted in the list of active processes immediately before the selected process, otherwise it is appended to the end of the list of active processes. This gives the user the possibility to influence the order of active processes (the order is irrelevant for the program, but it may be convenient for the user to have a certain order).

Initial conditions for an advective-diffusive reactor compartment can be specified by clicking the button '**Init. Cond.**' of the dialog box shown in Fig. 3.41. This action opens the dialog box shown in Fig. 3.44. This dialog box shows a list of all initial conditions

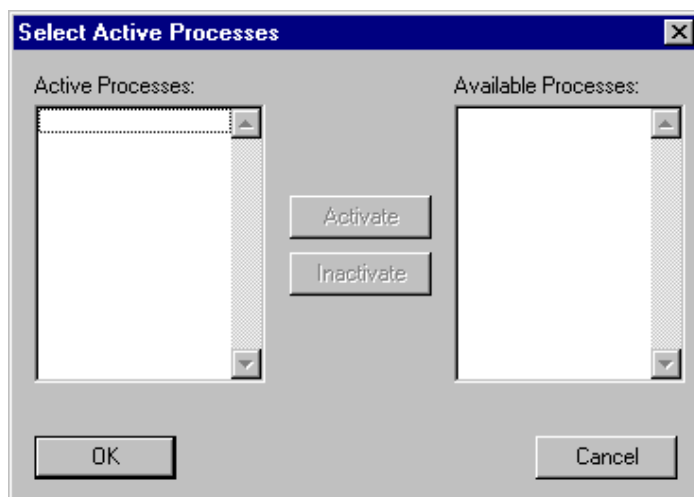


Figure 3.43: Dialog box for activating and inactivating processes in a compartment.

already specified. Each line of the list box contains the name of a variable, followed by the zone of the compartment for which the initial condition is specified (in brackets) and by the algebraic expression specifying the initial value. For each combination of a variable with a zone only one unique initial condition can be specified. The list of initial conditions can be edited using the buttons ‘**Add**’, ‘**Edit**’ and ‘**Delete**’. If an initial condition is selected while adding a new initial condition, the new initial condition is inserted in the list immediately before the selected initial condition, otherwise it is appended to the end of the list of initial conditions. This gives the user the possibility to influence the order of the initial conditions (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The dialog box used for editing a single initial condition is shown in Fig. 3.45. In this dialog box the fields ‘**Variable**’ and ‘**Init. Cond.**’ allow the user to select a variable and to specify an initial condition. The field ‘**Zone**’ is inactive, because the advective-diffusive reactor compartment contains only one zone **Water Body**. Initial conditions for any type of variables can be specified, but only initial conditions for state variables are used by the program. The reason for allowing to define initial conditions for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of initial conditions of the compartments. Initial conditions of an advective-diffusive reactor compartment may not depend on state variables and they can only depend on the program variables **Calculation Number**, **Time** and **Space Coordinate X**.

Input to an advective-diffusive reactor compartment can be specified by clicking the button ‘**Input**’ of the dialog box shown in Fig. 3.41. This action opens the dialog box shown in Fig. 3.46. In this dialog box the user can select which type of input to edit. There exist two different types of inputs to an advective-diffusive reactor compartment. The radio button ‘**Inlet Input**’ can be used to describe water and substance flow at the inlet of the compartment and the radio button ‘**Lateral Input**’ makes it possible to specify water and substance inflow along the compartment. In the following paragraphs, the dialog boxes used to define each of these input types are described.

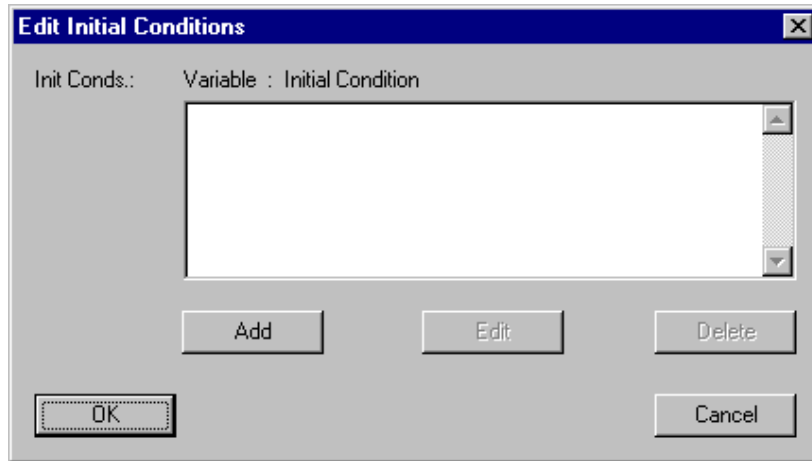


Figure 3.44: Dialog box for editing initial conditions for an advective-diffusive reactor compartment.



Figure 3.45: Dialog box for editing a single initial condition for an advective-diffusive reactor compartment.

Selection of the radio button '**Inlet Input**' in the dialog box shown in Fig. 3.46 opens the dialog box shown in Fig. 3.47. The edit field '**Water Inflow**' of this dialog box is used to specify the discharge of water into the compartment,  $Q_{in}$ , and the list box contains substance loadings,  $I_{in,C_i}$ , into the compartment. For each variable only one unique inlet loading can be specified. The list of inlet loadings can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If a loading is selected while adding a new loading, the new loading is inserted in the list immediately before the selected loading, otherwise it is appended to the end of the list of the loadings. This gives the user the possibility to influence the order of the loadings (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.48 shows the dialog box used to specify a single inlet loading of a substance. In this dialog box the fields '**Variable**' and '**Loading**' allow the user to select a variable and specify an inlet loading,  $I_{in,C_i}$ . Note that this input loading represents a mass per unit of time. An inlet loading to an advective-diffusive reactor compartment may depend on the program variables **Discharge** and on dynamic volume state variables. These variables return the discharge and the concentrations of dynamic volume state variables resulting from all advective links connected to the inlet

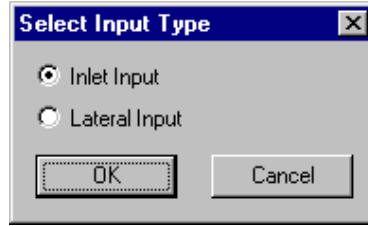


Figure 3.46: Dialog box for selecting an input type for an advective-diffusive reactor compartment.

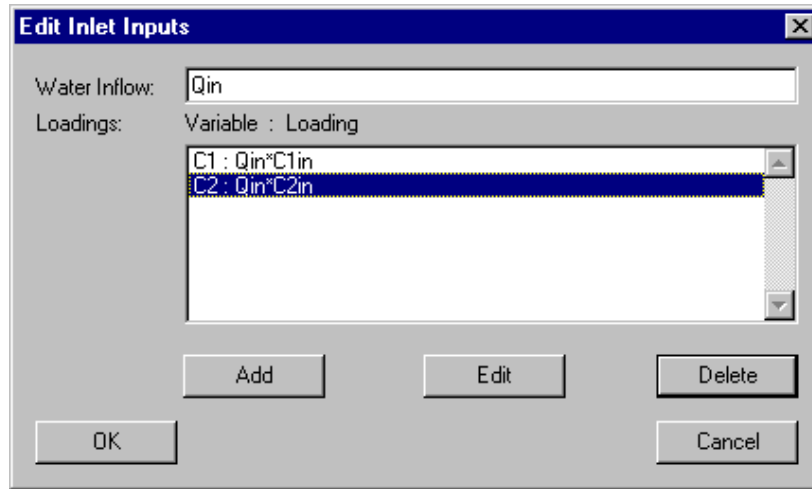


Figure 3.47: Dialog box for editing inlet inputs to an advective-diffusive reactor compartment.

of the reactor. Loadings for any type of variables can be specified, but only loadings for dynamic volume state variables are used by the program. The reason for allowing to define loadings for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of loadings of the compartments.

Selection of the radio button '**Lateral Input**' in the dialog box shown in Fig. 3.46 opens the dialog box shown in Fig. 3.49. The edit field '**Water Inflow**' of this dialog box is used to specify the discharge of water per unit length into the compartment,  $q$ , and the list box contains substance concentrations,  $C_{lat,i}$ , in the inflowing water. A positive value of  $q$  represents a flow into the compartment, a negative value represents an outflow. According to the equations (3.38) and (3.40) the inflow concentration is irrelevant if  $q$  is smaller than or equal to zero (outflow). For each variable only one unique lateral inflow concentration can be specified. The list of inflow concentrations can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If an inflow concentration is selected while adding a new inflow concentration, the new inflow concentration is inserted in the list immediately before the selected inflow concentration, otherwise it is appended to the end of the list of inflow concentrations. This gives the user the possibility to influence the order of the

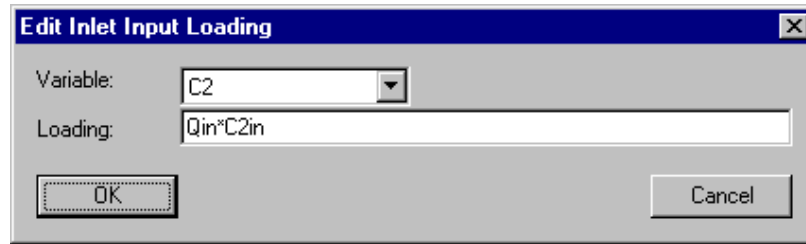


Figure 3.48: Dialog box for editing a single upstream input to an advective-diffusive reactor compartment.

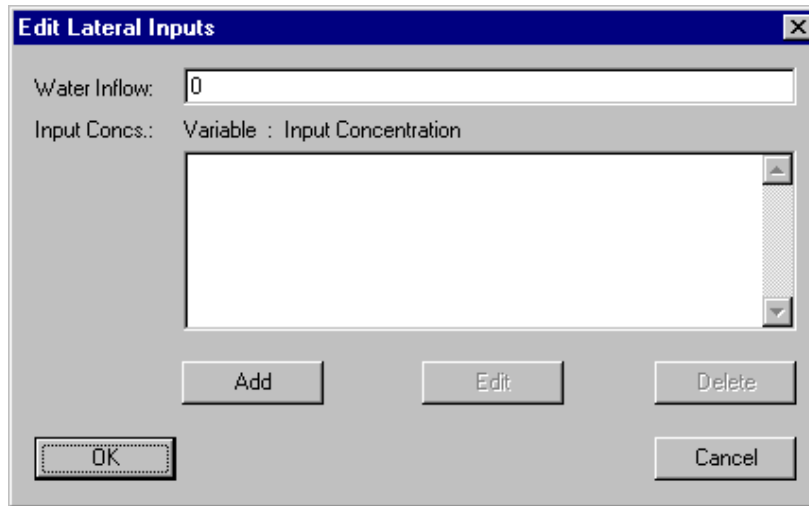


Figure 3.49: Dialog box for editing lateral inputs to an advective-diffusive reactor compartment.

inflow concentrations (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.50 shows the dialog box used to specify a single lateral inflow concentration of a dynamic volume state variable. In this dialog box the fields '**Variable**' and '**Inflow Conc.**' allow the user to select a variable and specify a lateral inflow concentration,  $C_{lat,i}$ . A lateral inflow concentration of an advective-diffusive reactor compartment may depend on the program variables **Discharge** and on state variables. These variables return the discharge and the current values of state variables in the compartment as a function of the location  $x$ . Inflow concentrations for any type of variables can be specified, but only inflow concentrations for dynamic volume state variables are used by the program. The reason for allowing to define inflow concentrations for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of lateral inflow concentrations of the compartments.

The button '**Acc.**' of the dialog box shown in Fig. 3.41 is used to specify the numerical accuracies of program variables. Fig. 3.51 shows the dialog box used for this purpose. It allows the user to specify relative and absolute accuracies of the program variable



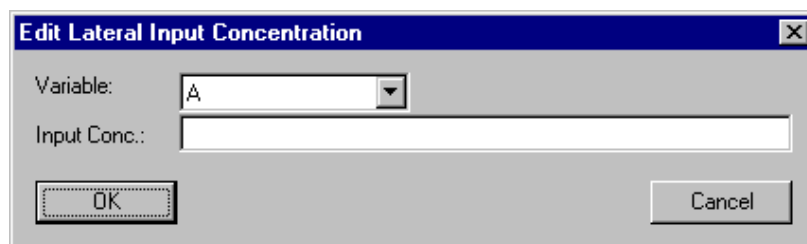


Figure 3.50: Dialog box for editing a single lateral inflow concentration to an advective-diffusive reactor compartment.

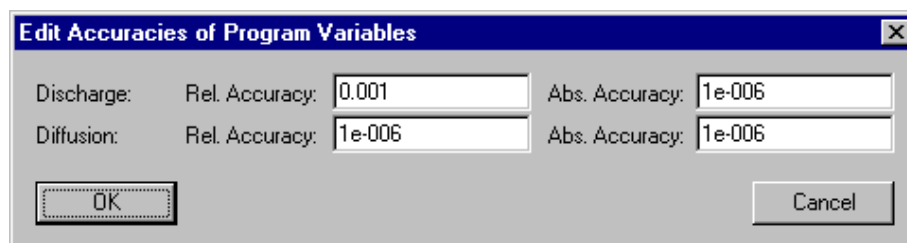


Figure 3.51: Dialog box for editing the accuracies of program variables of an advective-diffusive compartment.

‘**Discharge**’,  $Q$ , and for the ‘**Diffusion**’ coefficient,  $D$ , in the compartment. Good behaviour of the numerical algorithms is usually achieved if the absolute accuracy and the product of the relative accuracy times a typical value of the variable both are 4 to 6 orders of magnitude smaller than typical values of the variable.

In Table 3.6 the program variables available in a advective-diffusive compartment are summarized (for a complete overview of all program variables see Table 3.1 on page 17).

---

Calculation Number	Identifier for calculations (value set in the dialog boxes shown in Figs. 4.5 and 4.17).
Time	Simulation time.
Compartment Index	Identifier for compartments (value set in the dialog box shown in Fig. 3.41).
Zone Index	Identifier for zones within compartments (returns a value of 0).
Discharge	Volumetric flow rate.
Water Fraction	Volumetric fraction of water (returns a value of 1).
Space Coordinate X	Space coordinate along the compartment.
Cross Sectional Area	Area of water body perpendicular to the flow direction.

---

Table 3.6: Program variables available in the advective-diffusive compartment.

### 3.3.4 Saturated Soil Column Compartment

#### Overview

The saturated soil column compartment of AQUASIM can be used to describe advective-dispersive transport of dissolved substances in a saturated soil column, exchange processes with immobile regions consisting of serially connected mixed zones, and transformations of dissolved and sorbed substances. Within the soil column, fast sorption processes can be used to describe equilibrium sorption and slower sorption processes to model the effects of sorption kinetics. The use of any linear or nonlinear sorption isotherm is possible. The inlet and outlet of the soil column compartment can be advectively linked to other AQUASIM compartments.

#### Equations Solved by AQUASIM

In order to formulate the one-dimensional conservation laws

$$\frac{\partial \hat{\rho}}{\partial t} + \frac{\partial \hat{\mathbf{j}}}{\partial x} = \hat{\mathbf{r}} \quad (3.44)$$

compartment-specific expressions for the one-dimensional density,  $\hat{\rho}$  (amount of conserved quantity per unit compartment length), for the one-dimensional flux,  $\hat{\mathbf{j}}$  (amount of the conserved quantity transported per unit time), and for the one-dimensional source term,  $\hat{\mathbf{r}}$  (amount produced per unit compartment length and per unit time), must be derived.

The soil column compartment has a variable number of zones. Variables present in more than one zone are distinguished by the index  $mob$  for the mobile zone and  $im_{jk}$  for the mixed zone  $k$  of the immobile region  $j$ . Figure 3.52 shows schematically the arrangement and enumeration of the zones. The length of the soil column is resolved by

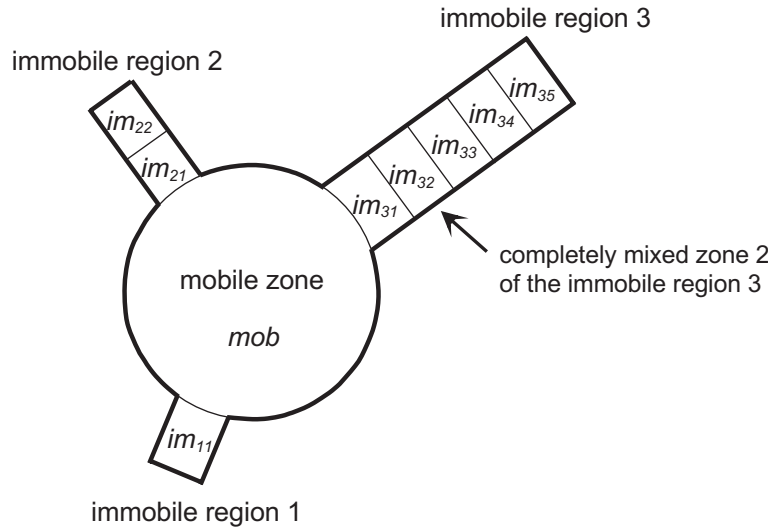


Figure 3.52: Schematic cross-section through the soil column indicating the enumeration of the immobile zones (in the example, there are 3 types of immobile regions consisting of 1, 2 and 5 mixed zones, respectively).

the space coordinate  $x$ .  $x$  can increase or decrease along the column, however, in the following the equations are formulated with increasing  $x$  in flow direction.

In order to formulate the equations for the saturated soil column, 5 types of components of a conservation law must be distinguished. The array of one-dimensional densities of these types of components is given as follows:

$$\hat{\rho} = \begin{pmatrix} A\theta_{mob} \\ A\theta_{mob}C_{mob,i} \\ A\theta_{im_{jk}}C_{im_{jk},i} \\ S_{mob,i} \\ S_{im_{jk},i} \end{pmatrix} \quad (3.45)$$

The first component of equation (3.45) describes the conservation of water volume within the mobile zone of the soil column (water is approximated to be incompressible, and the volume of the immobile zones remains constant). The one-dimensional density of water volume in the mobile zone of the column (volume in the mobile zone per unit length of the column) is given by the product of the cross-sectional area,  $A$ , and the porosity of the mobile zone,  $\theta_{mob}$  of the column. The second component of equation (3.45) describes dissolved substances in the mobile zone of the saturated soil column. Their one-dimensional densities are given as the product of the cross-sectional area,  $A$ , the porosity of the mobile zone,  $\theta_{mob}$ , and the laterally averaged concentration in the mobile zone,  $C_{mob,i}$ . The third component of equation (3.45) describes dissolved substances in the mixed zone  $k$  of the immobile region  $j$ . Their one-dimensional densities are given as the product of the cross-sectional area,  $A$ , the porosity of the zone,  $\theta_{im_{jk}}$ , and the laterally averaged concentration,  $C_{im_{jk},i}$ . The two last components of equation (3.45) describe substances sorbed to surfaces in contact with the mobile zone or with the mixed zone  $k$  of the immobile region  $j$ , respectively.

The value of the variable  $A$  must be specified as a function of the distance along the compartment,  $x$ . Its value is then accessible by the program variable **Cross Sectional Area**. The variables  $C_{mob,i}$  and  $C_{im_{jk},i}$  are represented by dynamic volume state variables, and the variables  $S_{mob,i}$  and  $S_{im_{jk},i}$  are represented by dynamic surface state variables. For equilibrium state variables, algebraic equations, specified as equilibrium processes, are solved in all zones everywhere along the column.

The one-dimensional fluxes of the substances with one-dimensional densities as described by equation (3.45) are given as follows:

$$\hat{\mathbf{j}} = \begin{pmatrix} Q \\ QC_{mob,i} - A\theta_{mob}E\frac{\partial C_{mob}}{\partial x} \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (3.46)$$

In this equation  $Q$  refers to the volumetric discharge through the mobile zone of the saturated soil column, and  $E$  is the coefficient of longitudinal dispersion. There are no fluxes in the direction along the column within the immobile zones and sorbed substances are not transported at all. The lateral diffusive exchange processes of dissolved substances between the mobile zone and the immobile regions and within the immobile regions are described as source terms in the next paragraph.

The following one-dimensional source terms are required to complete the set of equations for the saturated soil column:

$$\hat{\mathbf{r}} = \begin{pmatrix} q \\ -\sum_{j=1}^{n_r} q_{ex,im_{j1},i}(C_{mob,i} - f_{ex,im_{j1},i}C_{im_{j1},i}) + A\theta_{mob}rC_{mob} \\ + \left(\frac{1+\text{sign}(q)}{2}\right)qC_{lat,i} + \left(\frac{1-\text{sign}(q)}{2}\right)qC_{mob,i} \\ \begin{cases} q_{ex,im_{j1},i}(C_{mob,i} - f_{ex,im_{j1},i}C_{im_{j1},i}) \\ - q_{ex,im_{j2},i}(C_{im_{j1},i} - f_{ex,im_{j2},i}C_{im_{j2},i}) \\ + A\theta_{im_{j1}}rC_{im_{j1},i} \end{cases} & \text{for } k = 1 \\ \begin{cases} q_{ex,im_{jk},i}(C_{im_{jk-1},i} - f_{ex,im_{jk},i}C_{im_{jk},i}) \\ - q_{ex,im_{jk+1},i}(C_{im_{jk},i} - f_{ex,im_{jk+1},i}C_{im_{jk+1},i}) \\ + A\theta_{im_{jk}}rC_{im_{jk},i} \end{cases} & \text{for } 1 < k < n_{z,j} \\ \begin{cases} q_{ex,im_{jn_{z,j}},i}(C_{im_{jn_{z,j}-1},i} - f_{ex,im_{jn_{z,j}},i}C_{im_{jn_{z,j}},i}) \\ + A\theta_{im_{jn_{z,j}}}rC_{im_{jn_{z,j}},i} \end{cases} & \text{for } k = n_{z,j} \\ rS_{mob,i} \\ rS_{im_{jk},i} \end{pmatrix} \quad (3.47)$$

The first component of equation (3.47) describes the lateral water inflow as volume per unit length of the column,  $q$ . The second component describes the effect on the mobile zone concentrations of the exchange between the mobile zone and the first mixed zones  $im_{11}$  to  $im_{n_r,1}$  of the  $n_r$  immobile regions that are adjacent to the mobile zone, the effect of transformation processes in the mobile zone and the effect of lateral inflows or outflows ( $C_{lat,i}$  is the concentration in the lateral inflow  $q$ ). The third component describes the effect of the exchange with neighbouring zones of the  $n_{z,j}$  serially connected mixed zones of the immobile region  $j$ . The first row of this component describes the exchange of the first mixed zone with the mobile zone and the second mixed zone, the second row the exchange of an intermediate mixed zone with the neighbouring mixed zones, and the last row the exchange of the last mixed zone with the second to last mixed zone of the immobile region  $j$ . All exchange processes are assumed to be proportional to the difference between the concentration in one of the adjacent zones and a concentration multiplied with a conversion factor,  $f_{ex,im_{jk},i}$ , in the other zone. The consideration of a conversion factor makes the description of systems with different solvents possible. If all zones are filled with the same solvent, all conversion factors are unity. The substance-specific exchange coefficient is  $q_{ex,im_{jk},i}$  represents the volume exchange rate per unit length of the soil column for the substance described by the concentration  $C_i$ .

Application of the general law for differential conservation laws (3.44) to the definitions given by the equations (3.45) to (3.47) leads to the following set of 5 differential equations.

The first equation describes water flow through the soil column:

$$\frac{\partial Q}{\partial x} = q \quad (3.48)$$

the spatial gradient of the discharge,  $Q$ , is determined by the lateral inflow,  $q$ . Positive values of  $q$  (inflow) increase the downstream discharge, negative values (outflow) decrease the downstream discharge.

The second equation describes the behaviour of dissolved substances in the mobile zone of the column:

$$\begin{aligned} \frac{\partial C_{mob,i}}{\partial t} = & -\frac{1}{A\theta_{mob}} \frac{\partial}{\partial x} (QC_{mob,i}) + \frac{1}{A\theta_{mob}} \frac{\partial}{\partial x} \left( A\theta_{mob} E \frac{\partial C_{mob,i}}{\partial x} \right) \\ & - \sum_{j=1}^{n_r} \frac{q_{ex,im_{j1},i}}{A\theta_{mob}} (C_{mob,i} - f_{ex,im_{j1},i} C_{im_{j1},i}) + r_{C_{mob,i}} \\ & + \left( \frac{1+\text{sign}(q)}{2} \right) \frac{q}{A\theta_{mob}} C_{in,i} + \left( \frac{1-\text{sign}(q)}{2} \right) \frac{q}{A\theta_{mob}} C_{mob,i} \quad (3.49) \end{aligned}$$

The concentration is affected by advection with the water flow (first term), dispersion (second term), exchange with the immobile zones adjacent to the mobile zone (third term), transformation processes (fourth term) and by lateral inflow or outflow (fifth and sixth terms).

The third equation describes the behaviour of dissolved substances in the mixed zone  $k$  of the immobile region  $j$ :

$$\frac{\partial C_{im_{jk},i}}{\partial t} = \begin{cases} \frac{q_{ex,im_{j1},i}}{A\theta_{im_{j1}}} (C_{mob,i} - f_{ex,im_{j1},i} C_{im_{j1},i}) \\ \quad - \frac{q_{ex,im_{j2},i}}{A\theta_{im_{j1}}} (C_{im_{j1},i} - f_{ex,im_{j2},i} C_{im_{j2},i}) \\ \quad + r_{C_{im_{j1},i}} & \text{for } k = 1 \\ \frac{q_{ex,im_{jk},i}}{A\theta_{im_{jk}}} (C_{im_{jk-1},i} - f_{ex,im_{jk},i} C_{im_{jk},i}) \\ \quad - \frac{q_{ex,im_{jk+1},i}}{A\theta_{im_{jk}}} (C_{im_{jk},i} - f_{ex,im_{jk+1},i} C_{im_{jk+1},i}) \\ \quad + r_{C_{im_{jk},i}} & \text{for } 1 < k < n_{z,j} \\ \frac{q_{ex,im_{jn_{z,j}},i}}{A\theta_{im_{jn_{z,j}}}} (C_{im_{jn_{z,j}-1},i} - f_{ex,im_{jn_{z,j}},i} C_{im_{jn_{z,j}},i}) \\ \quad + r_{C_{im_{jn_{z,j}},i}} & \text{for } k = n_{z,j} \end{cases} \quad (3.50)$$

The first row of this equation describes the behaviour of a dissolved substance within the first mixed zone of an immobile region, the second row the behaviour in an inner zone, and the third row that in an end zone of the immobile region.

The fourth and fifth equations describe the behaviour of sorbed substances in the mobile zone and the immobile zones respectively:

$$\frac{\partial S_{mob,i}}{\partial t} = r_{S_{mob,i}} \quad (3.51)$$

$$\frac{\partial S_{im_{jk},i}}{\partial t} = r_{S_{im_{jk},i}} \quad (3.52)$$

The concentrations are only influenced by transformation processes. Note that sorption must also be formulated as a transformation process transforming the dissolved species,  $C_i$ , to the sorbed species,  $S_i$ . When describing sorbing substances, the conventional notation

of  $C$  is mass of a substance per unit liquid volume and of  $S$  is mass of a substance per unit mass of the solid phase. If the concentrations are expressed as mass per unit of total column volume,  $C$  in zone  $zo$  must be converted to  $\theta_{zo}C$  (multiplication of  $C$  with the porosity of the zone), and  $S$  must be converted to  $S\rho_{solid}(1 - \theta)$  (multiplication with the density of the solid phase and with the volume fraction of the solid phase). Considering these conversion factors, sorption can be described by a dynamic sorption process with a process rate of

$$k_i \left( S_{eq,i}(C_i) - S_i \right) \quad (3.53a)$$

and with stoichiometric coefficients of

$$C_i : -\rho_{solid} \frac{1 - \theta}{\theta_{zo}} \quad (3.53b)$$

for the dissolved concentration (the index of the zone,  $zo$ , must be replaced by  $mob$  or  $im_{jk}$ ),  $C_i$ , and

$$S_i : 1 \quad (3.53c)$$

for the sorbed concentration,  $S_i$ . In these equations  $\rho_{solid}$  is the density of the solid material in the soil column,  $S_{eq,i}(C_i)$  is the equilibrium isotherm and the process describes relaxation of the actually sorbed concentration to the equilibrium concentration with a rate constant  $k_i$ . If  $k_i$  is set to a sufficiently large value, this model is a good approximation to equilibrium sorption.

In order to make the solution to the above system of differential equations unique, one boundary condition for the ordinary differential equation (3.48) and two boundary conditions for the partial differential equation (3.49) are required. The ordinary differential equations (3.50) to (3.52), that do not contain spatial derivatives, do not require boundary conditions.

The boundary condition for equation (3.48) that describes discharge through the soil column is given by

$$Q(x_s) = Q_{in} \quad (3.54a)$$

at the start point,  $x_s$ , of the column. According to equation (3.48), due to the lateral inflow,  $q$ , this results in a discharge of

$$Q(x_e) = Q(x_s) + \int_{x_s}^{x_e} q \, dx \quad (3.54b)$$

at the column outlet.

The boundary conditions for equation (3.49) are given by the continuity of the substance mass flows entering the column and by a ‘transmission boundary condition’ (Shamir and Harleman, 1967) at the outlet:

$$Q(x_s)C_{mob,i} - A\theta_{mob}E \frac{\partial C_{mob,i}}{\partial x} = I_{in,i} \quad (3.55a)$$

$$\frac{\partial^2 C_{mob,i}}{\partial x^2} = 0 \quad (3.55b)$$

where  $I_{in,i}$  is the total (given) mass input of substance  $i$  per unit of time. The second of these boundary conditions (3.55b) is omitted for dispersion-free transport.

### User Definitions

Within a saturated soil compartment a mobile zone and a user-defined number of immobile zones are distinguished. Variables and process rates can be made dependent on the zone by using the program variable **Zone Index**, which takes the value 0 in the mobile zone and a user-defined positive integer value in each of the immobile zones.

Figure 3.53 shows the dialog box used for defining or editing a soil column compartment. The edit field '**Name**' is used to specify the name of the compartment. Each

The dialog box 'Edit Saturated Soil Column Compartment' contains the following fields and controls:

- Name:** Text field containing 'column'.
- Comp. Index:** Text field containing '0'.
- Description:** Empty text field.
- Options:** Four buttons: 'Variables', 'Processes', 'Init. Cond.', and 'Input'.
- Start Coord.:** Text field containing '0'.
- End Coord.:** Text field containing '12.21'.
- Cross Sect.:** Text field containing 'A'.
- Mob. Vol. Fract.:** Text field containing 'theta-theta\_im'.
- Dispersion:** Radio buttons for 'without dispersion' (selected) and 'with dispersion'.
- Immobile Reg.:** A list box containing 'pores' with 'Add', 'Edit', and 'Delete' buttons to its right.
- Num. Grid Pts.:** Text field containing '15'.
- Resolution:** Radio buttons for 'low' and 'high' (selected), with an 'Acc.' button to the right.
- active for calculation:** A checked checkbox.
- Buttons:** 'OK' and 'Cancel' at the bottom.

Figure 3.53: Dialog box for editing a soil column compartment.

compartment needs a unique name as an identifier. A name of a compartment consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

The edit field '**Comp. Index**' can be used to specify a nonnegative inter number as a compartment index. This value can be accessed with the aid of the program variable **Compartment Index** to make variables or process rates dependent on the compartment.

To improve documentation of compartments, the edit field '**Description**' can optionally be used to store comments on specific implementation features of a compartment.

The buttons '**Variables**', '**Processes**', '**Init. Cond.**' and '**Input**' are used to activate and inactivate state variables, to activate and inactivate processes, to specify initial conditions and to define inputs to a compartment. These options are described later in this subsection.

The edit fields '**Start Coord.**' and '**End Coord.**' are used to define the location of the column inlet,  $x_s$ , and of the column outlet,  $x_e$ , respectively. The values of variables are resolved continuously with the space coordinate  $x$  between these two locations. Although all equations in this subsection are formulated with the  $x$ -axis pointing in flow direction



( $x_s < x_e$ ), the program works also correctly with an  $x$ -axis defined in the reverse direction ( $x_s > x_e$ ).

The edit field ‘**Cross Sect.**’ is used to specify the total cross-sectional area of the compartment,  $A$ , as a function of the longitudinal spatial coordinate  $x$ , which is accessible by the program variable **Space Coordinate X**. It is not allowed to use a time dependence or a dependence on state variables for the specification of the cross-sectional area.

The edit field ‘**Mob. Vol. Fract.**’ is used to specify the porosity of the mobile zone,  $\theta_{mob}$ , of the soil column. Similarly to the cross-sectional area,  $A$ ,  $\theta_{mob}$  can be a function of the space coordinate  $x$  and it is not allowed to use a time dependence in this variable.

As a next option, ‘**Dispersion**’, in the dialog box shown in Fig. 3.53 the user can select how to model dispersion. If the user selects the radio button ‘**without dispersion**’, a purely advective equation with  $D = 0$  is solved (note, however, that due to the spatial discretization numerical diffusion can occur). If the user selects the radio button ‘**with dispersion**’ a positive dispersion coefficient,  $D$ , must be specified. This coefficient can be made dependent on time, discharge or cross-sectional area by using the program variables **Time**, **Discharge** or **Cross-Sectional Area**.

The list box ‘**Immobile Reg.**’ contains the names of all immobile regions (index  $j$  in the equations above) already defined. The set of regions can be edited using the buttons ‘**Add**’, ‘**Edit**’ and ‘**Delete**’. This editing of immobile regions is described later in this subsection.

As last options, ‘**Num. Grid Pts.**’, ‘**Resolution**’ and ‘**Acc.**’, in the dialog box used for the definition of a saturated soil column compartment (Fig. 3.53) the user can select the number of grid points, the discretization order of the numerical algorithm and the accuracies of program variables. The number of grid points is used to specify by how many discrete points the continuous  $x$ -axis is approximated. If the number of grid points is set to  $n_{gp}$ , the column is resolved longitudinally into 2 boundary points and  $n_{gp} - 2$  grid points located in the middle of  $n_{gp} - 2$  cells of equal thickness. For this division of the  $x$ -axis the ‘**low resolution**’ method applies a simple first-order discretization scheme that is very robust but can have significant numerical diffusion. The ‘**high resolution**’ method uses a second-order discretization scheme that applies flux-limiters to avoid oscillations of the numerical solutions (Reichert, 1994b, chapter 6). The button ‘**Acc.**’ is used to specify the numerical accuracies of program variables as described later in this subsection.

The check box ‘**active for calculation**’ can be used to activate or inactivate the compartment from the calculations. This check box has the same functionality as the buttons **Activate** and **Inactivate** in the dialog box shown in Fig. 3.19.

As for each compartment, the user has to select which state variables are active. This is done by clicking the button ‘**Variables**’ of the dialog box shown in Fig. 3.53. This action opens the dialog box shown in Fig. 3.54. The two list boxes of this dialog box show the active variables and the available variables, respectively. The button ‘**Activate**’ is used to activate available variables selected in the right list box, and the button ‘**Inactivate**’ is used to inactivate active variables selected in the left list box. If an active variable is selected while activating another variable, the new active variable is inserted in the list of active variables immediately before the selected variable, otherwise it is appended to the end of the list of active variables. This gives the user the possibility to influence the order of active variables (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The list of active variables may contain variables of any type, but activation has only an effect to state variables. Inactive state variables return a value of zero. The reason for allowing other types of variables in the list of active state

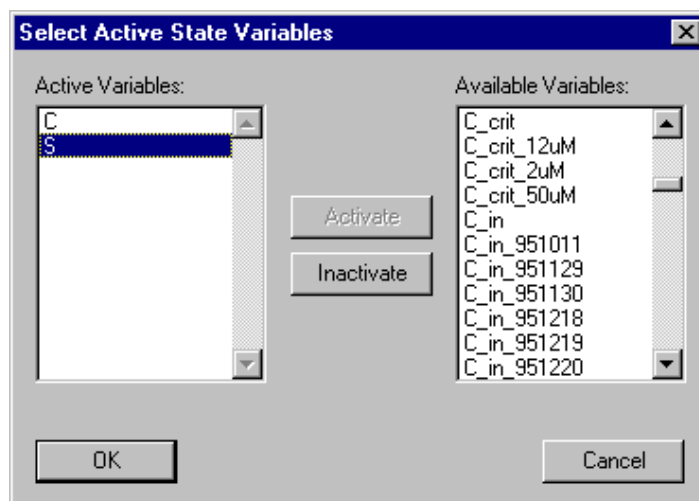


Figure 3.54: Dialog box for activating and inactivating state variables in a compartment.

variables is to facilitate switching between different models that do not contain the same state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of active variables of the compartments.

Similarly to the activation of state variables, the user has to select which processes are active in a compartment. This is done by clicking the button '**Processes**' of the dialog box shown in Fig. 3.53. This action opens the dialog box shown in Fig. 3.55. The two list

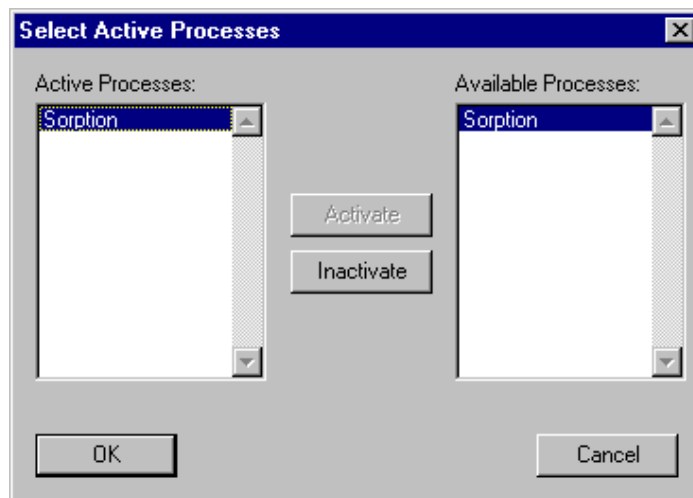


Figure 3.55: Dialog box for activating and inactivating processes in a compartment.

boxes in this dialog box show the active processes and the available processes, respectively. The button '**Activate**' is used to activate available processes selected in the right list box, and the button '**Inactivate**' is used to inactivate a active processes selected in the left list box. If an active process is selected while activating another process, the new active

process is inserted in the list of active processes immediately before the selected process, otherwise it is appended to the end of the list of active processes. This gives the user the possibility to influence the order of active processes (the order is irrelevant for the program, but it may be convenient for the user to have a certain order).

Initial conditions for a saturated soil column compartment can be specified by clicking the button '**Init. Cond.**' of the dialog box shown in Fig. 3.53. This action opens the dialog box shown in Fig. 3.56. This dialog box shows a list of all initial conditions already

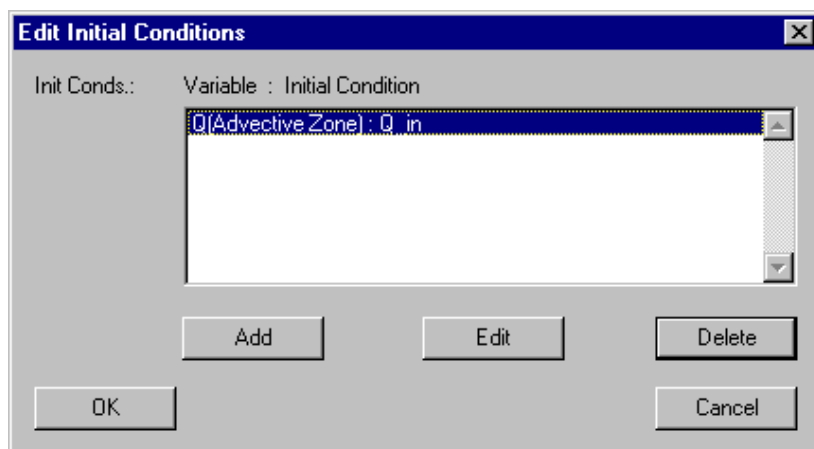


Figure 3.56: Dialog box for editing initial conditions for a saturated soil column compartment.

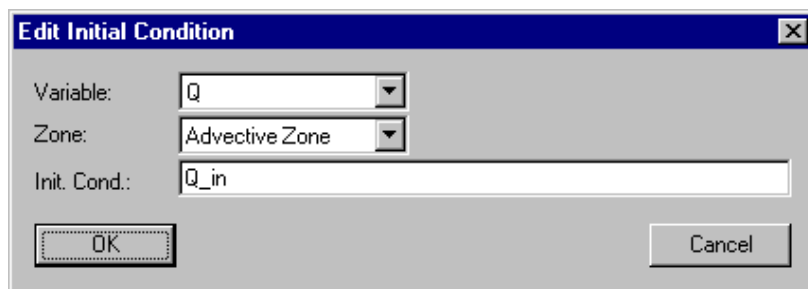


Figure 3.57: Dialog box for editing a single initial conditions for a saturated soil column compartment.

specified. Each line of the list box contains the name of a variable, followed by the zone of the compartment for which the initial condition is specified (in brackets) and by the algebraic expression specifying the initial value. For each combination of a variable with a zone only one unique initial condition can be specified. The list of initial conditions can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If an initial condition is selected while adding a new initial condition, the new initial condition is inserted in the list immediately before the selected initial condition, otherwise it is appended to the end of the list of initial conditions. This gives the user the possibility to influence the order of

the initial conditions (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The dialog box used for editing a single initial condition is shown in Fig. 3.57. In this dialog box the fields '**Variable**', '**Zone**' and '**Init. Cond.**' allow the user to select a variable and a zone and to specify an initial condition. Initial conditions for any type of variables can be specified, but only initial conditions for state variables are used by the program. The reason for allowing to define initial conditions for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of initial conditions of the compartments. Initial conditions of a saturated soil column compartment may not depend on state variables and they can only depend on the program variables **Calculation Number**, **Time** and **Space Coordinate X**.

Input to a saturated soil column compartment can be specified by clicking the button '**Input**' of the dialog box shown in Fig. 3.53. This action opens the dialog box shown in Fig. 3.58. In this dialog box the user can select which type of input to edit. There

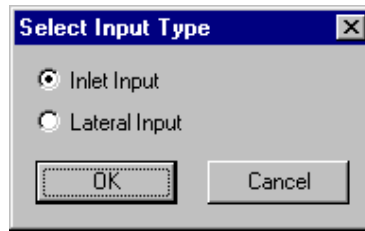


Figure 3.58: Dialog box for selecting an input type for a soil column compartment.

exist two different types of inputs to a saturated soil column compartment. The radio button '**Inlet Input**' can be used to describe water and substance flow at the inlet of the column and the radio button '**Lateral Input**' makes it possible to specify water and substance inflow along the column. In the following paragraphs, the dialog boxes used to define each of these input types are described.

Selection of the radio button '**Inlet Input**' in the dialog box shown in Fig. 3.58 opens the dialog box shown in Fig. 3.59. The edit field '**Water Inflow**' of this dialog box is used to specify the discharge of water into the compartment,  $Q_{in}$ , and the list box contains substance loadings,  $I_{in,C_i}$ , into the compartment. For each variable only one unique inlet loading can be specified. The list of inlet loadings can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If a loading is selected while adding a new loading, the new loading is inserted in the list immediately before the selected loading, otherwise it is appended to the end of the list of input loadings. This gives the user the possibility to influence the order of the input loadings (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.60 shows the dialog box used to specify a single inlet loading of a substance. In this dialog box the fields '**Variable**' and '**Loading**' allow the user to select a variable and specify an inlet loading,  $I_{in,C_i}$ . Note that this loading represents mass per unit of time. An inlet loading of a saturated soil column compartment may depend on the program variables **Discharge** and on dynamic volume state variables. These variables return the discharge and the concentrations of dynamic volume state variables resulting from all advective links connected to the inlet

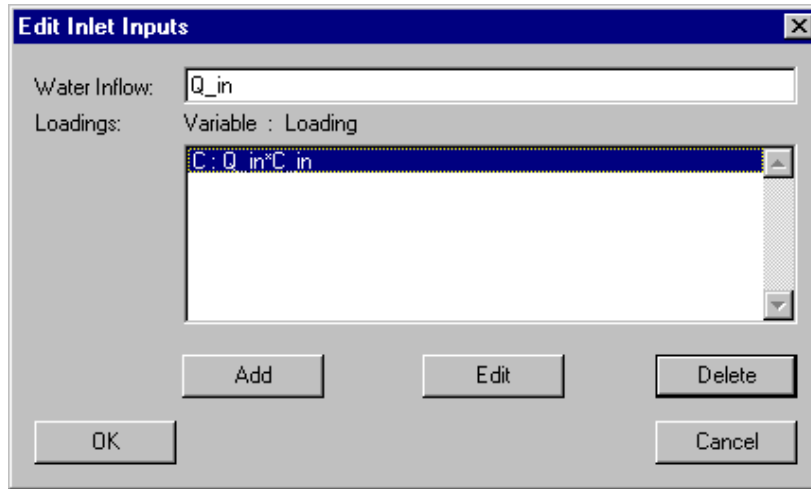


Figure 3.59: Dialog box for editing inlet inputs to a soil column compartment.

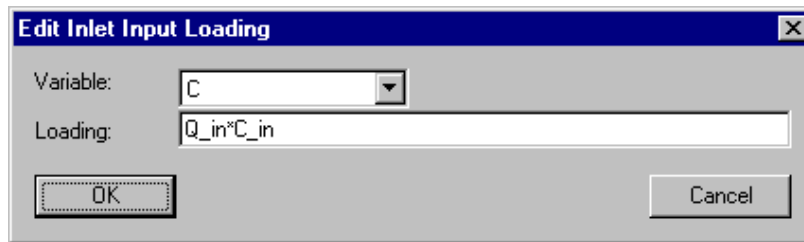


Figure 3.60: Dialog box for editing a single inlet input to a soil column compartment.

of the column. Input loadings for any type of variables can be specified, but only input loadings for dynamic volume state variables are used by the program. The reason for allowing to define loadings for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of loadings of the compartments.

