11

WEST®

described by Peter A. Vanrolleghem & Sylvie Gillot

NOTE: The issues and procedures outlined in this chapter are specific to the use of WEST with the *'simulation benchmark'* and in no way should be interpreted as necessary procedures for using WEST for any other purpose.

WEST is a multi-platform modelling and experimentation system for the simulation of any system that can be represented by algebraic and differential equations. In WEST, ease of modelling a wide range of systems with maximum reuse of model knowledge is combined with high calculation performance. So far, WEST mainly has been used in the context of wastewater treatment research and is currently available for the following operating systems:

- Win 32: Windows 95, Windows 98, Windows NT 4.0 & Windows 2000
- Linux: Redhat, SuSE, Slackware, …
- SGI Irix 6.5
- IBM AIX 4.2
- Sun Solaris 2.6

The modelling environment is geared to support the rapid development of complex system models. Models are constructed by introducing model components (e.g. reactors, splitters, settlers, sensors, controllers, …) on a hierarchical graphical editor (HGE), linking the material and information flows between them and selecting the appropriate models from a modelbase. The open (plain text format) and hierarchically structured modelbase contains an extensive set of wastewater treatment component models and can be easily augmented with user-defined models. For instance, a new type of sensor can be quickly created by extending the basic sensor model available in the

WEST® is a registered product of:
HEMMIS N.V. Koning Leopold III-laan 2, B-8500 Kortrijk, Belgium
tel: +32 (0)56 37 26 37
fax: +32-(0)56 37 23 24
e-mail:info@hemmis.be
web: www.hemmiswest.com

modelbase with the specific properties of the new sensor to be modelled (e.g. the variable it measures, the noise type, measuring frequency). The model specification language (MSL Vs. 3.1) in which models are written, has been defined in such a way that particular information (e.g default parameter values or initial conditions, typical parameter ranges, variable constraint, units) can be included in the modelbase. In this way modelling is easier for non-expert users. Further, for the expert user procedures coded in C can be called from the MSL-code allowing, for instance, the use of existing high-performance numerical libraries.

Once the model is constructed, the WEST model transformer is launched and, in doing so, attention is diverted from user-friendly modelling support to raw calculation speed. WEST uses compiled code to achieve maximum calculation performance. Before compilation is initiated, however, the MSL-code of the different model components is combined into a set of algebraic and differential equations to which symbolic manipulation is applied. This allows a number of things including proper (dependency based) sorting of the equations, detection of algebraic loops (and if possible an immediate symbolic solution), and equation simplification. Finally, C-code is generated, compiled and linked.

The following sections describe the implementation of the *'simulation benchmark'* in WEST (Windows version 2.2 and higher), and outline the choices that are required to achieve benchmark consistent results.

The implementation of the benchmark into WEST is rather straightforward, as all benchmark specified models are available in WEST. Therefore, the implementation follows the normal procedure defined in detail on the web site (http://www.hemmiswest.com/) or in the WEST Quick Tour. Tuning WEST to the benchmark specifications simply consists in changing the default settings. A problem with the time interval of the output data, possibly influenced by the settings of the integration algorithm, is discussed.

11.1 CONFIGURATION ISSUES – WEST

The WEST layout of the *'simulation benchmark'* is shown in Figure 11.1. The benchmark configuration consists of five *activated_sludge_units*, two *two_splitters*, two *two_combiners*, a *secondary_clarifier*, two *loop_breakers*, a *CtoF*, a *FtoC*, a *sensor*, an *input*, an *output* and a *waste*. Once the layout has been created, models are assigned. Table 11.1 lists the WEST models that should be selected.

Table 11.1: WEST models to be used with the'*simulation benchmark'*.