Selection of the radio button '**Lateral Input**' in the dialog box shown in Fig. 3.58 opens the dialog box shown in Fig. 3.61. The edit field '**Water Inflow**' of this dialog box is used to specify the discharge of water per unit length into the mobile zone of the column,  $q$ , and the list box contains substance concentrations,  $C_{lat,i}$ , in the inflowing water. A positive value of  $q$  represents a flow into the column, a negative value represents an outflow. According to the equations (3.47) and (3.49) the inflow concentration is irrelevant if  $q$  is smaller than or equal to zero (outflow). For each variable only one unique lateral inflow concentration can be specified. The list of inflow concentrations can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If an inflow concentration is selected while adding a new inflow concentration, the new inflow concentration is inserted in the list immediately before the selected inflow concentration, otherwise it is appended to the end of the list of inflow concentrations. This gives the user the possibility to influence the

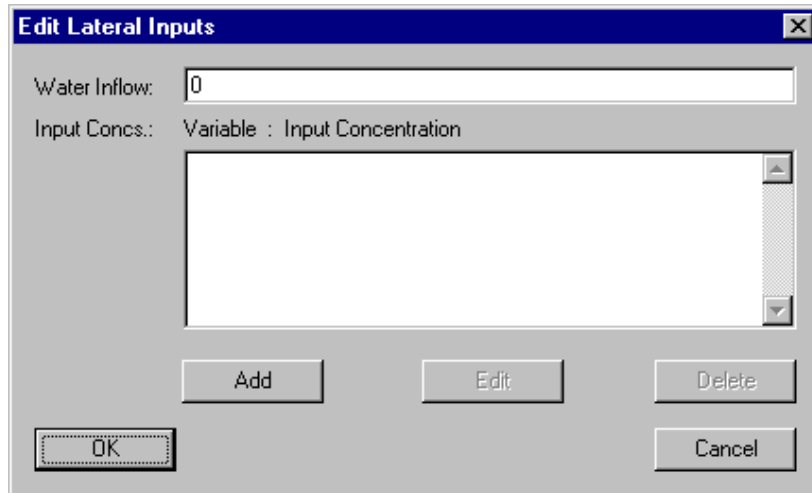


Figure 3.61: Dialog box for editing lateral inputs to a soil column compartment.

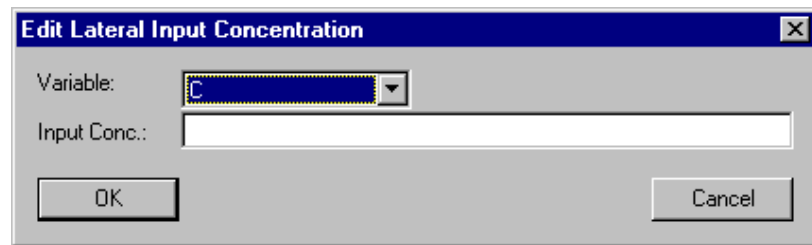


Figure 3.62: Dialog box for editing a single lateral input to a soil column compartment.

order of the inflow concentrations (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.62 shows the dialog box used to specify a single lateral inflow concentration of a dynamic volume state variable. In this dialog box the fields '**Variable**' and '**Inflow Conc.**' allow the user to select a variable and specify a lateral inflow concentration,  $C_{lat,i}$ . A lateral inflow concentration of a saturated soil column compartment may depend on the program variables **Discharge** and on state variables. These variables return the discharge and the current values of state variables in the compartment as a function of the location  $x$ . Inflow concentrations for any type of variables can be specified, but only inflow concentrations for dynamic volume state variables are used by the program. The reason for allowing to define inflow concentrations for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of lateral inflow concentrations of the compartments.

The buttons '**Add**', '**Edit**' and '**Delete**' of the dialog box shown in Fig. 3.53 allow to define immobile zones. Several immobile regions, each of which with an arbitrary number of serially connected mixed zones can be defined. Figure 3.63 shows the dialog box used for defining an immobile region. The edit field '**Name**' is used to specify the name of the immobile region. Each immobile region needs a unique name as an identifier. A name of

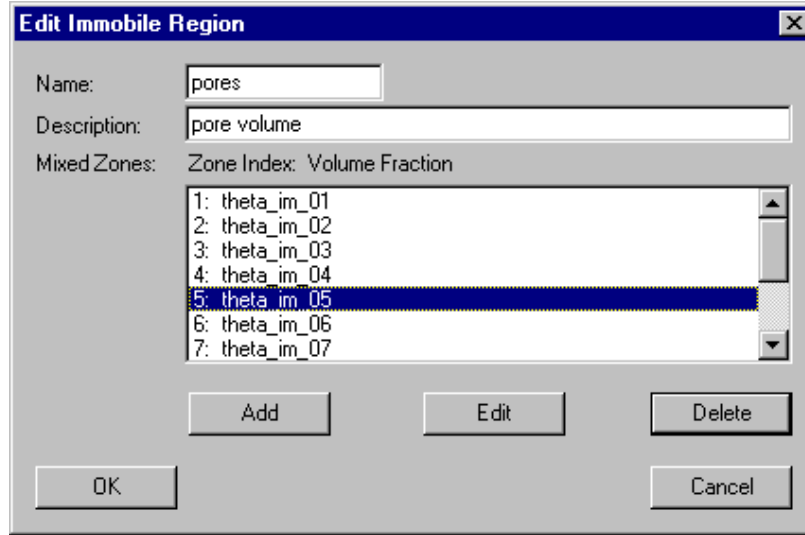


Figure 3.63: Dialog box for editing an immobile region of a saturated soil column compartment.

an immobile region consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

To improve documentation of immobile regions, the edit field '**Description**' can optionally be used to store comments on specific implementation features of the zone. The list box of the dialog box shown in Fig. 3.63 shows the division of the immobile region into mixed zones. At least one mixed zone is required in order to specify a valid immobile region. The list of mixed zones can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If a mixed zone is selected while adding a new zone, the new zone is inserted in the list immediately before the selected zone, otherwise it is appended to the end of the list of mixed zones. This gives the user the possibility to influence the order of the mixed zones. The Fig. 3.64 shows the dialog box used for the definition of mixed zone. The edit field '**Zone Index**' can be used to specify a nonnegative inter number as a zone index. This value can be accessed with the aid of the program variable **Zone Index** to make variables or process rates dependent on the zone (in the mobile zone, the program variable **Zone Index** returns zero). The edit field '**Vol. Fract.**' is used to specify the volume fraction of the zone,  $\theta_{im_{jk}}$ . Note that the sum of the volume fraction of the mobile zone and all volume fractions of the immobile zones must be smaller than unity. The list box of the dialog box shown in Fig. 3.64 shows the exchange coefficients,  $q_{ex,im_{jk},i}$ , and the conversion factors,  $f_{ex,im_{jk},i}$ , of the substances exchanged between the current mixed zone and the mixed zone immediately adjacent to the current mixed zone in direction to the mobile zone (for the first mixed zone this is the exchange coefficient with the mobile zone). The conversion factor can be used to describe transitions between different solvents in the pores. If all pores are filled with the same solvent, the conversion factors are unity. The list of exchange coefficients and conversion factors can be edited with the buttons '**Add**', '**Edit**' and '**Delete**'. If an exchange coefficient is selected while adding a new exchange coefficient, the new exchange coefficient is inserted in the list immediately before the selected exchange coefficient, otherwise it is appended to the

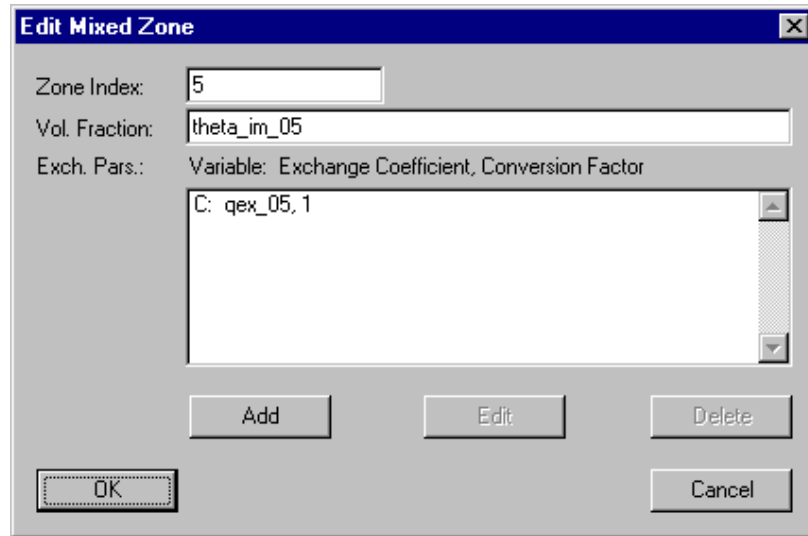


Figure 3.64: Dialog box for editing a mixed zone of a soil column compartment.

end of the list of exchange coefficients. This gives the user the possibility to influence the order of the exchange coefficients (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The Figure 3.65 shows the dialog box used for defining exchange coefficients and conversion factors. In this dialog box the fields

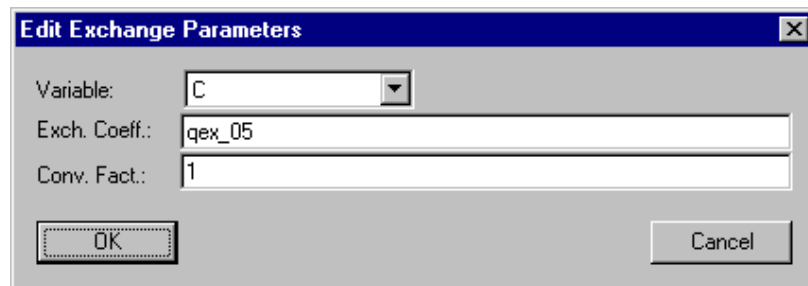


Figure 3.65: Dialog box for editing an exchange coefficient of a mixed zone of a soil column compartment.

‘**Variable**’, ‘**Exch. Coeff.**’ and ‘**Conv. Fact.**’ allow the user to select a variable and specify an exchange coefficient,  $q_{ex,im_{jk},i}$ , and a conversion factor,  $f_{ex,im_{jk},i}$ . Exchange coefficients for any type of variables can be specified, but only exchange coefficients for dynamic volume state variables are used by the program. The reason for allowing to define inflow concentrations for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of lateral inflow concentrations of the compartments.

The button ‘**Acc.**’ of the dialog box shown in Fig. 3.53 is used to specify the numerical accuracies of program variables. Fig. 3.66 shows the dialog box used for this purpose.



It allows the user to specify relative and absolute accuracies of the program variable ‘**Discharge**’,  $Q$ , and of the ‘**Dispersion**’ coefficient,  $E$ , in the compartment. Good behaviour of the numerical algorithms is usually achieved if the absolute accuracy and the product of the relative accuracy times a typical value of the variable both are 4 to 6 orders of magnitude smaller than typical values of the variable.

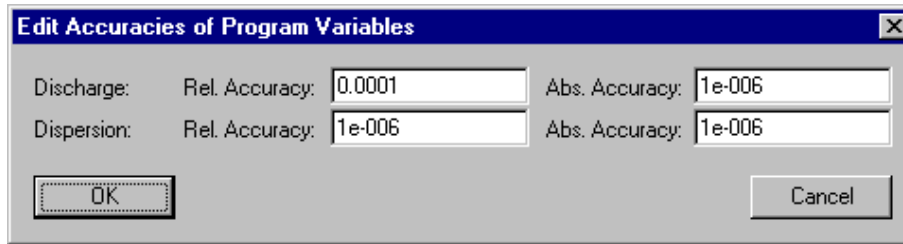


Figure 3.66: Dialog box for editing the accuracies of program variables of a saturated soil column compartment.

In Table 3.7 the program variables available in a saturated soil column compartment are summarized (for a complete overview of all program variables see Table 3.1 on page 17).

Calculation Number	Identifier for calculations (value set in the dialog boxes shown in Figs. 4.5 and 4.17).
Time	Simulation time.
Compartment Index	Identifier for compartments (value set in the dialog box shown in Figs. 3.53).
Zone Index	Identifier for zones within compartments (returns a value of 0 in the mobile zone and a value set in the dialog box shown in Fig. 3.64 in the mixed immobile zones).
Discharge	Volumetric flow rate.
Water Fraction	Volumetric fraction of water (returns the value of the porosity).
Space Coordinate X	Space coordinate along the compartment.
Cross Sectional Area	Area of water body perpendicular to the flow direction (advective-diffusive compartment, saturated soil column compartment, river section compartment).

Table 3.7: Program variables available in the saturated soil column compartment.

### 3.3.5 River Section Compartment

#### Overview

The river section compartment of AQUASIM can be used to describe river hydraulics, advective-dispersive transport of substances dissolved or suspended in the water column, exchange of substances between the water column and the sediment, and transformation processes of substances in the water column or the sediment in a river section without abrupt hydraulic controls. In the presence of significant hydraulic structures or tributaries, several river sections can be linked advectively to model the river reach of interest. The description of the river section is one-dimensional. This means that all variables are averaged over the river cross section and the depth of the sediment is not resolved. Such a description is in many cases adequate when the main interest is to model transformation processes over relatively long distances, but it makes the application of the river section compartment to local mixing phenomena impossible.

#### Equations Solved by AQUASIM

One-dimensional river hydraulics can be described by a set of two partial differential equations representing a mass and a momentum balance (de St. Venant, 1871; Chow, 1959; Henderson, 1966; Yen, 1973; Yen, 1979; French, 1985). The two most important approximations to these so-called St. Venant equations, the kinematic and diffusive wave approximations (Yen, 1979), are implemented in AQUASIM to describe river hydraulics. The equations for river hydraulics are coupled with advection-diffusion equations to describe transport of substances dissolved or suspended in the water. An empirical, substance-independent diffusion coefficient is used to describe the longitudinal mixing effect due to dispersion.

In order to formulate the one-dimensional conservation laws

$$\frac{\partial \hat{\rho}}{\partial t} + \frac{\partial \hat{j}}{\partial x} = \hat{r} \quad (3.56)$$

compartment-specific expressions for the one-dimensional density,  $\hat{\rho}$  (amount of conserved quantity per unit compartment length), for the one-dimensional flux,  $\hat{j}$  (amount of the conserved quantity transported per unit time), and for the one-dimensional source term,  $\hat{r}$  (amount produced per unit compartment length and per unit time), must be derived.

For the case of a river section compartment, 3 types of components of a conservation law must be distinguished. The array of one-dimensional densities of these types of components is given as follows:

$$\hat{\rho} = \begin{pmatrix} A \\ AC_i \\ S_i \end{pmatrix} \quad (3.57)$$

The first component of equation (3.57) describes the conservation of water volume within the river (water is approximated to be incompressible). The one-dimensional density of water volume in the river (volume per unit length) is given by the wetted cross-sectional area,  $A$ , of the water body. The second component of equation (3.57) describes substances transported with the water flow along the river. Their one-dimensional densities are given as the product of the cross-sectional area,  $A$ , of the water body and the laterally averaged

concentration,  $C_i$ . The last component of equation (3.57) describes substances settled to the bottom or sorbed to surfaces of the river bed.

The value of the variable  $A$  must be specified as a function of the distance along the river,  $x$ , and of the elevation of the water level,  $z_0$ . Its current value is then generally accessible by the program variable **Cross Sectional Area**. The variables  $C_i$  are represented by dynamic volume state variables, and the variables  $S_i$  are represented by dynamic surface state variables. For equilibrium state variables algebraic equations specified as equilibrium processes are solved everywhere along the compartment.

The one-dimensional fluxes of the substances with one-dimensional densities as described by equation (3.57) are given as follows:

$$\hat{\mathbf{j}} = \begin{pmatrix} Q \\ QC_i - AE \frac{\partial C_i}{\partial x} \\ 0 \end{pmatrix} \quad (3.58)$$

In this equation  $Q$  refers to the volumetric discharge through the compartment, and  $E$  is the coefficient of longitudinal dispersion. Calculation of river hydraulics requires the formulation of the cross-sectionally averaged friction force as an empirical function of averaged flow properties. Usually, instead of the friction force, the non-dimensional friction slope,  $S_f$ , the ratio of the friction force to the gravity force of the water body, is parameterized empirically as a function of wetted cross-sectional area, wetted perimeter and discharge. The discharge  $Q$ , for the kinematic or the diffusive approximation to the St. Venant equations, is then given as the solution to one of the following equations, respectively:

$$Q_{kin} : \quad -\frac{\partial z_B}{\partial x} = S_f \quad (3.59a)$$

$$Q_{diff} : \quad -\frac{\partial z_0}{\partial x} = S_f \quad (3.59b)$$

where  $z_B$  and  $z_0$  are the elevation of the river bed and of the the water surface, respectively. The applicability of the kinematic approximation given by the equation (3.59a) is limited by the necessity of a geometry of the river bed that is monotonically decreasing and simple enough that the bed slope  $-\partial z_B/\partial x$  can reasonably be defined. This approximation assumes the driving gravity force to be equilibrated by the friction force everywhere along the river. In addition to the gravity and the friction forces, the diffusive approximation given by the equation (3.59b) considers longitudinal pressure gradients occurring when the water level is not parallel to the river bed. In contrast to the kinematic approach, the diffusive approximation allows to describe backwater effects of weirs or other hydraulic controls and it can be applied if the river bed is not monotonically decreasing.

The following one-dimensional source terms are required to complete the set of equations for the river section compartment:

$$\hat{\mathbf{r}} = \begin{pmatrix} q \\ Ar_{C_i} + \left(\frac{1+\text{sign}(q)}{2}\right) qC_{lat,i} + \left(\frac{1-\text{sign}(q)}{2}\right) qC_i \\ r_{S_i} \end{pmatrix} \quad (3.60)$$

The first component of equation (3.60) describes the lateral water inflow as volume per unit length of the compartment,  $q$ . This variable can be used to describe the integral effect of small tributaries or the effect of groundwater in- or exfiltration. Larger tributaries must be modelled as upstream input boundary conditions to a new river section compartment. The second component of equation (3.60) describes the effect of transformation processes in the compartment and the effect of lateral inflows or outflows ( $C_{lat,i}$  is the concentration in the lateral inflow,  $q$ ). The last component describes the effect of transformation processes on settled or sorbed substances or on growing organisms.

Application of the general law for differential conservation laws (3.56) to the definitions given by the equations (3.57) to (3.60) leads to the following set of 3 differential equations.

The first equation describes water flow through the compartment:

$$\frac{\partial A}{\partial t} + \frac{\partial Q}{\partial x} = q \quad (3.61)$$

The temporal change in the wetted cross-sectional area,  $A$ , is determined by the spatial gradient of the discharge,  $Q$ , and by the lateral inflow,  $q$ . A negative spatial gradient of the discharge causes an increase in the wetted cross-sectional area because more water flows into a river segment from upstream then leaves it downstream. Obviously, a positive lateral inflow also increases the wetted cross-sectional area. Note that the equation (3.61) must be solved simultaneously with one of the equations (3.59a) or (3.59b).

The second equation describes the behaviour of substances transported with the water flow:

$$\begin{aligned} \frac{\partial(AC_i)}{\partial t} = & -\frac{\partial}{\partial x}(QC_i) + \frac{\partial}{\partial x}\left(AE\frac{\partial C_i}{\partial x}\right) + ArC_i \\ & + \left(\frac{1+\text{sign}(q)}{2}\right)qC_{lat,i} + \left(\frac{1-\text{sign}(q)}{2}\right)qC_i \end{aligned} \quad (3.62)$$

The concentration is affected by advection with the water flow (first term), longitudinal dispersion (second term), transformation processes (third term) and by lateral inflow or outflow (fourth and fifth term, respectively).

The third equation describes the behaviour of settled or sorbed substances or of sessile organisms:

$$\frac{\partial S_i}{\partial t} = r_{S_i} \quad (3.63)$$

The concentration is only affected by transformation processes. Note that settling or sorption must also be formulated as a transformation process from the dissolved or suspended species,  $C_i$ , to the sorbed or settled species,  $S_i$ .

In order to make the solution to the above system of differential equations unique, one or two boundary conditions are necessary for equation (3.61) (depending on the choice of the kinematic or the diffusive approximation), and two boundary conditions are required for the partial differential equation (3.62). The ordinary differential equation (3.63) that does not contain spatial derivatives does not require boundary conditions.

The first boundary condition for the equation (3.61) that describes discharge through the river section is given by

$$Q(x_s) = Q_{in} \quad (3.64a)$$

at the start point,  $x_s$ , of the river section. In the case of the diffusive approximation according to equation (3.59b) which makes it possible to describe backwater effects, an additional downstream boundary condition is required. Such a boundary condition can be specified as a given (discharge dependent) water level

$$z_0(x_e) = z_{given}(Q) \quad (3.64b)$$

at the end of the river section. Besides an arbitrary function describing a strategy of weir regulation, AQUASIM provides the options to select a normal ( $z_{0,N}$ ), or a critical ( $z_{0,C}$ ) end water level. These water levels are given as the solutions to the following algebraic equations:

$$z_{0,N} : S_f = -\partial z_B / \partial x \quad (3.65a)$$

$$z_{0,C} : \frac{Q^2 w}{g A^3} = 1 \quad (3.65b)$$

The equation (3.65a) describes the water level as used in the kinematic approximation to the St. Venant equations. This water level is also approached in the diffusive case with a prismatic river bed at distances far away from hydraulic controls. This boundary condition can be used if the calculation ends within a uniform reach where the water level is not influenced by hydraulic structures. The critical depth is a boundary condition reasonable at drops.

The boundary conditions for equation (3.62) are given by the continuity of the substance mass flows entering the river section and by a ‘transmission boundary condition’ (Shamir and Harleman, 1967) at the end of the compartment:

$$Q(x_s)C_i - AE \frac{\partial C_i}{\partial x} = I_{in,C_i} \quad (3.66a)$$

$$\frac{\partial^2 C_i}{\partial x^2} = 0 \quad (3.66b)$$

where  $I_{in,C_i}$  is the total (given) mass input of substance described by the concentration  $C_i$  per unit of time. The second of these boundary conditions (3.66b) is omitted for dispersion-free transport.

### User Definitions

Figure 3.67 shows the dialog box used for defining or editing a river section compartment. The edit field ‘**Name**’ is used to specify the name of the compartment. Each compartment needs a unique name as an identifier. A name of a compartment consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

The edit field ‘**Comp. Index**’ can be used to specify a nonnegative inter number as a compartment index. This value can be accessed with the aid of the program variable **Compartment Index** to make variables or process rates dependent on the compartment.

To improve documentation of compartments, the edit field ‘**Description**’ can optionally be used to store comments on specific implementation features of a compartment.

Figure 3.67: Dialog box for editing a river section compartment.

The buttons ‘**Variables**’, ‘**Processes**’, ‘**Init. Cond.**’ and ‘**Input**’ are used to activate and inactivate state variables, to activate and inactivate processes, to specify initial conditions and to define inputs to a compartment. These options are described later in this subsection.

The edit fields ‘**Start Coord.**’ and ‘**End Coord.**’ are used to define the location of the upstream and downstream ends of the river section,  $x_s$  and  $x_e$ , respectively. The values of variables are resolved continuously with the space coordinate  $x$  between these two locations. Although all equations in this subsection are formulated with the  $x$ -axis pointing in flow direction ( $x_s < x_e$ ), the program works also correctly with an  $x$ -axis defined in the reverse direction ( $x_s > x_e$ ).

The edit field ‘**Grav. Acceleration**’ is used to specify the value of the gravitational acceleration,  $g$ , in the units as used by the user of the program. Typical values of the gravitational acceleration are  $9.81\text{m/s}^2$ ,  $1.27\text{m/h}^2$  or  $7.32 \cdot 10^{10}\text{m/d}^2$ . The gravitational acceleration is required in order to calculate the critical water level (3.65b) that can be selected as a downstream boundary condition if the diffusive wave approximation is selected to calculate river hydraulics.

The three edit fields ‘**Cross. Sect.**’, ‘**Perimeter**’ and ‘**Width**’ are used to specify the geometry of the river bed described by the cross-sectional area,  $A$ , the wetted perimeter,  $P$ , and the surface width,  $w$ , as functions of the coordinate  $x$  along the river and the elevation of the water level,  $z_0$ . These two variables are accessible by the program variables **Space Coordinate X** and **Water Level Elevation**. A cross-sectional area

that depends on a state variable or does not depend on the water level elevation is illegal. As an example of a simple river geometry, a rectangular channel with slope  $S_0$  can be described by the three functions

$$A(x, z_0) = (z_0 + S_0 x)w \quad (3.67a)$$

for the cross-sectional area

$$P(x, z_0) = w + 2(z_0 + S_0 x) \quad (3.67b)$$

for the wetted perimeter and

$$w(x, z_0) = w \quad (3.67c)$$

for the (constant) water surface width.

The edit field **‘Frict. Slope’** is used to specify an empirical formula for the non-dimensional friction slope. The two most widely used expressions for the friction slope are

$$S_f = \frac{1}{K_{st}^2} \frac{1}{R^{4/3}} \frac{Q^2}{A^2} = n^2 \frac{1}{R^{4/3}} \frac{Q^2}{A^2} \quad (3.68a)$$

according to Manning-Strickler, and

$$S_f = \frac{f}{8g} \frac{1}{R} \frac{Q^2}{A^2} \quad (3.68b)$$

according to Darcy-Weisbach (Chow, 1959; Henderson, 1966; French, 1985). In these equations  $K_{st}$  is the friction coefficient according to Strickler,  $n$  is the friction coefficient according to Manning,

$$R = \frac{A}{P} \quad (3.69)$$

is the hydraulic radius, and  $f$  is the non-dimensional friction factor.

As a next option in the dialog box shown in Fig. 3.67 the user can select how to model **‘Dispersion’**. If the user selects the radio button **‘without dispersion’**, a purely advective equation with  $E = 0$  is solved (note, however, that due to the spatial discretization numerical diffusion can occur). If the user selects the radio button **‘with dispersion’** a positive dispersion coefficient,  $E$ , must be specified in the edit field below the radio buttons. An estimation of the dispersion coefficient can be calculated as follows (Fischer et al., 1979):

$$E \approx 0.011 \frac{w^2 Q^2}{u^* d A^2} \quad (3.70)$$

where

$$u^* \approx \sqrt{gdS_f} \quad (3.71)$$

is the shear velocity and

$$d = \frac{A}{w} \quad (3.72)$$

is the mean river depth.

The next option of the dialog box shown in Fig. 3.67 is to select the method used for calculating river hydraulics. The user has the choice between the radio buttons '**kinematic**' and '**diffusive**', which refer to the kinematic wave approximation according to equation (3.59a) and the diffusive wave approximation to the St. Venant equations according to equation (3.59b). In the case of the diffusive approximation, an end water level must be specified as a downstream boundary condition. The user can select between the radio buttons '**normal**', '**critical**' and '**given**' which refer to a normal end water level according to equation (3.65a), a critical end water level according to equation (3.65b) or an end water level specified by the user in the edit field below the radio buttons as a function of time and discharge.

As last options, '**Num. Grid Pts.**', '**Resolution**' and '**Acc.**', in the dialog box used for the definition of a river section compartment (Fig. 3.67) the user can select the number of grid points, the discretization order of the numerical algorithm and the accuracies of program variables. The number of grid points is used to specify by how many discrete points the continuous  $x$ -axis is approximated. If the number of grid points is set to  $n_{gp}$ , the river section is resolved longitudinally into 2 boundary points and  $n_{gp} - 2$  grid points located in the middle of  $n_{gp} - 2$  cells of equal thickness. For this division of the  $x$ -axis the '**low resolution**' method applies a simple first-order discretization scheme that is very robust but can have significant numerical diffusion. The '**high resolution**' method uses a second-order discretization scheme that applies flux-limiters to avoid oscillations of the numerical solutions (Reichert, 1994b, chapter 6). The button '**Acc.**' is used to specify the numerical accuracies of program variables as described later in this subsection.

The check box '**active for calculation**' can be used to activate or inactivate the compartment from the calculations. This check box has the same functionality as the buttons **Activate** and **Inactivate** in the dialog box shown in Fig. 3.19.

As for each compartment, the user has to select which state variables are active. This is done by clicking the button '**Variables**' of the dialog box shown in Fig. 3.67. This action opens the dialog box shown in Fig. 3.68. The two list boxes of this dialog box show the

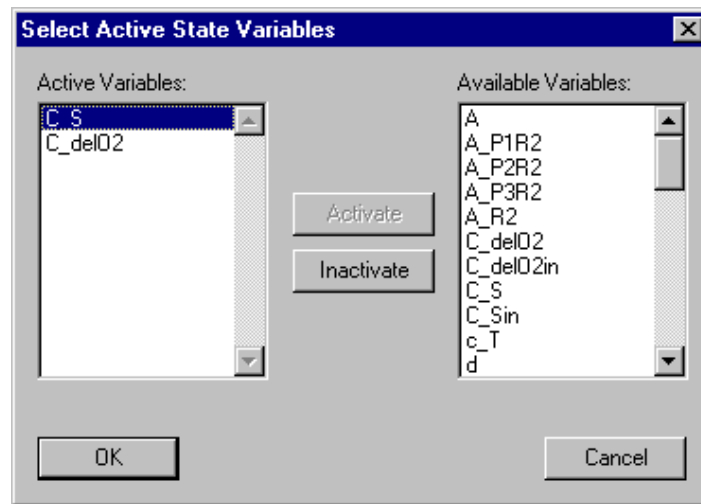


Figure 3.68: Dialog box for activating and inactivating state variables in a compartment.



active variables and the available variables, respectively. The button '**Activate**' is used to activate available variables selected in the right list box, and the button '**Inactivate**' is used to inactivate active variables selected in the left list box. If an active variable is selected while activating another variable, the new active variable is inserted in the list of active variables immediately before the selected variable, otherwise it is appended to the end of the list of active variables. This gives the user the possibility to influence the order of active variables (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The list of active variables may contain variables of any type, but activation has only an effect to state variables. Inactive state variables return a value of zero. The reason for allowing other types of variables in the list of active state variables is to facilitate switching between different models that do not contain the same state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of active variables of the compartments.

Similarly to the activation of state variables, the user has to select which processes are active in a compartment. This is done by clicking the button '**Processes**' of the dialog box shown in Fig. 3.67. This action opens the dialog box shown in Fig. 3.69. The two list

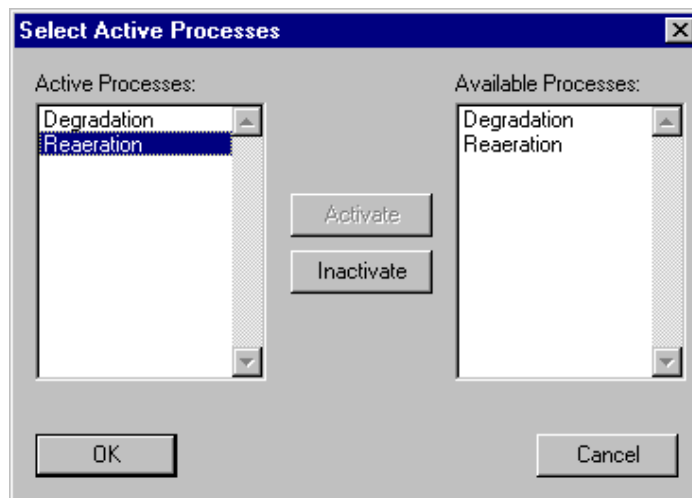


Figure 3.69: Dialog box for activating and inactivating processes in a compartment.

boxes in this dialog box show the active processes and the available processes, respectively. The button '**Activate**' is used to activate available processes selected in the right list box, and the button '**Inactivate**' is used to inactivate a active processes selected in the left list box. If an active process is selected while activating another process, the new active process is inserted in the list of active processes immediately before the selected process, otherwise it is appended to the end of the list of active processes. This gives the user the possibility to influence the order of active processes (the order is irrelevant for the program, but it may be convenient for the user to have a certain order).

Initial conditions for a river section compartment can be specified by clicking the button '**Init. Cond.**' of the dialog box shown in Fig. 3.67. This action opens the dialog box shown in Fig. 3.70. This dialog box shows a list of all initial conditions already specified. Each line of the list box contains the name of a variable, followed by the zone of the compartment for which the initial condition is specified (in brackets) and by the

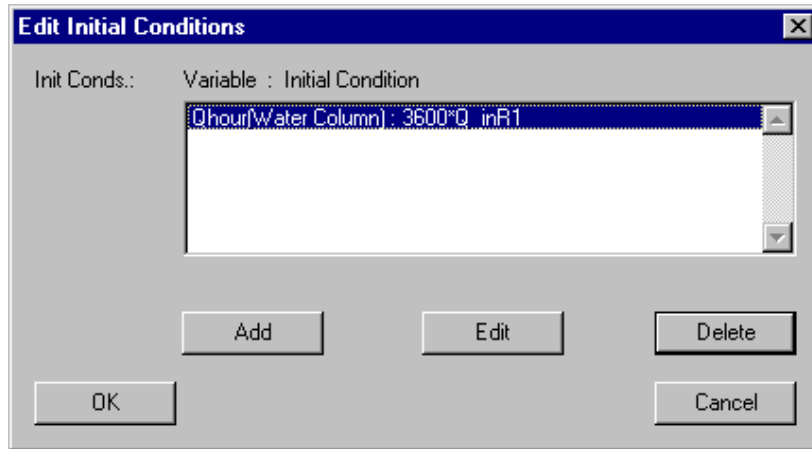


Figure 3.70: Dialog box for editing initial conditions for a river section compartment.

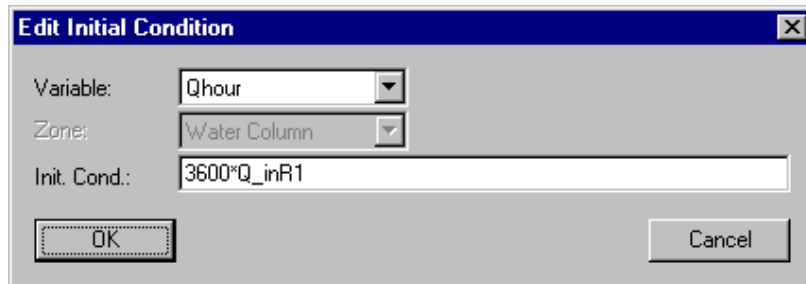


Figure 3.71: Dialog box for editing a single initial condition for a river section compartment.

algebraic expression specifying the initial value. For each combination of a variable with a zone only one unique initial condition can be specified. The list of initial conditions can be edited using the buttons ‘**Add**’, ‘**Edit**’ and ‘**Delete**’. If an initial condition is selected while adding a new initial condition, the new initial condition is inserted in the list immediately before the selected initial condition, otherwise it is appended to the end of the list of initial conditions. This gives the user the possibility to influence the order of the initial conditions (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The dialog box used for editing a single initial condition is shown in Fig. 3.71. In this dialog box the fields ‘**Variable**’ and ‘**Init. Cond.**’ allow the user to select a variable and to specify an initial condition. The field ‘**Zone**’ is inactive, because the river section compartment contains only one zone **Water Column**. Initial conditions for any type of variables can be specified, but only initial conditions for state variables and for the program variable **Discharge** are used by the program. An initial condition for this program variable is required. The reason for allowing to define initial conditions for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of initial conditions of the

compartments. Initial conditions of a river section compartment may not depend on state variables and they can only depend on the program variables **Calculation Number**, **Time** and **Space Coordinate X**.

Input to a river section compartment can be specified by clicking the button '**Input**' of the dialog box shown in Fig. 3.67. This action opens the dialog box shown in Fig. 3.72. In this dialog box the user can select which type of input to edit. There exist two different

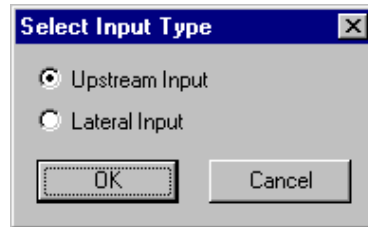


Figure 3.72: Dialog box for selecting an input type for a river section compartment.

types of inputs to a river section compartment. The radio button '**Upstream Input**' can be used to describe water and substance flow at the upstream end of the river section and the radio button '**Lateral Input**' makes it possible to specify water and substance inflow along the river. In the following paragraphs, the dialog boxes used to define each of these input types are described.

Selection of '**Upstream Input**' in the dialog box shown in Fig. 3.72 opens the dialog box shown in Fig. 3.73. The edit field '**Water Inflow**' of this dialog box is used to specify

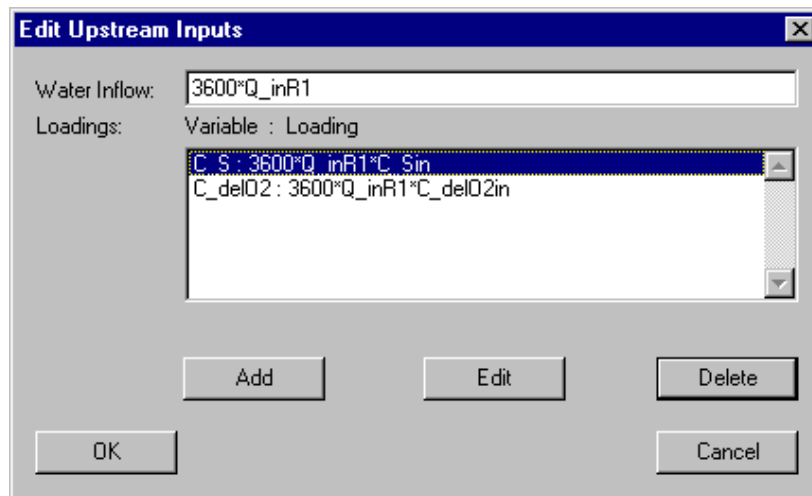


Figure 3.73: Dialog box for editing upstream inputs to a river section compartment.

the discharge of water into the compartment,  $Q_{in}$ , and the list box contains substance loadings,  $I_{in,C_i}$ , into the river section. For each variable only one unique upstream input loading can be specified. The list of upstream substance loadings can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If a loading is selected while adding a new loading,

the new loading is inserted in the list immediately before the selected loading, otherwise it is appended to the end of the list of input loadings. This gives the user the possibility to influence the order of the loadings (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.74 shows the dialog box used to specify a single upstream loading of a substance. In this dialog box the fields ‘**Variable**’

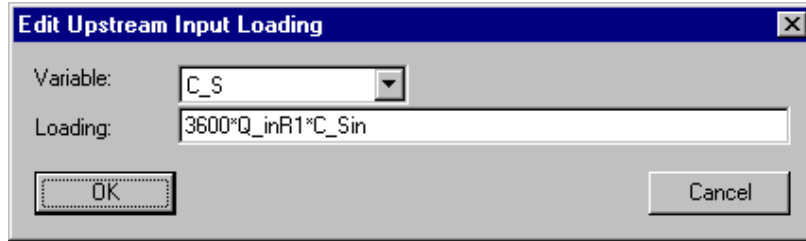


Figure 3.74: Dialog box for editing a single upstream input to a river section compartment.

and ‘**Loading**’ allow the user to select a variable and to specify an upstream loading,  $I_{in,C_i}$ . Note that this loading represents mass per unit of time. An upstream loading to a river section compartment may depend on the program variables **Discharge** and on dynamic volume state variables. These variables return the discharge and the current values of dynamic volume state variables resulting from all advective links connected to the upstream end of the river section. Input loadings for any type of variables can be specified, but only input loadings for dynamic volume state variables are used by the program. The reason for allowing to define loadings for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of input loadings of the compartments.

Selection of the radio button ‘**Lateral Input**’ in the dialog box shown in Fig. 3.72 opens the dialog box shown in Fig. 3.75. The edit field ‘**Water Inflow**’ of this dialog box is used to specify the discharge of water per unit length into the river section,  $q$ , and the list box contains substance concentrations,  $C_{lat,i}$ , in the inflowing water. A positive value of  $q$  represents a flow into the compartment, a negative value represents an outflow. According to the equations (3.60) and (3.62) the inflow concentration is irrelevant if  $q$  is smaller than or equal to zero (outflow). For each variable only one unique lateral inflow concentration can be specified. The list of inflow concentrations can be edited using the buttons ‘**Add**’, ‘**Edit**’ and ‘**Delete**’. If an inflow concentration is selected while adding a new inflow concentration, the new inflow concentration is inserted in the list immediately before the selected inflow concentration, otherwise it is appended to the end of the list of inflow concentrations. This gives the user the possibility to influence the order of the inflow concentrations (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.76 shows the dialog box used to specify a single lateral inflow concentration of a dynamic volume state variable. In this dialog box the fields ‘**Variable**’ and ‘**Inflow Conc.**’ allow the user to select a variable and specify a lateral inflow concentration,  $C_{lat,i}$ . A lateral inflow concentration of a river section compartment may depend on the program variables **Discharge** and on state variables. These variables return the discharge and the current values of state variables in the compartment as a function of the location  $x$ . Inflow concentrations for any type

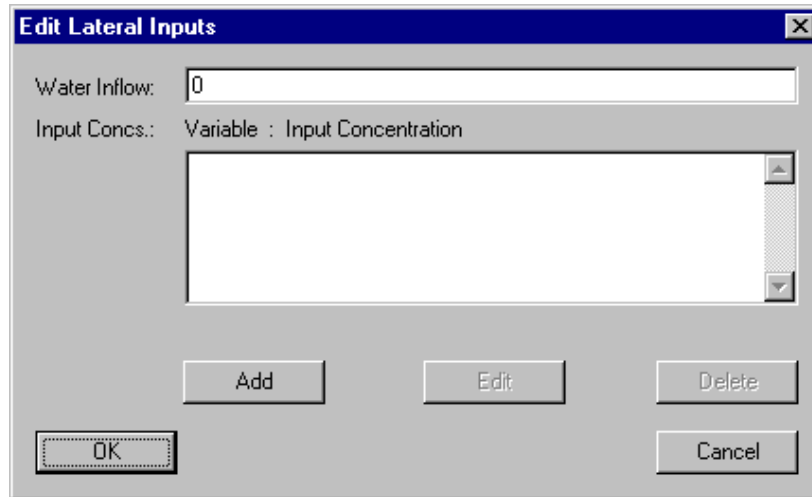


Figure 3.75: Dialog box for editing lateral inputs to a river section compartment.

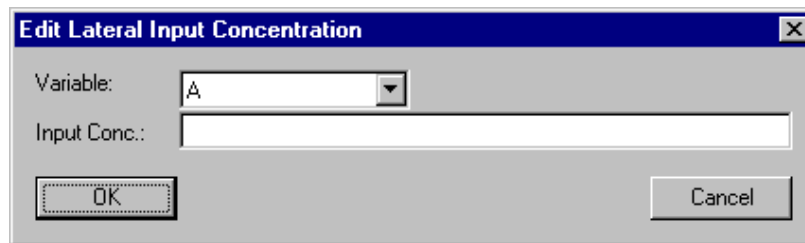


Figure 3.76: Dialog box for editing a single lateral input to a river section compartment.

of variables can be specified, but only inflow concentrations for dynamic volume state variables are used by the program. The reason for allowing to define inflow concentrations for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of lateral inflow concentrations of the compartments.

The button ‘**Acc.**’ of the dialog box shown in Fig. 3.70 is used to specify the numerical accuracies of program variables. Fig. 3.77 shows the dialog box used for this purpose. It allows the user to specify relative and absolute accuracies of the program variables ‘**Discharge**’,  $Q$ , ‘**Cross-Sectional Area**’,  $A$ , ‘**Water Level**’ elevation,  $z_0$ , and for the ‘**Dispersion**’ coefficient,  $E$ , in the compartment. Good behaviour at the numerical algorithms is usually achieved if the absolute accuracy and the product of the relative accuracy times a typical value of the variable both are 4 to 6 orders of magnitude smaller than typical values of the variable.

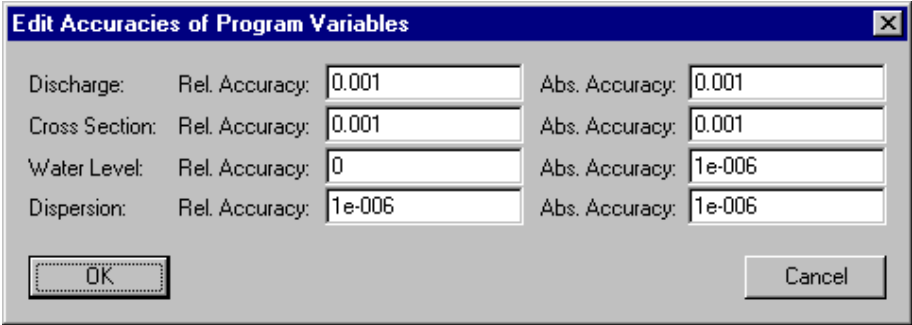


Figure 3.77: Dialog box for editing the accuracies of program variables of a river section compartment.

In Table 3.8 the program variables available in a river section compartment are summarized (for a complete overview of all program variables see Table 3.1 on page 17).

Calculation Number	Identifier for calculations (value set in the dialog boxes shown in Figs. 4.5 and 4.17).
Time	Simulation time.
Compartment Index	Identifier for compartments (value set in the dialog box shown in Fig. 3.67).
Zone Index	Identifier for zones within compartments (returns a value of 0).
Discharge	Volumetric flow rate.
Water Fraction	Volumetric fraction of water (returns a value of 1).
Space Coordinate X	Space coordinate along the compartment.
Water Level Elevation	Elevation of water level above an absolute reference level.
Cross Sectional Area	Area of water body perpendicular to the flow direction.
Perimeter Length	Length of the interface between water and the river bed perpendicular to the flow velocity.
Surface Width	Length of the interface between water and the atmosphere perpendicular to the flow velocity.
Friction Slope	Nondimensional friction force: Friction force divided by gravity force.

Table 3.8: Program variables available in the river section compartment.

### 3.3.6 Lake Compartment

#### Overview

The lake compartment of AQUASIM can be used to describe the stratification of the water column, vertical mixing and advection of substances dissolved or suspended in the water column, sedimentation and resuspension of particles, exchange of dissolved substances between water column and pore water of the top sediment layer, advective and diffusive exchange between an arbitrary number of sediment layers, and transformation processes in the water column as well as in the sediment layers. A one-dimensional description is used that averages all variables over horizontal cross sections. This limits the applicability of this compartment to situations in which the dimensions of the lake, the stratification and the time scales of the investigated processes make a horizontally averaged description reasonable. The current version of the lake compartment has no connections to advective or diffusive links, so that it can only be used to describe a single lake with given inputs and processes.

#### Equations Solved by AQUASIM

The lake equations solved by AQUASIM consist of a combination of a conventional advection-diffusion equation for the water column (Ulrich et al., 1995) with a sediment model describing an arbitrary number of sediment layers, and with a  $k$ - $\epsilon$  turbulence model (Rodi, 1980; Rodi, 1987; Burchard and Baumert, 1995) that has been extended by a simple model of energy storage in seiche motion in the lake basin. The user can specify the coefficient of vertical turbulent diffusion as a given function of time and space, but it is also possible to use a parameterization depending on the stability of the water column and on turbulent kinetic energy and dissipation. In this subsection, the full set of equations is described. However, in order to make the use of simpler models easier and faster, the sediment submodel and the turbulence submodel can be inactivated independently of each other.

In order to formulate the one-dimensional conservation laws

$$\frac{\partial \hat{\rho}}{\partial t} + \frac{\partial \hat{\mathbf{j}}}{\partial z} = \hat{\mathbf{r}} \quad (3.73)$$

compartment-specific expressions for the one-dimensional density,  $\hat{\rho}$  (amount of conserved quantity per unit compartment length), for the one-dimensional flux,  $\hat{\mathbf{j}}$  (amount of the conserved quantity transported per unit time), and for the one-dimensional source term,  $\hat{\mathbf{r}}$  (amount produced per unit compartment length and per unit time), must be derived.

Several zones are distinguished in the lake compartment: one zone for the water column and one zone for each sediment layer. Variables present in all zones are distinguished by the index  $L$  for the lake water column and the index  $S_j$  for the sediment layer  $j$ . The vertical dimension of the lake is resolved by the space coordinate  $z$ . In the following, the lake equations are formulated with the  $z$ -axis pointing upwards, however, the program runs with both definitions of the direction of the  $z$ -axis.

In order to formulate the lake equations, 8 types of components of a conservation law must be distinguished. The array of one-dimensional densities of these types of components

is given as follows:

$$\hat{\rho} = \begin{pmatrix} A \\ AU \\ A\rho k \\ A\rho\epsilon \\ AC_{L,i} \\ AX_{L,i} \\ \frac{dA}{dz}h_{sed,j}\theta_j C_{S_j,i} \\ \frac{dA}{dz}h_{sed,j}X_{S_j,i} \end{pmatrix}. \quad (3.74)$$

The first component of equation (3.74) describes the conservation of the water volume (water is approximated to be incompressible). The one-dimensional density of water volume (volume per unit of depth) is given by the cross-sectional area,  $A$ , of the lake. The second component of equation (3.74) describes a horizontal water flow induced by the surface shear of the wind. This component is used in mixing models to calculate the production of turbulent kinetic energy by shear forces of wind induced water flow. The one-dimensional density of horizontal water discharge is given as the product of the cross-sectional area,  $A$ , and the horizontally averaged (horizontal) flow velocity,  $U$ . The third component of equation (3.74) describes turbulent kinetic energy. The one-dimensional density of turbulent kinetic energy is given as the product of the cross-sectional area of the lake,  $A$ , the density of water,  $\rho$ , and the turbulent kinetic energy,  $k$  (turbulent kinetic energy per unit mass of water). The fourth component of equation (3.74) is an equation for the dissipation,  $\epsilon$ , of turbulent kinetic energy. This quantity together with  $k$  can be used to estimate the coefficient of turbulent diffusion,  $K_z$ , of substances dissolved or suspended in the water column (see below). The fifth and the sixth components of equation (3.74) describe dissolved and suspended substances in the water column of the lake. Both of these one-dimensional densities are given as the product of the cross-sectional area of the lake,  $A$ , and the horizontally averaged volumetric concentration of the substance. In the following, concentrations of dissolved substances are described with the symbol  $C$ , concentrations of particles with the symbol  $X$ . The index  $L$  refers to the water column of the lake and the index  $i$  indicates, that an arbitrary number of such substances can be used simultaneously. The last two components of equation (3.74) describe the concentrations of dissolved substances and of particles in the pore volume of sediment layers of the lake. The one-dimensional density of dissolved substances is given as the product of the gradient of the cross-sectional area of the lake,  $dA/dz$ , the thickness of the sediment layer,  $h_{sed,j}$ , the porosity of the sediment layer,  $\theta_j$ , and the concentration of the dissolved substance in the pore water of the sediment layer averaged over the depth of the sediment layer and over the lake border at a given depth,  $C_{S_j}$ . The one-dimensional density of particles is given as the product of the gradient of the cross-sectional area of the lake,  $dA/dz$ , the depth of the sediment layer,  $h_{sed,j}$ , and the concentration of the particles in the sediment layer averaged over the depth of the sediment layer and over the lake border at a given depth,  $X_{S_j}$ . The gradient  $dA/dz$  in these expressions specifies the sediment area per unit depth of the lake (this area is approximated by its horizontal projection). Note that the sediment submodel of AQUASIM does not resolve the depths of the sediment continuously. Instead, an arbitrary number of sediment layers can be defined by the user to roughly resolve



the sediment depths. These sediment layer always describe the top of the sediment. The influence of the sediment underneath these layers can be described by substance fluxes into and out of the bottom sediment layer described by AQUASIM.

The variables  $A$ ,  $U$ ,  $k$  and  $\epsilon$  in the first four components of the one-dimensional density array (3.74) are accessible by the program variables **Cross-Sectional Area**, **Horizontal Velocity**, **Turbulent Kinetic Energy** and **Dissipation**. The variables  $C_{L,i}$ ,  $C_{S,i}$ ,  $X_{L,i}$  and  $X_{S,i}$  are represented by dynamic volume state variables with dissolved ( $C_{L,i}$  and  $C_{S,i}$ ) or particulate ( $X_{L,i}$  and  $X_{S,i}$ ) properties.

The one-dimensional fluxes of the substances with one-dimensional densities as described by equation (3.74) are given as follows:

$$\hat{\mathbf{j}} = \begin{pmatrix} Q \\ -A\nu_t \frac{\partial U}{\partial z} + QU \\ -A \frac{\nu_t}{\sigma_k} \frac{\partial(\rho k)}{\partial z} + Q\rho k \\ -A \frac{\nu_t}{\sigma_\epsilon} \frac{\partial(\rho\epsilon)}{\partial z} + Q\rho\epsilon \\ -AK_z \frac{\partial C_{L,i}}{\partial z} + QC_{L,i} \\ -AK_z \frac{\partial X_{L,i}}{\partial z} + QX_{L,i} - Aw_{sed,i}X_{L,i} \\ 0 \\ 0 \end{pmatrix}, \quad (3.75)$$

where

$$\nu_t = PrK_z \quad (3.76)$$

is the turbulent viscosity,  $\sigma_k$  and  $\sigma_\epsilon$  are factors to change diffusion of  $k$  and  $\epsilon$  from turbulent viscosity,  $K_z$  is the coefficient of turbulent diffusion of substances dissolved or suspended in the water column,  $Pr$  is the Prandtl number,  $Q$  is the vertical discharge induced by water in- or outflows in the lake depth (positive upwards), and  $w_{sed,i}$  is the sedimentation velocity of the particles  $X_i$  (positive downwards). Terms of equation (3.75) that contain  $\nu_t$  or  $K_z$  describe vertical mixing by turbulent diffusion, those containing  $Q$  describe vertical advection with water flow, and the term containing  $w_{sed,i}$  vertical movement due to sedimentation. There is no vertical flux in the sediment, because the  $z$ -axis describes the vertical coordinate in the lake and not the depth within the sediment. The exchange between sediment layers is specified with the aid of source terms as given below.

The following one-dimensional source terms are required to complete the set of lake

equations:

$$\hat{\mathbf{r}} = \begin{pmatrix} q \\ -\left(\frac{1+\text{sign}\left(\frac{\partial U}{\partial z}\right)}{2}\right) \frac{dA}{dz} \nu_t \frac{\partial U}{\partial z} + Ar_U \\ A\rho P + A\rho G - A\rho\epsilon + \frac{dA}{dz} P_{bottom} + A\rho p_{int} \\ -\left(\frac{1+\text{sign}\left(\frac{\partial(\rho k)}{\partial z}\right)}{2}\right) \frac{dA}{dz} \nu_t \frac{\partial(\rho k)}{\partial z} \\ A\rho r_\epsilon - \left(\frac{1+\text{sign}\left(\frac{\partial(\rho k)}{\partial z}\right)}{2}\right) \frac{dA}{dz} \nu_t \frac{\partial(\rho\epsilon)}{\partial z} \\ -\frac{dA}{dz} \frac{D_i\theta_1}{\frac{h_{sed,1}}{2}} (C_{L,i} - C_{S1,i}) + Ar_{C_{L,i}} + \left(\frac{1+\text{sign}(q)}{2}\right) qC_{lat,i} + \left(\frac{1-\text{sign}(q)}{2}\right) qC_{L,i} \\ -\left(\frac{1+\text{sign}(w_{sed,i})}{2}\right) \frac{dA}{dz} w_{sed,i} X_{L,i} + \frac{dA}{dz} k_{res} X_{S,i} + Ar_{X_{L,i}} \\ + \left(\frac{1+\text{sign}(q)}{2}\right) qX_{lat,i} + \left(\frac{1-\text{sign}(q)}{2}\right) qX_{L,i} \\ \frac{dA}{dz} i_{ex,C_{S_j,i}} + \frac{dA}{dz} h_{sed,j} r_{C_{S_j,i}} \\ \frac{dA}{dz} i_{ex,X_{S_j,i}} + \frac{dA}{dz} h_{sed,j} r_{X_{S_j,i}} \end{pmatrix} \quad (3.77)$$

where

$$P = \nu_t \left( \frac{\partial U}{\partial z} \right)^2 \quad (3.78)$$

is the shear production of turbulent kinetic energy,

$$G = \begin{cases} \left( \frac{\nu_t}{Pr} - K_{z,min} \right) \frac{g}{\rho} \frac{\partial \rho}{\partial z} & \text{for } \frac{\nu_t}{Pr} > K_{z,min} \\ 0 & \text{else} \end{cases} \quad (3.79)$$

is the buoyancy production (or destruction) of turbulent kinetic energy (buoyancy production and destruction are set to zero if the turbulent diffusion coefficient is very small; this avoids negative values of turbulent kinetic energy in a stably stratified lake with no wind forcing),  $i_{ex,C_{S_j,i}}$  is the flux of the dissolved component  $i$  into the sediment layer  $j$

given as

$$i_{ex,CS_j,i} = \begin{cases} \frac{D_i \theta_1}{\frac{h_1}{2}} (C_L - C_{S1,i}) \\ \quad + \frac{D_i \min(\theta_1, \theta_2)}{\frac{h_1+h_2}{2}} (C_{S2,i} - C_{S1,i}) & \text{for } j = 1 \\ \\ \frac{D_i \min(\theta_{j-1}, \theta_j)}{\frac{h_{j-1}+h_j}{2}} (C_{S_{j-1},i} - C_{S_j,i}) \\ \quad + \frac{D_i \min(\theta_j, \theta_{j+1})}{\frac{h_j+h_{j+1}}{2}} (C_{S_j,i} - C_{S_{j+1},i}) & \text{for } 1 < j < n_{sed} \\ \\ \frac{D_i \min(\theta_{n_{sed}-1}, \theta_{n_{sed}})}{\frac{h_{n_{sed}-1}+h_{n_{sed}}}{2}} (C_{S_{n_{sed}-1},i} - C_{S_{n_{sed}},i}) \\ \quad + i_{sed,C_i} & \text{for } j = n_{sed} \end{cases} \quad (3.80)$$

(assuming the number of layers,  $n_{sed}$ , to be larger than 2 and neglecting advection due to sedimentation), and  $i_{ex,CS_j,i}$  is the flux of the particulate component  $i$  into the sediment layer  $j$  given as

$$i_{ex,CS_j,i} = \begin{cases} X_{L,i} w_{sed,i} - k_{res} X_{S1,i} \\ \quad - \left( \frac{1+\text{sign}(F_{vol,tot,2})}{2} \right) \frac{F_{vol,tot,2}}{1-\theta_1} X_{S1,i} \\ \quad + \left( \frac{1-\text{sign}(F_{vol,tot,2})}{2} \right) \frac{F_{vol,tot,2}}{1-\theta_2} X_{S2,i} & \text{for } j = 1 \\ \\ \left( \frac{1+\text{sign}(F_{vol,tot,j})}{2} \right) \frac{F_{vol,tot,j}}{1-\theta_{j-1}} X_{S_{j-1},i} \\ \quad - \left( \frac{1-\text{sign}(F_{vol,tot,j})}{2} \right) \frac{F_{vol,tot,j}}{1-\theta_j} X_{S_j,i} \\ \quad - \left( \frac{1+\text{sign}(F_{vol,tot,j+1})}{2} \right) \frac{F_{vol,tot,j+1}}{1-\theta_j} X_{S_j,i} \\ \quad + \left( \frac{1-\text{sign}(F_{vol,tot,j+1})}{2} \right) \frac{F_{vol,tot,j+1}}{1-\theta_{j+1}} X_{S_{j+1},i} & \text{for } 1 < j < n_{sed} \\ \\ \left( \frac{1+\text{sign}(F_{vol,tot,n_{sed}})}{2} \right) \frac{F_{vol,tot,n_{sed}}}{1-\theta_{n_{sed}-1}} X_{S_{n_{sed}-1},i} \\ \quad - \left( \frac{1-\text{sign}(F_{vol,tot,n_{sed}})}{2} \right) \frac{F_{vol,tot,n_{sed}}}{1-\theta_{n_{sed}}} X_{S_{n_{sed}},i} \\ \quad - \left( \frac{1+\text{sign}(F_{vol,tot,n_{sed}+1})}{2} \right) \frac{F_{vol,tot,n_{sed}+1}}{1-\theta_{n_{sed}}} X_{S_{n_{sed}},i} \\ \quad + \left( \frac{1-\text{sign}(F_{vol,tot,n_{sed}+1})}{2} \right) \frac{F_{vol,tot,n_{sed}+1}}{1-\theta_{n_{sed}+1}} X_{S_{n_{sed}},i} & \text{for } j = n_{sed} \end{cases} \quad (3.81)$$

with the total volume flux of particles from layer  $j-1$  to layer  $j$ ,  $F_{vol,tot,j}$  (layer 0 corresponds to the water column, layer  $n_{sed}+1$  to the sediment below the modelled sediment layers where the particle concentrations are assumed to be equal to those in the sediment layer  $n_{sed}$ ), given as

$$F_{vol,tot,1} = \sum_i \frac{X_{L,i} w_{sed,i}}{\rho_{X_i}} + F_{vol} - (1-\theta_1) k_{res} \quad (3.82a)$$

$$F_{vol,tot,j+1} = F_{vol,tot,j} + h_{sed,j} \sum_i \frac{r_{X_{S_j},i}}{\rho_{X_i}} \quad (3.82b)$$

The first component of equation (3.77) describes the lateral water inflow as volume per unit lake depth,  $q$ .

The second component of equation (3.77) describes the loss of horizontal velocity,  $U$ , at the sediment surface during vertical diffusive downward transport (the factor  $(1 + \text{sign}(dU/dz))/2$  makes this term only to be effective if velocity is transported downwards) and it includes a user-defined source term  $r_U$  for velocity that can be interpreted as a pressure gradient per unit of water mass.

The third component of equation (3.77) describes the one-dimensional source terms for turbulent kinetic energy. The first term describes production of turbulent kinetic energy by the shear forces induced by a gradient of the wind-driven horizontal velocity,  $U$ . The second term describes production or consumption of turbulent kinetic energy by turbulent mixing in the presence of stratification. This buoyancy production term is negative for the case of a stable stratification ( $\partial\rho/\partial z < 0$ ), because mixing increases the potential energy of the water column. In contrast, during phases of convective mixing ( $\partial\rho/\partial z > 0$ ), turbulent kinetic energy is produced. The third term of the third component of equation (3.77) describes loss of turbulent kinetic energy by dissipation,  $\epsilon$ . The fourth and fifth terms describe production of turbulent kinetic energy by bottom friction,  $P_{bottom}$ , and internal shear,  $p_{int}$ , of seiche oscillations, respectively, and the last term describes loss of turbulent kinetic energy,  $k$ , at the sediment surface during vertical diffusive downward transport (the factor  $(1 + \text{sign}(d(\rho k)/dz))/2$  makes this term only to be effective if turbulent kinetic energy is transported downwards).

The fourth component of equation (3.77) describes the one-dimensional source terms for dissipation of turbulent kinetic energy. The first term is a general source term to be defined by the program user. The expression for this term for the standard  $k$ - $\epsilon$  model is given as

$$r_\epsilon = c_1(P + c_3 G) \frac{\epsilon}{k} - c_2 \frac{\epsilon^2}{k} \quad (3.83)$$

with the nondimensional model parameters  $c_1$ ,  $c_2$  and  $c_3$  (Rodi, 1980; Rodi, 1987; Burchard and Baumert, 1995). The second term of the fourth component describes loss of dissipation,  $\epsilon$ , at the sediment surface during vertical diffusive downward transport (the factor  $(1 + \text{sign}(d(\rho k)/dz))/2$  makes this term only to be effective if dissipation is transported downwards).

The fifth component of equation (3.77) describes the effect of the exchange of dissolved substances between water column and the pore water of the first sediment layer, the effect of transformation processes, and the effect of water in- or outflows on the mass of dissolved substances in the water column. In this source term,  $D_i$  is the molecular diffusion coefficient of the dissolved substance  $i$  in the sediment pore water,  $r_{C_{L,i}}$  is the production rate per unit time and unit water volume due to transformation processes, and  $C_{lat,i}$  is the concentration of the dissolved substance  $i$  in the lateral inflow. The terms  $(1 \pm \text{sign}(q))/2$  switch between the concentration in the inflow,  $C_{lat,i}$ , for an inflow ( $q > 0$ ) and the concentration in the lake,  $C_{L,i}$ , for an outflow ( $q < 0$ ).

The sixth component of equation (3.77) describes the effect of sedimentation, resuspension, transformation processes and in- or outflows on the mass of particles in the water column. The first two terms describe the loss and gain of particles by sedimentation and

resuspension, respectively. In the second term,  $k_{res}$  describes the resuspension velocity of the sediment. The third term of the fifth component of equation (3.77) describes production of particles by transformation processes and the last two terms describe changes in particle concentration in the lake by in- and outflows, analogously to the dissolved substances in the fourth component.

The seventh component of equation (3.77) describes changes in mass of dissolved substances in the sediment layer  $j$  due to diffusive exchange with neighbouring sediment layers and due to transformation processes. The effect of advection on dissolved substances is neglected (because always the top sediment layers are described by AQUASIM, changes in sediment depth leads to an advective exchange between the sediment layers).

The eighth component of equation (3.77) describes changes in mass of particulate substances in the sediment layer  $j$  due to advection caused by sedimentation or by transformation processes.

In addition to the continuous conservation laws defined above, the energy stored in seiche oscillations is calculated by AQUASIM according to the following equation:

$$\frac{dE_{seiche}}{dt} = P_{wind} - A(z_0)P_{bottom} - \int_{z_B}^{z_0} \rho A p_{int} dz \quad (3.84)$$

where  $E_{seiche}$  is the energy stored in seiche oscillations and  $P_{wind}$  is the power of wind excitation of the seiche motion.

Application of the general law for differential conservation laws (3.73) to the definitions given by the equations (3.74) to (3.77) leads to the following set of 8 differential equations.

The first equation describes vertical water flow in the lake:

$$\frac{\partial Q}{\partial z} = q \quad . \quad (3.85)$$

The spatial gradient of the discharge,  $Q$ , is determined by the lateral inflow per unit lake depth,  $q$ . Positive values of  $q$  (inflow into the lake) increase the upward discharge, negative values (outflows) decrease the upward discharge.

The second equation describes the behaviour of the horizontal flow velocity:

$$\frac{\partial U}{\partial t} = \frac{1}{A} \frac{\partial}{\partial z} \left( A \nu_t \frac{\partial U}{\partial z} \right) - \left( \frac{1 + \text{sign}(\frac{\partial U}{\partial z})}{2} \right) \frac{1}{A} \frac{dA}{dz} \nu_t \frac{\partial U}{\partial z} - \frac{1}{A} \frac{\partial}{\partial z} (QU) + r_U \quad (3.86)$$

The horizontal flow velocity spreads diffusively (first term), it is dissipated if diffusive vertical downward transport ends at the sediment (second term), it follows the advective motion of the water column (third term), and an arbitrary pressure gradient,  $r_U$ , can be specified by the program user (fourth term).

The third equation describes the behaviour of turbulent kinetic energy:

$$\begin{aligned} \frac{\partial k}{\partial t} = \frac{1}{A} \frac{\partial}{\partial z} \left( A \frac{\nu_t}{\sigma_k} \frac{\partial k}{\partial z} \right) - \left( \frac{1 + \text{sign}(\frac{\partial k}{\partial z})}{2} \right) \frac{1}{A} \frac{dA}{dz} \frac{\nu_t}{\sigma_k} \frac{\partial k}{\partial z} \\ - \frac{1}{A} \frac{\partial}{\partial z} (Qk) + P + G - \epsilon + \frac{1}{A} \frac{dA}{dz} \frac{P_{bottom}}{\rho} + p_{int} \end{aligned} \quad (3.87)$$

In order to derive this equation from the equations (3.74) to (3.77), the gradient of the density,  $\rho$ , was neglected. Turbulent kinetic energy spreads diffusively (first term), it is

dissipated if diffusive vertical downward transport ends at the sediment (second term), it follows the advective motion of the water column (third term), it is produced by the shear forces of the horizontal flow (fourth term), it is produced (in an unstable water column) or consumed (in a stable water column) by turbulent mixing (fifth term), it is dissipated (sixth term), it is produced by bottom friction of seiche oscillations (seventh term), and it is produced by the shear of seiche oscillations produced within the water column (eighth term).

The fourth equation describes the behaviour of dissipation:

$$\frac{\partial \epsilon}{\partial t} = \frac{1}{A} \frac{\partial}{\partial z} \left( A \frac{\nu_t}{\sigma_\epsilon} \frac{\partial k}{\partial z} \right) - \left( \frac{1 + \text{sign}(\frac{\partial \epsilon}{\partial z})}{2} \right) \frac{1}{A} \frac{dA}{dz} \frac{\nu_t}{\sigma_\epsilon} \frac{\partial \epsilon}{\partial z} - \frac{1}{A} \frac{\partial}{\partial z} (Qk) + r_\epsilon \quad (3.88)$$

In order to derive this equation from the equations (3.74) to (3.77), the gradient of the density,  $\rho$ , was neglected. Dissipation spreads diffusively (first term), it is eliminated if diffusive vertical downward transport ends at the sediment (second term), it follows the advective motion of the water column (third term), and it is produced by a source term (fourth term). It is usual to use equation (3.83) in order to parameterize this source term of the  $\epsilon$  equation.

The fifth equation describes the behaviour of dissolved substances in the water column:

$$\begin{aligned} \frac{\partial C_{L,i}}{\partial t} = & \frac{1}{A} \frac{\partial}{\partial z} \left( AK_z \frac{\partial C_{L,i}}{\partial z} \right) - \frac{1}{A} \frac{\partial}{\partial z} (QC_{L,i}) - \frac{1}{A} \frac{dA}{dz} \frac{D_i \theta_1}{\frac{h_{sed,1}}{2}} (C_{L,i} - C_{S_1,i}) \\ & + r_{C_{L,i}} + \left( \frac{1 + \text{sign}(q)}{2} \right) \frac{q}{A} C_{lat,i} + \left( \frac{1 - \text{sign}(q)}{2} \right) \frac{q}{A} C_{L,i} \end{aligned} \quad (3.89)$$

Dissolved substances spread diffusively (first term), they follow the advective motion of the water column (second term), they are exchanged with the pore water of the first sediment layer (third term), they can be converted by transformation processes (fourth term), and they are influenced by inflows and outflows (fifth and sixth terms, respectively).

The sixth equation describes the behaviour of particles in the water column:

$$\begin{aligned} \frac{\partial X_{L,i}}{\partial t} = & \frac{1}{A} \frac{\partial}{\partial z} \left( AK_z \frac{\partial X_{L,i}}{\partial z} \right) - \frac{1}{A} \frac{\partial}{\partial z} (QX_{L,i}) \\ & + \frac{\partial}{\partial z} (w_{sed,i} X_{L,i}) + \left( \frac{1 - \text{sign}(w_{sed,i})}{2} \right) \frac{dA}{dz} w_{sed,i} X_{L,i} + \frac{1}{A} \frac{dA}{dz} k_{res} X_{S_1,i} \\ & + r_{X_{L,i}} + \left( \frac{1 + \text{sign}(q)}{2} \right) \frac{q}{A} X_{lat,i} + \left( \frac{1 - \text{sign}(q)}{2} \right) \frac{q}{A} X_{L,i} \end{aligned} \quad (3.90)$$

Particles spread diffusively (first term), they follow the advective motion of the water column (second term), they sink through the water column with their sedimentation velocity (third term), they can be resuspended from the sediment (fourth term), they can be converted by transformation processes (fifth term), and they are influenced by inflows and outflows (sixth and seventh terms, respectively).

The seventh equation describes the behaviour of dissolved substances in the pore water of sediment layer  $j$ :

$$\frac{\partial C_{S_j,i}}{\partial t} = \frac{i_{ex,C_{S_j,i}}}{\theta_j h_{sed,j}} + \frac{r_{C_{S_j,i}}}{\theta_j} \quad (3.91)$$

Substances dissolved in the pore water of the sediment layer are exchanged with the neighbouring layers (first term) and they are changed by transformation processes (second

term). Note that the process rates must be formulated as mass per unit of time and per unit of total sediment volume (and not only per pore water volume). This facilitates the formulation of mass conversion for transformation processes from the particulate to the dissolved phase.

The last equation describes the behaviour of particles in the sediment:

$$\frac{\partial X_{S_{j,i}}}{\partial t} = \frac{i_{ex,X_{S_{j,i}}}}{h_{sed,j}} + r_{X_{S,i}} \quad (3.92)$$

The particle concentrations in the sediment layer are exchanged with the neighbouring layers (first term) and they are changed by transformation processes (second term).

In order to make the solution of the above system of differential equations unique, one boundary condition for the ordinary differential equation (3.85) and two boundary conditions for each of the partial differential equations (3.86) to (3.90) are required. The ordinary differential equations (3.91) and (3.92) that do not contain spatial derivatives do not require boundary conditions.

The boundary condition for the equation (3.85) describing vertical discharge at the lake bottom is given by

$$Q(z_B) = 0 \quad . \quad (3.93a)$$

According to the equation (3.85) this results in a discharge of

$$Q(z_0) = \int_{z_B}^{z_0} q \, dz \quad (3.93b)$$

at the lake surface. This discharge is assumed to leave (if it is positive) or to enter (if it is negative) the lake at its surface in order to keep the lake volume constant.

The boundary conditions for the equation (3.86) describing the wind-driven horizontal flow are given by

$$U(z_B) = 0 \quad (3.94a)$$

and

$$\rho \nu_t \frac{\partial U}{\partial z}(z_0) = \tau_{surf} \quad (3.94b)$$

where  $\tau_{surf}$  is the wind shear at the lake surface.

The boundary conditions for the equation (3.87) describing the turbulent kinetic energy are approximated to be no-flux conditions:

$$\frac{\partial k}{\partial z}(z_B) = 0 \quad (3.95a)$$

and

$$\frac{\partial k}{\partial z}(z_0) = 0 \quad (3.95b)$$

These boundary conditions neglect changes in turbulent kinetic energy due to effects of the lake surface and the lake bottom.

The boundary conditions for the equation (3.88) describing the dissipation of turbulent kinetic energy are approximated to be no-flux conditions:

$$\frac{\partial \epsilon}{\partial z}(z_B) = 0 \quad (3.96a)$$

and

$$\frac{\partial \epsilon}{\partial z}(z_0) = 0 \quad (3.96b)$$

These boundary conditions neglect changes in dissipation of turbulent kinetic energy due to effects of the lake surface and the lake bottom.

The boundary conditions for the equation (3.89) describing the behaviour of dissolved substances in the water column of the lake are given by the continuity of the fluxes over the boundary:

$$K_z \frac{\partial C_{L,i}}{\partial z}(z_B) = \theta k_{diff,i} (C_{L,i}(z_B) - C_{S,i}(z_B)) \quad (3.97a)$$

and

$$K_z \frac{\partial C_{L,i}}{\partial z}(z_0) = i_{surf,C_i} \quad (3.97b)$$

where  $i_{surf,C_i}$  is the (given) flux of the dissolved substance  $i$  through the lake surface as mass per unit lake surface area and per unit time (positive).

The boundary conditions for the equation (3.90) describing the behaviour of particles in the water column of the lake are given by the continuity of the fluxes over the boundary:

$$K_z \frac{\partial X_{L,i}}{\partial z}(z_B) = -k_{res} X_{S,i} \quad (3.98a)$$

and

$$K_z \frac{\partial X_{L,i}}{\partial z}(z_0) = i_{surf,X_i} \quad (3.98b)$$

where  $i_{surf,X_i}$  is the (given) flux of substance  $X_i$  through the lake surface as mass per unit lake surface area and per unit time (positive).



## User Definitions

Within a lake compartment two zones, the **water column** and the **sediment layer**, are distinguished. Variables and process rates can be made dependent on the zone by using the program variable **Zone Index** which takes the value 0 in the water column and 1 in the sediment.

Figure 3.78 shows the dialog box used for defining or editing a lake compartment. The

The dialog box 'Edit Lake Compartment' contains the following fields and controls:

- Name:** LakeZuerich
- Comp. Index:** 0
- Description:** (empty text field)
- Options:** Buttons for Variables, Processes, Init. Cond., and Input.
- Properties of:** Buttons for Particulate Variables and Dissolved Variables.
- Grav. Accel.:** 7.32e+010
- Top Coord.:** 0
- Bottom Coord.:** 136
- Cross Sect.:** A\_Zuerich
- Density:** rho\_w
- Turb. Diffusion:** Kz\_Zuerich
- Modes:**
  - ☐ without Sediment
  - ☒ with Sediment
  - ☒ without TKE
  - ☐ with TKE
- Buttons:** Sed. Prop., TKE Prop., Acc.
- Num. Grid Pts.:** 70
- Resolution:** ☒ low, ☐ high
- ☒ active for calculation
- Buttons:** OK, Cancel

Figure 3.78: Dialog box for editing a lake compartment.

edit field '**Name**' is used to specify the name of the compartment. Each compartment needs a unique name as an identifier. A name of a compartment consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

The edit field '**Comp. Index**' can be used to specify a nonnegative integer number as a compartment index. This value can be accessed with the aid of the program variable **Compartment Index** to make variables or process rates dependent on the compartment.

To improve documentation of compartments, the edit field '**Description**' can optionally be used to store comments on specific implementation features of a compartment.

The buttons '**Variables**', '**Processes**', '**Init. Cond.**' and '**Input**' are used to activate and inactivate state variables, to activate and inactivate processes, to specify initial conditions and to define s to a compartment. These options are described later in this subsection.

The buttons '**Particulate Variables**' and '**Dissolved Variables**' are used to define properties of dissolved or particulate substances for dynamic volume state variables. These options are also described later in this subsection.

The edit field '**Grav. Acceleration**' is used to specify the value of the gravitational acceleration,  $g$ , in the units as used by the user of the program. Typical values of the

gravitational acceleration are  $9.81\text{m/s}^2$ ,  $1.27 \cdot 10^8\text{m/h}^2$  or  $7.32 \cdot 10^{10}\text{m/d}^2$ . The gravitational acceleration is required in order to calculate the stability or Brunt Vaisälä frequency,  $N^2$ , defined by equation (3.99) below.

The edit fields **‘Top Coord.’** and **‘Bottom Coord.’** are used to define the location of the water surface,  $z_0$ , and of the deepest point of the lake,  $z_B$ , respectively. Although all equations in this subsection are formulated with the  $z$ -axis pointing upwards ( $z_0 > z_B$ ), the program works also correctly with a reverse  $z$ -axis ( $z_0 < z_B$ ). Both of these locations,  $z_0$  and  $z_B$ , are fixed numbers. It is not possible to describe varying water levels or changes in volume due to sedimentation with the current version of the lake compartment of AQUASIM.

The edit field **‘Cross. Sect.’** is used to specify the cross-sectional area of the lake,  $A$ , as a function of the vertical coordinate  $z$  which is accessible by the program variable **Space Coordinate Z**. For most applications to real systems the cross-sectional area  $A$  will be defined as a real list variable with the argument  $z$  and the name of this real list variable is then entered in this field. However, it is also possible to specify the cross-sectional area as an algebraic expression. Note that the cross-sectional area must be a non-increasing function with increasing depth of the lake and that the value of the surface cross-sectional area at  $z_B$  must still be positive. It is not allowed to use a time dependence in the cross-sectional area.

The edit field **‘Density’** is used to specify the density,  $\rho$ , of water as a function of temperature and concentrations of dissolved and suspended substances. An irrelevant constant value can be entered here, if the coefficient of turbulent diffusion,  $K_z$ , is given without reference to the stratification of the water column and if the value of the stability or Brunt Vaisälä frequency

$$N^2 = -\frac{g}{\rho} \frac{\partial \rho}{\partial z} \quad (3.99)$$

is not used. In the case of a freshwater lake, the following formula can be used to approximate the density of the lake water (Bührer and Ambühl, 1975):

$$\rho \approx 999.84298 \text{ kg/m}^3 + 10^{-3} \text{ kg/m}^3 \cdot \left( 65.4891 \text{ }^\circ\text{C}^{-1} T - 8.56272 \text{ }^\circ\text{C}^{-2} T^2 + 0.059385 \text{ }^\circ\text{C}^{-3} T^3 \right) \cdot \left( 1 + \beta_\kappa \kappa_{20} \right) \quad (3.100)$$

where  $T$  is temperature,  $\beta_\kappa$  is given as

$$\beta_\kappa \approx 0.705 \cdot 10^{-6} \text{ cm}/\mu\text{S} \quad (3.101)$$

and  $\kappa_{20}$  is the electrical conductivity at  $20^\circ\text{C}$  that can be calculated from the conductivity at temperature  $T$ ,  $\kappa_T$ , according to

$$\kappa_{20} \approx \kappa_T \cdot \left( 1.72118 - 0.0541369 \text{ }^\circ\text{C}^{-1} T + 1.14842 \cdot 10^{-3} \text{ }^\circ\text{C}^{-2} T^2 - 1.222651 \cdot 10^{-5} \text{ }^\circ\text{C}^{-3} T^3 \right) \quad (3.102)$$

The edit field **‘Diffusion’** is used to specify the coefficient of turbulent diffusion,  $K_z$ .  $K_z$  can be given as any function of time, space, stability etc.. Some formulations for  $K_z$  are described in the following. The coefficient of turbulent diffusion can be parameterized as a simple function of space and time:

$$K_z = \text{given function of } z \text{ and } t. \quad (3.103a)$$

The parameters of such a function can be estimated by AQUASIM using measured data of temperature or substances without or with known transformation processes. An alternative can be to parameterize the coefficient of vertical turbulent diffusion using the stability or Brunt Vaisälä frequency,  $N^2$  (3.99) that is accessible as the program variable **Brunt Vaisala Frequency**. A parameterization often used in the literature is given by

$$K_z = \begin{cases} \frac{a}{(N^2)^b} & \text{if } \frac{a}{(N^2)^b} \leq K_{z,max} \\ K_{z,max} & \text{if } N^2 \leq 0 \text{ or } \frac{a}{(N^2)^b} > K_{z,max} \end{cases} \quad (3.103b)$$

The parameters  $a$  and  $b$  of this expression can be estimated by AQUASIM using several temperature profiles of the lake. An additional option is to use the turbulence submodel to estimate the coefficient of vertical turbulent diffusion. A possible parameterization is

$$K_z = c_\mu \frac{k^2}{\epsilon} \cdot \frac{1}{Pr} \quad (3.103c)$$

The factor  $1/Pr$  is necessary in order to convert from the turbulent viscosity,  $\nu_t$ , to the diffusion coefficient for substances and temperature.

There are four **‘Modes’** of calculation for the lake compartment that can be selected with the radio buttons **‘without Sediment’**, **‘with Sediment’**, **‘without TKE’** and **‘with TKE’**. If the sediment submodel is turned on, its options must be specified in a dialog box appearing after clicking the button **‘Sed. Prop.’**. Similarly, if the submodel for the calculation of turbulent kinetic energy (TKE) is turned on, its options must be specified by clicking the button **‘TKE Prop.’**. The dialog boxes for the specification of these submodels are discussed later in this subsection.

As last options, **‘Num. Grid Pts.’**, **‘Resolution’** and **‘Acc.’**, in the dialog box used for the definition of a lake compartment (Fig. 3.78) the user can select the number of grid points, the discretization order of the numerical algorithm and the accuracies of program variables. The number of grid points is used to specify by how many discrete points the continuous  $z$ -axis is approximated. If the number of grid points is set to  $n_{gp}$ , the depth of the lake is resolved by 2 boundary points and  $n_{gp} - 2$  grid points located in the middle of  $n_{gp} - 2$  cells of equal thickness. For this division of the  $z$ -axis the **‘low resolution’** method applies a simple first-order discretization scheme that is very robust but can have significant numerical diffusion. The **‘high resolution’** method uses a second-order discretization scheme that applies flux-limiters to avoid oscillations of the numerical solutions (Reichert, 1994b, chapter 6). The button **‘Acc.’** is used to specify the numerical accuracies of program variables as described later in this subsection.

As for each compartment, the user has to select which state variables are active. This is done by clicking the button **‘Variables’** of the dialog box shown in Fig. 3.78. This action opens the dialog box shown in Fig. 3.79. The two list boxes of this dialog box show the active variables and the available variables, respectively. The button **‘Activate’** is used to activate available variables selected in the right list box, and the button **‘Inactivate’** is used to inactivate active variables selected in the left list box. If an active variable is selected while activating another variable, the new active variable is inserted in the list of active variables immediately before the selected variable, otherwise it is appended to the end of the list of active variables. This gives the user the possibility to influence the order

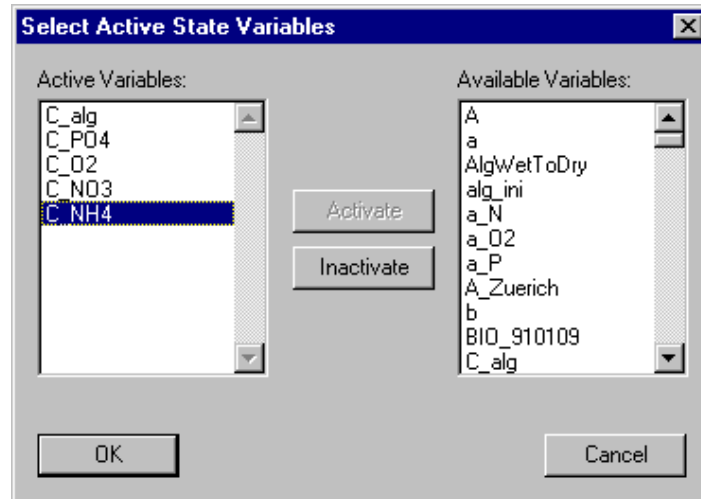


Figure 3.79: Dialog box for activating and inactivating state variables in a compartment.

of active variables (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The list of active variables may contain variables of any type, but activation has only an effect to state variables. Inactive state variables return a value of zero. The reason for allowing other types of variables in the list of active state variables is to facilitate switching between different models that do not contain the same state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of active variables of the compartments.

Similarly to the activation of state variables, the user has to select which processes are active in a compartment. This is done by clicking the button '**Processes**' of the dialog box shown in Fig. 3.78. This action opens the dialog box shown in Fig. 3.80. The two list

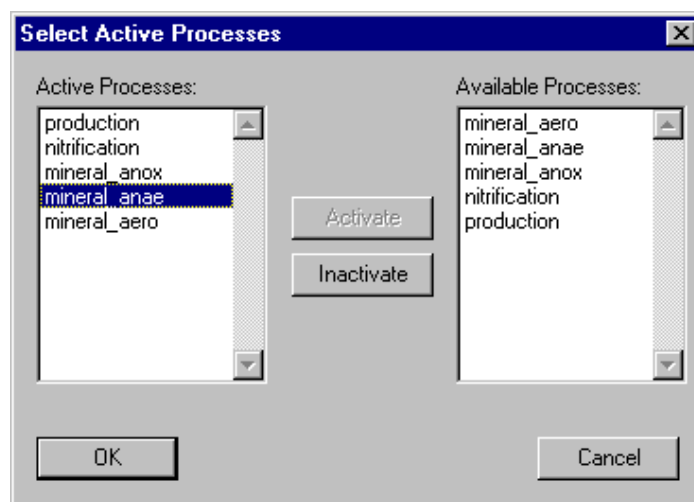


Figure 3.80: Dialog box for activating and inactivating processes in a compartment.

boxes in this dialog box show the active processes and the available processes, respectively. The button **‘Activate’** is used to activate available processes selected in the right list box, and the button **‘Inactivate’** is used to inactivate a active processes selected in the left list box. If an active process is selected while activating another process, the new active process is inserted in the list of active processes immediately before the selected process, otherwise it is appended to the end of the list of active processes. This gives the user the possibility to influence the order of active processes (the order is irrelevant for the program, but it may be convenient for the user to have a certain order).

Initial conditions for a lake compartment can be specified by clicking the button **‘Init. Cond.’** of the dialog box shown in Fig. 3.78. This action opens the dialog box shown in Fig. 3.81. This dialog box shows a list of all initial conditions already specified.

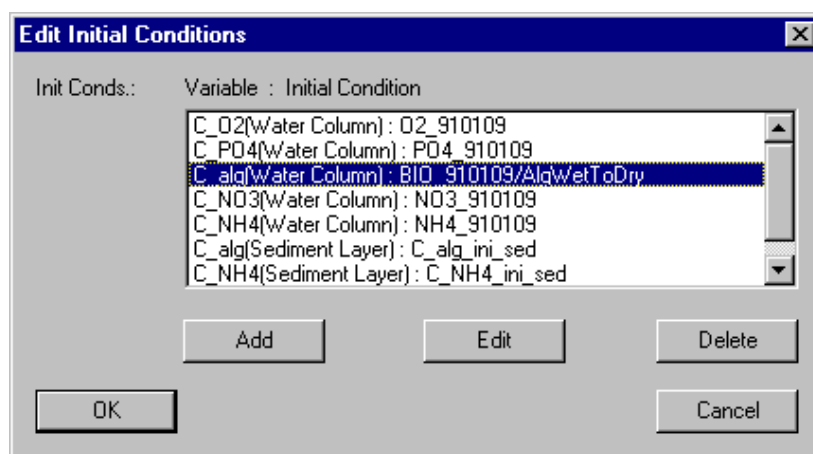


Figure 3.81: Dialog box for editing initial conditions for a lake compartment.

Each row of the list box contains the name of a variable, followed by the zone of the compartment for which the initial condition is specified (in brackets) and by the algebraic expression specifying the initial value. For each combination of a variable with a zone only one unique initial condition can be specified. The list of initial conditions can be edited using the buttons **‘Add’**, **‘Edit’** and **‘Delete’**. If an initial condition is selected while adding a new initial condition, the new initial condition is inserted in the list immediately before the selected initial condition, otherwise it is appended to the end of the list of initial conditions. This gives the user the possibility to influence the order of the initial conditions (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). The dialog box used for editing a single initial condition is shown in Fig. 3.82. In this dialog box the fields **‘Variable’**, **‘Zone’** and **‘Init. Cond.’** allow the user to select a variable and a zone and to specify an initial condition. Note that in the lake compartment the two zones **Water Column** and **Sediment Layer** are available. Initial conditions for any type of variables can be specified, but only initial conditions for state variables and for the program variables **Brunt Vaisala Frequency** ( $N^2$ ), **Horizontal Velocity** ( $U$ ), **Turbulent Kinetic Energy** and **Energy of Seiche Oscillation** are used by the program. The reason for allowing to define initial conditions for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement

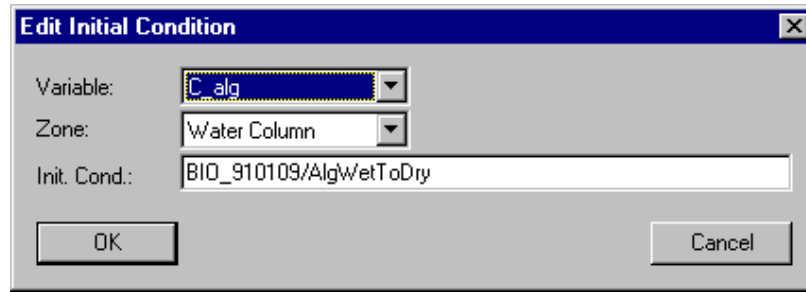


Figure 3.82: Dialog box for editing a single initial conditions for a lake compartment.

ot editing the lists of initial conditions of the compartments. Initial conditions of a lake compartment may not depend on state variables and they can only depend on the program variables Calculation Number, Time and Space Coordinate Z.

Input to a lake compartment can be specified by clicking the button ‘**Input**’ of the dialog box shown in Fig. 3.78. This action opens the dialog box shown in Fig. 3.83. In this

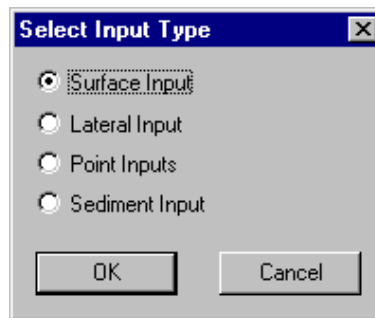


Figure 3.83: Dialog box for selecting an type for a lake compartment.

dialog box the user can select which type of input to edit. There exist four different types of inputs to a lake compartment. The radio button ‘**Surface Input**’ can be used to describe water and substance exchange over the lake surface, the radio buttons ‘**Lateral Input**’ and ‘**Point Input**’ are two alternative ways of describing water and substance in- or outflow distributed over the depth of the lake, and the radio button ‘**Sediment Input**’ is used to describe substance exchange between the sediment layer described by the model and deeper sediment layers. In the following paragraphs, the dialog boxes used to define each of these input types are described.

Selection of the radio button ‘**Surface Input**’ in the dialog box shown in Fig. 3.83 opens the dialog box shown in Fig. 3.84. This dialog box contains a list box for defining the substance mass fluxes,  $i_{surf,C_i}$  and  $i_{surf,X_i}$ , accross the lake surface. For each variable only one unique surface input flux can be specified. The surface input fluxes can be edited using the buttons ‘**Add**’, ‘**Edit**’ and ‘**Delete**’. If an input flux is selected while adding a new input flux, the new input flux is inserted in the list immediately before the selected input flux, otherwise it is appended to the end of the list of input fluxes. This gives the user the possibility to influence the order of the input fluxes (the order is irrelevant for

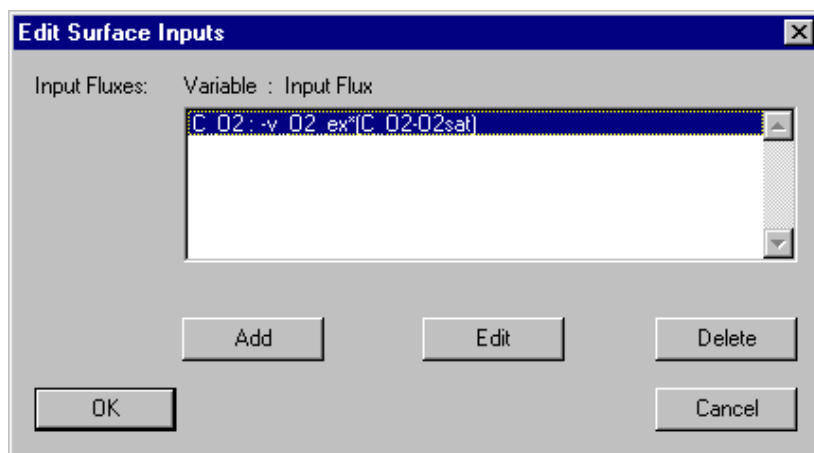


Figure 3.84: Dialog box for editing surface inputs to a lake compartment.