Process	WEST Model
Biological (ASU)	FixVolumeASU
Settling (Clarifier)	SecondaryTakacsSolublesPropagator
Splitters	AbsTwoSplitter
Combiners	TwoCombiner
Loop breaker	DifferentialLoopBreaker
Waste	WasteFlux
Sensor	BenchmarkSensor

A benchmark specific model (*BenchmarkSensor*) has been created in order to calculate the effluent quality index, the effluent quality variance and the effluent violations (number of violations and percentage of time the plant is in violation) as the simulation runs.

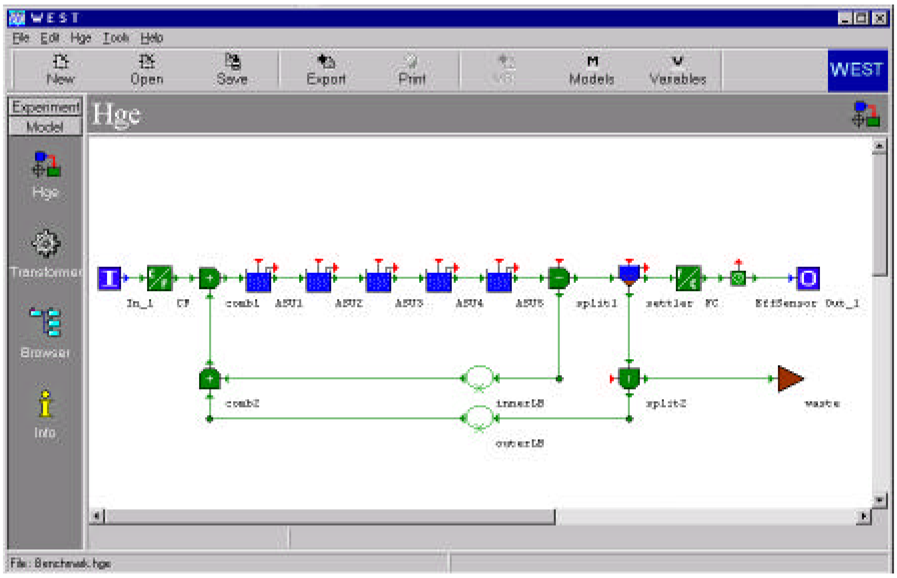


Figure 11.1: Interface layout of the open loop *'simulation benchmark'* in the Hierarchical Graphical Editor (HGE) in WEST.

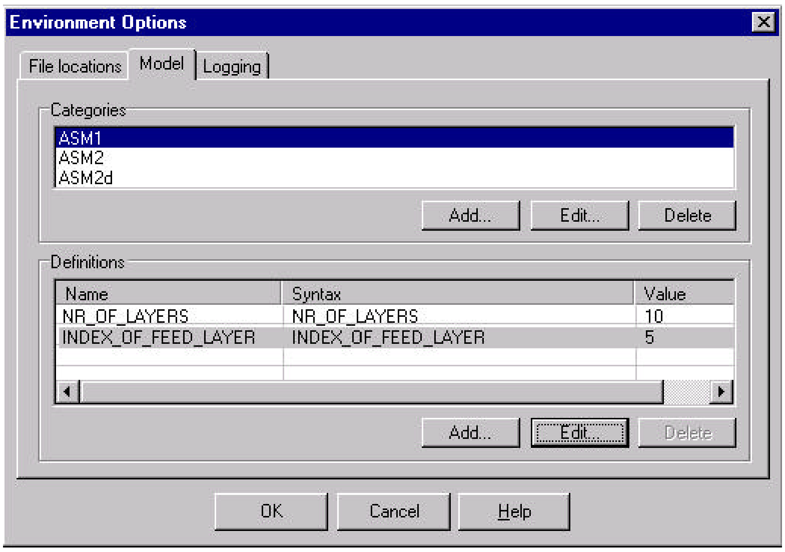
Setting up the *'simulation benchmark'* configuration and assigning the models is done in the modelling environment of WEST. The model is then compiled and a dynamic link library (WEST model library, extension WML) is created using the transformer module. The WML library can then be loaded in the experimentation environment where the model parameters and initial conditions are set.