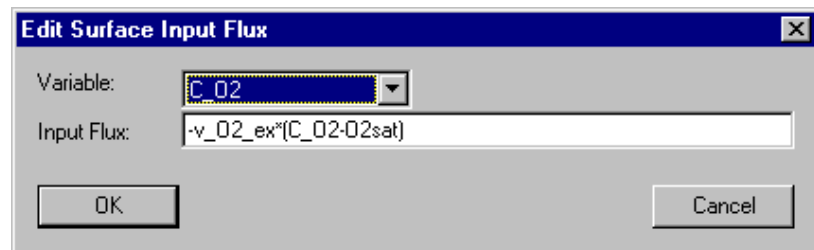


Figure 3.85: Dialog box for editing a single surface input to a lake compartment.

the program, but it may be convenient for the user to have a certain order). Figure 3.85 shows the dialog box used to specify a single surface input flux of a dynamic volume state variable. In this dialog box the fields ‘**Variable**’ and ‘**Input Flux**’ allow the user to select a variable and to specify a surface input flux,  $i_{surf,C_i}$  or  $i_{surf,X_i}$ . Note that this input flux represents a mass flux per unit lake surface area and per unit of time. A positive value of a surface flux represents a flux into the lake, a negative value a flux out of the lake. A surface input flux of a lake compartment may depend on dynamic volume state variables. These variables return the values of state variables in the lake immediately at the surface. The use of state variables makes it possible to model diffusive fluxes through the surface boundary layer. Input fluxes for any type of variables can be specified, but only input fluxes for dynamic volume state variables are used by the program. The reason for allowing to define input fluxes for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of input fluxes of the compartments.

Selection of the radio button ‘**Lateral Input**’ in the dialog box shown in Fig. 3.83 opens the dialog box shown in Fig. 3.86. The edit field ‘**Water Inflow**’ of this dialog box is used to specify the discharge of water per unit depth into the lake,  $q$ , and the list box contains substance concentrations,  $C_{lat,i}$  or  $X_{lat,i}$ , in the inflowing water. A positive value

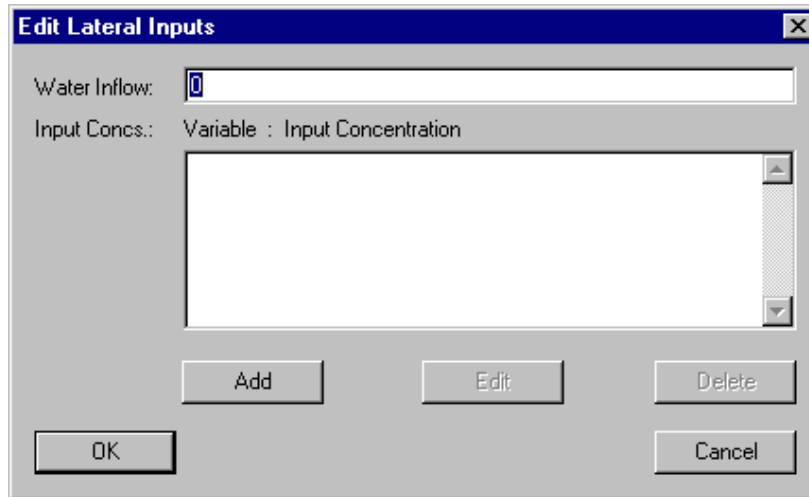


Figure 3.86: Dialog box for editing lateral inputs to a lake compartment.

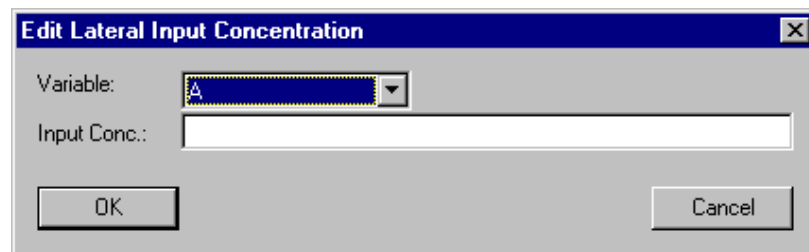


Figure 3.87: Dialog box for editing a single lateral input to a lake compartment.

of  $q$  represents a flow into the lake, a negative value represents an outflow. According to the equations (3.77), (3.89) and (3.90) the inflow concentration is irrelevant if  $q$  is smaller than or equal to zero (outflow). For each variable only one unique lateral inflow concentration can be specified. The list of inflow concentrations can be edited using the buttons ‘Add’, ‘Edit’ and ‘Delete’. If an inflow concentration is selected while adding a new inflow concentration, the new inflow concentration is inserted in the list immediately before the selected inflow concentration, otherwise it is appended to the end of the list of inflow concentrations. This gives the user the possibility to influence the order of the inflow concentrations (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.87 shows the dialog box used to specify a single lateral inflow concentration of a dynamic volume state variable. In this dialog box the fields ‘Variable’ and ‘Inflow Conc.’ allow the user to select a variable and specify a lateral inflow concentration,  $C_{lat,i}$ . A lateral inflow concentration of a lake compartment may depend on the program variables **Discharge** and on state variables. These variables return the discharge and the current values of state variables in the compartment as a function of the depth  $z$ . Inflow concentrations for any type of variables can be specified, but only inflow concentrations for dynamic volume state variables are used by the program. The reason for allowing to define inflow concentrations for variables of other types is to



facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of lateral inflow concentrations of the compartments.

Selection of the radio button '**Point Inputs**' in the dialog box shown in Fig. 3.83 opens the dialog box shown in Fig. 3.88. This dialog box contains a list of the names

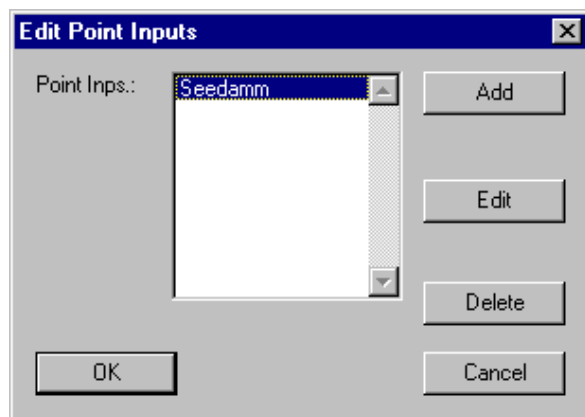


Figure 3.88: Dialog box for editing point inputs to a lake compartment.

of all point inputs to the lake (typically rivers or technical discharges). This list can be edited with the aid of the buttons '**Add**', '**Edit**' and '**Delete**'. Figure 3.89 shows the dialog box used to specify a single point input to a lake. The edit field '**Name**' is used to specify the name of the point input. Each point input needs a unique name as an identifier. A name of a point input consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. To improve documentation of point inputs the edit field '**Description**' can optionally be used to store comments on specific implementation features of a compartment. The edit fields '**Mean Depth**' and '**Range**' are used to determine the vertical location of the input. The input is equidistributed over the depth range from **Mean Depth** minus one half of **Range** to **Mean Depth** plus one half of **Range** (the input depth range is bounded by the lake bottom and the lake surface). The definitions of **Mean Depth** and **Range** can depend on time in order to simulate temporary events of higher density of a river during a storm event. The edit field '**Water Inflow**' is used to specify the discharge of water of the point inflow. A positive value of this variable represents an inflow into the lake, a negative value an outflow. The list box '**Concentrations**' contains the concentrations of substances represented by dynamic volume state variables. For each variable only one unique inflow concentration can be specified. The inflow concentrations specified here are irrelevant if the water inflow is negative or zero. The list of inflow concentrations can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If an inflow concentration is selected while adding a new concentration, the new inflow concentration is inserted in the list immediately before the selected concentration, otherwise it is appended to the end of the list of inflow concentrations. This gives the user the possibility to influence the order of the inflow concentrations (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.90 shows the dialog box used to specify an inflow concentration for a selected state variable to a lake. In this dialog box the

The dialog box is titled "Edit Point Input". It contains the following fields and controls:

- Name:** Seedamm
- Description:** inflow Seedamm
- Mean Depth:** 10
- Range:** 20
- Water Inflow:** inflow\_Qd\_91
- Input Concs.:** Variable : Input Concentration
- Input Concentration List:**
  - C\_PO4 : inflow\_PO4/1000
  - C\_NO3 : inflow\_NO3/1000** (highlighted)
  - C\_NH4 : inflow\_NH4/1000
- Buttons:** Add, Edit, Delete, OK, Cancel

Figure 3.89: Dialog box for editing a single point input to a lake compartment.

The dialog box is titled "Edit Point Input Concentration". It contains the following fields and controls:

- Variable:** C\_NO3
- Input Conc.:** inflow\_NO3/1000
- Buttons:** OK, Cancel

Figure 3.90: Dialog box for editing a point input concentration for a lake compartment.

fields '**Variable**' and '**Inflow Conc.**' allow the user to select a variable and specify an inflow concentration. Inflow concentrations for any type of variables can be specified, but only inflow concentrations for dynamic volume state variables are used by the program. The reason for allowing to define inflow concentrations for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of inflow concentrations of the compartments.

Selection of the radio button '**Sediment Input**' in the dialog box shown in Fig. 3.83 opens the dialog box shown in Fig. 3.91. This dialog box contains a list box for defining the substance mass fluxes,  $i_{sed,C_i}$  and  $i_{sed,X_i}$ , into the lake sediment. For each variable only one unique sediment input flux can be specified. The sediment input fluxes can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'. If a sediment input flux is selected while adding a new sediment input flux, the new input flux is inserted in the list

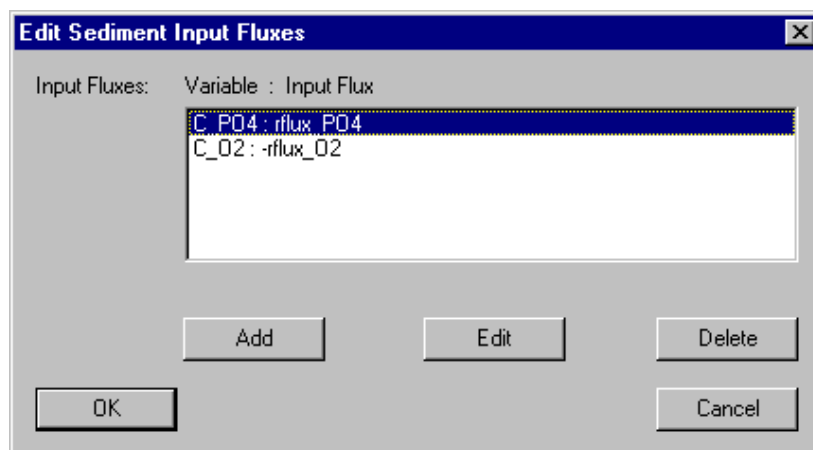


Figure 3.91: Dialog box for editing sediment inputs to a lake compartment.

immediately before the selected input flux, otherwise it is appended to the end of the list of sediment input fluxes. This gives the user the possibility to influence the order of the sediment input fluxes (the order is irrelevant for the program, but it may be convenient for the user to have a certain order). Figure 3.92 shows the dialog box used to specify a single sediment input flux of a dynamic volume state variable. In this dialog box the

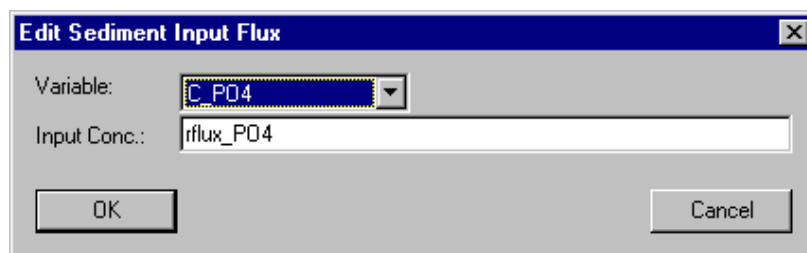


Figure 3.92: Dialog box for editing a single sediment input to a lake compartment.

fields '**Variable**' and '**Input Flux**' allow the user to select a variable and to specify a surface input flux,  $i_{sed,C_i}$  or  $i_{sed,X_i}$ . Note that this input flux represents a mass flux per unit sediment area and per unit of time. A positive value of a sediment flux represents a flux into the sediment, a negative value a flux out of the sediment. A sediment input flux of a lake compartment may depend on dynamic volume state variables. These variables return the values of state variables in the lake sediment. The use of state variables makes it possible to model diffusive fluxes from deeper sediment layers to the sediment layer described by the lake model. Input fluxes for any type of variables can be specified, but only input fluxes for dynamic volume state variables are used by the program. The reason for allowing to define input fluxes for variables of other types is to facilitate the users switching between models with different state variables. A user can change a state variable to a variable of another type without the requirement of editing the lists of input fluxes of the compartments.

The button '**Particulate Variables**' of the dialog box shown in Fig. 3.78 allows the

user to specify properties of particulate variables for dynamic volume state variables. Figure 3.93 shows the dialog box used for this purpose. It contains a list box with the

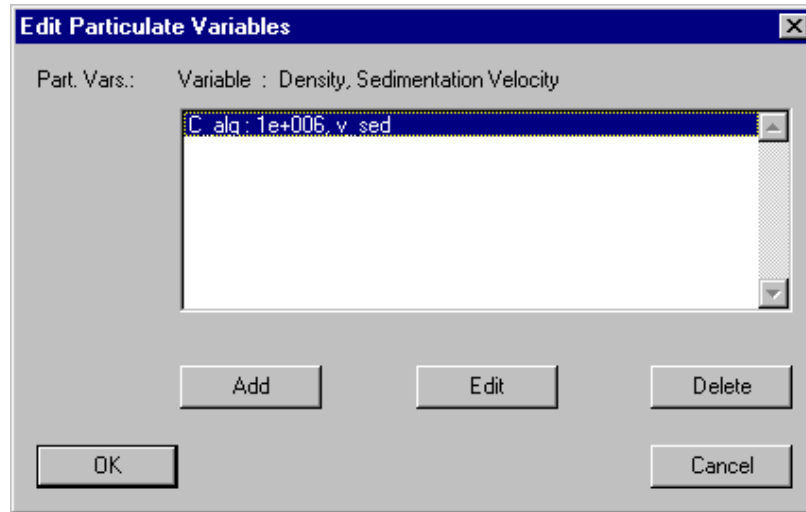


Figure 3.93: Dialog box for editing particulate variables of a lake compartment.

names, densities and sedimentation velocities of all defined particulate variables. Note that a variable can only be once in both lists of particulate and dissolved variables. The buttons ‘Add’, ‘Edit’ and ‘Delete’ allow the user to edit this list. Figure 3.94 shows the dialog box used to specify the properties of a particulate variable in a lake. The edit fields

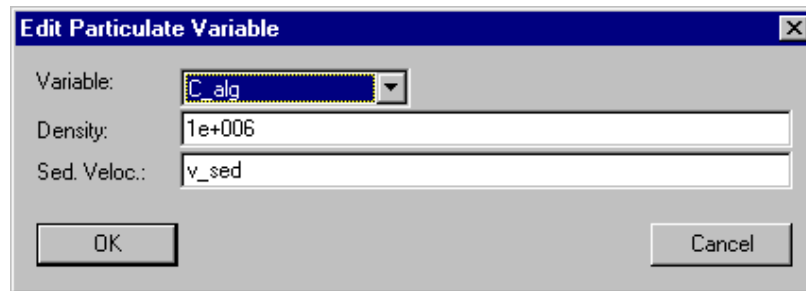


Figure 3.94: Dialog box for editing a single particulate variable for a lake compartment.

‘Variable’, ‘Density’ and ‘Sedimentation Velocity’ allow the user to select a variable and to specify a density,  $\rho_{X_i}$ , and a sedimentation velocity,  $w_{sed,i}$ . The density  $\rho_{X_i}$  is used in equation (3.82) to calculate the volume flux of particles between the sediment layers. A positive sedimentation velocity,  $w_{sed,i}$ , describes downward movement. According to the differential equations described above, this downward movement includes movement through the water column and settling to the sediment if the lake cross-sectional area decreases with increasing depth of the lake. If for some special purposes, a negative sedimentation velocity is used, this indicates upward movement. In this case, an increase of the lake cross-sectional area with decreasing depth leads to a dilution of the moving particles. Note that properties of particulate variables can be specified for any type of

variables, but that only properties of dynamic volume state variables have an effect. The larger class of variables is allowed in order to facilitate the users to switch between models with different state variables.

The button ‘**Dissolved Variables**’ of the dialog box shown in Fig. 3.78 allows the user to specify properties of dissolved variables for dynamic volume state variables. Figure 3.95 shows the dialog box used for this purpose. It contains a list box with the names and

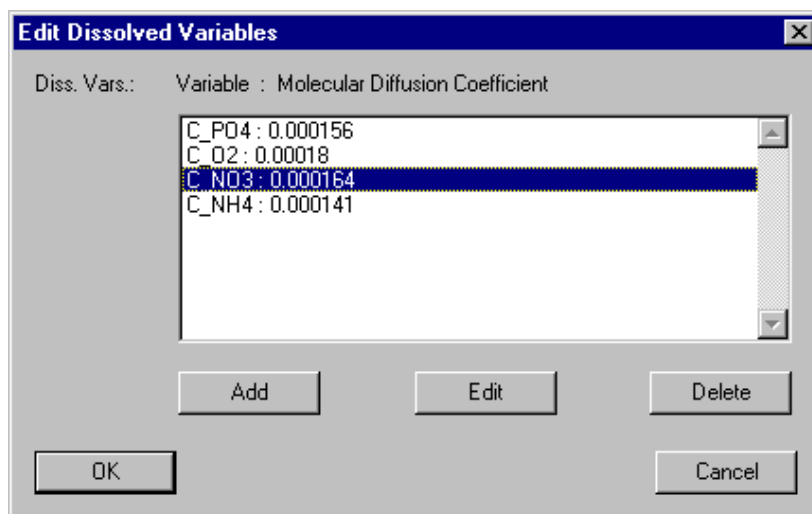


Figure 3.95: Dialog box for editing dissolved variables of a lake compartment.

molecular diffusion coefficients of all defined dissolved variables. Note that a variable can only be once in both lists of particulate and dissolved variables. The buttons ‘**Add**’, ‘**Edit**’ and ‘**Delete**’ allow the user to edit this list. Figure 3.96 shows the dialog box used to specify the properties of a dissolved variable in a lake. The edit fields ‘**Variable**’ and

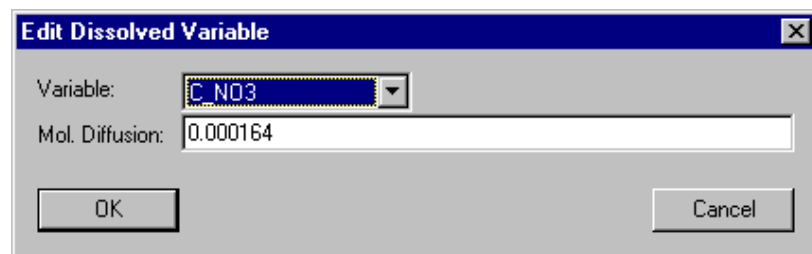


Figure 3.96: Dialog box for editing a single dissolved variable of a lake compartment.

‘**Mol. Diffusion**’ allow the user to select a variable and to specify a molecular diffusion coefficient,  $D$ . This diffusion coefficient determines the exchange of dissolved substances between the sediment layers and between the top sediment layer and the water column of the lake. Note that properties of dissolved variables can be specified for any type of variables, but that only properties of dynamic volume state variables have an effect. The larger class of variables is allowed in order to facilitate the users to switch between models with different state variables.

The button '**Sed. Prop.**' of the dialog box shown in Fig. 3.78 is used to specify the properties of the sediment submodel of the lake compartment. The definitions of this submodel are only relevant if the radio button '**with Sediment**' is selected. Fig. 3.97 shows the dialog box used for this purpose. The edit field '**Vol. Flux**' is used to specify a

Figure 3.97: Dialog box for editing the sediment submodel of a lake compartment.

volume flux of particles to the sediment. This volume flux makes it possible to consider a particle flux to the sediment of particles that are not modelled as state variables. In the edit field '**Resusp. Veloc.**' a resuspension velocity can be specified. The resuspension velocity can be used to model resuspension events. The list box '**Sediment Layers**' shows the properties of all sediment layers currently defined. The sediment layers can be edited with the aid of the buttons '**Add**', '**Edit**' and '**Delete**' below the list box. The dialog box shown in Fig. 3.98 is used to edit the properties of a single sediment layer. For

Figure 3.98: Dialog box for editing a sediment layer.

each sediment layer a '**Zone Index**', the sediment layer '**Thickness**' and the '**Porosity**' must be specified.

The button '**TKE Prop.**' of the dialog box shown in Fig. 3.78 is used to specify the properties of the submodel for the calculation of turbulent kinetic energy. The definitions

of this submodel are only relevant if the radio button ‘**with TKE**’ is selected. Fig. 3.99 shows the dialog box used for this purpose. The edit field ‘**Prandtl Number**’ is used to

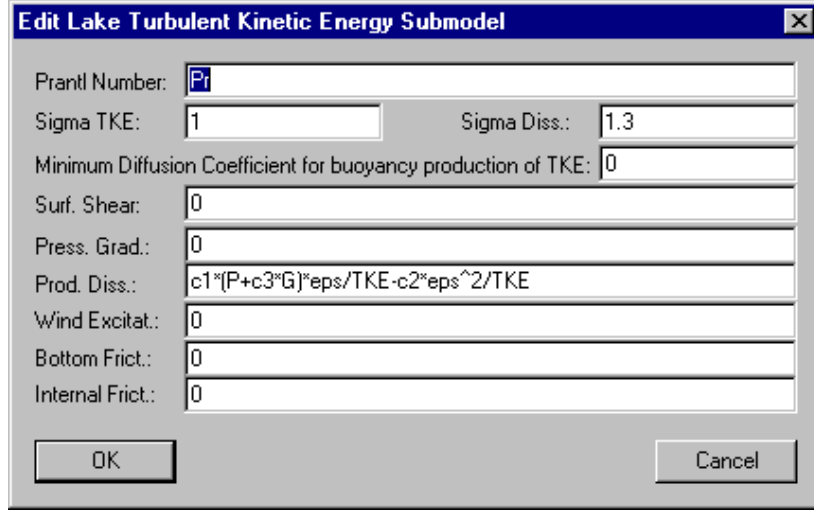


Figure 3.99: Dialog box for editing the turbulent kinetic energy submodel of a lake compartment.

define the value of the Prandtl number,  $Pr$ .

The numbers ‘**Sigma TKE**’ and ‘**Sigma Diss.**’ are used to define the values of the modification parameters  $\sigma_k$  and  $\sigma_\epsilon$  for diffusion of turbulent kinetic energy and dissipation, respectively.

The edit field ‘**Minimum Diffusion Coefficient for buoyancy production of TKE**’ is used to specify the value of  $K_{z,min}$  used in equation (3.79).

The edit field ‘**Surf. Shear**’ is used to specify the shear at the lake surface,  $\tau_{surf}$ . The surface shear is used to drive the horizontal motion described by the velocity  $U$ . It appears in the boundary condition (3.94b). The surface shear can be estimated as follows (Amorocho and de Vries, 1980):

$$\tau_{surf} = \rho_{air} c_z U_{w,10m}^2 \quad (3.104)$$

with

$$c_z \approx \begin{cases} 0.001 & \text{for } U_{w,10m} \leq 7 \text{ m/s} \\ 0.001 + 0.00015 \text{ s/m } (U_{w,10m} - 7 \text{ m/s}) & \text{for } 7 \text{ m/s} < U_{w,10m} \leq 17 \text{ m/s} \\ 0.0025 & \text{for } 17 \text{ m/s} < U_{w,10m} \leq 20 \text{ m/s} \end{cases} \quad (3.105)$$

The edit field ‘**Press. Grad.**’ is used to specify the source term  $r_U$  for the horizontal velocity used in equations (3.77) and (3.86).

The edit field ‘**Prod. Diss.**’ is used to specify the production of dissipation of turbulent kinetic energy. The expression used in the standard  $k$ - $\epsilon$  model is

$$c_1 (P + c_3 G) \frac{\epsilon}{k} - c_2 \frac{\epsilon^2}{k} \quad (3.106)$$

The edit fields ‘**Wind Excitation**’, ‘**Bottom Frict.**’ and ‘**Internal Frict.**’ are used to specify the parameters  $P_{wind}$ ,  $P_{bottom}$  and  $p_{int}$ . Empirical formulations for these parameters are required (Goudsmit et al., 1996).

Variable	Rel. Accuracy	Abs. Accuracy
Discharge:	0.0001	0.0001
Horiz. Velocity:	0.0001	1000
Turb. Kin. E.:	0.0001	100
Dissipation:	0.0001	100
Seiche Energy:	0.001	0.001

Figure 3.100: Dialog box for editing the accuracies of program variables of a lake compartment.

The button ‘**Acc.**’ of the dialog box shown in Fig. 3.78 is used to specify the numerical accuracies of program variables. Fig. 3.100 shows the dialog box used for this purpose. It allows the user to specify relative and absolute accuracies of the variables ‘**Discharge**’,  $Q$ , ‘**Horizontal Velocity**’,  $U$ , ‘**Turbulent Kinetic Energy**’,  $k$ , ‘**Dissipation**’,  $\epsilon$ , and ‘**Seiche Energy**’,  $E_{seiche}$ , in the compartment. Good behaviour of the numerical algorithms is usually achieved if the absolute accuracy and the product of the relative accuracy times a typical value of the variable both are 4 to 6 orders of magnitude smaller than typical values of the variable.

The check box ‘**active for calculation**’ can be used to activate or inactivate the compartment from the calculations. This check box has the same functionality as the buttons **Activate** and **Inactivate** in the dialog box shown in Fig. 3.19.



In Table 3.9 the program variables available in a lake compartment are summarized (for a complete overview of all program variables see Table 3.1 on page 17).

Calculation Number	Identifier for calculations (value set in the dialog boxes shown in Figs. 4.5 and 4.17).
Time	Simulation time.
Compartment Index	Identifier for compartments (value set in the dialog box shown in Fig. 3.78).
Zone Index	Identifier for zones within compartments (returns 0 in the water column and the value specified in the dialog box 3.98 in the sediment layers).
Discharge	Volumetric flow rate.
Water Fraction	Volumetric fraction of water (returns a value of 1).
Space Coordinate Z	Depth coordinate in the compartment.
Cross Sectional Area	Area of water body perpendicular to the flow direction (returns the horizontal area of the lake).
Density	Density of the water.
Area Gradient	Gradient of the cross-sectional area as a function of depth.
Brunt Vaisälä Frequency	Stability frequency of the water column.
Horizontal Velocity	Velocity of horizontal wind-driven flow.
Turbulent Kinitic Energy (TKE)	Turbulent kinetic energy (TKE) per unit mass of water in the lake.
Shear Production of TKE	Production of TKE due to shear of horizontal velocity.
Buoyancy Production of TKE	Production or loss of TKE due to density differences.
Dissipation	Dissipation of turbulent kinetic energy.
Energy of Seiche Oscillation	Total energy stored in Seiche motion.

Table 3.9: Program variables available in the lake compartment.

## 3.4 Links

The compartments introduced in section 3.3 can be connected by links in order to model water and substance exchange between the compartments. Two types of links are distinguished:

- **Advective Links** are used to describe water flow and advective substance transport from one compartment to another.
- **Diffusive Links** model diffusive boundary layers or membranes between compartments which can be diffusively penetrated by certain substances.

Figure 3.101 shows the dialog box for editing links. This dialog box is opened with the

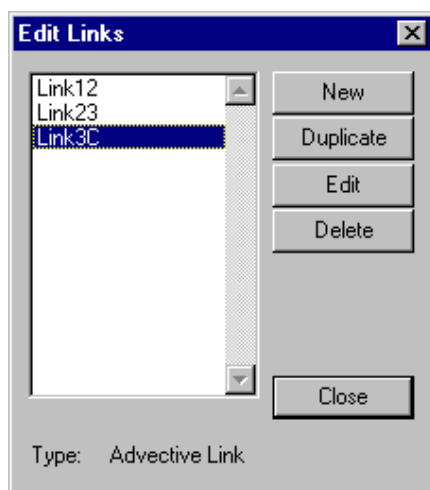


Figure 3.101: Dialog box for editing links.

**Links** command in the **Edit** menu shown in Figure 3.2. It is of modeless type in order to facilitate the editing process. The names of all links already defined are listed alphabetically in the list box of this dialog box. The type of the currently selected compartment is indicated at the bottom of this dialog box. The buttons of this dialog box allow the user to perform the following operations with links: **New** links may be created directly or old links may be **duplicated** (in both cases the new link needs a new name). The data items of a link can be **edited**. Finally, it is possible to **delete** links. The buttons **Duplicate**, **Edit** and **Delete** are inactive as long as no link is selected. Clicking the **Close** button results in closing this dialog box. It can be reopened by choosing the **Links** command in the **Edit** menu shown in Fig. 3.2.

After clicking the button **New** in the dialog box shown in Fig. 3.101 the link type can be selected in the dialog box shown in Fig. 3.102. The two types of links shown in this dialog box are described in more detail in the following two subsections.



Figure 3.102: Dialog box for selecting the type of a link.

### 3.4.1 Advective Link

#### Overview

An advective link connects an outflow connection of a compartment with an inflow connection of another compartment. An advective link can have an arbitrary number of bifurcations through which any fractions of water and substance flow can bifurcate to a further compartment. To each outflow connection of a compartment at most one advective link can be connected. To each inflow connection of a compartment an arbitrary number of outflows or bifurcations of advective links can be connected. Instead of connecting outflows or bifurcations of advective links to an inflow connection of a compartment, the connection can be left open so that the flows through such a link or bifurcation leave the modelled part of the system.

#### Equations Solved by AQUASIM

The water inflow,  $Q_{in}$ , as well as the mass flow of substances into the link,  $I_{in,i}$ , are given by the outflow from the compartment to which the inflow of the link is connected. The water and substance flows through the bifurcation  $j$ ,  $Q_{bif,j}$  and  $I_{bif,j,i}$ , can be specified by the user as functions of the water inflow,  $Q_{in}$ , and of the substance inflow concentrations.

$$C_{in,i} = \frac{I_{in,i}}{Q_{in}} \quad (3.107)$$

The index  $i$  runs over all dynamic volume state variable (which model concentrations of substances transported with the water flow). The water and substance outflows of the link are then calculated by AQUASIM according to the equations

$$Q_{ef} = Q_{in} - \sum_{j=1}^{n_{bif}} Q_{bif,j} \quad (3.108)$$

and

$$I_{ef,i} = I_{in,i} - \sum_{j=1}^{n_{bif}} I_{bif,j,i} \quad (3.109)$$

respectively.

The program user is responsible that the sum of the bifurcating water flows (which all must be nonnegative) does not exceed the water inflow to the link. In contrast, the bifurcating substance flows may be negative (modeling substance input from the bifurcation)

and the sum of all bifurcating fluxes may be larger than the input flux (this leads to a negative input flux to the compartment to which the link is connected; in some cases, this may be meaningful, e.g. to model the effect of an oxygen saturation deficit).

### User Definitions

Figure 3.103 shows the dialog box used for defining or editing an advective link. The edit

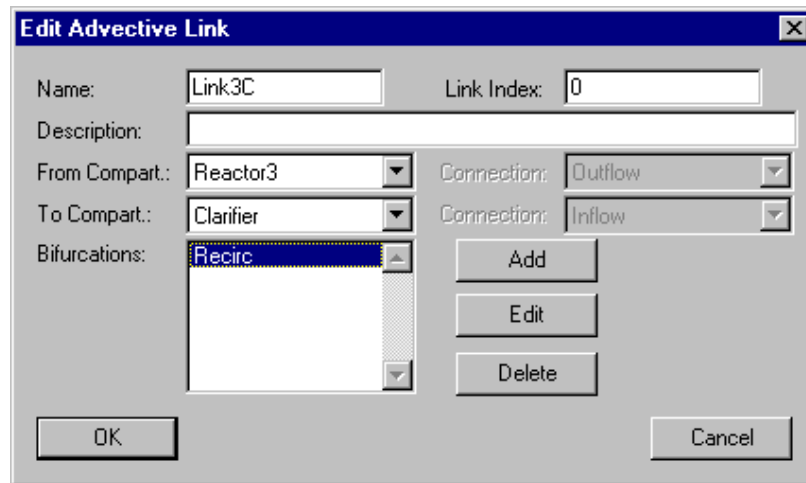


Figure 3.103: Dialog box for editing an advective link.

field '**Name**' is used to specify the name of the link. Each link needs a unique name as an identifier. A name of a link consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

The edit field '**Link Index**' can be used to specify a nonnegative number as a link index. This value can be accessed with the aid of the program variable `Link Index` to make variables dependent on the link.

To improve documentation of links, the edit field '**Description**' can optionally be used to store comments on specific implementation features of a link.

The list button '**From Compartment**' allows the user to select the compartment from which the advective fluxes feed into the link. If the selected compartment has more than one advective output connection, the list button '**Connection**' to the right of the list button **From Compartment** allows the user to select the connection. Because the outflow from the connection of the compartment selected here determines water and substance fluxes into the link, a compartment must be selected with the aid of the list button **From Compartment**.

The list buttons '**To Compartment**' and '**Connection**' (to the right of the list button **To Compartment**) allow the user to select to which input connection of which compartment the water and substance fluxes not bifurcating at the link are flowing. The field **To Compartment** can be left blank in which case these fluxes leave the modelled system.

The list box '**Bifurcations**' contains the names of all bifurcations that have been defined for the link. These bifurcations can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'.

The dialog box used for editing a bifurcation is shown in Figure 3.104.

Figure 3.104: Dialog box for editing a bifurcation of an advective link.

The edit field '**Name**' is used to specify the name of the bifurcation. Each link needs a unique name as an identifier. A name of a link consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

To improve documentation of links, the edit field '**Description**' can optionally be used to store comments on specific implementation features of a link.

The list buttons '**To Compartment**' and '**Connection**' allow the user to select to which input connection of which compartment the water and substance fluxes of the bifurcation are flowing. The field '**To Compartment**' can be left blank in which case these fluxes leave the modelled system.

The bifurcating '**Water Flow**',  $Q_{bif}$ , can be specified as an algebraic expression of globally available variables, the program variables, **Time**, **Calculation Number** and **Link Index** and of the program variable **Discharge** which evaluates here to the inflow to the link.

Flows of substances the concentrations of which are represented by dynamic volume state variables, can either be selected to divide '**with water flow**' or to be '**as given below**' in the list box of mass '**Fluxes**'. In the first case the mass fluxes of all dynamic volume state variables divide proportional to the division of water flow (as it is typical for dissolved substances), in the second case the mass fluxes of all substances transported through the bifurcation have to be specified. This second option can be used to define mass separation e.g. increase in concentrations of particulate substances at the bottom outlet of a settler.

If the mass fluxes are selected to be given, they are displayed in the list box and can be edited using the buttons '**Add**', '**Edit**' and '**Delete**'.

Figure 3.105 shows the dialog box used for defining a mass flux for a substance the

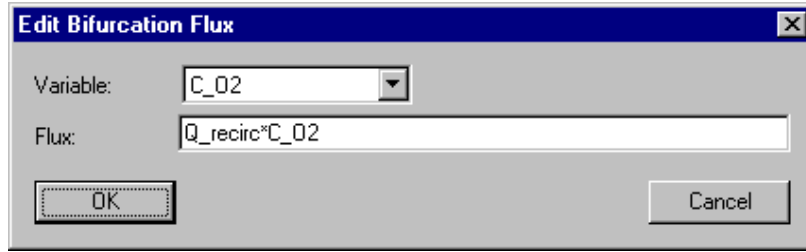


Figure 3.105: Dialog box for editing a mass flux within a bifurcation of an advective link.

concentration of which is represented by a dynamic volume state variable. The list button ‘**Variable**’ is used to select the state variable, in the edit field ‘**Flux**’ the mass flux,  $I_{bif,i}$  can be specified as an algebraic expression. In this expression, globally available variables, the program variables **Time**, **Calculation Number** and **Link Index**, the program variable **Discharge** representing the inflow to the link (not the bifurcating flow), and dynamic volume state variables representing link-inflow concentrations, can be used. Bifurcation mass fluxes can be defined for any type of variable, but only those for dynamic volume state variables have an effect.

### 3.4.2 Diffusive Link

#### Overview

A diffusive link connects two diffusive connections of compartments. It is used to model diffusive mass exchange of substances between compartments through membranes or boundary layers. There is no water flow through a diffusive link.

#### Equations Solved by AQUASIM

The mass flux of a substance  $i$  through a diffusive link from compartment 1 to compartment 2 is given by the equation

$$I_i = q_{ex,i}(f_i C_{i,1} - C_{i,2}) \quad (3.110)$$

where  $C_{i,1}$  and  $C_{i,2}$  are the concentrations of substance  $i$  in the compartment 1 and 2, respectively,  $q_{ex,i}$  is the exchange coefficient, and  $f_i$  is a conversion factor that allows for the description of phase transitions. This conversion factor is unity, if both compartments 1 and 2 contain the same solvent. If compartment 1 is a mixed reactor compartment modelling a gas phase, compartment 2 models a water phase, the conversion factor is the inverse of the non-dimensional Henry coefficient of substance  $i$ :

$$f_i = \frac{1}{H_i} \quad (3.111)$$

In many cases the mass exchange coefficient,  $q_{ex,i}$ , is given as the product of the surface area of the membrane or boundary layer,  $A$ , and a mass transfer coefficient,  $k_i$

$$q_{ex,i} = A k_i \quad (3.112)$$

which itself is often given as the ratio of the diffusion coefficient  $D_i$  of the substance in the boundary layer or membrane and the thickness  $L_M$  of the boundary layer or membrane

$$k_i = D_i L_M \quad (3.113)$$

Instead of these expressions, an empirical exchange coefficient can be used.

### User Definitions

Figure 3.106 shows the dialog box used for defining or editing a diffusive link.

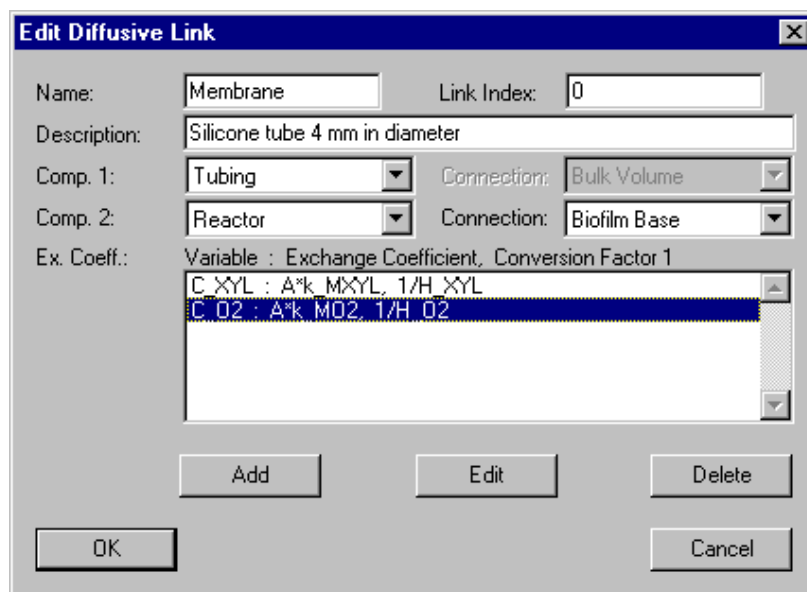


Figure 3.106: Dialog box for editing a diffusive link.

The edit field ‘**Name**’ is used to specify the name of the link. Each link needs a unique name as an identifier. A name of a link consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

The edit field ‘**Link Index**’ can be used to specify a nonnegative number as a link index. This value can be accessed with the aid of the program variable **Link Index** to make variables dependent on the link.

To improve documentation of links, the edit field ‘**Description**’ can optionally be used to store comments on specific implementation features of a link.

The list buttons ‘**Compartment 1**’ and ‘**Compartment 2**’ allow the user to select the two compartments to be linked diffusively. If a selected compartment has more than one diffusive connection, the list button ‘**Connection**’ can be used to select one of these. An arbitrary number of diffusive links can be connected to the same diffusive connection of a compartment. If both compartments contain the same solvent, it is irrelevant which compartment is connected as compartment 1 and which as compartment 2. This is only relevant if the conversion factor of a substance is not unity. In this case it has to be noted that the concentration in compartment 1 is multiplied with the conversion factor in equation (3.110).

The list box in the dialog box shown in Fig. 3.106 shows the list of exchange coefficients,  $q_{ex,i}$ , and conversion factors,  $f_i$ . This list can be edited using the buttons ‘Add’, ‘Edit’ and ‘Delete’.

Figure 3.107 shows the dialog box used for defining the exchange coefficient and the

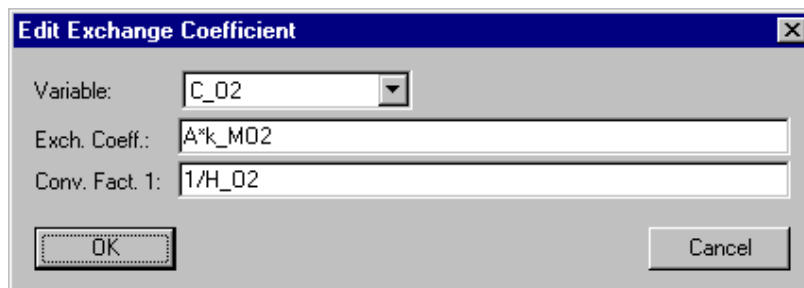


Figure 3.107: Dialog box for editing an exchange coefficient of a diffusive link.

conversion factor of a substance. The list button ‘**Variable**’ is used to select the state variable, in the edit field ‘**Exchange Coefficient**’, the exchange coefficient,  $q_{ex,i}$ , can be specified and in the edit field ‘**Conversion Factor 1**’, the value of the conversion factor,  $f_i$ , for the substance concentration in compartment 1 can be given. The program variables **Time**, **Calculation Value** and **Link Index** can be used for these definitions in addition to all globally available variables. Exchange coefficients and conversion factors can be defined for any type of variables, but only exchange coefficients for dynamic volume state variables have an effect.



### 3.5 Numerical Parameters

In order to integrate the differential equations of the user specified model in AQUASIM, as a first step, the partial differential equations are discretized in space. Then, the spatially discretized partial differential equations together with the ordinary differential equations and the algebraic equations are integrated numerically in time with the algorithm DASSL (Petzold, 1983) which is based on the implicit (backward differencing) variable-step, variable-order Gear integration technique (Gear, 1971b; Gear, 1971a; Gear, 1971c).

In the dialog box **Edit Numerical Parameters** shown in Fig. 3.108, which is opened with the **Numerical Parameters** command in the **Edit** menu, the general numerical parameters of the time integration algorithm DASSL (Petzold, 1983) can be edited.

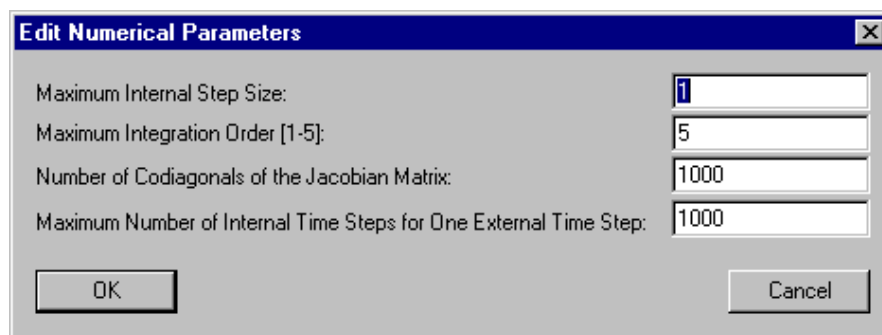


Figure 3.108: Dialog box for editing numerical parameters.

The first parameter, '**Maximum Internal Step Size**', makes it possible to bound the step size used internally by the numerical integration algorithm. The output step size specified in the dialog box **Edit Calculation Definition** shown in Fig. 4.5 only defines the points of time at which the model results are stored in memory to be available for plotting and postprocessing of results. The internal step size used by the integration algorithm is chosen dynamically in order to optimise the integration efficiency while maintaining the requested integration accuracy (see below for how to specify the integration accuracy of state and program variables). The maximum size of these internal time steps is ten times the output step size. In addition, the step size is bounded by the number specified in the field **Maximum Internal Step Size** of the dialog box shown in Fig. 3.108. A too small value of this numerical parameter makes the integration inefficient (slow) because too many steps must be performed by the algorithm. Under most circumstances, a very large value has no negative effect (because the step actually used is then bounded by the accuracy requirement for the solution). There is one very important exception from this general rule: If a system has a very smooth temporal behaviour that is interrupted by very short excitations (inputs or process rates), then, due to the large time step selected during the smooth period, the integration algorithm may step over some of the excitations. If one integration step leads to an evaluation of the input or process rate during an excitation period, the algorithm recognises the problem and repeats the step by a sequence of smaller steps in order to resolve the time course of the excitation accurately. However, if no step leads to an evaluation of the input or process rate during the excitation period, the algorithm steps over the excitation and fails to model the system correctly. In such

situations, the numerical parameter **Maximum Internal Step Size** can be used to bound the internal step size to a value (slightly) smaller than the duration of the excitation period in order to avoid this problem.

The second parameter, '**Maximum Integration Order**', in the dialog box shown in Fig. 3.108 makes it possible to bound the order of the temporal discretization scheme. By default the integration order is chosen dynamically between one and five in order to optimise the integration efficiency while maintaining the requested integration accuracy (see below for how to specify the integration accuracy of state and program variables). A smaller maximum integration order leads to smaller time steps required to maintain the integration accuracy. For this reason it is usually advisable to leave the value of the numerical parameter **Maximum Integration Order** at its default value of five.

The third parameter, '**Number of Codiagonals of the Jacobian Matrix**', can be used to increase the calculation efficiency for systems with a linear geometry. The implicit integration algorithm applied by DASSL (Petzold, 1983) requires the evaluation of the matrix of partial derivatives of the components of the right hand side of the system of ordinary differential equations (obtained after spatial discretization) with respect to the components of the state vector of the system, the so-called Jacobian matrix. The state vector of the system consists of the sequence of all active state variables and those program variables that are calculated by time integration or as the solution of algebraic equations at all grid points. For systems with a linear geometry (e.g. a river), the time derivative of a component of the state vector depends only on the components representing other state or program variables at the same or at the neighbouring grid points. This leads to a Jacobian matrix in which all elements that are not zero are close to the main diagonal of the matrix. The limitation of the number of codiagonals to twice the number of state or program variables at one grid point then increases the efficiency of the algorithm considerably because all elements further away from the main diagonal (which are zero) are not evaluated by the program. It is evident that such a limitation to a banded Jacobian by selecting a small number in the edit field **Number of Codiagonals of the Jacobian Matrix** leads to problems in the case of recirculation (because then the time derivative at the first grid point depends on the values in the last grid point, what leads to nonzero elements of the Jacobian matrix far away from its main diagonal). If the **Number of Codiagonals of the Jacobian Matrix** is given a too small value, the algorithm will fail to converge and the integration is stopped with an error message. A value at least equal to the number of equations minus one leads to the use of the full Jacobian matrix. It is not dangerous to try to find a good value of this numerical parameter because if convergence is obtained, convergence is always to the correct solution even if the Jacobian matrix is not correct (this is a very useful property of the Gear algorithm which makes it possible to use the same Jacobian matrix for several time steps and update it only if convergence becomes slow). The recommended value for this parameter is twice the number of state and program variables integrated in the compartment for systems with a linear geometry, and a value larger than the number of equations for a system with recirculations (this leads to the use of the full Jacobian matrix).

The fourth parameter, '**Maximum Number of Internal Time Steps for One External Time Step**', is useful for the detection of integration problems. In many cases, the fact that a huge number of steps is required for one output time step, is an indication of an integration problem. For this reason, this number can be limited in the edit field **Maximum Number of Internal Time Steps for One External Time Step**. If a solution is calculated with a very large output time step for relaxation followed by a

period with higher numerical resolution, this numerical parameter can lead to unwanted interrupts of the integration during the initialization phase. In this case, the value of this parameter can be increased to avoid this behaviour. In most other cases, however, an interruption of a simulation caused by this parameter is a strong indication for numerical integration problems.

In addition to the general numerical parameters of the time integration algorithm shown in Fig. 3.108, the required integration accuracies of the state variables and of those program variables that are integrated by AQUASIM, are additional important numerical parameters as well. The integration accuracy of the state variables can be specified in the edit fields **Rel. Accuracy** and **Abs. Accuracy** of the dialog box shown in figure 3.5. In the dialog boxes used for the definition of the compartments (see Figs. 3.21, 3.29, 3.41, 3.53, 3.67 and 3.78), the button **Acc.** allows the program users to specify the integration accuracy of program variables which are integrated in the current compartment (see Figs. 3.28, 3.40, 3.51, 3.66, 3.77 and 3.100).

As a last point, in the dialog boxes used for editing compartments (see Figs. 3.29, 3.41, 3.53, 3.67 and 3.78) the **Number of Grid Points**) and the **Resolution** of the algorithm can be selected by the user. **Low Resolution** corresponds to a first-order discretization in space, **High Resolution** corresponds to a discretization of at least second order combined with a flux limiter that is used to avoid numerical oscillations.

## 3.6 Deleting Calculated States

Calculations, as described in chapter 4, lead to the storage of calculated states of the user-defined system. In this section it is described, how the user can check which states are currently stored in memory and how selected states can be deleted.

In the dialog box **Delete Calculated States** shown in Fig. 3.109, which is used to check which states are available and to delete selected states, is opened with the **Delete States** command in the **Edit** menu. Selective deletion of calculated states may be advan-

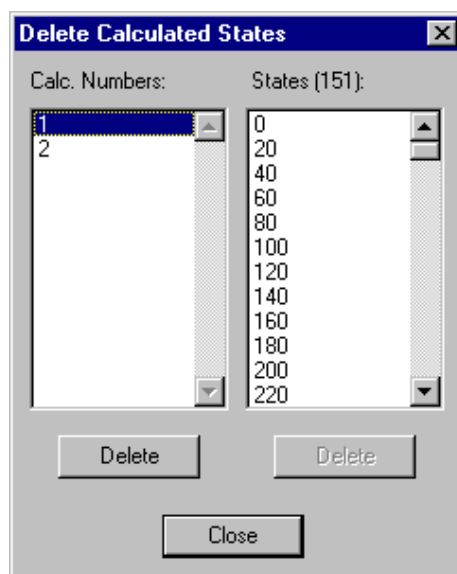


Figure 3.109: Dialog box for deleting calculated states.

tageous to save memory during program operation or disk space if the calculated states are saved together with the system definitions (if no calculated states are to be saved, it is not necessary to delete them before saving, because the user is asked during the saving process if the calculated states are to be saved in addition to the system definitions). As an example, the user can delete all states with the exception of the last one in order to be able to continue the simulation from a relatively small saved file (calculated states may enlarge the file size considerably).

The left list box of the dialog box shown in Fig. 3.109 shows the calculation numbers for which calculated states are currently available. The right list box of this dialog box shows the states that are available for the calculation number selected in the left list box (the right list box is empty if in the left list box no or more than one calculation numbers are selected). The button '**Delete**' below the left list box makes it possible to delete all calculated states for all selected calculation numbers, the button '**Delete**' below the right list box can be used to delete the selected calculated states for the calculation number selected in the left list box.

Note that initializing a simulation for a given calculation number also leads to the deletion of all calculated states for this calculation number and that editing operations which lead to a change in the number of differential equations to be solved (e.g. addition/deletion of a compartment, activation/inactivation of state variables, changes of the numbers of

grid points of compartments etc.) also leads to the deletion of all calculated states for all calculation numbers.

## Chapter 4

# Simulation and Data Analysis

Fig. 4.1 shows the menu `Calc` of AQUASIM. The item ‘**Simulation**’ of this menu is

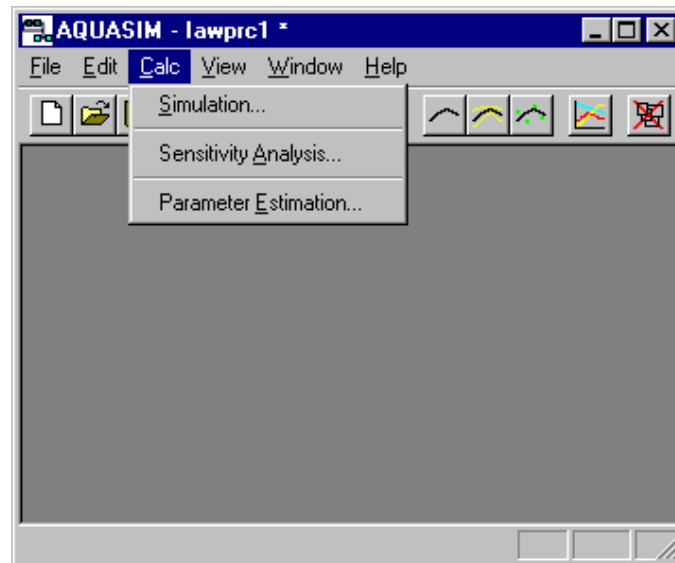


Figure 4.1: `Calc` menu.

used to define and execute simulations for the user-defined model, the item ‘**Sensitivity Analysis**’ is used to define and execute sensitivity analyses, and the item ‘**Parameter Estimation**’ is used to define parameter estimation procedures and to estimate model parameters. These three items of the menu `Calc` are described in detail in the three sections of this chapter.

During all calculation operations, information on the system of equations currently solved, on the progress of the integration, sensitivity analysis or parameter estimation algorithm, and on numerical problems is written to the log file `aquasim.log` in the program directory of AQUASIM. In the case of numerical problems this information may be very useful in order to help to locate the source of the error. Some help for the interpretation of the information contained on this file is given in section 6.3. The file is overwritten every time AQUASIM is started.

## 4.1 Simulation

Execution of a simulation is equivalent to numerically integrating a system of ordinary and partial differential equations in time and simultaneously solving the algebraic equations. In AQUASIM, as a first step, the partial differential equations are discretized in space. Then, the spatially discretized partial differential equations together with the ordinary differential equations and the algebraic equations are integrated numerically in time with the algorithm DASSL (Petzold, 1983) which is based on the implicit (backward differencing) variable-step, variable-order Gear integration technique (Gear, 1971b; Gear, 1971a; Gear, 1971c).

Spatial discretization of partial differential equations is done using conservative finite difference schemes (LeVeque, 1990). The differential conservation laws 3.6, 3.35, 3.44, 3.56, and 3.73

$$\frac{\partial \hat{\rho}}{\partial t} = -\frac{\partial \hat{j}}{\partial z} + \hat{r} \quad (4.1)$$

are discretized as

$$\frac{d}{dt} \hat{\rho}(x_i, t) = -\frac{\hat{j}_{num}(x_{i+1/2}, t) - \hat{j}_{num}(x_{i-1/2}, t)}{x_{i+1/2} - x_{i-1/2}} + \hat{r}(x_i, t) \quad (4.2)$$

using the division of the spatial coordinate axis in grid points as shown in Fig. 4.2. The user

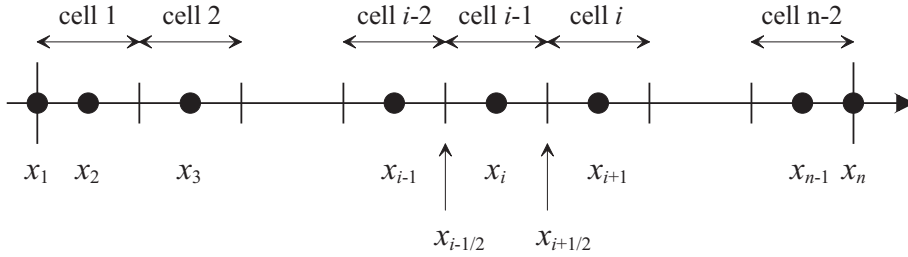


Figure 4.2: Division of the spatial coordinate axis in cells and grid points for spatial discretization of partial differential equations (note that the number of cells is equal to the number of grid points minus 2 and that the grid spacing is smaller at the boundaries;  $n$  is the number of grid points specified in the dialog boxes shown in Figs. 3.29, 3.41, 3.53, 3.67 and 3.78).

has the choice between a low resolution (first-order) and a high resolution (second-order with flux limiter) numerical flux  $\hat{j}_{num}$ . For the high resolution flux, the van Leer flux limiter is applied (van Leer, 1974; Sweby, 1984; LeVeque, 1990). Because time integration uses a high order discretization, numerical diffusion for the first-order technique can be estimated to be

$$D_{num} \approx \frac{v \Delta x}{2} \quad (4.3)$$

where  $v$  is the advective transport velocity and

$$\Delta x = \frac{x_e - x_s}{n - 2} \quad (4.4)$$

is the grid spacing ( $x_e - x_s$  is the length of the compartment and  $n$  the number of grid points).

The main advantage of the Gear integration technique for the solution of systems of ordinary differential equations (Gear, 1971b; Gear, 1971a; Gear, 1971c; Hindmarsh, 1983) is its property of being stiffly stable. The implementation DASSL (Petzold, 1983) of this technique, which is used in AQUASIM, has the advantage that not only systems of ordinary differential equations, but differential-algebraic systems can be solved. Stiff systems of differential equations are systems for which the transients towards the solution to be calculated approach this solution on a much shorter time scale as the time scale for variations of the solution of interest. The main idea of achieving a stiffly stable integration technique is using backward differencing schemes. The advantage of backward versus forward differencing can easily be illustrated with a comparison of the explicit and the implicit Euler algorithm. If the system of ordinary differential equations is written in the form

$$\frac{d\mathbf{y}}{dt} = \mathbf{F}(\mathbf{y}, t) \quad (4.5)$$

the explicit Euler technique is given as

$$\mathbf{y}(t_{j+1}) = \mathbf{y}(t_j) + (t_{j+1} - t_j)\mathbf{F}(\mathbf{y}(t_j), t_j) \quad (4.6)$$

and the implicit Euler technique as

$$\mathbf{y}(t_{j+1}) = \mathbf{y}(t_j) + (t_{j+1} - t_j)\mathbf{F}(\mathbf{y}(t_{j+1}), t_{j+1}) \quad (4.7)$$

Figure 4.3 shows the numerical solution of a stiff system of differential equations with the explicit technique (4.6) (left) and with the implicit technique (4.7) (right). Because the

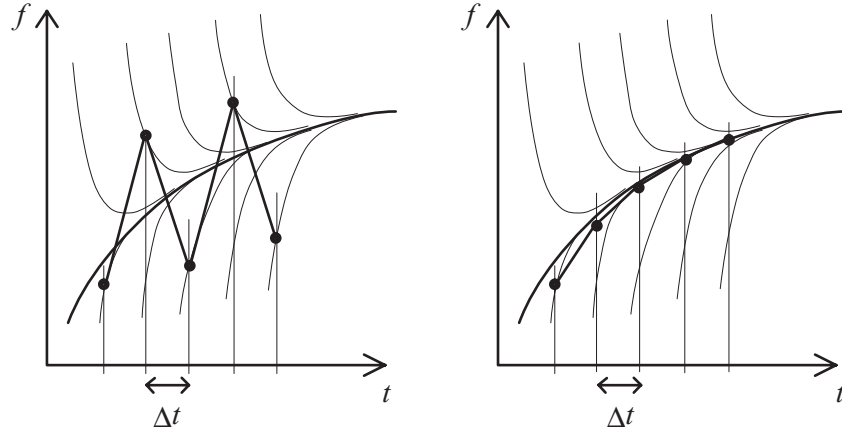


Figure 4.3: Numerical solution of a stiff system of differential equations using the explicit (left) and the implicit (right) Euler algorithm with the same size of the time step.

numerical solution with the implicit technique is tangent to the true solution at the end of each time step and not at the beginning, the technique can use much larger time steps. For stiff systems of differential equations this advantage leads to a faster integration although a implicit system of nonlinear equations must be solved at each time step (the explicit



technique requires only the evaluation of explicit functions at each time step). Whereas Fig. 4.3 demonstrates the main idea of achieving a stiffly stable integration algorithm by backward differencing, the Gear algorithm (Gear, 1971b; Gear, 1971a; Gear, 1971c) uses a more sophisticated generalization of this idea to a multi-step, variable step size and variable order (originally up to 6, in the implementation DASSL up to 5) algorithm.

The first disadvantage of the Gear integration technique and the implementation DASSL (Petzold, 1983) is its inability to step over discontinuities of inputs or process rates. Such step discontinuities can be approximated by a linear increase or decrease of the quantity of interest over a very short time interval (compared to the typical time scale of the problem of interest). With this approximation, the algorithm still has to decrease the time step considerably at the points where the time derivative is discontinuous (the integration order is also reduced to 1). As an alternative, a smoothed step function with continuous derivatives can be used, e.g. a step approximated by third order spline polynomials for which, in addition to the value of the function, also the first and second derivatives are continuous. A second disadvantage of the Gear algorithm is that it requires the evaluation of the jacobian matrix

$$\underline{\mathbf{J}} = \frac{\partial \mathbf{F}}{\partial \mathbf{y}} \quad (4.8)$$

for solving the nonlinear system of algebraic equations at each time step. The implementation DASSL (Petzold, 1983) allows only the choice between using the full or a banded jacobian matrix. For linearly arranged systems of compartments (and, in AQUASIM, if the alphabetic order of compartments corresponds to the sequence of flow) the option of a banded jacobian matrix, with a number of codiagonals (cf. section 3.5) equals to twice the number of equations solved at one grid point, leads to a very efficient solution procedure. However, in the case of branched systems or systems with recirculations, due to (typically only a few) nonzero elements of the jacobian matrix far from its main diagonal, the option of a full jacobian matrix must be used. The evaluation of the jacobian matrix then consumes much computation time and can make the integration inefficient.

Figure 4.4 shows the dialog box used for defining and starting simulations. This dialog box is opened with the **Simulation** command in the **Calc** menu shown in Figure 4.1. It is of modeless type in order to facilitate performing simulations. A simulation consists of one or more calculations. The names of all active calculations are listed in the left list box of the dialog box shown in Fig. 4.4, those of all available calculations in the right list box. The buttons between these two list boxes allow the user to perform the following operations: By clicking the button '**New**', new calculations can be defined. Alternatively, by clicking the button '**Duplicate**', the selected calculation can be duplicated. With the button '**Edit**', or by double-clicking the calculation name in the list box, a calculation can be edited. The button '**Delete**' allows the program users to delete calculations. The buttons **Duplicate**, **Edit** and **Delete** are inactive as long as no calculation is selected. With the button '**Activate**', calculations selected in the right list box can be activated, with the button '**Inactivate**', calculations selected in the left list box can be inactivated. Active calculations must have different values of their calculation numbers. This means that activation of a calculation is impossible, if there is already an active calculation with the same calculation number.

Execution of a simulation leads to the storage of calculated states of the user-defined model. Clicking the button '**Initialize**' deletes the calculated states for the calculation

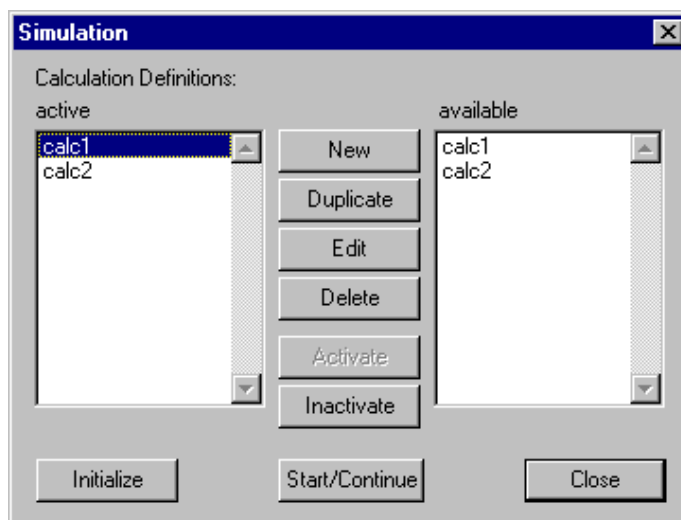


Figure 4.4: Dialog box for editing calculation definitions and executing simulations.

numbers of all active calculations and generates newly calculated initial states. Clicking this button is only necessary before a simulation, for which there exist already calculated states, is to be redone. The button **'Start/Continue'** is used to continue the active calculations from the last calculated states (if there are no calculated states, the simulation is automatically initialized before the dynamic simulation is started). The buttons **Initialize** and **Start/Continue** are inactive as long as there is no active calculation. Clicking the **Close** button results in closing this dialog box. It can be reopened by choosing the **Simulation** command in the **Calc** menu shown in Fig. 4.1.

As mentioned above, the buttons **New**, **Edit** and **Duplicate** of the dialog box shown in Fig. 4.4 are used to define calculations for the user-defined model. Figure 4.5 shows the dialog box used for defining and editing a single calculation. The edit field **'Name'** is used to specify the name of the calculation. Each calculation needs a unique name as an identifier. A name of a calculation consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

To improve documentation of calculations the edit field **'Description'** can optionally be used to store comments on the purpose of the calculation definition.

The edit field **'Calc. Number'** allows the program user to identify a calculation by a nonnegative integer number. Variables (and thus also processes, etc.) can be made dependent on this calculation number with the aid of the program variable **Calculation Number**. Active calculations must have different values of their calculation numbers.

Each calculation must be initialized before it can be started. In the edit field **'Initial Time'** the time at which a calculation is initialized can be specified.

The user has the choice between two types of the **'Initial State'** to be used by the program. The default option **'given, made consistent'** applies the initial state as given by the user (a value of zero is the default initial condition for all state variables for which the user did not explicitly specify an initial condition). If this initial state violates constraints on boundary conditions or algebraic equations of equilibrium processes, the program tries to fulfill these constraints by small modifications to the initial state given by

**Edit Calculation Definition**

Name:

Description:

Calc. Number:

Initial Time:

Initial State: ☒ given, made consistent ☐ steady state

Output Steps: Step Size:  Num. of Steps:

Status: ☒ active for simulation ☒ active for sensitivity analysis

Figure 4.5: Dialog box for editing a calculation definition.

the user. This procedure is called to make the initial state consistent. With the selection of the alternative option '**steady state**' the program tries to find the steady state solution of the user-defined model (under external parameter values evaluated at the initial time) and uses this steady state solution as the initial condition. Note, however, that not all models have a steady state solution and that even if such a solution exists, the numeric algorithm used by AQUASIM may fail to find it. In such a situation, the convergence can be improved by a good choice of initial conditions which are used as starting values for the iterative search process for the steady state solution. If this does not help the steady state solution must be found by relaxation, i.e. by executing a dynamic simulation with constant boundary conditions.

In the list box '**Output Steps**' the user can define sequences of time steps that are executed consecutively. To do this, the '**Step Size**' and '**Number of Steps**' must be entered in the edit fields below the list box and then one of the buttons **Add** or **Replace** must be clicked. If no row of the list box is selected, '**Add**' leads to insertion of the new step sequence at the end of the list, otherwise the new step sequence is inserted before the selected row. '**Replace**' leads to the replacement of the selected row by the step sequence specified in the edit fields below the list box. Clicking the button '**Delete**' leads to the deletion of the selected rows of the list. Note that the output steps defined for a calculation specify the points of time at which the calculated states are stored in memory and are available later on for plotting results and for exporting results to other programs. The internal step size used by the integration algorithm, however, is determined according to the requirement of fulfilling the accuracy requirements of the state and program variables as described in section 3.5. This internal step size is usually much smaller than the output step size, however, it can increase to ten times the output step size if all variables

change only smoothly in time (the solution at the output steps is then determined by interpolation). The size of the internally used time step can be bounded as described in section 3.5 (this may be useful in order to avoid that the algorithm steps over short excitations).

Finally, the check boxes ‘**active for simulation**’ and ‘**active for sensitivity analysis**’ are used to activate the calculation for the specified task.

During the execution of a simulation, the dialog box shown in Fig. 4.6 shows the

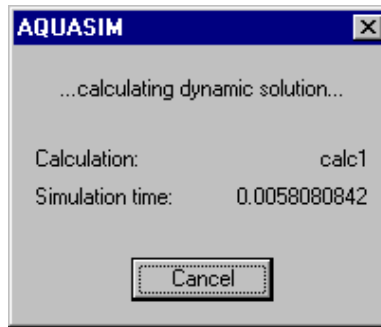


Figure 4.6: Dialog box for interrupting a simulation.

current calculation and the current simulation time and it allows the user to interrupt the simulation.

## 4.2 Sensitivity Analysis

As mentioned in the introduction, sensitivity analysis combines the tasks of identifiability analysis and uncertainty analysis.

The goal of **identifiability analysis** is to check if model parameters can be uniquely determined with the aid of the available data and to estimate the uncertainty of the parameter estimates. This can be done by estimating the standard errors and correlation coefficients of parameters during the parameter estimation procedure as described in section 4.3. An additional source of information on the identifiability of model parameters are the sensitivity functions discussed in this section. The following four sensitivity functions are distinguished by AQUASIM:

$$\boxed{\delta_{y,p}^{a,a} = \frac{\partial y}{\partial p}} \quad (4.9a)$$

$$\boxed{\delta_{y,p}^{r,a} = \frac{1}{y} \frac{\partial y}{\partial p}} \quad (4.9b)$$

$$\boxed{\delta_{y,p}^{a,r} = p \frac{\partial y}{\partial p}} \quad (4.9c)$$

$$\boxed{\delta_{y,p}^{r,r} = \frac{p}{y} \frac{\partial y}{\partial p}} \quad (4.9d)$$

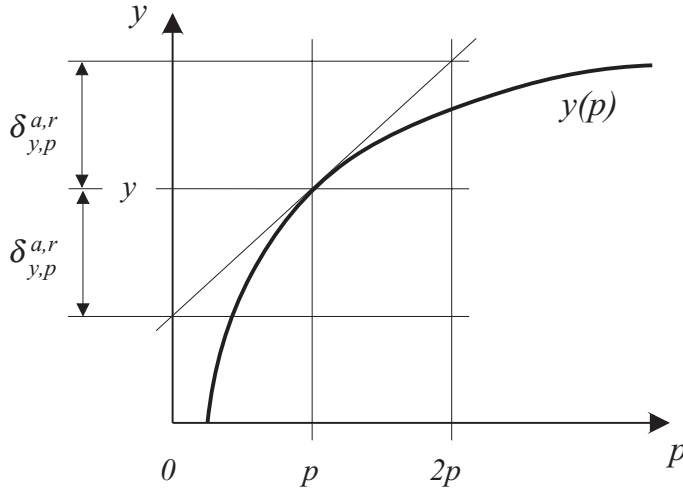


Figure 4.7: Interpretation of the absolute-relative sensitivity function  $\delta_{y,p}^{a,r} = p \partial y / \partial p$ .

In these functions,  $y$  is an arbitrary variable calculated by AQUASIM and  $p$  is a model parameter represented by a constant variable or by a real list variable. The absolute-absolute sensitivity function (4.9a) measures the absolute change in  $y$  per unit of change

in  $p$ , the relative-absolute sensitivity function (4.9b) measures the relative change in  $y$  per unit of change in  $p$ , the absolute-relative sensitivity function (4.9c) measures the absolute change in  $y$  for a 100 % change in  $p$ , and the relative-relative sensitivity function (4.9d) measures the relative change in  $y$  for a 100 % change in  $p$ . All these changes are calculated in linear approximation only. The most useful sensitivity functions are the absolute-relative sensitivity function (4.9c) and the relative-relative sensitivity function (4.9d), because their units do not depend on the unit of the parameter. This makes quantitative comparisons of the effect of different parameters  $p$  on a common variable  $y$  possible. Because the relative-relative sensitivity function (4.9d) is non-dimensional, this sensitivity function can not only be used to compare the effect of different parameters on a common variable, but also the effects of different parameters on different variables. However, the disadvantage of this relative-relative comparison is that it gives not very useful results if the value of the variable  $y$  becomes small during a simulation (because large relative changes of a value that is close to zero are not relevant). Fig. 4.7 illustrates the meaning of the absolute-relative sensitivity function (4.9c). The linear approximation to the change in  $y$  for a 100 % change in  $p$  can easily be constructed with the aid of the tangent to the function  $y(p)$  in  $p$ . The larger the values of sensitivity functions and the more pronounced the differences in shape of the sensitivity functions (within the range of available data), the more accurately are the parameters identifiable.

Figure 4.8 shows an example of the time dependence of the sensitivity functions of

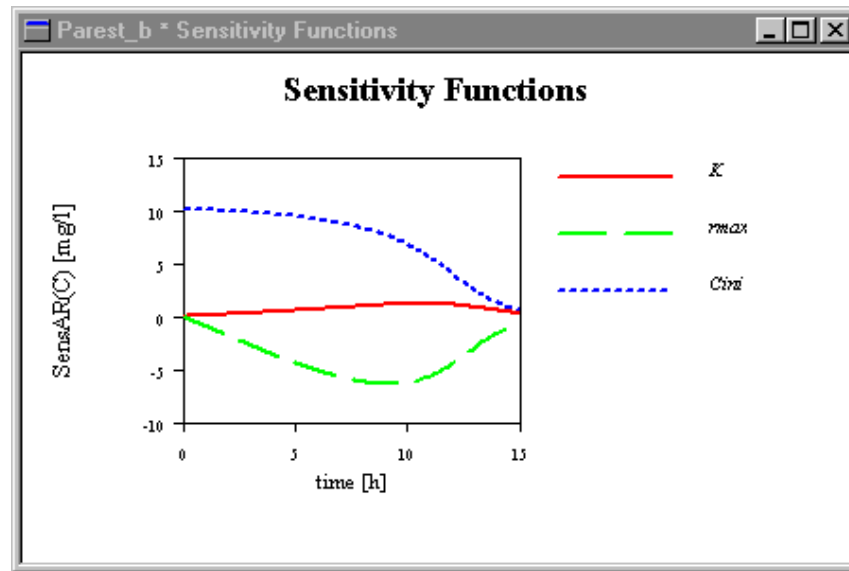


Figure 4.8: Example of a plot of sensitivity functions of a calculated concentration with respect to three model parameters.

a calculated concentration with respect to three model parameters  $K$ ,  $r_{max}$  and  $C_{ini}$  (look at the tutorial for a complete description of this example). It is evident that the parameter  $C_{ini}$  is identifiable from measured concentrations, because for small values of the time,  $t$ , the calculated concentration is only sensitive to this parameter. The sensitivity functions of the parameters  $K$  and  $r_{max}$  have a similar shape (the signs indicate that the calculated concentrations increase with increasing values of  $K$ , but they decrease

with increasing values of  $r_{max}$ ). This leads to a correlation between the estimates of these parameters (changes in calculated concentrations caused by a change in  $K$  can approximately be compensated by an appropriate change in  $r_{max}$ ). Furthermore, the much smaller sensitivity of the calculated concentration to the parameter  $K$  in comparison to the sensitivity to the parameter  $r_{max}$  leads to a larger uncertainty of the estimate of  $K$  than that of  $r_{max}$ .

In **uncertainty analysis**, the uncertainty of model parameters is propagated to the uncertainty of model results (model structure uncertainty is not addressed as an automated task of AQUASIM, however, predictions with different model structures can be compared manually). In the current version of AQUASIM only the simplest error propagation method is implemented, the linearized propagation of standard deviations of uncorrelated parameters. The error propagation formula using the linearized model and neglecting the parameter correlation is given by

$$\sigma_y = \sqrt{\sum_{i=1}^m \left( \frac{\partial y}{\partial p_i} \right)^2 \sigma_{p_i}^2} \quad (4.10)$$

where  $p_i$  are the uncertain model parameters,  $\sigma_{p_i}$  are their standard deviations,  $y(p_1, \dots, p_m)$  is the solution of the model equations for a given variable at a given location and time, and  $\sigma_y$  is the approximate standard deviation of the model result. Figure 4.9 shows a

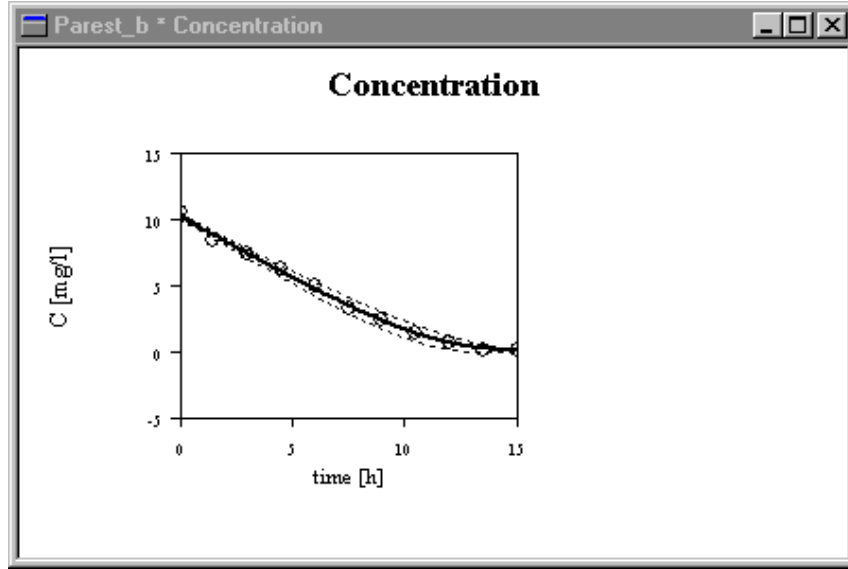


Figure 4.9: Example of a plot of simulation results with estimated error bounds.

plot of measured data and the simulation result with error bounds limiting the range of values of the result plus and minus one standard deviation. The error contribution of each parameter is given as

$$\delta_{y,p}^{err} = \frac{\partial y}{\partial p} \sigma_p \quad (4.11)$$

The error contributions of the three parameters of the example shown in the Figs. 4.8 and 4.9 are shown in Fig. 4.10. It is interesting that the magnitudes of the error contributions

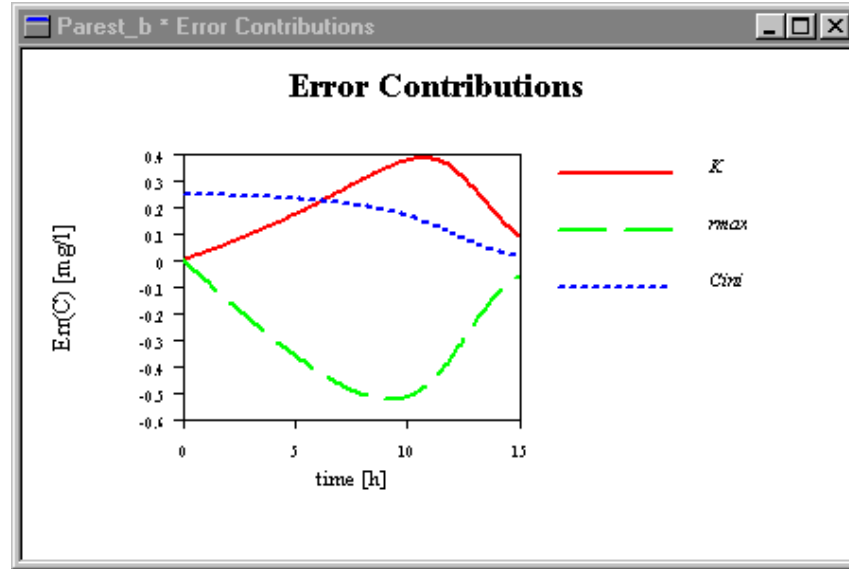


Figure 4.10: Example of a plot of error contributions of three model parameters on a calculated concentration.

due to the parameters  $K$  and  $r_{max}$  are of a similar size despite the difference in sensitivity shown in Fig. 4.8. The reason for this effect is the higher uncertainty in the estimate of  $K$  resulting from the smaller sensitivity.

The derivatives required for calculating the sensitivity functions (4.9a) to (4.9d), the standard deviations of calculated variables according to equation (4.10), and the contributions of parameter uncertainties to the total uncertainty according to equation (4.11) are calculated using the finite difference approximation

$$\frac{\partial y}{\partial p_i} \approx \frac{y(p_i + \Delta p_i) - y(p_i)}{\Delta p_i} \quad (4.12)$$

where  $\Delta p_i$  is chosen to be 1 % of the standard deviation  $\sigma_{p_i}$  of the parameter  $p_i$ . In order to make the program as flexible as possible, the solutions for the basic parameter values ( $p_1, \dots, p_m$ ) and for all sets of parameter values, where one of the  $p_i$ 's is replaced by  $p_i + \Delta p_i$  are calculated one after the other and stored simultaneously. Using these  $m + 1$  stored states, all the expressions (4.9a) to (4.9d), (4.10), and (4.11) can be evaluated easily for any variable  $y$ .

Figure 4.11 shows the dialog box used for defining and starting a sensitivity analysis. This dialog box is opened with the **Sensitivity Analysis** command in the **Calc** menu shown in Figure 4.1.

The two upper list boxes in this dialog box show the active and the available 'Parameters' (constant variables and real list variables), respectively. With the aid of the buttons 'Activate' and 'Inactivate', parameters selected in the right list box can be



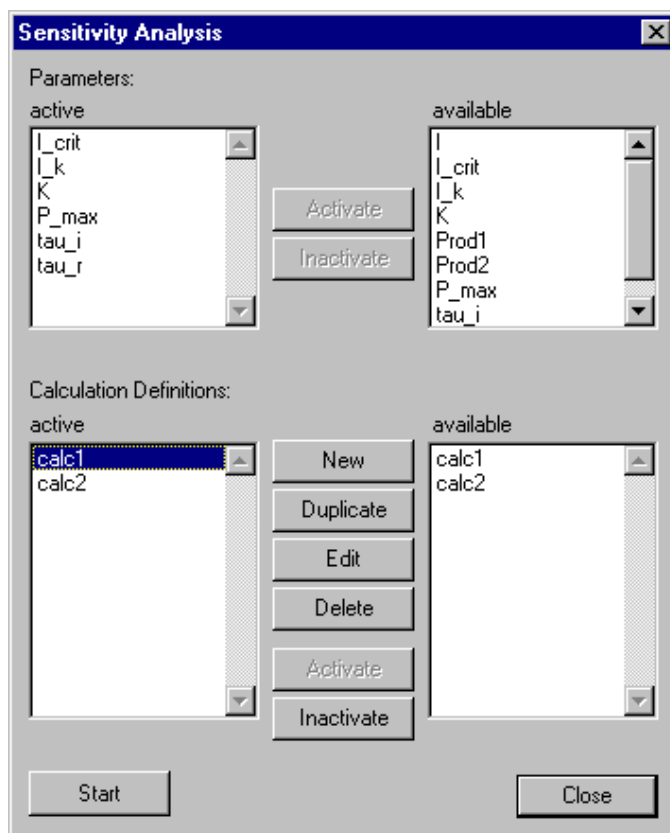


Figure 4.11: Dialog box for editing a sensitivity analysis.

activated and parameters selected in the left list box can be inactivated (this is equivalent to toggle the check box **active for sensitivity analysis** in the dialog boxes shown in Figs. 3.7 and 3.9).

The two lower list boxes show the active and the available '**Calculations**', respectively. These are the same calculation definitions as are needed to execute simulations as described in section 4.1. The buttons '**New**', '**Duplicate**', '**Edit**', and '**Delete**', used for editing calculations, have the same functionality as the same buttons described for the dialog box shown in Fig. 4.4 in section 4.1. The calculations are edited as described in section 4.1 with the aid of the dialog box shown in Fig. 4.5. With the aid of the buttons '**Activate**' and '**Inactivate**', calculations selected in the right list box can be activated and calculations selected in the left list box can be inactivated (this is equivalent to toggle the check box **active for sensitivity analysis** in the dialog box shown in Fig. 4.5).

The button '**Start**' is used to start the sensitivity analysis, the button '**Close**' to close this modeless dialog box. After clicking the button **Start** the user is asked to specify the name of a file for a sensitivity ranking. If the user clicks **Cancel** in the file open dialog box, no sensitivity ranking is produced (this may be advantageous to save computation time if the ranking is not required). During execution of the sensitivity analysis the dialog box shown in Fig. 4.12 is displayed. This dialog box shows the progress of the calculation and it allows the user to interrupt the sensitivity analysis. When all simulations have been performed and if a sensitivity ranking file has been specified, the dialog box shown

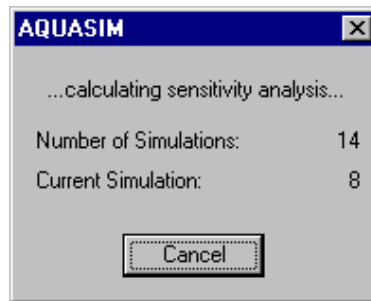


Figure 4.12: Dialog box for interrupting a sensitivity analysis during calculation.

in Fig. 4.13 appears. Now the program calculates a ranking of the sensitivity functions as

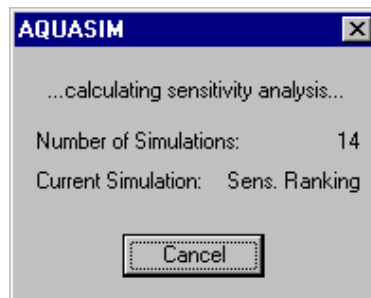


Figure 4.13: Dialog box for interrupting a sensitivity ranking (without losing the other sensitivity analysis results).

described below. This process can be interrupted (losing the ranking) without losing the calculated states.

The sensitivity ranking file looks as follows:

```
*****
AQUASIM Version 2.0 (win/batch) - Sensitivity Analysis File
*****
```

Date and time of listing: 07/28/1997 09:01:42

Ranking of mean absolute sensitivities and error contributions:

```
Calculation Number: 1
Compartment: Reactor
Zone: Bulk Volume
Variable: C
```

	Parameter:	Sens AR:	Parameter:	Error Contr.:
		[mg/l]		[mg/l]
1	Cini1	5.606	rmax1	0.1093
2	rmax1	3.166	K	0.1069
3	K	0.7936	Cini1	0.1032
4	Cini2	0	Cini2	0
5	rmax2	0	rmax2	0

Calculation Number: 2  
 Compartment: Reactor  
 Zone: Bulk Volume  
 Variable: C

	Parameter:	Sens AR:	Parameter:	Error Contr.:
		[mg/l]		[mg/l]
1	Cini2	0.0769	K	0.0071
2	rmax2	0.0632	rmax2	0.0031
3	K	0.0530	Cini2	0.0030
4	rmax1	0	rmax1	0
5	Cini1	0	Cini1	0

It gives a ranking of the averages of the absolute values of the absolute-relative sensitivity functions (4.9c) and of the error contributions (4.11) for all state variables in all zones of all compartments and for all calculations. For models with many parameters, this ranking is very useful to quickly find out which are the most important parameters influencing the state variables in the different compartments.

### 4.3 Parameter Estimation

Model parameters represented by constant variables can be estimated by AQUASIM by minimizing the sum of the squares of the weighted deviations between measurements and calculated model results

$$\chi^2(\mathbf{p}) = \sum_{i=1}^n \left( \frac{y_{meas,i} - y_i(\mathbf{p})}{\sigma_{meas,i}} \right)^2 \quad (4.13)$$

In this equation  $y_{meas,i}$  is the  $i$ -th measurement,  $\sigma_{meas,i}$  is its standard deviation,  $y_i(\mathbf{p})$  is the calculated value of the model variable corresponding to the  $i$ -th measurement and evaluated at the time and location of this measurement,  $\mathbf{p} = (p_1, \dots, p_m)$  are the model parameters, and  $n$  is the number of data points. The measurements  $y_{meas,i}$  (for  $i = 1, \dots, n$ ) must be represented by real list variables with the argument either the program variable **Time** or the program variable corresponding to the space coordinate of the compartment in which the comparison takes place. The standard deviations  $\sigma_{meas,i}$  can be defined individually for each data point or globally for all data points of each real list variable in the dialog box used for editing real list variables shown in Fig. 3.9. The sum (4.13) extends over all data points of all real list variables specified as fit targets as shown below. Simultaneous comparisons of data for measurements corresponding to different variables, compartments and zones are possible. AQUASIM performs a minimization of the sum of squares (4.13) with the constraints

$$p_{min,i} \leq p_i \leq p_{max,i} \quad (4.14)$$

where  $p_{min,i}$  and  $p_{max,i}$  are the minimum and maximum of the constant variable representing  $p_i$ . The values for  $p_{min,i}$  and  $p_{max,i}$  are specified in the dialog box used for editing constant variables shown in Fig. 3.7.

Due to the possible nonlinearity of the model equations and due to the numerical integration procedure, the sum (4.13) must be minimized numerically. The user has the choice between two numerical minimization algorithms: The simplex algorithm (Nelder and Mead, 1965) and the secant algorithm (Ralston and Jennrich, 1978). Both of these techniques are well-suited for the minimization of numerically integrated equations, because they avoid the calculation of derivatives of the solutions with respect to the parameters.