11.2 MODEL ISSUES – WEST

The *'simulation benchmark'* specifies two process models: the IAWQ Activated Sludge #1 – ASM1 (Henze *et al*., 1987) for the biological processes and the double-exponential settling function of Takács *et al*. (1991) for the settling process. Both process models are available in WEST, as seen in Table 11.1. The following sections outline the change in the default parameters required to meet the *'simulation benchmark'* specifications.

11.2.1 Biological Process Model

To work with the ASM1 state variables, *ASM1* should be selected as the WEST *Model Category* (Figure 11.2). Table 11.2 lists the state variables and symbols used in ASM1 and the WEST *FixVolumeASU* model. Note that in WEST, the state variables have mass units, but their concentrations are also available in the model.



Name	Syntax	Value
NR_OF_LAYERS	NR_OF_LAYERS	10
INDEX_OF_FEED_LAYER	INDEX_OF_FEED_LAYER	5

Figure 11.2: The definition of biological conversion model (ASM1) and settler properties in WEST.

Table 11.2: Comparison of the state variable symbols used in ASM1 and the WEST *FixVolumeASU* model.

State variable	ASM1 Symbol	WEST Symbol	WEST Units
Soluble inert organic matter	S_I	*S_I*	g COD
Readily biodegradable substrate	S_s	*S_S*	g COD
Particulate inert organic matter	X_I	*X_I*	g COD
Slowly biodegradable substrate	X_S	*X_S*	g COD
Active heterotrophic biomass	$X_{B,H}$	*X_BH*	g COD
Active autotrophic biomass	$X_{B,A}$	*X_BA*	g COD
Particulate products arising from biomass decay	X_P	*X_P*	g COD
Oxygen	S_O	*S_O*	g COD
Nitrate & nitrite nitrogen	S_{NO}	*S_NO*	g N
NH_4^+ & NH_3 nitrogen	S_{NH}	*S_NH*	g N
Soluble biodegradable organic nitrogen	S_{ND}	*S_ND*	g N
Particulate biodegradable organic nitrogen	X_{ND}	*X_ND*	g N

Table 11.3 and Table 11.4 list the stoichiometric and the kinetic parameters to be used with the WEST *FixVolumeASU* model. The parameter values are identical to those specified in the *'simulation benchmark'*.

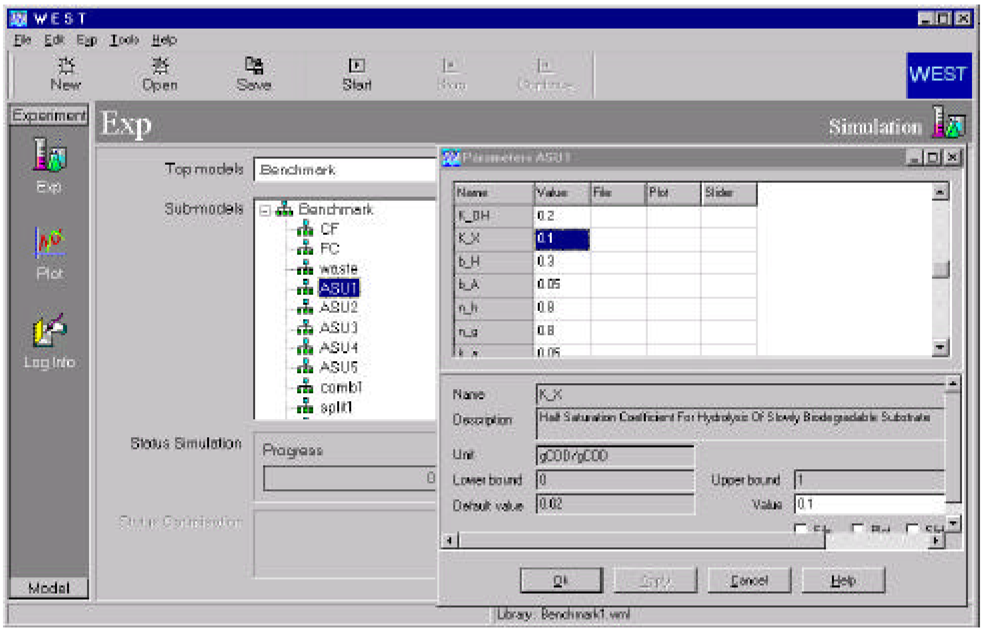
Table 11.3: *'Simulation benchmark'* stoichiometric parameters and WEST *FixVolumeASU* model.