The **simplex** technique starts with a set of  $m + 1$  ( $m$  is the number of parameters) arrays of parameter values building a simplex with nonzero volume in parameter space (in AQUASIM, this set consists of one array with the user-defined start values and  $m$  arrays in which exactly one parameter value differs from its start value). At each iteration step, the simplex of arrays of parameter values is replaced by a new simplex according to the following strategy (Nelder and Mead, 1965) illustrated in Fig. 4.14 for the case of two parameters to be estimated. The parameter array leading to the highest value of  $\chi^2$  is first reflected at the plane defined by the other parameter arrays (see Fig. 4.14 (1); this is a rough guess for the downhill direction). If the value of  $\chi^2$  at this reflected array of parameter values is between the lowest and the highest value of the old simplex, this reflected array of parameter values is accepted and replaces the array leading to the highest value of  $\chi^2$ . If the value of  $\chi^2$  at the reflected parameter array is smaller than the lowest value on the old simplex, a new parameter array with an increased step size is calculated (see Fig. 4.14 (2)) by an expansion of the simplex. If the value of  $\chi^2$  at this

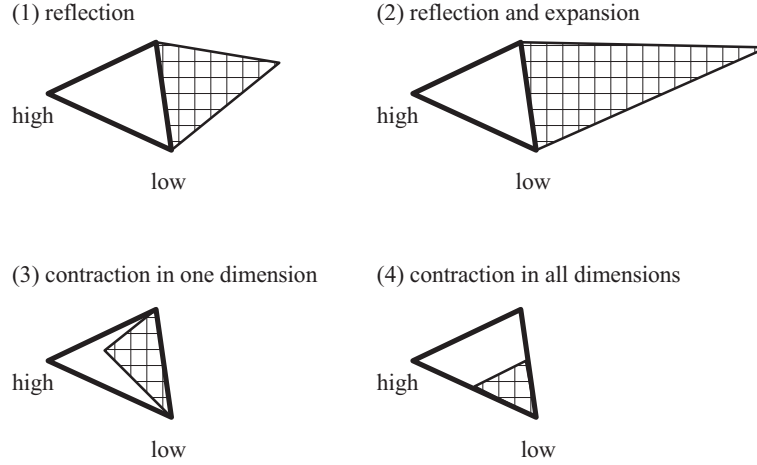


Figure 4.14: Basic steps of the simplex algorithm for function minimization illustrated for the case of two parameters. The solid line simplex is replaced by one of the hashed simplexes.

new array of parameter values is smaller than the lowest value on the old simplex it is accepted, otherwise the reflected array (without expansion) is accepted (in both cases the new array replaces the array leading to the highest value of  $\chi^2$  of the old simplex). If the value of  $\chi^2$  at the reflected point is higher than the highest value at the old simplex, a new array of parameter values is constructed by contracting the point with the highest value of  $\chi^2$  of the old simplex in direction to the other points of the simplex (see Fig. 4.14 (3)). If this leads to a value of  $\chi^2$  that is smaller than the highest value at the old simplex, this array is accepted, otherwise, all arrays of parameter values of the old simplex with exception of that leading to the lowest value of  $\chi^2$  are contracted in direction to the array leading to the lowest value (see Fig. 4.14 (4)). This procedure is repeated until all values of  $\chi^2$  at the points of the simplex are close enough together in order to fulfill the convergence criterion. This unconstrained simplex technique is extended to constrained minimization by the coordinate transformation of the parameters

$$p'_i = \tan \left( \frac{\pi}{2} \frac{2p_i - p_{max,i} - p_{min,i}}{p_{max,i} - p_{min,i}} \right) \quad (4.15)$$

shown in Fig. 4.15. Application of the unconstrained technique to the parameters  $p'_i$  is equivalent to constrained minimization of the parameters  $p_i$ . If the (unconstrained) estimates of the parameters  $p'_i$  are found the parameters  $p_i$  can be calculated by the inverse formula

$$p_i = \frac{1}{2}(p_{max,i} + p_{min,i}) + (p_{max,i} - p_{min,i}) \frac{\arctan(p'_i)}{\pi} \quad (4.16)$$

Similarly to the simplex technique described above, the **secant** technique starts with a set of  $m + 1$  arrays of parameters. But now the secant technique uses the special form of the function  $\chi^2$  (4.13) to be minimized. There is exactly one linear function in the parameters that passes through all function values  $y_i(\mathbf{p})$  for all  $m + 1$  parameter sets. For this linear function, the expression (4.13) is a quadratic form with exactly one well-defined minimum. The idea of the secant technique (Ralston and Jennrich, 1978) is to

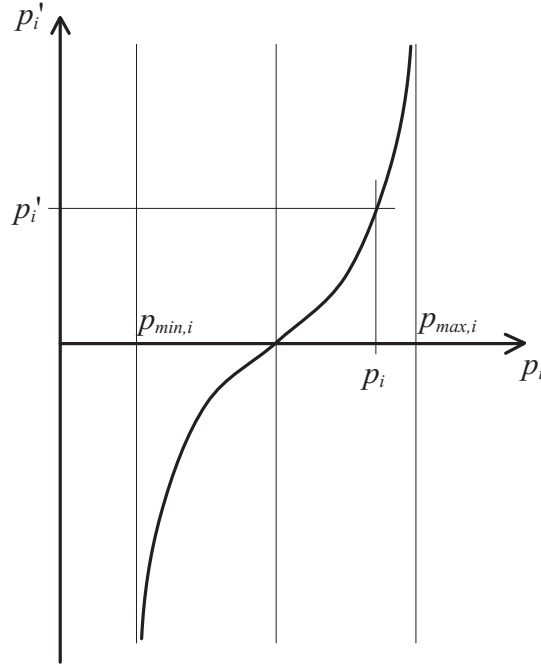


Figure 4.15: Scaled tan-function used to transform the bounded interval  $p_{min,i}, p_{max,i}$  of the parameter  $p_i$  to the whole real axis for the parameter  $p'_i$ .

replace one of the  $m + 1$  parameter arrays with the array leading to the minimum for the linear approximation of the model functions  $y_i(\mathbf{p})$  at each iteration step. In order to take into account the constraints of the parameters, the secant technique was combined with the active set technique (Gill et al., 1981) for the implementation in AQUASIM. In the active set technique the inequality constraints (4.14) are replaced by a number of (active) equality constraints for which some of the parameters are exactly on one of their boundary values. After each iteration step, this set of active constraints is checked for validity and changes to this set are made if this seems to be necessary.

Because it looks for a downward direction in a very robust way, the simplex technique may be applied even to a poorly defined parameter estimation process with starting values of the parameters far from those leading to the minimum of (4.13). In contrast, the secant method has more problems with bad starting values and poorly defined minima of  $\chi^2$ , but it leads to much faster end convergence close to a well-defined minimum. Estimates for the standard errors of the estimated parameters and for the parameter correlation matrix are only calculated by the secant method, and only if no parameter estimated is on one of the bounds  $p_{min,i}$  or  $p_{max,i}$  of the parameter.

Figure 4.16 shows the dialog box used for defining and starting a parameter estimation. This dialog box is opened with the **Parameter Estimation** command in the **Calc** menu shown in Figure 4.1.

The two upper list boxes in this dialog box show the active and the available '**Parameters**' to be estimated (constant variables), respectively. With the aid of the buttons '**Activate**' and '**Inactivate**', parameters selected in the right list box can be activated and parameters selected in the left list box can be inactivated (this is equivalent to tog-

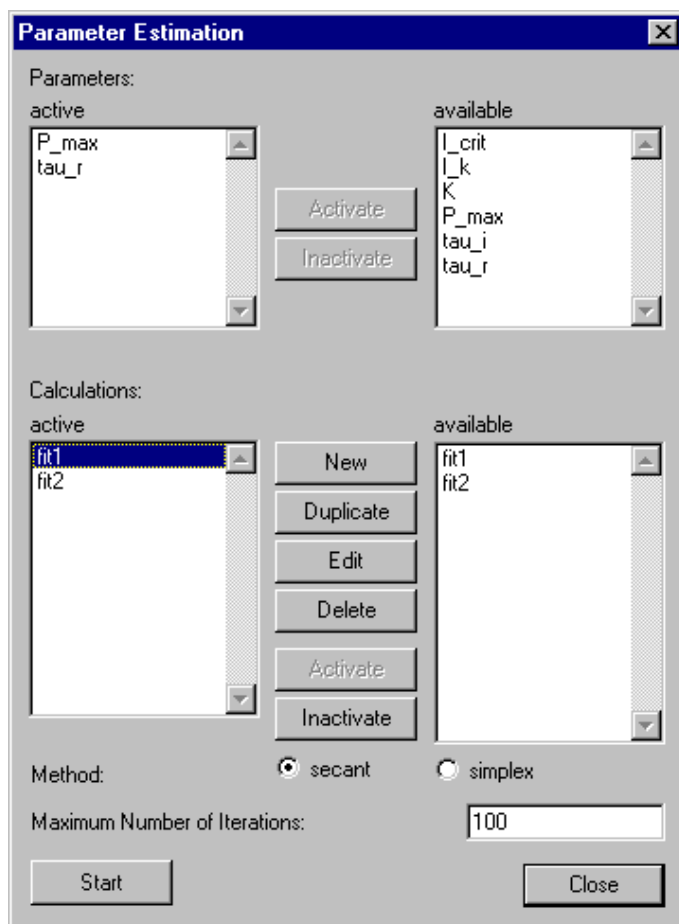


Figure 4.16: Dialog box for editing a parameter estimation.

gle the check box **active for parameter estimation** in the dialog box used for editing constant variables shown in Fig. 3.7).

The two lower list boxes show the active and the available **‘Calculations’**, respectively. The calculation definitions for parameter estimation can be edited with the aid of the buttons **‘New’**, **‘Duplicate’**, **‘Edit’** and **‘Delete’** as discussed below. With the aid of the buttons **‘Activate’** and **‘Inactivate’**, calculations selected in the right list box can be activated and calculations selected in the left list box can be inactivated (this is equivalent to toggle the check box **active for parameter estimation** in the dialog box shown in Fig. 4.17).

The radio buttons **‘secant’** and **‘simplex’** can be used to select the numerical minimization **‘Method’** above. The basic ideas and references for both possible selections are given above.

In the edit field **‘Maximum Number of Iterations’**, the number of iterations can be bounded. A value of zero leads to the calculation of the value of  $\chi^2$  without trials to improve this value.

The button **‘Start’** is used to start the parameter estimation, the button **‘Close’** to close this modeless dialog box. After clicking the button **Start** the user is asked to specify

the name of a file on which the parameter and  $\chi^2$  values for all iteration steps are reported and which contains a final summary of parameter estimates, estimated standard errors and correlation coefficients (for the secant method not leading to estimates on the bounds of the constraining interval only), and a list of the contributions of each data series to the total value of  $\chi^2$  (see below for an example of such a file).

As discussed above, the buttons **New**, **Edit** and **Duplicate** in the dialog box shown in Fig. 4.16 are used to define calculations for parameter estimation. Figure 4.17 shows the dialog box used for defining and editing a single calculation. The items **Name**, **Description**,

Figure 4.17: Dialog box for editing a parameter estimation calculation.

**Calc. Number**, **Initial Time** and **Initial State** have the same meaning as for calculations for simulation and sensitivity analysis in the dialog box shown in Fig. 4.5:

The edit field '**Name**' is used to specify the name of the calculation. Each calculation needs a unique name as an identifier. A name of a calculation consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit.

To improve documentation of calculations the edit field '**Description**' can optionally be used to store comments on the purpose of the calculation definition.

The edit field '**Calc. Number**' allows the program user to identify a calculation by a nonnegative integer number. Variables (and thus also processes, etc.) can be made dependent on this calculation number with the aid of the program variable **Calculation Number**. Active calculations must have different values of their calculation numbers.

Each calculation must be initialized before it can be started. In the edit field '**Initial Time**' the time at which a calculation is initialized can be specified.

The user has the choice between two types of the '**Initial State**' to be used by



the program. The default option '**given, made consistent**' applies the initial state as given by the user (a value of zero is the default initial condition for all state variables for which the user did not explicitly specify an initial condition). If this initial state violates constraints on boundary conditions or algebraic equations of equilibrium processes, the program tries to fulfill these constraints by a minimal number of modifications to the initial state given by the user. This procedure is called to make the initial state consistent. With the selection of the alternative option '**steady state**' the program tries to find the steady state solution of the user-defined model (under external parameter values evaluated at the initial time) and uses this steady state solution as the initial condition. Note, however, that not all models have a steady state solution and that even if such a solution exists, the numeric algorithm used by AQUASIM may fail to find it. In such a situation, the convergence can be improved by a good choice of initial conditions which, under this option for the initial state, are used as starting values for the iterative search process for the steady state solution. If this does not help, the steady state solution must be found by relaxation, i.e. by executing a dynamic simulation with constant boundary conditions.

After these five items which have the same meaning as those discussed for simulation and sensitivity analysis (Fig. 4.5), instead of step sizes and numbers of steps, the user must specify **Fit Targets** (the step sizes are selected automatically by the program to meet the points in time at which data are available). All fit targets already defined are listed in the list box of the dialog box shown in Fig. 4.17. Each fit target consists of a data series and a specification of which variable evaluated in which zone of which compartment and at which point in time or space should be compared with the data. The list of fit targets is edited with the buttons '**Add**', '**Edit**' and '**Delete**' below the list box.

Finally, the calculation can be made '**active for parameter estimation**', if no other calculation with the same calculation number is already active.

The Fig. 4.18 shows the dialog box used to specify a fit target. This dialog box is

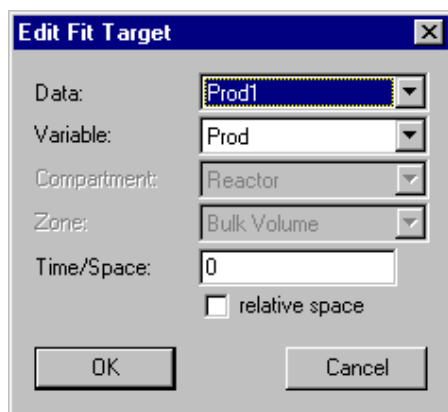


Figure 4.18: Dialog box for editing a fit target.

opened by clicking one of the buttons **Add** or **Edit** of the dialog box shown in Fig. 4.17.

In the first field '**Data**' of this dialog box a data series can be selected. This data series is represented by a real list variable with the argument either the program variable **Time** or the program variable corresponding to the space coordinate of the compartment in which the comparison with the calculated variable takes place.

The next four fields of the dialog box shown in Fig. 4.18 are used to define which calculated value of the model should be compared with the data series specified in the first field.

In the field '**Variable**', a variable must be selected. In many cases measured data is compared with values of a state variable of the model, but in AQUASIM, any variable can be used. E.g. a formula variable can be used for conversion of units, for building combinations of state variables (e.g. a sum of state variables that is measured), etc..

In the field '**Compartment**' the compartment must be selected in which the variable to be compared with the data has to be evaluated. This field is inactive if there exists only one compartment.

In the field '**Zone**' the zone of the compartment must be selected in which the variable to be compared with the data has to be evaluated. This field is inactive if there exists only one zone in the compartment selected above.

In the last field **Time/Space** the time must be specified at which the comparison of the spatial profile data with the calculated variable, or the spatial location at which the comparison with the time series data must take place. Note that the value entered here is interpreted as a spatial location if the argument of the real list variable selected in the field **Data** is the program variable **Time**, and as a point of time if the argument of the real list variable selected in the field **Data** is the program variable corresponding to the spatial location of the compartment selected in the field **Compartment**. If the check box '**relative space**' is ticked, spatial locations must be given in relative coordinates, otherwise in absolute coordinates (this check box has no influence for time specifications which are always absolute). In the case of a comparison with time series data, the value entered in the field **Time/Space** must be within the legal coordinate range of the compartment. For absolute spatial locations, the legal range is defined in the definition of the compartment selected in the field **Compartment**. Relative locations must be between 0 and 1. If the compartment is a mixed reactor compartment (which has no spatial coordinate), the value entered in the field **Time/Space** is ignored.

During execution of the parameter estimation the dialog box shown in Fig. 4.19 is displayed. This dialog box shows the initial value of  $\chi^2$ , the best (smallest) value already

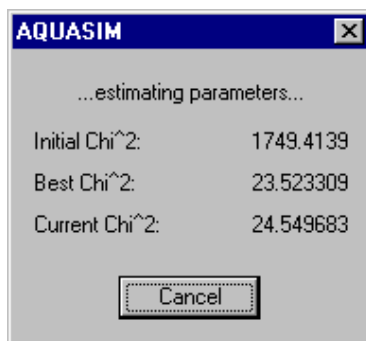


Figure 4.19: Dialog box for interrupting a parameter estimation.

found, and the value for the current parameter set. This dialog box allows the user to interrupt the calculation.

In case of normal termination of the parameter estimation algorithm, the dialog box shown in Fig. 4.20 is displayed to give the user a brief survey of the fit results. The first

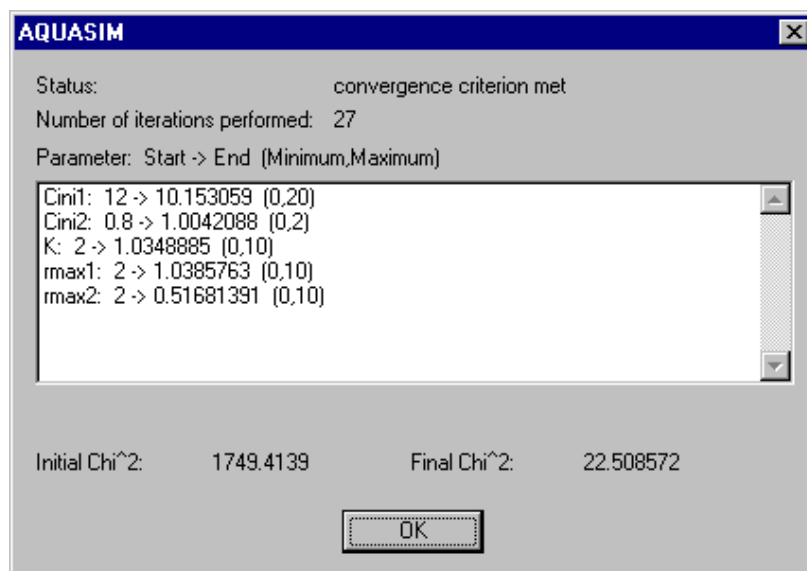


Figure 4.20: Dialog box for displaying a summary of fit results.

row of this dialog box shows the termination status of the algorithms, the second row the number of iterations performed. In the following list box for each parameter that was active for parameter estimation, its initial value, its final value (found by the parameter estimation algorithm), and its minimum and maximum is given. Finally, below the list box, the initial and the final value of  $\chi^2$  are given in order to give the user a quick overview on the improvement that could be achieved by the parameter estimation procedure.

More details on the performance of the parameter estimation algorithm is given on the file specified after clicking the button **Start** in the dialog box shown in Fig. 4.16. This file looks as follows:

```
*****
AQUASIM Version 2.0 (win/mfc) - Parameter Estimation File
*****
```

Date and time of listing: 07/29/1997 13:56:09

```
Number of parameters   = 5
Number of data points  = 22
Estimation method      = secant
Parameters:
```

Name	Unit	Start	Minimum	Maximum
Cini1	mg/l	12	0	20
Cini2	mg/l	0.8	0	2
K	mg/l	2	0	10
rmax1	mg/l/h	2	0	10
rmax2	mg/l/h	2	0	10

Calculations:

Cini1 [mg/l]	Cini2 [mg/l]	K [mg/l]	rmax1 [mg/l/h]	rmax2 [mg/l/h]	Chi <sup>2</sup>
12	0.8	2	2	2	1749.41
12.2	0.8	2	2	2	1700.74
12	0.82	2	2	2	1725.51
12	0.8	2.1	2	2	1656.36
12	0.8	2	2.1	2	1928.34
12	0.8	2	2	2.1	1795.93
11.9411	0.80557	2.3	2.01325	2.01687	1516.94
11.8094	0.82432	2.9	2.06794	2.13019	1191.95
.	.	.	.	.	.
10.1604	1.00113	1.04662	1.04293	0.51529	22.6634
10.1531	1.00421	1.03489	1.03858	0.51681	22.5086

Parameter estimation successfully finished (convergence criterion met)

Cini1 [mg/l]	Cini2 [mg/l]	K [mg/l]	rmax1 [mg/l/h]	rmax2 [mg/l/h]
-----------------	-----------------	-------------	-------------------	-------------------

Estimated values of the parameters:

10.1531 1.00421 1.03489 1.03858 0.51681

Estimated standard errors of the parameters:

0.22133 0.03057 0.27284 0.07084 0.09739

Estimated correlation matrix of the parameters:

1	0.17923	0.53889	0.77737	0.52576
0.17923	1	0.34116	0.30991	0.46943
0.53889	0.34117	1	0.92990	0.98009
0.77737	0.30991	0.92990	1	0.901655
0.52576	0.46943	0.98009	0.90166	1

Contribution of data series to Chi<sup>2</sup>

Calculation:	Data Series:	Chi <sup>2</sup> ini:	Chi <sup>2</sup> end:
Fit1	Cmeas1	801.34	12.1706
Fit2	Cmeas2	948.074	10.338
		-----	-----
		1749.41	22.5086

Number of steps performed = 27

Number of simulations performed = 39

After the header, on this file the number of parameters and the number of data points active for the current parameter estimation and the method selected for the numerical min-

imization algorithm are indicated. Then the file contains a listing of all active parameters with unit, start value, minimum and maximum. Then a listing of the parameter values and of the values of  $\chi^2$  for all calculations performed during the parameter estimation process is given. After this listing, the parameter estimates are given. If the estimation algorithm was the secant method and no parameter estimate was on the border of its legal range, an estimate of the standard error and of the parameter correlation matrix is also given. Then, in addition to the value of  $\chi^2$  which represents the sum of the deviations for all fit targets, the contributions of all fit targets to the final value of  $\chi^2$  are given individually. Finally, the number of iterative steps and the number of simulations performed, are given.

## Chapter 5

# Visualization of Results

Fig. 5.1 shows the menu **View** of AQUASIM. The item '**Results**' of this menu is used



Figure 5.1: View menu.

to specify plot definitions, to plot results to the screen or in PostScript or Encapsulated PostScript format to a file, and to list results to a text file for external postprocessing. The item '**Toolbar**' is used to activate or hide the AQUASIM toolbar, and the item '**Status Bar**' is used to activate or hide the status bar of AQUASIM (the status bar is not used in the current version of AQUASIM).

Figure 5.2 shows the dialog box used for editing plot definitions and for plotting and listing results. This dialog box is opened with the command **Results** of the menu **View** shown in Fig. 5.1. The list box in this dialog box contains a list of plot definitions. Each of these plot definitions contains general plot specifications such as meaning, labels and scaling of axes and a list of curves to be drawn in the plot. Each of these curves contains specifications such as the type of the curve (value, sensitivity function or error contribution of a variable), the variable, the calculation number, the compartment, the

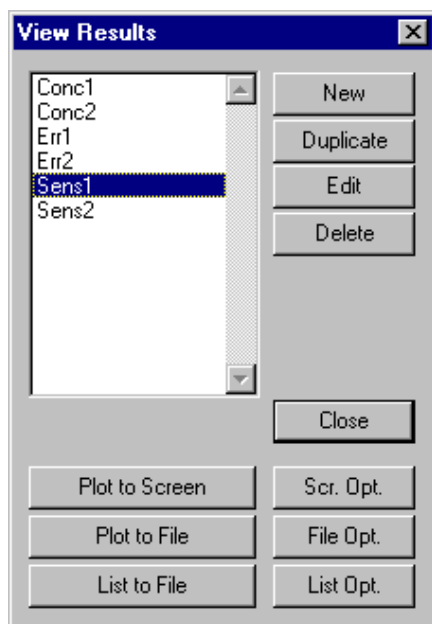


Figure 5.2: Dialog box for editing plot definitions and plotting and listing results.

zone, the spatial location (if the abscissa of the plot is time) or the time (if the abscissa of the plot is the spatial coordinate of the compartment) at which the curve should be drawn, and the signature of the curve. All these definitions can be specified independently of the existence of the calculated states necessary for actually drawing the curves. The recommended strategy is therefore to spend some time to carefully specify a series of plot definitions which can later on be used very efficiently to present an overview of the current simulation results. The buttons at the right of the list box of plot definitions allow the user to perform the following operations: By clicking the button **'New'**, new plot definitions can be specified. Alternatively, by clicking the button **'Duplicate'**, the selected plot definition can be duplicated. With the button **'Edit'**, or by double-clicking the plot definition name in the list box, a plot definition can be edited. The button **'Delete'** allows the program user to delete plot definitions. The buttons **Duplicate**, **Edit** and **Delete** are inactive as long as no plot definition is selected. The dialog boxes used for editing plot definitions are discussed later in this section (Figs. 5.3 to 5.5).

The three buttons **'Plot to Screen'**, **'Plot to File'**, and **'List to File'** are used to draw a plot corresponding to the currently selected plot definition to the screen, to write it in PostScript or Encapsulated PostScript format to a file, and to write it in text format to a file, respectively. The PostScript or Encapsulated PostScript file must be handled by the user. Typically, a PostScript file is sent to a printer with the aid of an appropriate program (available as a shareware program), an Encapsulated PostScript file is included as a figure into a text processing program. All the three actions described above are only possible, if calculated data for at least one curve of the currently selected plot definition exists and if no curve contains undefined values as they can occur e.g. by division by zero or by taking logarithms of negative numbers in the variable to be plotted.

The three buttons **'Scr. Opt.'**, **'File Opt.'**, and **'List Opt.'** are used to specify

general options for plotting to the screen, for plotting to a PostScript file, and for listing plot data to a text file, respectively. The dialog boxes used for specifying these options are discussed later in this section (Figs. 5.6 to 5.8).

Figure 5.3 shows the dialog box used for defining or editing a plot definition. This

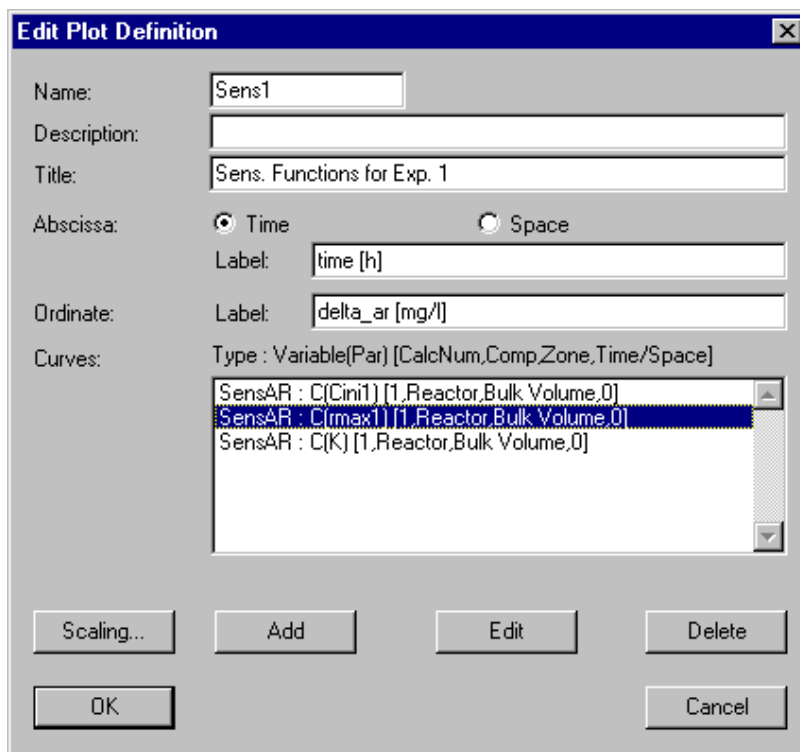


Figure 5.3: Dialog box for editing a plot definition.

dialog box is opened by clicking one of the buttons **New**, **Edit** or **Duplicate** of the dialog box shown in Fig. 5.2.

Each plot definition needs a unique '**Name**' as an identifier. A name of a plot definition consists of a sequence of letters (A-Z,a-z), digits (0-9) and underline characters (\_). The first character may not be a digit. To improve documentation of plot definitions, a '**Description**' can be given optionally.

The edit field '**Title**' is used to specify the title of the plot.

The radio buttons '**Abscissa**' can be used to select the meaning of the abscissa. The user has the choice between the abscissa '**Time**' which is used for plotting time series at given locations in the system described with AQUASIM and the absissa '**Space**' used for plotting spatial profiles at given points of time (this last option is not meaningful for mixed reactors that have no spatial variation in state variables).

The edit fields '**Label**' can be used to specify a label for the **Abscissa** and for the **Ordinate**, respectively.

A plot definition contains an arbitrary number of definitions of '**Curves**'. All curve definitions are listed in the list box of the dialog box shown in Fig. 5.3. The buttons '**Add**', '**Edit**' and '**Delete**' below this list box are used to edit the list of curve definitions. If no



curve definition is selected, clicking the button **Add** leads to the insertion of a new curve definition at the end of the list of curve definitions. If a curve definition is selected, the new curve definition is inserted before the selected curve definition.

Finally, the button '**Scaling**' is used to choose options for bounds and ticks of the axes of the plot.

Figure 5.4 shows the dialog box used for defining or editing a curve definition within a plot. This dialog box is opened by clicking one of the buttons **Add** or **Edit** of the dialog

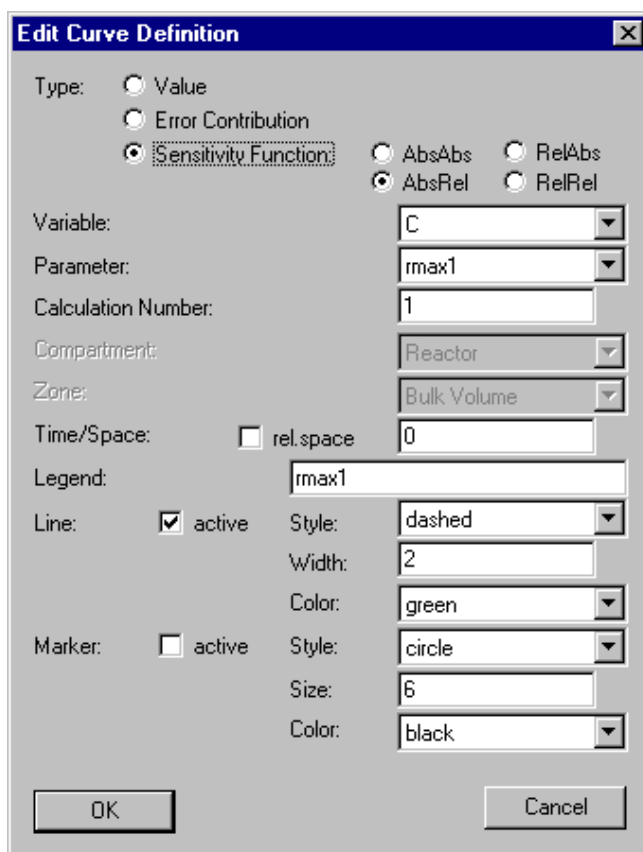


Figure 5.4: Dialog box for editing the definition of a curve within a plot.

box shown in Fig. 5.3.

The radio buttons '**Type**' are used to select the type of the curve. The user has the choice between the following options: The radio button '**Value**' is selected in order to plot the value of a variable. The value of all variables can be plotted if calculated states corresponding to the specifications given below are available. If not only a simulation, but a sensitivity analysis has been performed, the curve for the value of the variable is supplemented by thin lines indicating the value plus and minus one standard deviation calculated according to the linear error propagation formula which ignores parameter correlations (4.10). Usually, if the abscissa of the plot is **Time**, the curve for the value of a variable is plotted as linear interpolation between values evaluated at each output time step specified in the dialog box shown in Fig. 4.5 (for plots drawn after a simulation or a

sensitivity analysis) or by the time steps of the data specified as fit targets in the dialog box shown in Fig. 4.18 (for plots drawn after a parameter estimation). If the abscissa of the plot is **Space**, similarly the grid points of the spatial resolution of the compartment are used. There is one important exception from this general rule: If the abscissa of the plot is **Time** and the variable specified in the curve definition is a real list variable with the program variable **Time** as the argument, or if the abscissa of the plot is **Space** and the variable specified in the curve definition it is a real list variable with the program variable corresponding to the spatial extent of the compartment as the argument, the data pairs of the real list are used instead of the interpolated values at the simulation times or grid points. This makes it possible to plot measured data without interpolation (see also Figs. 5.9 and 5.10). Selection of one of the radio buttons '**Error Contrib.**' and '**Sens. Function**' leads only to a curve if a sensitivity analysis corresponding to the specifications below has been performed. Selection of the radio button **Error Contrib.** leads to the plot of the error contribution of a parameter  $p$  to the total error of a variable  $y$  given by equation (4.11). Selection of the radio button **Sensitivity Function** leads to the plot of a sensitivity function. Selection of the radio button '**AbsAbs**' leads to the plot of the absolute-absolute sensitivity function given by equation (4.9a), selection of '**RelAbs**' to the plot of the relative-absolute sensitivity function given by equation (4.9b), selection of '**AbsRel**' to the plot of the absolute-relative sensitivity function given by equation (4.9c), and selection of '**RelRel**' to the plot of the relative-relative sensitivity function given by equation (4.9d).

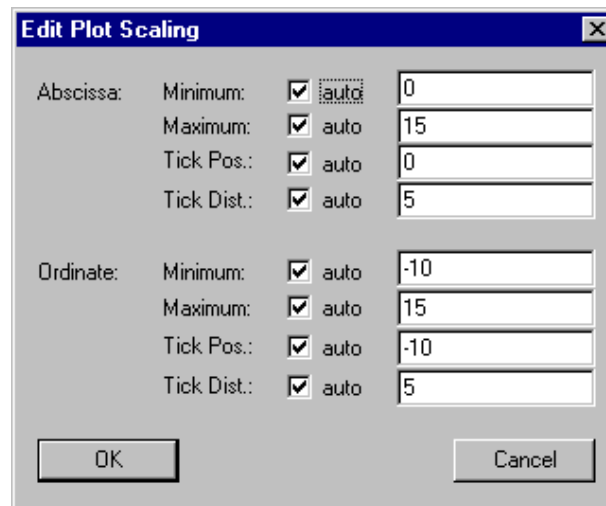


Figure 5.5: Dialog box for editing scaling of plot axes.

As a next item in the dialog box used for the specification of a curve definition shown in Fig. 5.4, a '**Variable**' must be selected.

If the type of the curve is **Error Contrib.** or **Sensitivity Function**, a constant variable or a real list variable must be selected as a '**Parameter**'. The curve can then only be plotted, if a sensitivity analysis with this **Parameter** as an active parameter has been performed. This field is inactive, if the curve is of the type **Value**.

The edit field '**Calculation Number**' is used to select a calculation identified by its calculation number (specified in one of the dialog boxes shown in Figs. 4.5 and 4.17).

Because calculations for different calculation numbers can be stored in memory simultaneously and because variables and process rates can be made depend on the calculation number with the aid of the program variable **Calculation Number**, curves for the same variable at the same location under different process hypotheses can be compared in the same plot.

As a next specification, a '**Compartment**' within which the curve must be evaluated can be selected. This field is inactive if only one compartment is defined.

If the compartment selected above contains more than one zone, a '**Zone**' must be selected. This field is inactive if the compartment contains only one zone.

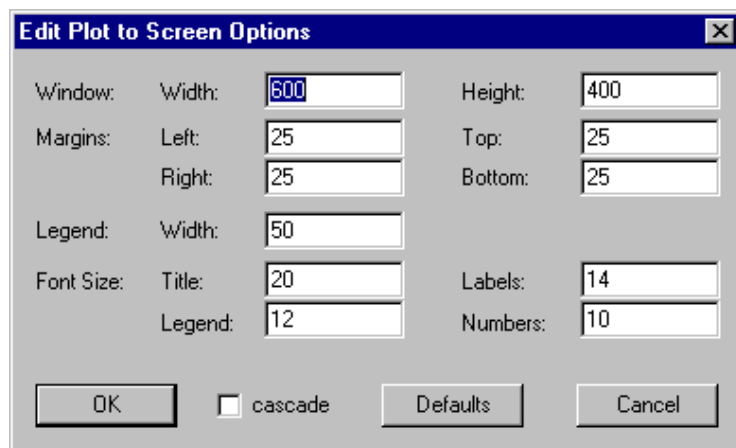


Figure 5.6: Dialog box for editing plot to screen options.

Finally, the edit field '**Time/Space**' is used to complete the specification of when and where to evaluate the curve. If the abscissa of the plot is **Time**, a spatial location valid for the compartment selected above must be given here. If the check box '**rel. space**' is ticked, relative coordinates between 0 and 1 must be given, otherwise absolute spatial locations must be specified. The values of the variable of function to be plotted are then linearly interpolated between the values at the neighbouring grid points of the compartment. In the case of a mixed reactor compartment, the value of this edit field is ignored, because there is no spatial resolution. If the abscissa of the plot is **Space** the edit field **Time/Space** is used to specify the time at which the curve is evaluated. In this case the check box **rel. space** is ignored (points in time must always be given as absolute values). The values are linearly interpolated between the neighbouring output time steps specified in one of the dialog boxes shown in Figs. 4.5 and 4.17.

In the edit field '**Legend**' the user can specify a legend entry for the curve.

The appearance of the curve can be specified by '**Line**' and '**Marker**' attributes. The check boxes '**active**' are used to turn on and off the lines between data pairs and the markers at the data pairs, respectively. As described above, the data pairs of plots with the abscissa **Time** are usually determined by the output time steps of the calculation, the data pairs of plots with the abscissa **Space** by the grid points of the compartment. If the value of a real list variable with the program variable corresponding to the space coordinate of the compartment as its argument is plotted in a plot with the abscissa **Time**, or if the value of a real list variable with the program variable **Time** as its argument is plotted in a plot with the abscissa **Space**, instead, the data pairs of the real list variable

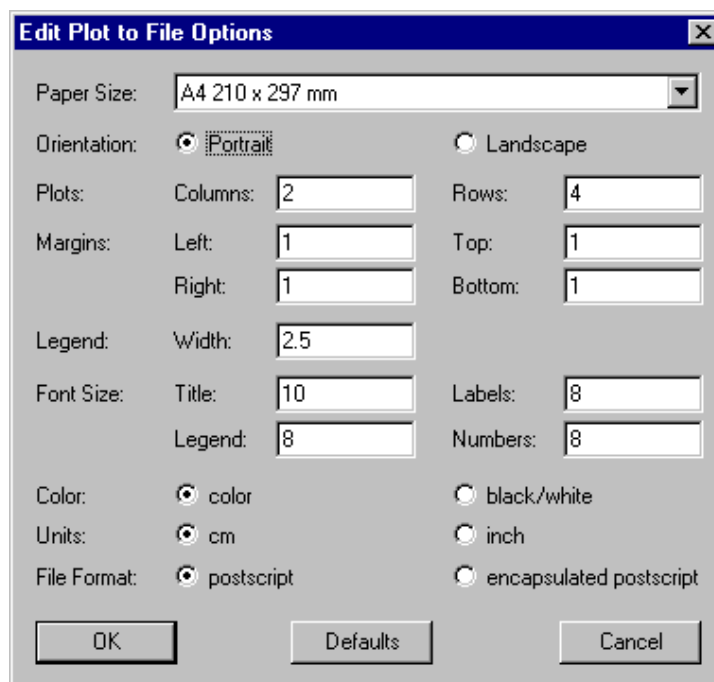


Figure 5.7: Dialog box for editing plot to file options.

are used. In order to plot the actually used interpolated values (that may be different from linear interpolation according to the selection of the **Interpolation Method** in the dialog box shown in Fig. 3.9), a formula variable with the name of the real list variable as its algebraic expression may be defined and specified as the variable to be plotted (this is in fact the way the data for the plot shown in Fig. 3.8 has been generated). Several line attributes, grouped as **'Style'**, **'Width'** and **'Color'** attributes, can be used to obtain a clear distinction between plotted curves. Similarly, for markers, **'Style'**, **'Size'** and **'Color'** attributes are available.

Figure 5.5 shows the dialog box used to specify the scaling of the plot. This dialog box is opened by clicking the button **Scaling** of the dialog box shown in Fig. 5.3.

For both, **'Abscissa'** and **'Ordinate'**, a **'Minimum'**, a **'Maximum'**, a **'Tick Position'** and a **'Tick Distance'** can either be specified or selected to be automatically determined. The possibility of specifying a tick position makes the positions of the ticks independent of the bounds of the axis.

Figure 5.6 shows the dialog box used to edit the options for plotting to the screen. This dialog box is opened by clicking the button **Scr. Opt.** of the dialog box shown in Fig. 5.3.

It allows the program users to specify the **'Width'** and the **'Height'** of the plot window, the width of the **'Margins'** and the **'Legend Width'**. In addition, the user can select the **'Font Size'** for the plot **'Title'**, for the axes **'Labels'**, for the **'Legend'** and for the tick **'Numbers'**. All measurement units are pixels.

Figure 5.7 shows the dialog box used to edit the options for plotting in PostScript

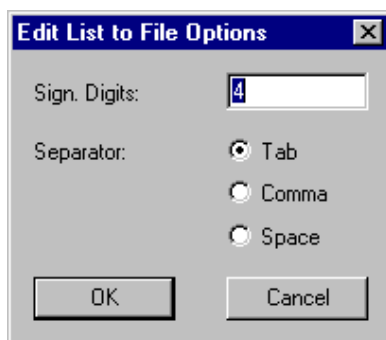


Figure 5.8: Dialog box for editing list to file options.

or Encapsulated PostScript format to a file. This dialog box is opened by clicking the button **File Opt.** of the dialog box shown in Fig. 5.3.

It allows the program user to select the **'Paper Size'** and the **'Orientation'** and to specify the number of **'Columns'** and **'Rows'** of plots on the sheet, the **'Margins'**, the **'Legend Width'** and the **'Font Size'** for the **'Title'**, the axes **'Labels'**, the **'Legend'** and the tick **'Numbers'**.

Furthermore, the user can select to plot in **'color'** or **'black/white'**, to use **'cm'** or **'inch'** as measurement units for distances and to select **'PostScript'** or **'Encapsulated PostScript'** as the file format (PostScript is suited for printing, Encapsulated Postscript for including AQUASIM plots in other documents).

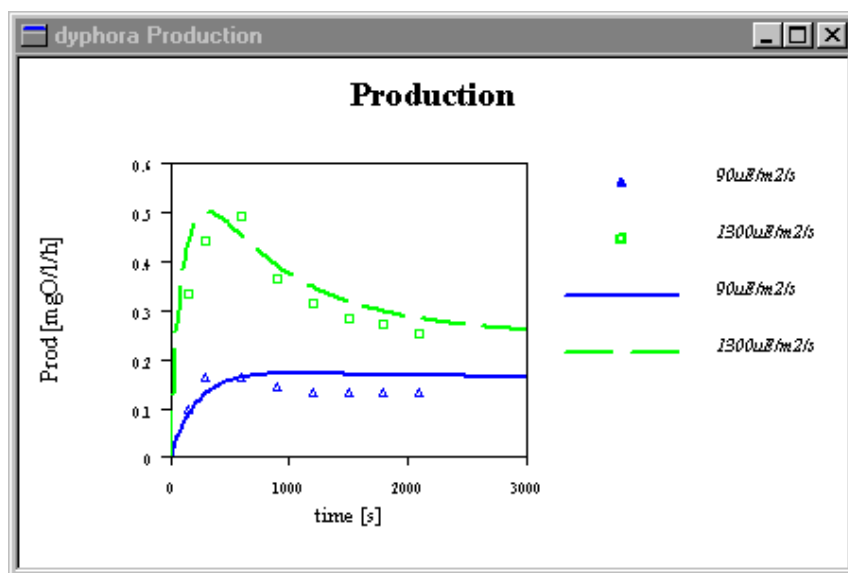


Figure 5.9: Example of a plot window on the screen.

Figure 5.8 shows the dialog box used to edit the options for listing calculated results corresponding to a plot definition to a text file. This dialog box is opened by clicking the

button **List Opt.** of the dialog box shown in Fig. 5.3.

The user can select the number of '**Significant Digits**' and the **Separator** between numbers on the output file.

Figure 5.9 shows a plot window on the screen. This plot window is generated by clicking the button **Plot to Screen** in the dialog box shown in Fig. 5.2. It contains four curves. Two of them, represented by markers, are real list variables with the program variable **Time** as the argument. In this case, the data pairs of the real list variables are plotted. The other two curves, represented by a dashed and a solid line, respectively, are simulation results and are plotted with a high temporal resolution. Figure 5.10 shows the same plot written in Encapsulated PostScript format to a file and imported to this document. This file is obtained by selecting the option **Encapsulated PostScript** in the dialog box shown in Fig. 5.7 and then clicking the button **Plot to Screen** in the dialog box shown in Fig. 5.2. Finally, the following listing shows the plot of two of the curves shown above as obtained by clicking the button **List to File** in the dialog box shown in Fig. 5.2.

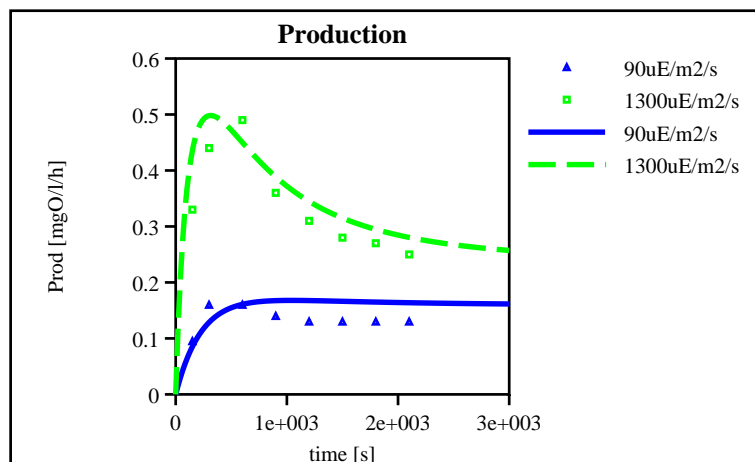


Figure 5.10: Example of an Encapsulated PostScript plot.

```
*****
AQUASIM Version 2.0 (win/mfc) - List File
*****
```

Date and time of listing: 05/05/1998 11:47:00

Name of plot: Val\_Prod

Type:	Argument	Value	Argument	Value
Variable:	t	Prod1	t	Prod
Parameter:				
CalcNum:		0		1
Compart.:		Reactor		Reactor
Zone:		Bulk Volume		Bulk Volume
Time/Space:		0		0
Unit:	s	mgO/l/h	s	mgO/l/h
Legend:		90uE/m2/s		90uE/m2/s
	150	0.095	0	0
	300	0.16	20	0.01492
	600	0.16	40	0.02858
	900	0.14	60	0.04106
	1200	0.13	80	0.05248
	1500	0.13	100	0.06291
	1800	0.13	120	0.07246
	2100	0.13	140	0.08118
			160	0.08915
			180	0.09644
			200	0.1031
			220	0.1092
			240	0.1147
			260	0.1198
			280	0.1244
			300	0.1286
			320	0.1325
			340	0.136
			360	0.1392
			380	0.1421
			400	0.1447
			420	0.1471
			440	0.1493
			460	0.1513
			480	0.1531
			500	0.1548
			.	.
			.	.

## Chapter 6

# Appendix

### 6.1 Character Interface Version

Besides the window interface version of AQUASIM described in the other sections of this manual, there exists a character interface version with the same functionality except that it is not able to draw plots to the screen. This version is designed for editing AQUASIM system files and for executing simulations, sensitivity analyses and parameter estimations via simple character-oriented terminal connections and for automating editing tasks with the aid of scripts of the operating system. For normal usage of the program, the window interface version is much more attractive.

Because the menu and dialog structure of the character interface version is exactly the same as that of the window interface version, the other chapters of this manual can be used as a manual for the character interface version also. In the following, some specific hints on the usage of the character interface version are given and a simple example demonstrates how to start and exit the character interface version.

In the character interface version, user input is prompted with a sign ‘*i*’ and each user input must be followed by pressing ‘Return’. Menu items have to be selected by typing the number preceding the item and pressing ‘Return’. Within menus and submenus, ‘Return’ without giving an input before leads back to the previous menu level. Default values proposed in square brackets can be accepted by pressing ‘Return’ without giving a new input before.

The following sequence shows how to start and exit the character interface version of AQUASIM (the two menus correspond to the menu bar of the window shown in Fig. 1.1 and to the menu **File** shown in Fig. 2.1):



```
>aquasimc
*****
*                                     *
* AQUASIM - Simulation of Aquatic Systems *
*                                     *
*****
```

```
AQUASIM - Main Menu
```

```
-----
```

```
1   =   File
2   =   Edit
3   =   Calc
4   =   View
>1
```

```
AQUASIM - File Menu
```

```
-----
```

```
1   =   New
2   =   Open
3   =   Close
4   =   Save
5   =   Save As
6   =   Revert to Saved
7   =   Print
8   =   About

E   =   Exit

B   =   Back
>e
```

```
End of program AQUASIM (normal termination)
```

## 6.2 Batch Version

The batch version of AQUASIM is designed for executing demanding calculations on a compute server as batch jobs. In contrast to the window and character interface versions of AQUASIM, the batch version does not provide full functionality. In particular, it is not possible to define or edit models. The idea of the batch version is to perform one of the following tasks for a model defined with the aid of an interactive program version and stored on an AQUASIM system file:

- Perform a simulation.
- Perform a sensitivity analysis.
- Perform a parameter estimation.
- Calculate values of  $\chi^2$ .
- Calculate results for given parameter sets and given variables.
- Plot results to a PostScript or Encapsulated PostScript file.
- List results to a text file.

On systems for which command line arguments can be specified, these jobs can be started with the following command lines

```
aquasimb jobfile
aquasimb -s logfile loadfile savefile [scmdfile]
aquasimb -a logfile loadfile savefile [sensfile]
aquasimb -e logfile loadfile savefile fitfile
aquasimb -c logfile loadfile vcmdfile chifile
aquasimb -r logfile loadfile vcmdfile rcmdfile resfile
aquasimb -p logfile loadfile pcmdfile
aquasimb -l logfile loadfile lcmdfile
```

The parameters used in these command lines have the following meaning:

<b>jobfile</b>	AQUASIM job file. This file can contain one or more command lines in the style of the other command lines. These command lines are executed one after the other so that a job specified in a command line can use the savefile of one of the previous jobs as its loadfile. When used within such a job file, the program name <b>aquasimb</b> may be omitted and the line can directly start with <b>-s</b> , <b>-a</b> , <b>-e</b> , <b>-c</b> , <b>-p</b> or <b>-l</b> . Within a job file a <b>#</b> -sign terminates input for the current line and makes it possible to add comments.
<b>-s</b>	Perform a simulation.
<b>-a</b>	Perform a sensitivity analysis.
<b>-e</b>	Perform a parameter estimation.
<b>-c</b>	Perform a calculation of $\chi^2$ -values.
<b>-r</b>	Calculate results for given parameter sets and given variables.
<b>-p</b>	Plot results to a PostScript file.

- l** List results to a text file.
- logfile** Name of an output text file for job log information. The information written to this file may be useful in case of problems during calculation (CAUTION: an existing file with this name is overwritten without warning!).
- loadfile** Name of the input AQUASIM system file containing the model definitions. This file must be created with one of the interactive program versions.
- savefile** Name of an output AQUASIM system file on which the results are stored. This file can be opened with an interactive program version or from a succeeding batch job for processing results (CAUTION: an existing file with this name is overwritten without warning!).
- scmdfile** Name of an (optional) input text file (simulation command file) containing the instructions for initialization and step sizes of simulations. If no such file is provided, the active calculations specified in the simulation dialog box are executed. If a simulation command file is provided, for each calculation number for which simulations should be performed, a line providing the two numbers  
`calcnm inittime`  
 (calcnm = calculation number, inittime = initial time) must be provided. This line is followed by an arbitrary number of lines of the form  
`calcnm timestep numstes`  
 (calcnm = calculation number, timestep = output step size for time integration, numstes = number of steps to be performed).
- sensfile** Name of an (optional) output text file to which a ranking of mean absolute sensitivity and error functions is written (CAUTION: an existing file with this name is overwritten without warning!).
- fitfile** Name of an output text file to which detailed results of the parameter estimation are written (CAUTION: an existing file with this name is overwritten without warning!).
- vcmdfile** Name of an input text file (parameter value file) specifying the names and values of parameters. The first line of this file must contain the names of the parameters (constant variables), on succeeding lines values of these parameters must be specified, each line containing one set of values for all parameters (the value of each parameter must be within the legal range of the parameter bounded by its minimum and maximum).
- chifile** Name of an output text file to which the results of the calculations of  $\chi^2$ -values are written. After a header, each line of this file contains the parameter values and the corresponding value of  $\chi^2$  (CAUTION: an existing file with this name is overwritten without warning!).
- rcmdfile** Name of an input text file (result definition file) defining when and where to calculate results. This file must contain lines of the following form:  
`var calcnm comp zoneind time space relabs`  
 where var is the name of a variable, calcnm is the calculation number of an active calculations, comp is the name of a compartment, zoneind is an allowed zone index for this compartment, time is a point of time within

the simulation time range, **space** is a location within the compartment, and **relabs** is equal to the letter 'a' for absolute space coordinates or equal to the letter 'r' for relative space coordinates. The space coordinate must also be specified for mixed reactor compartments, however, in this case its value is ignored.

- resfile** Name of an output text file to which the results of the calculations are written. After a header, each line of this file contains the parameter values and the corresponding values for all result definitions specified in the result definition file (CAUTION: an existing file with this name is overwritten without warning!).
- pcmdfile** Name of an input text file (plot command file) specifying the names of the plot definitions that are used to generate plots and the file names to which the plots are written (in PostScript format). A plot command file contains an arbitrary number of lines of the form
- plot psfile**
- where **plot** is the name of a plot definition and **psfile** is the name of a PostScript file to which the plot is appended.
- lcmdfile** Name of an input text file (list command file) specifying the names of the plot definitions that are used to generate listings of results and the file names to which these results are written (in ascii format). A list command file contains an arbitrary number of lines of the form
- plot lisfile**
- where **plot** is the name of a plot definition and **lisfile** is the name of a text file to which the results are appended.

If the batch version of AQUASIM is started without command line arguments (e.g. on systems that do not support this feature) the program looks for a file **aquasim.job** in the current directory. If it finds such a file, this file is interpreted as an AQUASIM job file, if no file with this name exist, the programs terminates with an error message.

As an example, an AQUASIM job file containing the lines

```
-e fit1.log test.aqu test_res.aqu test1.fit
-e fit2.log test_res.aqu test_res.aqu test2.fit
-e fit3.log test_res.aqu test_res.aqu test3.fit
-s sim.log test_res.aqu test_res.aqu sim.cmd
-p plot.log test_res.aqu plot.cmd
```

executes a parameter estimation based on the AQUASIM system file **test.aqu**, restarts this estimation twice, then for the fitted parameter values starts a simulation according to the specification given in the simulation command file **sim.cmd** (e.g. for improving the plot resolution) and plots the results according to the specifications given in the plot command file **plot.cmd** to PostScript files.

## 6.3 Troubleshooting

AQUASIM was developed in the computer and systems sciences department of EAWAG primarily for internal use. Since EAWAG is an institution for research and teaching and not a software company, it is not possible for us to provide end-user support. Instead, the following possibilities are offered:

- In this chapter hints are given for self-diagnosis and solution of problems (section 6.3.1).
- A user group is maintained as an e-mail list through which questions can be sent to other AQUASIM users. In this way, AQUASIM users can find help within the user community of AQUASIM (section 6.3.2).
- Bug reports and suggestions for program improvements are welcome at any time. Although there is no technical support on program use, we try to fix all bugs and provide bug-fixed program versions to licensed program users (section 6.3.3).
- Courses on AQUASIM use are given occasionally at EAWAG.

A topical overview of AQUASIM-related information, such as on the most recent program version, on references of program applications, on planned AQUASIM courses, and on known bugs or problems can be found on the EAWAG home page at <http://www.eawag.ch>.

### 6.3.1 Problem Self-Diagnosis

In this subsection, hints are given for how to find the cause of problems in program use. In the first subsection, problems of loading AQUASIM system files are discussed, in the second subsection problems of editing models, and in the third subsection problems during calculations.

#### Problems of Loading Files

##### *Format of AQUASIM System Files*

Before discussing possible problems of loading files, in this paragraph some explanations to the file format used by AQUASIM are given. In order to guarantee compatibility between different operating systems, AQUASIM systems are stored as ASCII text files. Line breaks guarantee compatibility with file editors and with e-mail programs. During the loading process all special characters (ASCII code below 32, e.g. line break and carriage return characters) are stripped from the file and the file is converted to a simple character stream (this is important because line breaks are represented differently on different operating systems). AQUASIM items are structured with the aid of curly braces (`{` and `}`). Text between two successive opening or closing braces is ignored, only text between an opening and a closing brace of the same level without any braces in between is interpreted by AQUASIM. A program `convedit` (convert AQUASIM system file for editing with a text editor) is provided with AQUASIM (in the subdirectory `bin`) which converts the originally more compact file format to a format that is much easier to read (line breaks and indents are inserted at each opening brace and between closing braces of different level) and is equivalent for AQUASIM (there is no tool to reverse this process, because this is not necessary since AQUASIM can read the converted file; the original format can easily be restored by loading and saving the file with AQUASIM). This tool may be used to allow advanced AQUASIM users to analyse causes for file loading problems or even to edit an AQUASIM system file directly with a text editor or a programming language (e.g. with Perl scripts). This last possibility may be efficient for the automation of repeated complicated editing processes. However, it is also dangerous, because with the aid of a text editor or a scripting language minor errors can make a file unreadable by AQUASIM.

##### *Problems During the Loading Process*

The most important cause for problems in loading files is that a file from one of the very old versions AQUASIM 1.0x or from a newer AQUASIM version than that used for loading is tried to be loaded. Files from AQUASIM 1.0x must be converted to the file format for AQUASIM 1.1 with the aid of the conversion program `conv1011` which is delivered with AQUASIM (in the subdirectory `bin`). Files from AQUASIM 1.1x and from pre-releases of AQUASIM 2.0 can then be read by AQUASIM 2.0, however, it is not possible to save a system in an older file format with AQUASIM 2.0. Files from newer versions than AQUASIM 2.0 cannot be loaded with AQUASIM 2.0.

AQUASIM system files are compatible between all operating systems supported by AQUASIM. One problem may occur on the Mac if an AQUASIM system file is transferred to the Mac from another operating system. If the file type is not `AQU_`, the file is not displayed in the file open dialog box of AQUASIM. For this reason, after the transfer to the Mac, the file type must be set to this value (AQUASIM system files created on the Mac automatically obtain this file type). This can be done in various ways:

- The file is moved on the application **AQUASIM Typewriter** which is delivered with the Mac distribution of AQUASIM.
- The file is loaded with the character interface version of the Mac and saved by this program version.
- The file type is changed to **AQU\_** with another tool, e.g. with the Mac resource editor.

After a parameter estimation, the estimated standard deviations may become undefined (due to inversion of an ill-defined matrix). On some operating systems (especially on the Mac), we did not succeed in finding a function to check for such undefined values. If on these systems such an undefined value is saved on the AQUASIM system file, AQUASIM is not able to read this value during loading. Because the save file format of AQUASIM is a text file, such an undefined value can be found with a text editor and can be corrected (see comments on the file format above). The string indicating an undefined value may depend on the operating system (e.g. **NaN**).

### Problems of Editing a Model

Errors of two major types can occur during model editing: Errors assigned to an edit field of the active dialog box and more general errors of incompatibility of the newly edited structure with the rest of the system definitions.

Errors assigned to an edit field can usually easily be identified. Figure 6.1 shows an example of such an error. The error message indicates an error in the edit field **Expression**.

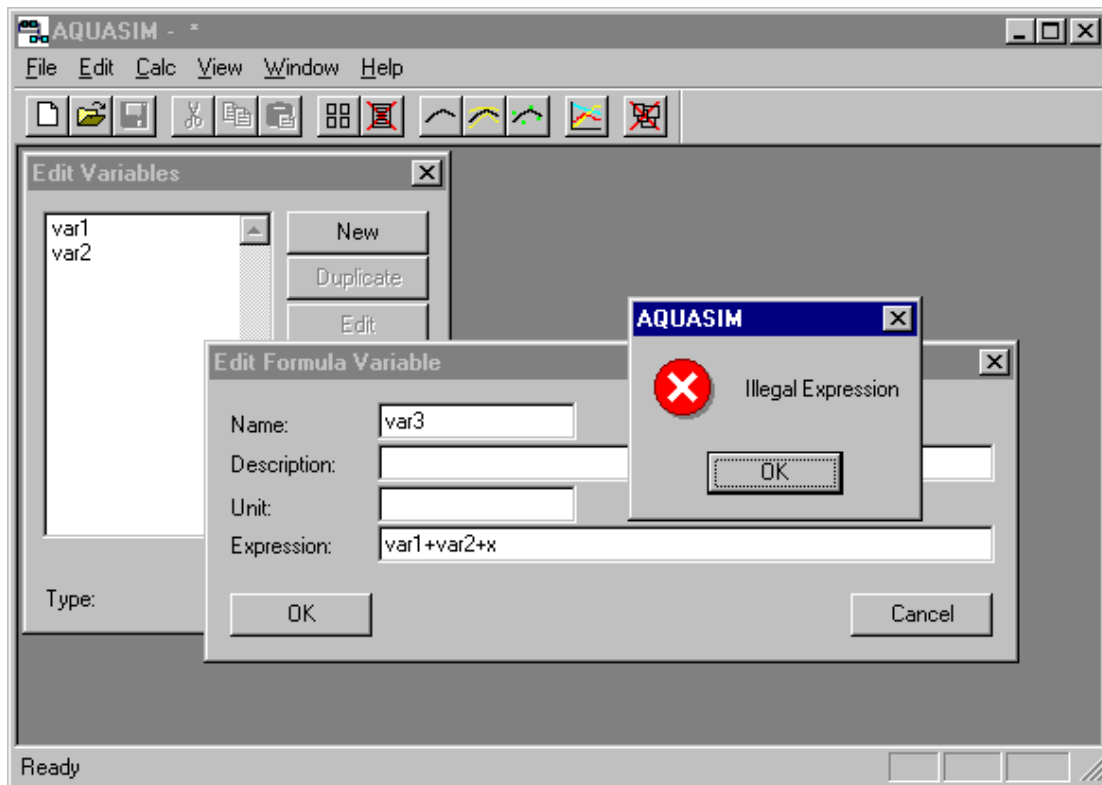


Figure 6.1: Example of an error assigned to an edit field of the dialog box.

If this error message occurs it is recommended to first check the existence of all variables used in the edit field and to check the correctness of the algebraic expression. In the present case (Fig. 6.1), as is seen in the dialog box **Edit Variables** in the background, the variable **x** used in the algebraic expression of the formula variable does not exist. If syntax and existence checks do not help uncovering the problem, it is recommended to carefully read the explaining text to the referenced edit field in the user manual. Besides errors in algebraic equations, illegal dependencies may occur. Examples of illegal dependencies are use of a program variable that is not meaningful in the context of the current compartment or the absence of obligatory dependencies.

Inconsistency errors of the newly defined object with existing system definitions are more difficult to find. In many cases, as shown in the example shown in Fig. 6.2, the name of the object may already exist. Also in this case, it is recommended to carefully read the comments to the dialog box in the user manual.



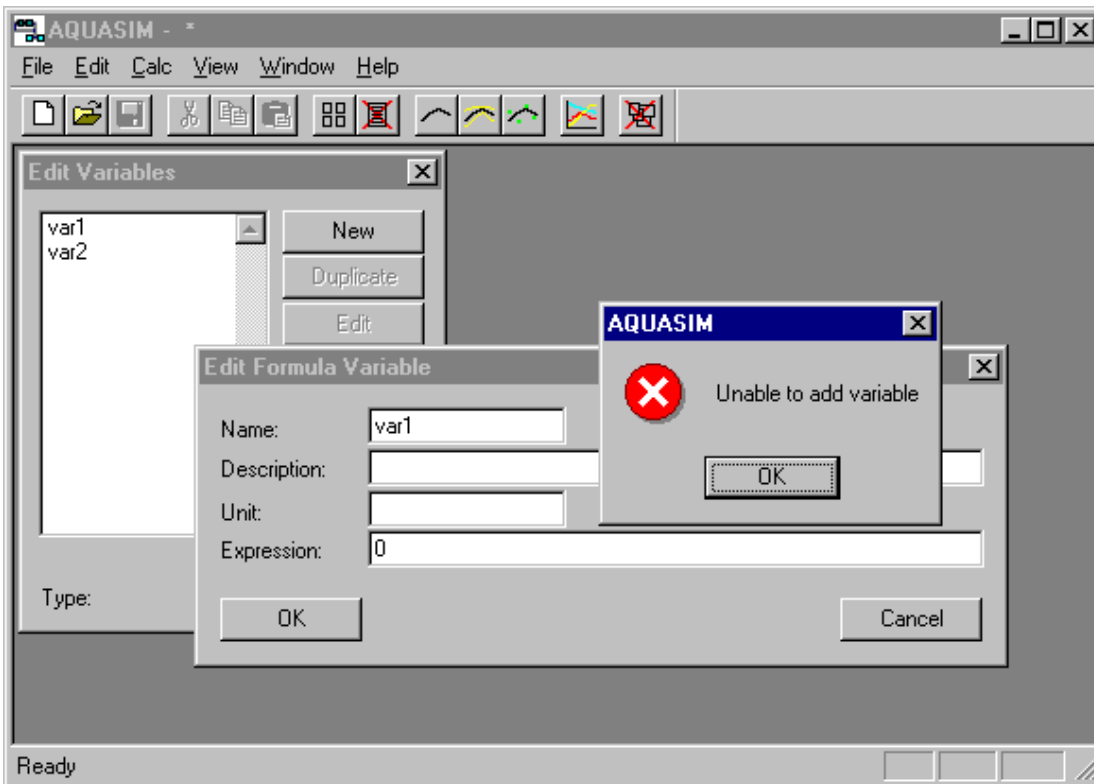


Figure 6.2: Example of an error assigned to an edit field of the dialog box.

## Problems During Calculation

### *The Log File of AQUASIM*

The most important source of information on problems during calculation is the log file of AQUASIM. For each interactive section, a log file `aquasim.log` is written to the program directory of AQUASIM (the log file from the previous session is overwritten). As explained in section 6.2 for batch jobs, the user can specify the name of the log file in the command line or in the job file. The contents of the log file can usually be ignored. However, it may contain useful information in case of problems during calculations. The following listing shows a typical log file written during an interactive simulation of the activated sludge example delivered with AQUASIM and described in section 8.2 of the report on the concepts of AQUASIM (Reichert, 1994b).

```
*****
AQUASIM Version 2.0 (win/mfc) - Log File
*****
```

```
Start of Session: 05/12/1998 20:58:43
```

```
Calculation of consistent initial condition
```

```
-----
```

```
Number of codiagonals considered:      1000
Time of initial condition:             -50
```

```
05/12/1998 22:03:04 Start of calculation
05/12/1998 22:03:04 End of calculation
```

```
Number of equations:                  50
Number of integers needed for calculation: 50
Number of reals needed for calculation: 5400
```

```
Number of steps taken:                0
Number of evaluations of the jacobian: 0
```

```
Dynamic calculation
```

```
-----
```

```
Number of codiagonals considered:      1000
Maximum internal step size:            1
```

```
05/12/1998 22:03:04 Start of calculation
05/12/1998 22:03:04 Integration at time -50
05/12/1998 22:03:07 Integration at time -49.98
05/12/1998 22:03:08 Integration at time -49.96
05/12/1998 22:03:08 Integration at time -49.94
05/12/1998 22:03:08 Integration at time -49.92
05/12/1998 22:03:08 Integration at time -49.9
```

```
.
.
```

```

05/12/1998 22:03:13 Integration at time 0.9
05/12/1998 22:03:13 Integration at time 0.92
05/12/1998 22:03:13 Integration at time 0.94
05/12/1998 22:03:13 Integration at time 0.96
05/12/1998 22:03:13 Integration at time 0.98
05/12/1998 22:03:13 Integration at time 1
05/12/1998 22:03:13 End of calculation

Number of equations:                50
Number of integers needed for calculation: 70
Number of reals needed for calculation: 2990

Current integration time:            1.0055
Current step size:                   0.00915549
Current order of integration:        3
Number of steps taken:               420
Number of function evaluations:       864
Number of evaluations of the jacobian: 48
Number of error test failures:        10
Number of convergence test failures:  2

```

First, information on the initialization process is given. The value of the numerical parameter, **Number of Codiagonals of the Jacobian Matrix** (cf. section 3.5), and the time at which the initial condition is evaluated, are given.

Then, start time and end time of the initialization procedure are given.

As a next group of parameters, the number of equations solved by the numerical algorithm and the memory requirements for the initialization process, expressed as a number of integer variables and a number of real variables, are listed.

The number of steps and the number of evaluations of the jacobian matrix required for the calculation of the initial state conclude the information on the initialization process.

Information on dynamic calculations starts with the values of the numerical parameters **Number of Codiagonals of the Jacobian Matrix** and **Maximum Internal Step Size** (cf. section 3.5).

The next block of information shows the time consumption of the simulation by giving the time at each output time step.

As a next group of parameters, the number of equations solved by the numerical algorithm and the memory requirements for the application of the differential-algebraic system solver DASSL (Petzold, 1983), expressed as a number of integer variables and a number of real variables, are listed.

The next block of information shows the current (at the end of the simulation) values of integration time, step size and integration order, gives a summary on the number of integration steps taken, of function evaluations and of evaluations of the jacobian matrix, and gives the numbers of error test failures and convergence test failures of the algorithm DASSL (Petzold, 1983). Especially these last two numbers may be interesting for simulation performance improvements. An error test failure indicates that the algorithm tries to perform a too large time step and it has to repeat the step with a smaller step size in order to fulfill the accuracy criteria of all state and program variables (cf. section 3.5). A change of the accuracies of state and program variables (cf. section 3.5) may decrease the number of error test failures. A convergence test failure indicates that the algorithm had to decrease the size of the time step because it was not able to find the solution of

the nonlinear system of algebraic equations for the solution after the time step (the iterative equation solver did not converge; cf. section 4.1). One cause for a large number of convergence test failures may be a too low value of the numerical parameter **Number of Codiagonals of the Jacobian Matrix**.

#### *Problems of Wrong Calculation Results*

Due to the robustness of the integration algorithm used by DASSL (Petzold, 1983) it is very rare that wrong simulation results occur. To our knowledge the only cases in which wrong calculation results occur are cases related to the automatic selection of integration time steps. If the integration time step is large in comparison to a short term change in system behaviour, the algorithm may step over such a change in behavior. The consequence of this problem is that the change in behaviour is not recognised by the calculation.

There are two very important examples of this problem. The first is related to short-term external excitation of a system, the second to short term changes in process rates due to changes in the values of variables calculated from the solution.

a) If the driving forces of a system (process rates and inputs) usually only vary slowly in time but show some short-term excitations, then, during the slow variation phase, the integration algorithm gradually increases the size of its time step. If then a short-term excitation with a duration shorter than the time step occurs, the algorithm may hit the excitation or it may step over the excitation. In the first case the algorithm recognises the integration problem and it follows the excitation accurately by decreasing the integration step size appropriately. In the second case the algorithm steps over the excitation without recognising it and the simulation result becomes erroneous. The simplest way of solving this problem is to decrease the value of the maximum internal step size of the integration algorithm to a value smaller than the duration of the excitation as described in section 3.5. If this slows down the integration too much during the smooth behaviour periods, an alternative may be to decrease the output time step to a value smaller than one tenth of the duration of the excitation during the excitation only (this also leads to better results for the plots), and let the output time step to be large during the smooth periods. This also works because the size of the internal time step is not allowed to increase to values larger than ten times the output time step.

b) If changes in reaction rates may occur abruptly as a function of a variable calculated by the program (e.g. a calculated concentration), a similar problem as described under a) may occur. The difference of the present problem to the one described above is that the time at which it occurs is unknown. This makes the second solution procedure proposed above inapplicable (decreasing the time step only in the critical ranges of time). The first solution (decreasing the internal time step) may improve the results, however, the appropriate size of the time step is more difficult to estimate, because it depends on the time scale of changes of the values of system variables which are calculated by the program. For this reason and to make the integration faster, it is better to solve this problem by changing the formulation of the rate expressions in a way that makes the problem detectable by the integration algorithm. This procedure is illustrated with a simple example: A rate expression of Monod type

$$r = \frac{r_{max}C}{K + C} \quad , \quad (6.1)$$

where  $r$  is the degradation rate of the substance described by the concentration  $C$ ,  $r_{max}$  is the maximum degradation rate and  $K$  is the concentration at which the degradation

rate is half of its maximum value, may cause such a problem if the value of  $K$  is very small in comparison to typical initial values of  $C$  and if degradation continues to zero. For large concentrations ( $C \gg K$ ) the degradation rate is approximately equal to  $r_{max}$ , for decreasing concentrations it tends to zero. However, for large negative concentrations, the degradation rate is also equal to  $r_{max}$ . If now the integration algorithm makes huge time steps because the decrease of  $C$  with a constant rate  $r_{max}$  is very simple to integrate, the concentration may jump over the whole range of small and negative rates to large negative values of  $C$ , where the rate is again approximately equal to  $r_{max}$ , and the degradation process continues (to larger negative values of  $C$ ). This problem only occurs if typical concentrations  $C$  are much larger than  $K$  and a fast decrease of  $C$  to zero is possible. An elegant way to solve this problem is to extend the function given above in a different way to negative concentrations (to avoid the problem that again  $r_{max}$  is returned). If the rate is extended linearly by

$$r = \begin{cases} \frac{r_{max}C}{K+C} & \text{for } C \geq 0 \\ \frac{r_{max}}{K}C & \text{for } C \leq 0 \end{cases} \quad (6.2)$$

the problem is avoided without changing the function in the domain of (reasonable) positive concentrations  $C$ . If now the algorithm steps to negative concentrations, it obtains negative degradation rates that differ significantly from the positive value of  $r_{max}$ . Such a rapid change in the rate during a single time step is not acceptable for the error test criterion and the step size is decreased. For this reason, the correct smooth approach to zero is achieved and negative concentrations do not occur in the final model results (they may only occur during integration steps that are not accepted).

In most cases, errors in the user input are the cause for wrong simulation results. It is advantageous to print out the system definitions and to check these definitions carefully in order to find the cause of the problem. In addition it may be advantageous to plot important system variables that do not show correct behaviour and to turn off processes one after the other in order to find out which process caused the problem. Then this process can be studied in more detail.

#### *Problems of Integration Interrupts*

If the integration algorithm detects a serious problem that cannot be solved by reducing the integration step size and order, the integration process is interrupted with the message box shown in Fig. 6.3. If this dialog box appears, in most cases there will be a message on

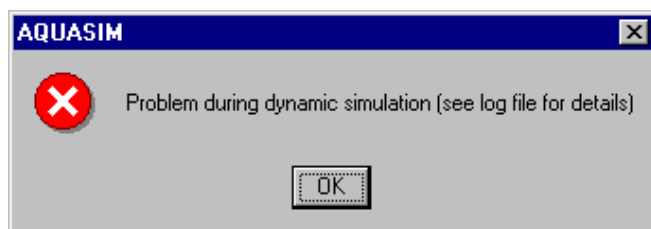


Figure 6.3: Example of a dialog box indicating a numerical problem.

the log file of AQUASIM that may give hints with respect to the nature of the problems.

Two types of error should be distinguished: Error messages by the integration algorithm DASSL (Petzold, 1983) which start with the keyword ‘DASSL’ and error messages by AQUASIM. Some examples of such error messages are discussed in the following two subsections.

#### DASSL Errors

DASSL errors are more difficult to interpret than AQUASIM errors because no specific reference to AQUASIM system definitions can be given.

The DASSL error message

```
05/13/1998 15:27:39 Start of calculation
05/13/1998 15:27:39 Integration at time 0
DASSL- AT CURRENT T = 0.00990773 1000 STEPS
DASSL- TAKEN ON THIS CALL BEFORE REACHING TOUT.
05/13/1998 15:27:39 End of calculation
```

indicates that more time steps are required to reach the next point of output time than was specified as the value of the numerical parameter **Maximum Number of Internal Time Steps for One External Time Step** (in the present example, the value of this parameter was set equal to 1000; cf. section 3.5). This limitation of the number of internal steps for one external step was made in order to show numerical problems without losing too much computing time. Under usual circumstances the requirement for more than 1000 internal steps for one external step indicates the existence of a numerical problem. There are two important exceptions from this general rule: If a dynamic simulation starts with a relaxation to an equilibrium state and this is done with one output step only (because the relaxation dynamics is not of interest), a large number of internal steps may be required in order to calculate the single external output step. The message shown above may also appear during a parameter estimation, for which a simulation could be run without problems. This is the case if the temporal distance between data points is much larger than the output interval selected for the simulation. For such cases it is necessary to increase the numerical parameter **Maximum Number of Internal Time Steps for One External Time Step** as discussed in section 3.5. In other cases it may be more appropriate to try to find out why the integration algorithm performs poorly (with very small time steps).

The DASSL error message that is most difficult to interpret is the following:

```
05/13/1998 15:32:44 Start of calculation
05/13/1998 15:32:44 Integration at time 0
05/13/1998 15:32:44 Integration at time 0.1
05/13/1998 15:32:44 Integration at time 0.2
05/13/1998 15:32:44 Integration at time 0.3
05/13/1998 15:32:44 Integration at time 0.4
05/13/1998 15:32:44 Integration at time 0.5
05/13/1998 15:32:44 Integration at time 0.6
05/13/1998 15:32:44 Integration at time 0.7
05/13/1998 15:32:44 Integration at time 0.8
05/13/1998 15:32:44 Integration at time 0.9
DASSL- AT T = 1 AND STEPSIZE H = 2.06912e-024
DASSL- THE ERROR TEST FAILED REPEATEDLY
DASSL- OR ABS(H) = HMIN.
05/13/1998 15:32:44 End of calculation
```

This or a similar error message results if the algorithm repeatedly tried to reduce the integration step size without being able to fulfill the error test criterion. It is then the difficult task of the program user to find the cause of the numerical problem. The point in time at which the problem occurs may help to find the cause. A rate or an input could be discontinuous at this point in time. It may be useful to look at time series plots of important system variables in order to detect strange behaviour at this point in time. It may also be useful to change the accuracy of program and state variables in order to find out if the problem occurs always at the same time (cf. section 3.5).

#### AQUASIM Errors

In most cases it is easier to find the cause for AQUASIM error messages. The following example shows an error message that occurs when in the advective-diffusive compartment a negative diffusion coefficient is specified:

```
05/13/1998 15:44:46 Start of calculation
ADVCOMP numerical problem: Diffusion coefficient is not positive
05/13/1998 15:44:46 End of calculation
```

If the error message is not clear enough, it is recommended to read the section of the user manual in which the edit field of the referenced quantity is described.

### 6.3.2 Finding Help in the AQUASIM User Group

In order to facilitate communication on program use and application among AQUASIM users, an electronic user group is maintained at the EAWAG mail server.

To join the AQUASIM user group, send the mail message

```
subscribe aquasim-users <mail-address>
```

to

```
majordomo@eawag.ch
```

In the mail message `<mail-address>` must be replaced by the mail address that should be added to the user group. Because the mail is processed automatically, it is very important that the mail message has exactly the format given above and does not include any further comments. The text must be given in the body of the message and not as a subject. In order to leave the AQUASIM user group send the mail message

```
unsubscribe aquasim-users <mail-address>
```

to

```
majordomo@eawag.ch
```

Your mail address is then deleted from the user group list.

Questions to be distributed or responses to distributed questions (that should again be distributed) must then be sent to

```
aquasim-users@eawag.ch
```

All messages sent to this address are automatically distributed to all members of the list. Usually, it is recommended to send your responses to distributed questions also to the user group in order to allow all user group members to participate in the discussion. However, individual answers to the person who sent out the question, are also possible.



### 6.3.3 Reporting Program Bugs and Suggestions for Improvements

Please report program errors as well as suggestions for program improvements to Peter Reichert, EAWAG, CH-8600 Dübendorf, Switzerland, or by email to `peter.reichert@eawag.ch`. An accurate description of the error accompanied by an AQUASIM system file which leads to the problem is very important.

We will provide information on known program errors, on workarounds and on bug-fixed program versions on the EAWAG home page at <http://www.eawag.ch>.

# Bibliography

- Albrecht, A., Reichert, P., Beer, J., and Lück, A. (1995). Evaluation of the importance of reservoir sediments as sinks for reactor-derived radionuclides in riverine systems. *J. Environ. Radioactivity*, 28(3):239–269.
- Amorocho, J. and de Vries, J. (1980). A new evaluation of the wind stress coefficient over water surface. *J. Geophys. Res.*, 85:433–442.
- Arcangeli, J.-P. and Arvin, E. (1997a). Modelling of the cometabolic biodegradation of trichloroethylene by toluene-oxidizing bacteria in a biofilm system. *Environmental Science & Technology*, 31:3044–3052.
- Arcangeli, J.-P. and Arvin, E. (1997b). Modelling of the growth of a methanotrophic biofilm. *Water Science and Technology*, 36(1):199–204.
- Beaudoin, D., Bryers, J., Cunningham, A., and Peretti, S. (1997). Mobilization of broad host range plasmid from *Pseudomonas putida* to established biofilms of *Bacillus azotiformans*. II. modeling. *Biotechnology & Bioengineering*, 57(3):280–286.
- Bührer, H. and Ambühl, H. (1975). Die Einleitung von gereinigtem Abwasser in Seen. *Schweizerische Zeitschrift für Hydrologie*, 37:347–369.
- Burchard, H. and Baumert, H. (1995). On the performance of a mixed-layer model based on the  $k$ - $\epsilon$  turbulence closure. *J. Geophys. Res.*, 100(C5):8523–8540.
- Chow, V. (1959). *Open-channel hydraulics*. McGraw-Hill, New York.
- de St. Venant, M. (1871). Théorie du mouvement non permanent des eaux crues des rivières et à l'introduction des marées dans leur lit. *Comptes Rendus*, 73:147–154 & 237–240.
- Eisenbeis, M., Bauer-Kreisel, P., and Scholz-Muramatsu, H. (1997). Studies of the dechlorination of tetrachloroethene to *cis*-1,2-dichloroethene by *dehalospirillum multivorans* in biofilms. *Water Science and Technology*, 36(1):191–198.
- Fesch, C., Lehmann, P., Haderlein, S., Hinz, C., Schwarzenbach, R., and Flühler, H. (1998a). Effect of water content on solute transport in aggregated porous media. *J. Contaminant Hydrology*, 33(1-2):211–230.
- Fesch, C., Simon, W., Reichert, P., Haderlein, S., and Schwarzenbach, R. (1998b). Non-linear sorption and nonequilibrium transport of organic contaminants in saturated porous media: experiments, process identification and modeling. *J. Contaminant Hydrology*, 31(3-4):373–407.

- Filipe, C. and Daigger, G. (1997). Development of a revised metabolic model for the growth of phosphorus-accumulating organisms. *Water Environment Research*, 70(1):67–79.
- Fischer, H., List, E., Koh, C., Imberger, J., and Brooks, N. (1979). *Mixing in Inland and Coastal Waters*. Academic Press, New York.
- French, R. (1985). *Open-channel hydraulics*. McGraw-Hill, Stuttgart. New York.
- Gear, C. (1971a). Algorithm 407, DIFSUB for solution of ordinary differential equations. *Communications ACM*, 14(3):185–190.
- Gear, C. (1971b). The automatic integration of ordinary differential equations. *Communications ACM*, 14(3):176–179.
- Gear, C. (1971c). *Numerical initial value problems in ordinary differential equations*. Prentice Hall, Stuttgart. Englewood Cliffs, N.J.
- Gill, P., Murray, W., and Wright, M. (1981). *Practical Optimization*. Academic Press, London.
- Glod, G., Angst, W., Holliger, C., and Schwarzenbach, R. (1997a). Corrinoid-mediated reduction of Tetrachloroethene, Trichloroethene, and Trichlorofluoroethene in homogeneous aqueous solution: Reaction kinetics and reaction mechanisms. *Environ. Sci. Technol.*, 31:253–260.
- Glod, G., Brodmann, U., Angst, W., Holliger, C., and Schwarzenbach, R. (1997b). Cobalamin-mediated reduction of *cis*- and *trans*-dichloroethene, 1,1-dichloroethene, and vinyl chloride in homogeneous aqueous solution: reaction kinetics and mechanistic considerations. *Environ. Sci. Technol.*, 31:3154–3160.
- Goudsmit, G.-H., Reichert, P., and Wüest, A. (1996). Modelling of physical and biogeochemical properties in lakes using AQUASIM. In Müller, A., editor, *Hydroinformatics 96*, pages 779–786. Balkema, Rotterdam.
- Gujer, W. and Wanner, O. (1990). Modeling mixed population biofilms. In Characklis, W. and Marshall, K., editors, *Biofilms*, pages 397–443. John Wiley & Sons, New York.
- Henderson, F. (1966). *Open channel flow*. Macmillan, New York.
- Henze, M., Grady, C., Gujer, W., Marais, G., and Matsuo, T. (1986). Activated sludge model no. 1. Scientific and Technical Report 1, IAWPRC Task Group on Mathematical Modelling for Design and Operation of Biological Wastewater Treatment Processes, IAWPRC, London.
- Hindmarsh, A. (1983). ODEPACK. a systematized collection of ODE solvers. In Stepleman, R., editor, *Scientific Computing*, pages 55–64. IMACS / North-Holland.
- Horn, H. and Hempel, D. (1995). Mass transfer coefficients for an autotrophic and a heterotrophic biofilm system. *Wat. Sci. Tech.*, 32(8):199–204.
- Horn, H. and Hempel, D. (1997). Substrate utilization and mass transfer in an autotrophic biofilm system: Experimental results and numerical simulation. *Biotechnology and Bioengineering*, 53(4):363–371.

- Jancarkova, I., Larsen, T., and Gujer, W. (1997). Distribution of nitrifying bacteria in a shallow stream. *Water Science and Technology*, 36(8-9):161–166.
- Janning, K., Harremoës, P., and Nielsen, M. (1995). Evaluating and modelling the kinetics in a full scale submerged denitrification filter. *Wat. Sci. Tech.*, 32(8):115–123.
- Kuba, T., Murnleitner, E., van Loosdrecht, M., and Heijnen, J. (1996). A metabolic model for biological phosphorus removal by denitrifying organisms. *Biotechnology and Bioengineering*, 52(6):685–695.
- LeVeque, R. J. (1990). *Numerical Methods for Conservation Laws*. Birkhäuser, Basel.
- Londong, J., Borchardt, D., Firk, W., Reichert, P., Stein, M., and Strotmann, U. (1994). Nitrit in Fliessgewässern. *Korrespondenz Abwasser*, 11/94:2069–2076.
- Maryns, F. and Bauwens, W. (1997). The application of the activated sludge model no. 1 to a river environment. *Water Science and Technology*, 36(5):201–208.
- Mirpuri, R., Sharp, W., Villaverde, S., Jones, W., Lewandowski, Z., and Cunningham, A. (1997). Predictive model for Toluene degradation and microbial phenotypic profiles in flat plate vapor phase bioreactor. *Journal of Environmental Engineering*, 6/97:586–592.
- Murnleitner, E., Kuba, T., van Loosdrecht, M., and Heijnen, J. (1997). An integrated metabolic model for the aerobic and denitrifying biological phosphorus removal. *Biotechnology and Bioengineering*, 54(5):434–450.
- Musvoto, E., Wentzel, M., Loevenhals, R., and Ekama, G. (1997). Kinetic-based model for mixed weak acid/base systems. *Water SA*, 23(4):311–322.
- Nelder, J. and Mead, R. (1965). A simplex method for function minimization. *Computer Journal*, 7:308–313.
- Novack, B. and Sigg, L. (1997). Dissolution of Fe(II)(hydr)oxides by metal-EDTA complexes. *Geochimica et Cosmochimica Acta*, 61(5):951–963.
- Peeters, F., Kipfer, R., Hohmann, R., Hofer, M., Imboden, D., Kodenev, G., and Khozder, T. (1997). Modeling transport rates in lake Baikal: Gas exchange and deep water renewal. *Environ. Sci. Technol.*, 31:2973–2982.
- Petersen, E. (1965). *Chemical Reaction Analysis*. Prentice-Hall, Englewood Cliffs, New Jersey.
- Petzold, L. (1983). A description of DASSL: A differential/algebraic system solver. In Stepleman, R. e., editor, *Scientific Computing*, pages 65–68. IMACS / North-Holland, Amsterdam.
- Ralston, M. and Jennrich, R. (1978). DUD - a derivative-free algorithm for nonlinear least squares. *Technometrics*, 20(1):7–14.
- Reichert, P. (1994a). AQUASIM - A tool for simulation and data analysis of aquatic systems. *Wat. Sci. Tech.*, 30(2):21–30.

- Reichert, P. (1994b). Concepts underlying a computer program for the identification and simulation of aquatic systems. Schriftenreihe der EAWAG 7, Swiss Federal Institute for Environmental Science and Technology (EAWAG), CH-8600 Dübendorf, Switzerland.
- Reichert, P. (1995). Design techniques of a computer program for the identification of processes and the simulation of water quality in aquatic systems. *Environmental Software*, 10(3):199–210.
- Reichert, P. (1998). AQUASIM 2.0 – Tutorial. Technical report, Swiss Federal Institute for Environmental Science and Technology (EAWAG), CH-8600 Dübendorf, Switzerland.
- Reichert, P., von Schulthess, R., and Wild, D. (1995). The use of AQUASIM for estimating parameters of activated sludge models. *Wat. Sci. Tech.*, 31(2):135–147.
- Reichert, P. and Wanner, O. (1997). Movement of solids in biofilms: Significance of liquid phase transport. *Water Science and Technology*, 36(1):321–328.
- Rodi, W. (1980). Turbulence models and their applications in hydraulics. State of the art paper, International Association for Hydraulic Research (IAHR), Delft, Netherlands.
- Rodi, W. (1987). Examples of calculation methods for flow and mixing in stratified flows. *J. Geophys. Res.*, 92:5305–5328.
- Sanderson, S. and Stewart, P. (1997). Evidence of bacterial adaptation to monochloramine in *Pseudomonas aeruginosa* biofilms and evaluation of biocide action model. *Biotechnology & Bioengineering*, 56(2):201–209.
- Shamir, U. and Harleman, D. (1967). Numerical solutions for dispersion in porous mediums. *Water Resources Research*, 3(2):557–581.
- Siegrist, H., Krebs, P., Bühler, R., Purtschert, I., Röck, C., and Rufer, R. (1995). Denitrification in secondary clarifiers. *Wat. Sci. Tech.*, 31(2):205–214.
- Simon, W., Reichert, P., and Hinz, C. (1997). Properties of exact and approximate travelling wave solutions for transport with nonlinear and nonequilibrium sorption. *Water Resources Research*, 33(5):1139–1147.
- Suci, P., Vraný, J., and Mittelman, M. (1998). Investigation of interactions between antimicrobial agents and bacterial biofilms using attenuated total reflection Fourier transform infrared spectroscopy. *Biomaterials*, 19:327–339.
- Sweby, P. (1984). High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM J. Numer. Anal.*, 21(5):995–1011.
- Uehlinger, U., Bühner, H., and Reichert, P. (1996). Periphyton dynamics in a floodprone prealpine river: Evaluation of significant processes by modelling. *Freshwater Biology*, 36:249–263.
- Ulrich, M., Imboden, D., and Schwarzenbach, R. (1995). MASAS - A user-friendly simulation tool for modeling the fate of anthropogenic substances in lakes. *Environmental Software*, 10(3):177–198.

- van Leer, B. (1974). Towards the ultimate conservative difference scheme ii: Monotonicity and conservation combined in a second order scheme. *Journal of Computational Physics*, 14:361–370.
- von Gunten, U., Elovitz, M., and Kaiser, H. (1997). Characterization of ozonation processes with conservative and reactive tracers: Prediction of the degradation of micropollutants. *Analysis Magazine*, 25(7):29–31.
- von Schulthess, R. and Gujer, W. (1996). Release of nitrous oxide from denitrifying activated sludge: Verification and application of a mathematical model. *Water Research*, 30(3):521–530.
- von Schulthess, R., Wild, D., and Gujer, W. (1994). Nitric and nitrous oxides from denitrifying activated sludge at low oxygen concentration. *Wat. Sci. Tech.*, 30(6):123–132.
- Vrany, J., Stewart, P., and Suci, P. (1997). Comparison of Recalcitrance to Ciprofloxacin and Levofloxacin exhibited by *Pseudomonas aeruginosa* biofilms displaying rapid-transport characteristics. *Antimicrobial Agents and Chemotherapy*, 41(6):1352–1358.
- Wanner, O. (1994). Modeling of mixed-population biofilm accumulation. In Geesey, G., Levandowski, Z., and Flemming, H.-C., editors, *Biofouling and Biocorrosion in Industrial Water Systems*, pages 37–62. Lewis Publishers, Boca Raton.
- Wanner, O. (1995). New experimental findings and biofilm modelling concepts. *Wat. Sci. Tech.*, 32:133–140.
- Wanner, O. (1996). Modelling of biofilms. *Biofouling*, 10(1-3):31–41.
- Wanner, O., Cunningham, A., and Lundman, R. (1995). Modeling biofilm accumulation and mass transport in a porous medium under high substrate loading. *Biotechnology & Bioengineering*, 47:703–712.
- Wanner, O., Debus, O., and Reichert, P. (1994). Modelling the spatial distribution and dynamics of a xylene-degrading microbial population in a membrane-bound biofilm. *Wat. Sci. Tech.*, 29(10-11):243–251.
- Wanner, O. and Gujer, W. (1984). Competition in biofilms. *Wat. Sci. Tech.*, 17:27–44.
- Wanner, O. and Gujer, W. (1986). A multispecies biofilm model. *Biotechnology & Bioengineering*, 28:314–328.
- Wanner, O. and Reichert, P. (1996). Mathematical modelling of mixed-culture biofilms. *Biotechnology & Bioengineering*, 49:172–184.
- Wild, D., von Schulthess, R., and Gujer, W. (1994). Synthesis of denitrification enzymes in activated sludge: Modelling with structured biomass. *Wat. Sci. Tech.*, 30(6):113–122.
- Wild, D., von Schulthess, R., and Gujer, W. (1995). Structured modelling of denitrification intermediates. *Wat. Sci. Tech.*, 31(2):45–54.
- Yen, B. (1973). Open-channel flow equations revisited. *Journal of the Mechanical Engineering Division, ASCE*, 99:979–1009.

- Yen, B. (1979). Unsteady flow mathematical modeling techniques. In Shen, H., editor, *Modeling of Rivers*, pages 13–1–13–33. John Wiley, New York.