Description	ASM1 symbol	WEST symbol	WEST Value	Units
Fractions				
Fraction COD/TSS	f_{TC}	*F_TSS_COD*	0.75	g SS gCOD^{-1}
Heterotrophs				
Yield for heterotrophic biomass	Y_H	*Y_H*	0.67	gCOD gCOD^{-1}
Mass of nitrogen per mass of COD in biomass	i_{XB}	*i_X_B*	0.08	gN gCOD^{-1}
Fraction of biomass converted to inert matter	f_P	*f_P*	0.08	dimensionless
Mass of nitrogen per mass of COD in products	i_{XP}	*i_X_P*	0.06	gN gCOD^{-1}
Autotrophs				
Yield for autotrophic biomass	Y_A	*Y_A*	0.24	gCOD gN^{-1}

Figure 11.3 shows the *'simulation benchmark'* in the experimentation environment of WEST. From this window, the user can edit the model parameters and assign the *benchmark* specified values (Note in Figure 11.3 the properties that have been propagated from the modelbase into the experimentation environment).

Table 11.4: *'Simulation benchmark'* kinetic parameters and WEST *FixVolumeASU* model.

WEST description	ASM1 symbol	WEST symbol	WEST Value	Units
Heterotrophs				
Maximum specific growth rate for heterotrophic biomass	m_H	*mu_H*	4.0	day^{-1}
Half-saturation coefficient for heterotrophic biomass	K_S	*K_S*	10.0	g COD m^{-3}
Half saturation coefficient for hydrolysis of slowly biodegradable substrate	K_X	*K_X*	0.1	g COD g COD^{-1}
Decay coefficient for heterotrophic biomass	b_H	*b_H*	0.3	day^{-1}
Correction factor for anoxic hydrolysis	h_h	*n_h*	0.8	dimensionless
Correction factor for anoxic growth of heterotrophs	h_g	*n_g*	0.8	dimensionless
Maximum specific ammonification rate	k_a	*k_a*	0.05	m^3 (g COD · day)$^{-1}$
Maximum specific hydrolysis rate	k_h	*k_h*	3.0	g X_S (g COD · day)$^{-1}$
Autotrophs				
Maximum specific growth rate for autotrophic biomass	m_A	*mu_A*	0.5	day^{-1}
Decay coefficient for autotrophic biomass	b_A	*b_A*	0.05	day^{-1}
Ammonia half-saturation coefficient for autotrophic biomass	K_{NH}	*K_NH*	1.00	g NH_3-N m^{-3}
Switching Functions				
Oxygen half-saturation coefficient for heterotrophic biomass	$K_{O,H}$	*K_OH*	0.2	g O_2 m^{-3}
Nitrate half-saturation coefficient for denitrifying heterotrophic biomass	K_{NO}	*K_NO*	0.5	g NO_3-N m^{-3}
Oxygen half-saturation coefficient for autotrophic biomass	$K_{O,A}$	*K_OA*	0.4	g O_2 m^{-3}



Top models: Benchmark

Submodels: Benchmark, CF, FC, waste, ASU1, ASU2, ASU3, ASU4, ASU5, comb1, split1

Status Simulation: Progress 0

Status Optimisation

Name	Value	File	Plot	Slider
K_OH	0.2			
K_X	0.1			
b_H	0.3			
b_A	0.05			
n_h	0.8			
n_g	0.8			
k_a	0.05			

Name: K_X

Description: Half Saturation Coefficient For Hydrolysis Of Slowly Biodegradable Substrate

Unit: gCOD/gCOD

Lower bound: 0 Upper bound: 1

Default value: 0.02 Value: 0.1

□ File □ Plot □ Slider

Library: Benchmark1.wml

Figure 11.3: The COST Benchmark in the experimentation environment of WEST.