# Index

- about
  - file menu, 8
- abs
  - formula variable, 24
- abscissa
  - plot, 163
- absolute accuracy
  - program variable
    - advective-diffusive reactor compartment, 67
    - biofilm reactor compartment, 57
    - lake compartment, 123
    - mixed reactor compartment, 38
    - river section compartment, 96
    - saturated soil column compartment, 84
  - state variable, 14, 134
- accuracy
  - program variable, 132, 134, 142
    - advective-diffusive reactor compartment, 62, 67
    - biofilm reactor compartment, 51, 56
    - lake compartment, 110, 123
    - mixed reactor compartment, 35, 38
    - river section compartment, 91, 96
    - saturated soil column compartment, 76, 83
  - state variable, 14, 132, 134, 142
- acos
  - formula variable, 24
- active
  - for calculation
    - advective-diffusive reactor compartment, 62
    - biofilm reactor compartment, 51
    - lake compartment, 123
    - mixed reactor compartment, 35
    - river section compartment, 91
    - saturated soil column compartment, 76
  - for parameter estimation
    - calculation, 154
    - constant variable, 17, 153
  - for sensitivity analysis
    - calculation, 143
    - constant variable, 17, 147
    - real list variable, 21, 147
  - for simulation
    - calculation, 143
- active for parameter estimation
  - calculation, 154
  - constant variable, 17, 153
- active for sensitivity analysis
  - calculation, 143
  - constant variable, 17, 147
  - real list variable, 21, 147
- active for simulation
  - calculation, 143
- active processes
  - advective-diffusive reactor compartment, 62, 63
  - biofilm reactor compartment, 49, 52
  - lake compartment, 108, 111
  - mixed reactor compartment, 35, 36
  - river section compartment, 89, 92
  - saturated soil column compartment, 75, 77
- active state variables
  - advective-diffusive reactor compartment, 62
  - biofilm reactor compartment, 49, 51
  - lake compartment, 108, 110
  - mixed reactor compartment, 35
  - river section compartment, 89, 91
  - saturated soil column compartment, 75, 76
- add argument-value pairs
  - variable list variable, 22
- add data pairs
  - real list variable, 21



- advective link, 10, 125–129
  - bifurcation, 126, 127
  - description, 128
  - name, 128
  - substance flow, 126, 128
  - water flow, 126, 128
- connection, 127
- description, 127
- equations, 126–127
- link index, 127
- name, 127
- overview, 126
- state variable, 126, 128, 129
  - dynamic, 126, 128, 129
  - volume, 126, 128, 129
- substance flow, 126
- user definitions, 127–129
- water flow, 126
- advective-diffusive reactor compartment, 10, 32, 59–69
  - accuracy of program variables, 62, 67
  - active for calculation, 62
  - active processes, 62, 63
  - active state variables, 62
  - area, 59, 62
  - boundary condition, 61
  - compartment index, 62
  - cross section, 62
  - description, 62
  - diffusion coefficient, 60, 62, 68
  - discharge, 60
  - dispersion coefficient, 60, 62
  - end coordinate, 62
  - equations, 59–61
  - initial conditions, 62, 63
  - inlet input, 64, 65
  - input, 62, 64
    - inlet, 64, 65
    - lateral, 64, 66
  - lateral inflow, 60
  - lateral input, 64, 66
  - name, 61
  - number of grid points, 62
  - overview, 59
  - resolution, 62
  - space coordinate x, 62
  - start coordinate, 62
  - state variable, 59, 62, 64, 65, 67
    - dynamic, 59, 65, 67
    - equilibrium, 59
    - surface, 59
    - volume, 59, 65, 67
  - user definitions, 61–69
- applications of AQUASIM, 1
- aquasim.log, 8, 137, 181–183
- area
  - advective-diffusive reactor compartment, 59, 62
  - biofilm reactor compartment, 42, 50
  - lake compartment, 99, 109
  - river section compartment, 85, 89
  - saturated soil column compartment, 71, 76
- area gradient
  - lake compartment
    - program variable, 124
  - program variable, 17
- argument
  - real list variable, 18, 20
  - variable list variable, 22
- asin
  - formula variable, 24
- atan
  - formula variable, 24
- attachment
  - biofilm reactor compartment, 41
- attachment coefficient
  - surface
    - biofilm reactor compartment, 46, 56
  - volume
    - biofilm reactor compartment, 44, 56
- attachment velocity
  - biofilm reactor compartment, 45, 46
- attachment velocity of biofilm
  - biofilm reactor compartment
    - program variable, 58
  - program variable, 16
- batch version, 3, 4, 8, 173–175
- bifurcation
  - advective link, 126, 127
- biofilm area
  - biofilm reactor compartment, 50
- biofilm matrix
  - diffusive
    - biofilm reactor compartment, 45, 50

- rigid
  - biofilm reactor compartment, 46, 50
- biofilm reactor compartment, 10, 32, 41–58
  - accuracy of program variables, 51, 56
  - active for calculation, 51
  - active processes, 49, 52
  - active state variables, 49, 51
  - area, 42, 50
  - attachment, 41
  - attachment coefficient
    - surface, 46, 56
    - volume, 44, 56
  - attachment velocity, 45, 46
  - biofilm area, 50
  - biofilm matrix
    - diffusive, 45, 50
    - rigid, 46, 50
  - biofilm thickness, 41, 45
  - boundary condition, 46, 47
  - bulk volume, 41, 48, 50
  - compartment index, 49
  - confined reactor, 48, 50
  - density, 56
  - description, 49
  - detachment, 41
    - global velocity, 50
    - individual rate, 50
  - detachment coefficient
    - surface, 45, 56
    - volume, 44, 56
  - detachment velocity, 45
  - diffusion coefficient
    - biofilm matrix, 56
    - pore volume, 56
  - diffusive solid matrix, 50
  - dissolved substances, 41
  - dissolved variables, 49, 56
  - equations, 41–48
  - free volume
    - growth rate of, 43
  - initial conditions, 49, 52
  - input, 49, 53
  - liquid phase, 50
  - liquid phase volume fraction, 42
  - mass transfer resistance, 48, 56
  - name, 49
  - number of grid points, 51
  - overview, 41
  - particulate variables, 49, 55
  - pore volume, 41
  - porosity, 42
    - rate of change, 51
  - reactor type, 50
  - reactor volume, 48, 50
  - resolution, 51
  - rigid solid matrix, 50
  - solid matrix, 41
  - space coordinate  $z$ , 41
  - state variable, 42, 49–54
    - dynamic, 42, 49, 54
    - volume, 42, 49, 54
  - substratum, 41
  - surface attachment coefficient, 56
  - surface detachment coefficient, 56
  - suspended solids, 41, 50
  - unconfined reactor, 48, 50
  - user definitions, 49–58
  - velocity of biofilm matrix, 43
  - volume
    - bulk, 48
    - reactor, 48
  - volume attachment coefficient, 56
  - volume detachment coefficient, 56
  - zones, 41
- biofilm thickness
  - biofilm reactor compartment, 41, 45
    - program variable, 53, 57, 58
  - program variable, 16
- bottom coordinate
  - lake compartment, 109
- bottom friction
  - lake compartment, 103, 123
- boundary condition
  - advective-diffusive reactor compartment, 61
  - biofilm reactor compartment, 46, 47
  - lake compartment, 106, 107
  - river section compartment, 87, 88
  - saturated soil column compartment, 74
- Brunt Vaisälä frequency
  - lake compartment, 109, 110
    - program variable, 110, 112, 124
  - program variable, 17
- bulk volume

- biofilm reactor compartment, 41, 48, 50
    - program variable, 57, 58
  - mixed reactor compartment
    - program variable, 39
  - program variable, 16
- buoyancy production of TKE
  - lake compartment, 101, 103
    - program variable, 124
  - program variable, 17
- calc menu, 5, 137–160
- calculation, 148
  - parameter estimation, 155–156
    - calculation number, 155
    - description, 155
    - initial state, 155
    - initial time, 155
    - name, 155
  - sensitivity analysis, 140–143
    - calculation number, 141
    - description, 141
    - initial state, 141
    - initial time, 141
    - name, 140, 141
    - number of steps, 142
    - output steps, 142
    - step size, 142
  - simulation, 140–143
    - description, 141
    - initial state, 141
    - initial time, 141
    - name, 140, 141
    - number of steps, 142
    - output steps, 142
    - step size, 142
- calculation number
  - advective link
    - program variable, 128
  - advective-diffusive reactor compartment
    - program variable, 69
  - biofilm reactor compartment
    - program variable, 58
  - calculation
    - parameter estimation, 155
    - sensitivity analysis, 141
    - simulation, 141
  - curve, 165
  - lake compartment
    - program variable, 113, 124
  - mixed reactor compartment
    - program variable, 39
  - program variable, 15, 16
  - river section compartment
    - program variable, 97
  - saturated soil column compartment
    - program variable, 84
- character interface version, 3, 4, 171–172
- close
  - file menu, 7
- column number of argument
  - real list variable, 21
- column number of standard deviations
  - real list variable, 21
- column number of values
  - real list variable, 21
- compartment, 10, 32–124
  - advective-diffusive reactor, 10, 32, 59–69
    - biofilm reactor, 10, 32, 41–58
    - curve, 166
    - fit target, 157
    - lake, 10, 32, 98–124
    - mixed reactor, 10, 32, 34–40
    - probe variable, 26
    - river section, 10, 32, 85–97
    - saturated soil column, 10, 32, 70–84
- compartment index
  - advective-diffusive reactor compartment, 62
    - program variable, 62, 69
  - biofilm reactor compartment, 49
    - program variable, 49, 58
  - lake compartment, 108
    - program variable, 108, 124
  - mixed reactor compartment, 34
    - program variable, 34, 39
  - program variable, 16
  - river section compartment, 88
    - program variable, 88, 97
  - saturated soil column compartment, 75
    - program variable, 75, 84
- compartment type, 10, 32
- compartments

- edit menu, 11
- computing platforms, 4
- conductivity
  - lake compartment, 109
- confined reactor
  - biofilm reactor compartment, 48, 50
- connection
  - advective link, 127
- constant variable, 10, 12, 17, 21, 151
  - active for parameter estimation, 17, 153
  - active for sensitivity analysis, 17
  - description, 17
  - maximum, 17
  - minimum, 17
  - name, 17
  - standard deviation, 17
  - unit, 17
  - value, 17
- constant volume
  - mixed reactor compartment, 35
- continue
  - simulation, 141
- conversion factor
  - diffusive link, 129, 131
  - saturated soil column compartment, 83
- correlation coefficients, 144
- cos
  - formula variable, 24
- cosh
  - formula variable, 24
- critical water level
  - river section compartment, 91
- cross section
  - advective-diffusive reactor compartment, 62
  - lake compartment, 109
  - river section compartment, 89
  - saturated soil column compartment, 76
- cross-sectional area
  - advective-diffusive reactor compartment
    - program variable, 62, 69
  - lake compartment
    - program variable, 100, 124
  - program variable, 16
- river section compartment
  - program variable, 96, 97
- saturated soil column compartment
  - program variable, 76, 84
- curve
  - plot, 163
- DASSL, 132, 133, 139, 182, 183, 185
- data
  - fit target, 156
- data pairs
  - real list variable, 18
- deg
  - formula variable, 24
- delete argument-value pairs
  - variable list variable, 22
- delete data pairs
  - real list variable, 21
- delete states, 135–136
  - edit menu, 11
- density
  - biofilm reactor compartment, 56
  - lake compartment, 109, 119
    - program variable, 124
  - program variable, 17
  - saturated soil column compartment, 74
- description
  - advective link, 127
  - bifurcation, 128
  - advective-diffusive reactor compartment, 62
  - biofilm reactor compartment, 49
- calculation
  - parameter estimation, 155
  - sensitivity analysis, 141
  - simulation, 141
- constant variable, 17
- diffusive link, 130
- dynamic process, 29
- equilibrium process, 31
- formula variable, 23
- immobile region
  - saturated soil column compartment, 82
- lake compartment, 108
- mixed reactor compartment, 35
- plot, 163

- probe variable, 26
- program variable, 15
- real list variable, 20
- river section compartment, 88
- saturated soil column compartment, 75
- state variable, 14
- variable list variable, 21
- design of AQUASIM, 2
- detachment
  - biofilm reactor compartment, 41
  - global velocity
    - biofilm reactor compartment, 50
  - individula rate
    - biofilm reactor compartment, 50
- detachment coefficient
  - surface
    - biofilm reactor compartment, 45, 56
  - volume
    - biofilm reactor compartment, 44, 56
- detachment velocity of biofilm
  - biofilm reactor compartment, 45
    - program variable, 58
  - program variable, 16
- diffusion coefficient
  - advective-diffusive reactor compartment, 60, 62, 68
  - biofilm matrix
    - biofilm reactor compartment, 56
  - lake compartment
    - turbulent, 99
  - molecular
    - lake compartment, 103, 120
  - pore volume
    - biofilm reactor compartment, 56
  - turbulent
    - lake compartment, 100, 109
- diffusive approximation
  - river section compartment, 86, 91
- diffusive link, 10, 125, 129–131
  - conversion factor, 129, 131
  - description, 130
  - equations, 129–130
  - exchange coefficient, 129, 131
  - link index, 130
  - name, 130
  - overview, 129
  - state variable, 131
  - dynamic, 131
  - volume, 131
  - user definitions, 130–131
- diffusive solid matrix
  - biofilm reactor compartment, 50
- discharge
  - advective link
    - program variable, 128
  - advective-diffusive reactor compartment, 60
    - program variable, 62, 65, 67–69
  - biofilm reactor compartment
    - program variable, 54, 57, 58
  - lake compartment, 100, 104
    - program variable, 115, 123, 124
  - mixed reactor compartment
    - program variable, 37–39
  - program variable, 16
  - river section compartment, 86
    - program variable, 93, 95–97
  - saturated soil column compartment
    - program variable, 76, 79, 81, 84
- discretization
  - first order, 138
  - high resolution, 138
  - low resolution, 138
  - second order, 138
  - spatial, 138
- dispersion coefficient
  - advective-diffusive reactor compartment, 60, 62
  - river section compartment, 86, 90, 96
  - saturated soil column compartment, 71, 76, 84
- dissipation
  - lake compartment, 99, 103, 105
    - program variable, 100, 123, 124
  - program variable, 17
- dissolved substances
  - biofilm reactor compartment, 41
- dissolved variables
  - biofilm reactor compartment, 49, 56
  - lake compartment, 108, 120
- dynamic process, 10, 27–30
  - description, 29
  - name, 29
  - rate, 29
  - stoichiometric coefficient, 29

- dynamic state variable, 14
- edit
  - advective link, 126–129
  - advective-diffusive reactor compartment, 59–69
  - biofilm reactor compartment, 41–58
  - calculation
    - parameter estimation, 155–156
    - sensitivity analysis, 141–143
    - simulation, 141–143
  - compartments, 11, 32–124
  - constant variable, 17
  - delete states, 135–136
  - diffusive link, 129–131
  - dynamic process, 28–30
  - equilibrium process, 30–31
  - formula variable, 23–25
  - lake compartment, 98–124
  - links, 11, 125–131
  - mixed reactor compartment, 34–40
  - numerical parameters, 11, 132–134
  - probe variable, 26
  - processes, 11, 27–31
  - program variable, 15–17, 40
  - real list variable, 18–21
  - river section compartment, 85–97
  - saturated soil column compartment, 70–84
  - state variable, 14–15
  - system, 11
  - variable list variable, 21–22
  - variables, 11–26
- edit menu, 5, 9, 10, 12, 13, 27, 28, 136
- electronic mail, 8
- end coordinate
  - advective-diffusive reactor compartment, 62
  - river section compartment, 89
  - saturated soil column compartment, 75
- end row
  - real list variable, 21
- energy of seiche oscillation
  - lake compartment
    - program variable, 112, 123, 124
    - program variable, 17
- equation
  - equilibrium process, 31
- equilibrium process, 10, 27, 30–31
  - description, 31
  - equation, 31
  - name, 30
- equilibrium state variable, 14
- error
  - bounds of, 146
  - contribution to, 146
  - propagation of, 146
- error contribution, 146, 165
  - plot, 161
- error propagation, 146
- exchange coefficient
  - diffusive link, 129, 131
  - saturated soil column compartment, 82, 83
- exit
  - file menu, 8
- exp
  - formula variable, 24
- expression
  - formula variable, 23
- file menu, 4, 7–8
- file options, 162, 168
- first order, 138
- fit target
  - parameter estimation, 156
- flux limiter, 138
- format
  - print file, 8
    - long, 8
    - short, 8
  - system file, 177
- formula variable, 10, 12, 23–25
  - abs, 24
  - acos, 24
  - asin, 24
  - atan, 24
  - cos, 24
  - cosh, 24
  - deg, 24
  - description, 23
  - exp, 24
  - expression, 23
  - functions, 24–25
  - ln, 24

- log, 24
- log10, 24
- max, 24
- min, 24
- name, 23
- pi, 24
- rad, 24
- sign, 24
- sin, 24
- sinh, 24
- sqrt, 25
- tan, 25
- tanh, 25
- unit, 23
- free volume
  - growth rate of
    - biofilm reactor compartment, 43
- friction factor
  - river section compartment, 90
- friction slope
  - program variable, 17
  - river section compartment, 86, 90
    - Darcy-Weisbach, 90
    - Manning-Strickler, 90
  - program variable, 97
- Gear integration technique, 138–140
- gravitational acceleration
  - lake compartment, 108
  - river section compartment, 89
- growth velocity of biofilm
  - biofilm reactor compartment
    - program variable, 58
  - program variable, 16
- hardware platforms, 4
- high resolution, 138
- horizontal velocity
  - lake compartment, 99, 103, 104
    - program variable, 100, 112, 123, 124
  - program variable, 17
- identifiability analysis, 3, 144–146
- immobile region
  - saturated soil column compartment,
    - 70, 76, 81, 82
- inflow
  - mixed reactor compartment, 34
- initial conditions
  - advective-diffusive reactor compartment, 62, 63
  - biofilm reactor compartment, 49, 52
  - lake compartment, 108, 112
  - mixed reactor compartment, 35, 36
  - river section compartment, 89, 92
  - saturated soil column compartment,
    - 75, 78
- initial state
  - calculation
    - parameter estimation, 155
    - sensitivity analysis, 141
    - simulation, 141
- initial time
  - calculation
    - parameter estimation, 155
    - sensitivity analysis, 141
    - simulation, 141
- initialize simulation, 140
- inlet input
  - advective-diffusive reactor compartment, 64, 65
  - saturated soil column compartment,
    - 79
- input
  - advective-diffusive reactor compartment, 62, 64
  - biofilm reactor compartment, 49, 53
  - inlet
    - advective-diffusive reactor compartment, 64, 65
    - saturated soil column compartment,
      - 79
  - lake compartment, 108, 113
  - lateral
    - advective-diffusive reactor compartment, 64, 66
    - lake compartment, 113, 114
    - river section compartment, 94, 95
    - saturated soil column compartment,
      - 79, 80
  - mixed reactor compartment, 35, 37
  - point
    - lake compartment, 113, 116
  - river section compartment, 89, 94
  - saturated soil column compartment,
    - 75, 79
  - sediment

- lake compartment, 113, 117
- surface
  - lake compartment, 113
- upstream
  - river section compartment, 94
- interface velocity of biofilm
  - biofilm reactor compartment
    - program variable, 58
  - program variable, 16
- internal friction
  - lake compartment, 123
- internal shear
  - lake compartment, 103
- interpolation
  - multidimensional, 21
  - real list variable, 21
- interpolation method
  - linear, 18
  - real list variable, 18–19
  - smoothing, 19
  - spline, 18, 19
  - variable list variable, 22
- kinematic approximation
  - river section compartment, 86, 91
- label
  - plot, 163
- lake compartment, 10, 32, 98–124
  - accuracy of program variables, 110, 123
  - active for calculation, 123
  - active processes, 108, 111
  - active state variables, 108, 110
  - area, 99, 109
  - bottom coordinate, 109
  - bottom friction, 103, 123
  - boundary condition, 106, 107
  - Brunt Vaisälä frequency, 109, 110
  - buoyancy production of TKE, 101, 103
  - compartment index, 108
  - conductivity, 109
  - cross section, 109
  - density, 109, 119
  - description, 108
  - diffusion coefficient
    - molecular, 103, 120
    - turbulent, 99, 100, 109
  - discharge, 100, 104
  - dissipation, 99, 103, 105
  - dissolved variables, 108, 120
  - equations, 98–107
  - gravitational acceleration, 108
  - horizontal velocity, 99, 103, 104
  - initial conditions, 108, 112
  - input, 108, 113
    - lateral, 113, 114
    - point, 113, 116
    - sediment, 113, 117
    - surface, 113
  - internal friction, 123
  - internal shear, 103
  - lateral inflow, 103, 104
  - lateral input, 113, 114
  - molecular diffusion coefficient, 103, 120
  - name, 108
  - number of grid points, 110
  - overview, 98
  - particulate variables, 108, 118
  - point input, 113, 116
  - porosity
    - sediment layer, 121
  - Prandtl number, 100, 122
  - pressure gradient, 103, 122
  - production of dissipation, 122
  - resolution, 110
  - resuspension, 104
  - resuspension velocity, 121
  - sediment input, 113, 117
  - sediment layer, 99, 102, 105, 106
    - porosity, 99
    - thickness, 99
  - sediment submodel, 110, 121
  - sedimentation, 103
  - sedimentation velocity, 100, 119
  - seiche oscillation, 103, 104
  - shear production of TKE, 101
  - stability frequency, 109, 110
  - state variable, 100, 108, 110, 112, 114–116, 118–121
    - dynamic, 100, 108, 114–116, 118–120
    - volume, 100, 108, 114–116, 118–120
  - surface input, 113
  - surface shear, 122
  - temperature, 109



- thickness
  - sediment layer, 121
- top coordinate, 109
- turbulence submodel, 110, 121
- turbulent diffusion coefficient, 99, 100, 109
- turbulent kinetic energy, 99, 103, 104
- user definitions, 108–124
- volume flux, 121
- wind excitation, 104, 123
- zones, 98
- lateral inflow
  - advective-diffusive reactor compartment, 60
  - lake compartment, 103, 104
  - river section compartment, 87
  - saturated soil column compartment, 72
- lateral input
  - advective-diffusive reactor compartment, 64, 66
  - lake compartment, 113, 114
  - river section compartment, 94, 95
  - saturated soil column compartment, 79, 80
- legend
  - curve, 166
- line
  - curve, 166
- linear interpolation
  - real list variable, 18
  - variable list variable, 22
- link, 10, 125–131
  - advective link, 10, 125–129
  - diffusive link, 10, 125, 129–131
- link index
  - advective link, 127
    - program variable, 127, 128
  - diffusive link, 130
  - program variable, 16
- link type, 10
- links
  - edit menu, 11
- liquid phase
  - biofilm reactor compartment, 50
- liquid phase volume fraction
  - biofilm reactor compartment, 42
- list options, 162, 168
- list to file, 162
- ln
  - formula variable, 24
- loading files, 7
  - problems, 177–178
- location
  - probe variable, 26
- log
  - formula variable, 24
- log file, 8, 137, 181–183
- log10
  - formula variable, 24
- long
  - print file format, 8
- low resolution, 138
- mail
  - electronic, 8
- marker
  - curve, 166
- mass transfer resistance
  - biofilm reactor compartment, 48, 56
- max
  - formula variable, 24
- maximum
  - constant variable, 17
  - formula variable, 24
  - real list variable, 20
- maximum integration order, 133
- maximum internal step size, 132, 182, 183
- maximum number of internal time steps, 133, 185
- maximum number of iterations
  - parameter estimation, 154
- menu
  - calc, 5, 137–160
    - parameter estimation, 151–160
    - sensitivity analysis, 144–150
    - simulation, 138–143
  - edit, 5, 9–136
    - compartments, 32–124
    - delete states, 135–136
    - links, 125–131
    - numerical parameters, 132–134
    - processes, 27–31
    - variables, 12–26
- file, 4, 7–8
- view, 5, 161–170

- menu bar, 4
- min
  - formula variable, 24
- minimum
  - constant variable, 17
  - formula variable, 24
  - real list variable, 20
- mixed immobile zone
  - saturated soil column compartment, 70, 82
- mixed reactor compartment, 10, 32, 34–40
  - constant volume, 34
  - accuracy of program variables, 35, 38
  - active for calculation, 35
  - active processes, 35, 36
  - active state variables, 35
  - compartment index, 34
  - constant volume, 35
  - description, 35
  - equations, 34
  - inflow, 34
  - initial conditions, 35, 36
  - input, 35, 37
  - name, 34
  - outflow, 34
  - overview, 34
  - reactor type, 35
  - state variable, 34–38
    - dynamic, 34, 38
    - equilibrium, 34
    - surface, 34
    - volume, 34, 38
  - user definitions, 34–40
  - variable volume, 34, 35
  - volume, 34, 35
  - volumetric inflow, 34
  - volumetric outflow, 34
- mobile zone
  - saturated soil column compartment, 70
- molecular diffusion coefficient
  - lake compartment, 103, 120
- multidimensional interpolation, 21
- name
  - advective link, 127
  - bifurcation, 128
  - advective-diffusive reactor compartment, 61
  - biofilm reactor compartment, 49
  - calculation
    - parameter estimation, 155
    - sensitivity analysis, 140, 141
    - simulation, 140, 141
  - constant variable, 17
  - diffusive link, 130
  - dynamic process, 29
  - equilibrium process, 30
  - formula variable, 23
  - lake compartment, 108
  - mixed reactor compartment, 34
  - plot, 163
  - probe variable, 26
  - program variable, 15
  - real list variable, 20
  - river section compartment, 88
  - saturated soil column compartment, 75
    - immobile region, 81
  - state variable, 14
  - variable list variable, 21
- new
  - file menu, 7
- normal water level
  - river section compartment, 91
- number of codiagonals of the jacobian matrix, 133
- number of grid points
  - advective-diffusive reactor compartment, 62
  - biofilm reactor compartment, 51
  - lake compartment, 110
  - river section compartment, 91
  - saturated soil column compartment, 76
- number of steps
  - calculation
    - sensitivity analysis, 142
    - simulation, 142
- numerical algorithms
  - first order spatial discretization, 138
  - flux limiter technique, 138
  - Gear integration technique, 138, 140
  - high resolution spatial discretization, 138

- low resolution spatial discretization, 138
- secant minimization algorithm, 151–154
- second order spatial discretization, 138
- simplex minimization algorithm, 151, 153, 154
- numerical parameters, 132
  - edit menu, 11
  - maximum integration order, 133
  - maximum internal step size, 132, 182, 183
  - maximum number of internal time steps, 133, 185
  - number of codiagonals of the jacobian matrix, 133
- open
  - file menu, 7
- operating systems, 4
- options
  - list to file, 162, 168
  - plot to file, 162, 168
  - plot to screen, 162, 167
  - print to file, 8
- outflow
  - mixed reactor compartment, 34
- output steps
  - calculation
    - sensitivity analysis, 142
    - simulation, 142
- parameter, 165
- parameter estimation, 3, 17, 19, 21, 151–160
  - calculation, 155–156
    - active for parameter estimation, 154
  - fit target, 156
    - compartment, 157
    - data, 156
    - time/space, 157
    - variable, 157
    - zone, 157
  - maximum number of iterations, 154
  - start, 153, 154
- particulate variables
  - biofilm reactor compartment, 49, 55
  - lake compartment, 108, 118
- perimeter length
  - program variable, 16
  - river section compartment, 89
    - program variable, 97
- pi
  - formula variable, 24
- platforms, 4, 8
- plot
  - abscissa, 163
  - curve, 163
    - calculation number, 165
    - compartment, 166
    - legende, 166
    - line, 166
    - marker, 166
    - time/space, 166
    - type, 164
    - zone, 166
  - description, 163
  - label, 163
  - name, 163
  - scaling, 167
  - title, 163
- plot to file, 162
- plot to screen, 162
- point input
  - lake compartment, 113, 116
- pore volume
  - biofilm reactor compartment, 41
- porosity
  - biofilm reactor compartment, 42
  - mobile zone
    - saturated soil column compartment, 71, 76
  - rate of change
    - biofilm reactor compartment, 51
    - saturated soil column compartment, 71
  - sediment layer
    - lake compartment, 99, 121
- Prandtl number
  - lake compartment, 100, 122
- pressure gradient
  - lake compartment, 103, 122
- print file
  - format, 8
  - long, 8
  - short, 8
- print options

- file menu, 8
  - long, 8
  - short, 8
- print to file
  - file menu, 8
- probe variable, 10, 12, 26
  - compartment, 26
  - description, 26
  - location, 26
  - name, 26
  - unit, 26
  - variable, 26
  - zone, 26
- problem
  - during calculation, 181–186
  - editing a model, 179
  - loading files, 177–178
- process, 10, 27–31
  - dynamic process, 10, 27–30
  - equilibrium process, 10, 27, 30–31
- process matrix, 29
- process rate, 28, 29
- process type, 10
- processes
  - edit menu, 11, 27, 28
- production of dissipation
  - lake compartment, 122
- program variable, 10, 12, 15–17, 40
  - accuracy, 132, 134, 142
  - accuracy of
    - advective-diffusive reactor compartment, 62, 67
    - biofilm reactor compartment, 51, 56
    - lake compartment, 110, 123
    - mixed reactor compartment, 35, 38
    - river section compartment, 91, 96
    - saturated soil column compartment, 76, 83
- area gradient, 17
  - lake compartment, 124
- attachment velocity of biofilm, 16
  - biofilm reactor compartment, 58
- biofilm thickness, 16
  - biofilm reactor compartment, 53, 57, 58
- Brunt Vaisälä frequency, 17
  - lake compartment, 110, 112, 124
- bulk volume, 16
  - biofilm reactor compartment, 57, 58
  - mixed reactor compartment, 39
- buoyancy production of TKE, 17
  - lake compartment, 124
- calculation number, 15, 16
  - advective link, 128
  - advective-diffusive reactor compartment, 69
  - biofilm reactor compartment, 58
  - lake compartment, 113, 124
  - mixed reactor compartment, 39
  - river section compartment, 97
  - saturated soil column compartment, 84
- compartment index, 16
  - advective-diffusive reactor compartment, 62, 69
  - biofilm reactor compartment, 49, 58
  - lake compartment, 108, 124
  - mixed reactor compartment, 34, 39
  - river section compartment, 88, 97
  - saturated soil column compartment, 75, 84
- cross-sectional area, 16
  - advective-diffusive reactor compartment, 62, 69
  - lake compartment, 100, 124
  - river section compartment, 96, 97
  - saturated soil column compartment, 76, 84
- density, 17
  - lake compartment, 124
- description, 15
- detachment velocity of biofilm, 16
  - biofilm reactor compartment, 58
- discharge, 16
  - advective link, 128
  - advective-diffusive reactor compartment, 62, 65, 67–69
  - biofilm reactor compartment, 54, 57, 58
  - lake compartment, 115, 123, 124
  - mixed reactor compartment, 37–39
  - river section compartment, 93, 95–97
  - saturated soil column compartment, 76, 79, 81, 84
- dissipation, 17

- lake compartment, 100, 123, 124
- energy of seiche oscillation, 17
  - lake compartment, 112, 123, 124
- friction slope, 17
  - river section compartment, 97
- growth velocity of biofilm, 16
  - biofilm reactor compartment, 58
- horizontal velocity, 17
  - lake compartment, 100, 112, 123, 124
- interface velocity of biofilm, 16
  - biofilm reactor compartment, 58
- link index, 16
  - advective link, 127, 128
- name, 15
- perimeter length, 16
  - river section compartment, 97
- reactor volume, 16
  - biofilm reactor compartment, 57, 58
  - mixed reactor compartment, 35, 37–39
- reference, 15
- shear production of TKE, 17
  - lake compartment, 124
- space coordinate x, 16
  - advective-diffusive reactor compartment, 62, 69
  - river section compartment, 89, 97
  - saturated soil column compartment, 76, 84
- space coordinate z, 16
  - biofilm reactor compartment, 50, 58
  - lake compartment, 109, 113, 124
- surface width, 16
  - river section compartment, 97
- time, 16
  - advective link, 128
  - advective-diffusive reactor compartment, 62, 69
  - biofilm reactor compartment, 58
  - lake compartment, 124
  - mixed reactor compartment, 39
  - river section compartment, 97
  - saturated soil column compartment, 76, 84
- turbulent kinetic energy, 17
  - lake compartment, 100, 112, 123, 124
- unit, 15
- water fraction, 16
  - advective-diffusive reactor compartment, 69
  - biofilm reactor compartment, 57, 58
  - lake compartment, 124
  - mixed reactor compartment, 39
  - river section compartment, 97
  - saturated soil column compartment, 84
- water level elevation, 16
  - river section compartment, 89, 96, 97
- zone index, 16
  - advective-diffusive reactor compartment, 69
  - biofilm reactor compartment, 49, 58
  - lake compartment, 108, 124
  - mixed reactor compartment, 39
  - river section compartment, 97
  - saturated soil column compartment, 75, 82, 84
- rad
  - formula variable, 24
- rate, 28, 29
  - dynamic process, 29
- reactor
  - confined
    - biofilm reactor compartment, 48, 50
  - unconfined
    - biofilm reactor compartment, 48, 50
- reactor type
  - biofilm reactor compartment, 50
  - mixed reactor compartment, 35
- reactor volume
  - biofilm reactor compartment, 48, 50
    - program variable, 57, 58
  - mixed reactor compartment, 34
    - program variable, 35, 37–39
  - program variable, 16
- read data pairs
  - real list variable, 21
- real list variable, 10, 12, 18–21, 151
  - active for sensitivity analysis, 21
  - add data pairs, 21
  - argument, 18, 20
  - column number of argument, 21

- column number of standard deviations, 21
- column number of values, 21
- data pairs, 18
- delete data pairs, 21
- description, 20
- end row, 21
- interpolation, 21
- interpolation method, 18–19
  - linear, 18
  - smoothing, 19
  - spline, 18, 19
- maximum, 20
- minimum, 20
- name, 20
- read data pairs, 21
- replace data pairs, 21
- standard deviation, 18, 20, 21
  - absolute, 18, 20
  - global, 20
  - individual, 20, 21
  - relative, 18, 20
- start row, 21
- unit, 20
- write data pairs, 21
- reference
  - program variable, 15
- relative accuracy
  - program variable
    - advective-diffusive reactor compartment, 67
    - biofilm reactor compartment, 57
    - lake compartment, 123
    - mixed reactor compartment, 38
    - river section compartment, 96
    - saturated soil column compartment, 84
    - state variable, 14, 134
- replace argument-value pairs
  - variable list variable, 22
- replace data pairs
  - real list variable, 21
- resolution
  - advective-diffusive reactor compartment, 62
  - biofilm reactor compartment, 51
  - lake compartment, 110
  - river section compartment, 91
  - saturated soil column compartment, 76
- resuspension
  - lake compartment, 104
- resuspension velocity
  - lake compartment, 121
- revert to saved
  - file menu, 7, 8
- rigid solid matrix
  - biofilm reactor compartment, 50
- river bed elevation
  - river section compartment, 86
- river section compartment, 10, 32, 85–97
  - accuracy of program variables, 91, 96
  - active for calculation, 91
  - active processes, 89, 92
  - active state variables, 89, 91
  - area, 85, 89
  - boundary condition, 87, 88
  - compartment index, 88
  - critical water level, 91
  - cross section, 89
  - description, 88
  - diffusive approximation, 86, 91
  - discharge, 86
  - dispersion coefficient, 86, 90, 96
  - end coordinate, 89
  - equations, 85–88
  - friction slope, 86, 90
    - Darcy-Weisbach, 90
    - Manning-Strickler, 90
  - gravitational acceleration, 89
  - initial conditions, 89, 92
  - input, 89, 94
    - lateral, 94, 95
    - upstream, 94
  - kinematic approximation, 86, 91
  - lateral inflow, 87
  - lateral input, 94, 95
  - name, 88
  - normal water level, 91
  - number of grid points, 91
  - overview, 85
  - perimeter length, 89
  - resolution, 91
  - river bed elevation, 86
  - start coordinate, 89
  - state variable, 86, 89–91, 93, 95

- dynamic, 86, 95
  - equilibrium, 86
  - surface, 86
  - volume, 86, 95
  - upstream input, 94
  - user definitions, 88–97
  - water level elevation, 86
  - width, 89
- saturated soil column compartment, 10, 32, 70–84
- accuracy of program variables, 76, 83
  - active for calculation, 76
  - active processes, 75, 77
  - active state variables, 75, 76
  - area, 71, 76
  - boundary condition, 74
  - compartment index, 75
  - cross section, 76
  - density, 74
  - description, 75
    - immobile region, 82
  - dispersion coefficient, 71, 76, 84
  - end coordinate, 75
  - equations, 70–74
  - exchange coefficient, 82
  - immobile region, 70, 76, 81, 82
  - initial conditions, 75, 78
  - inlet input, 79
  - input, 75, 79
    - inlet, 79
    - lateral, 79, 80
  - lateral inflow, 72
  - lateral input, 79, 80
  - mixed immobile zone, 70, 82, 83
  - mobile zone, 70
  - name, 75
    - immobile region, 81
  - number of grid points, 76
  - overview, 70
  - porosity, 71
    - mobile zone, 71, 76
  - resolution, 76
  - sorption, 74
    - isotherm, 74
  - start coordinate, 75
  - state variable, 71, 75, 76, 79, 81, 83
    - dynamic, 71, 79, 83
    - equilibrium, 71
    - surface, 71
    - volume, 71, 79, 83
  - user definitions, 75–84
  - volume fraction, 82
    - mobile zone, 76
  - zone index, 82
  - zones, 70
- save
- file menu, 7, 8
- save as
- file menu, 7
- scaling
- plot, 167
- screen options, 162, 167
- secant algorithm, 151–154
- second order, 138
- sediment input
- lake compartment, 113, 117
- sediment layer
- lake compartment, 99, 102, 105, 106
- sediment submodel
- lake compartment, 110, 121
- sedimentation
- lake compartment, 103
- sedimentation velocity
- lake compartment, 100, 119
- seiche oscillation
- lake compartment, 103, 104
- sensitivity analysis, 3, 17, 20, 144–150, 164, 165
- start, 147, 148
- sensitivity function, 144, 165
- absolute-absolute, 144, 165
  - absolute-relative, 145, 165
  - plot, 161
  - relative-absolute, 145, 165
  - relative-relative, 145, 165
- sensitivity ranking, 148, 149
- shear production of TKE
- lake compartment, 101
  - program variable, 124
  - program variable, 17
- short
- print file format, 8
- sign
- formula variable, 24
- simplex algorithm, 151, 153, 154

- simulation, 2, 3, 138–143
  - continue, 141
  - start, 141
- sin
  - formula variable, 24
- sinh
  - formula variable, 24
- smoothing
  - real list variable, 19
  - variable list variable, 22
- solid matrix
  - biofilm reactor compartment, 41
- sorption
  - isotherm
    - saturated soil column compartment, 74
    - saturated soil column compartment, 74
- space coordinate x
  - advective-diffusive reactor compartment, 62
    - program variable, 62, 69
  - program variable, 16
  - river section compartment
    - program variable, 89, 97
  - saturated soil column compartment
    - program variable, 76, 84
- space coordinate z
  - biofilm reactor compartment, 41
    - program variable, 50, 58
  - lake compartment
    - program variable, 109, 113, 124
  - program variable, 16
- spatial discretization, 138
- spline interpolation
  - real list variable, 18, 19
  - variable list variable, 22
- sqrt
  - formula variable, 25
- stability frequency
  - lake compartment, 109, 110
- standard deviation, 146, 151
  - absolute
    - real list variable, 18, 20
  - constant variable, 17
  - global
    - real list variable, 20
  - individual
    - real list variable, 20, 21
  - relative
    - real list variable, 18, 20
- standard error, 144
- start
  - parameter estimation, 153, 154
  - sensitivity analysis, 148
  - simulation, 141
- start coordinate
  - advective-diffusive reactor compartment, 62
  - river section compartment, 89
  - saturated soil column compartment, 75
- start row
  - real list variable, 21
- state variable, 10, 12, 14–15
  - absolute accuracy, 14, 134
  - accuracy, 14, 132, 134, 142
  - advective link, 126, 128, 129
  - advective-diffusive reactor compartment, 59, 62, 64, 65, 67
  - biofilm reactor compartment, 42, 49–54
  - description, 14
  - diffusive link, 131
  - dynamic, 14
    - advective link, 126, 128, 129
    - advective-diffusive reactor compartment, 59, 65, 67
    - biofilm reactor compartment, 42, 49, 54
    - diffusive link, 131
    - lake compartment, 100, 108, 114–116, 118–120
    - mixed reactor compartment, 34, 38
    - river section compartment, 86, 95
    - saturated soil column compartment, 71, 79, 83
- equilibrium, 14
  - advective-diffusive reactor compartment, 59
  - mixed reactor compartment, 34
  - river section compartment, 86
  - saturated soil column compartment, 71
  - lake compartment, 100, 108, 110, 112,



- 114–116, 118–121
- mixed reactor compartment, 34–38
- name, 14
- relative accuracy, 14, 134
- river section compartment, 86, 89–91, 93, 95
- saturated soil column compartment, 71, 75, 76, 79, 81, 83
- surface, 14
  - advective-diffusive reactor compartment, 59
  - mixed reactor compartment, 34
  - river section compartment, 86
  - saturated soil column compartment, 71
- type, 14
- unit, 14
- volume, 14
  - advective link, 126, 128, 129
  - advective-diffusive reactor compartment, 59, 65, 67
  - biofilm reactor compartment, 42, 49, 54
  - diffusive link, 131
  - lake compartment, 100, 108, 114–116, 118–120
  - mixed reactor compartment, 34, 38
  - river section compartment, 86, 95
  - saturated soil column compartment, 71, 79, 83
- step size
  - calculation
    - sensitivity analysis, 142
    - simulation, 142
- stiff, 139
- stiffly stable, 139, 140
- stoichiometric coefficient, 28, 29
- stoichiometric matrix, 29
- substance flow
  - advective link, 126
  - bifurcation, 126, 128
- substratum
  - biofilm reactor compartment, 41
- supported platforms, 4, 8
- surface attachment coefficient
  - biofilm reactor compartment, 56
- surface detachment coefficient
  - biofilm reactor compartment, 56
- surface input
  - lake compartment, 113
- surface shear
  - lake compartment, 122
- surface state variable, 14
- surface width
  - river section compartment, 97
  - program variable, 97
- surface width
  - program variable, 16
- suspended solids
  - biofilm reactor compartment, 41, 50
- system
  - edit menu, 11
- tan
  - formula variable, 25
- tanh
  - formula variable, 25
- tasks of AQUASIM, 2–3
- temperature
  - lake compartment, 109
- thickness
  - sediment layer
    - lake compartment, 99, 121
- time
  - advective link
    - program variable, 128
  - advective-diffusive reactor compartment
    - program variable, 62, 69
  - biofilm reactor compartment
    - program variable, 58
  - lake compartment
    - program variable, 124
  - mixed reactor compartment
    - program variable, 39
  - program variable, 16
  - river section compartment
    - program variable, 97
  - saturated soil column compartment
    - program variable, 76, 84
- time/space
  - curve, 166
  - fit target, 157
- title
  - plot, 163
- top coordinate

- lake compartment, 109
- troubleshooting, 176–188
- turbulence submodel
  - lake compartment, 110, 121
- turbulent diffusion coefficient
  - lake compartment, 99, 100, 109
- turbulent kinetic energy
  - lake compartment, 99, 103, 104
    - program variable, 100, 112, 123, 124
  - program variable, 17
- type
  - compartment, 10, 32
  - curve, 164
  - link, 10
  - process, 10
  - state variable, 14
  - variable, 10, 12, 13
- uncertainty analysis, 3, 146–147
- unconfined reactor
  - biofilm reactor compartment, 48, 50
- unit
  - constant variable, 17
  - formula variable, 23
  - probe variable, 26
  - program variable, 15
  - real list variable, 20
  - state variable, 14
  - variable list variable, 21
- upstream input
  - river section compartment, 94
- user group, 176, 187
- user interfaces, 3–4
  - batch version, 3, 4, 8, 173–175
  - character interface version, 3, 4, 171–172
  - window interface version, 3, 4
- value
  - constant variable, 17
  - plot, 161
- variable, 10, 12–26
  - constant variable, 10, 12, 17, 21, 151
  - fit target, 157
  - formula variable, 10, 12, 23–25
  - probe variable, 10, 12, 26
  - program variable, 10, 12, 15–17, 40
  - real list variable, 10, 12, 18–21, 151
  - state variable, 10, 12, 14–15
  - variable list variable, 10, 12, 21–22
- variable list variable, 10, 12, 21–22
  - add argument-value pairs, 22
  - argument, 22
  - delete argument-value pairs, 22
  - description, 21
  - name, 21
  - replace argument-value pairs, 22
  - unit, 21
- variable type, 10, 12, 13
- variable volume
  - mixed reactor compartment, 35
- variables
  - dissolved
    - biofilm reactor compartment, 56
    - lake compartment, 120
  - edit menu, 11–13
  - particulate
    - biofilm reactor compartment, 55
    - lake compartment, 118
- velocity
  - biofilm matrix
    - biofilm reactor compartment, 43
  - horizontal, 17, 123, 124
- velocity of biofilm matrix
  - biofilm reactor compartment, 43
- view menu, 5, 161–170
- volume
  - bulk, 16, 39, 41, 57, 58
    - biofilm reactor compartment, 48
  - constant
    - mixed reactor compartment, 34, 35
  - mixed reactor compartment, 34, 35
  - pore, 41
  - reactor, 16, 35, 38, 39, 57, 58
    - biofilm reactor compartment, 48
    - mixed reactor compartment, 34
  - variable
    - mixed reactor compartment, 34, 35
- volume attachment coefficient
  - biofilm reactor compartment, 56
- volume detachment coefficient
  - biofilm reactor compartment, 56
- volume flux
  - lake compartment, 121
- volume fraction
  - mobile zone

- saturated soil column compartment, 76
  - saturated soil column compartment, 82
- volume state variable, 14
- volumetric inflow
  - mixed reactor compartment, 34
- volumetric outflow
  - mixed reactor compartment, 34
- water flow
  - advective link, 126
  - bifurcation, 126, 128
- water fraction
  - advective-diffusive reactor compartment
    - program variable, 69
  - biofilm reactor compartment
    - program variable, 57, 58
  - lake compartment
    - program variable, 124
  - mixed reactor compartment
    - program variable, 39
  - program variable, 16
  - river section compartment
    - program variable, 97
  - saturated soil column compartment
    - program variable, 84
- water level elevation
  - program variable, 16
  - river section compartment, 86
    - program variable, 89, 96, 97
- width
  - river section compartment, 89
- wind excitation
  - lake compartment, 123
  - molecular
    - lake compartment, 104
- window interface version, 3, 4
- write data pairs
  - real list variable, 21
- zone
  - curve, 166
  - fit target, 157
  - probe variable, 26
- zone index
  - advective-diffusive reactor compartment
- program variable, 69
- biofilm reactor compartment
  - program variable, 49, 58
- lake compartment
  - program variable, 108, 124
- mixed reactor compartment
  - program variable, 39
- program variable, 16
- river section compartment
  - program variable, 97
- saturated soil column compartment, 82
  - program variable, 75, 82, 84
- zones
  - biofilm reactor compartment, 41
  - lake compartment, 98
  - saturated soil column compartment, 70