11.2.2 Settling Model

The double-exponential settling velocity function of Takács *et al*. (1991) was chosen as the *'simulation benchmark'* settling model due to its wide use and apparent acceptability as a fair representation of the settling process. The appropriate benchmark model is the *SecondaryTakacsSolublesPropagator* secondary clarifier model and needs to be selected from the secondary settler modelbase. The parameters in the Takács function and the associated values are listed in Table 11.5.

Table 11.5: Parameters and values of the settling model.

Parameter description	Takács	WEST	Value	Units
	Symbol	Symbol		
Maximum settling velocity	n'_0	*v_0*	250	m day^{-1}
Maximum Vesilind settling velocity	n_0	*v_00*	474	m day^{-1}
Hindered zone settling parameter	r_h	*r_h*	0.000576	m^3 (g SS)$^{-1}$
Flocculant zone settling parameter	r_p	*r_p*	0.00286	m^3 (g SS)$^{-1}$
Non-settleable fraction	f_{ns}	*f_ns*	0.00228	Dimensionless

By default, the feed to the settler is introduced at layer 5 in WEST (Figure 11.3); this is consistent with the *benchmark* description (as layers are counted from top to bottom in WEST).

11.2.3 Loop Breaker

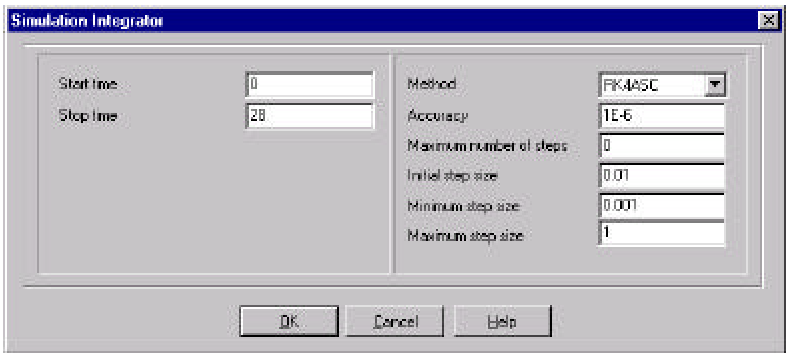
The two recycles lead to algebraic loops in the model (direct feed through, i.e. when an input to a submodel is directly dependent on its own output, for instance with the recycle flows). During model transformation, the WEST algebraic loop detector will try to solve this loop problem by introducing an implicit loop solver (a symbolic solution is not found automatically), but this leads to a dramatic increase in computation time as this implicit solver uses iteration. The proper alternative is to introduce an explicit loop breaker. In WEST one can choose either a time delay or a first order loop breaker. In the current WEST benchmark a first order loop breaker was chosen with the time constant set to 0.0005 days to maximise calculation performance and minimise the deviation from the benchmark consistent simulation results.

11.3 SIMULATION ISSUES – WEST

Before running simulations, users have to define an *integrator*, an *input file* and an *output file* (and possibly, plots).

11.3.1 Integrator

As was shown also in the Matlab studies (Chapter 8), the numerical integrator preferred for this type of problem is the WEST default 4th order Runge-Kutta with variable time step size (*RK4ASC*). The parameters to be used for the integrator are shown in Figure 11.4.



Simulation Integrator

Start time 0

Stop time 28

Method RK4ASC

Accuracy 1E-6

Maximum number of steps 0

Initial step size 0.01

Minimum step size 0.001

Maximum step size 1

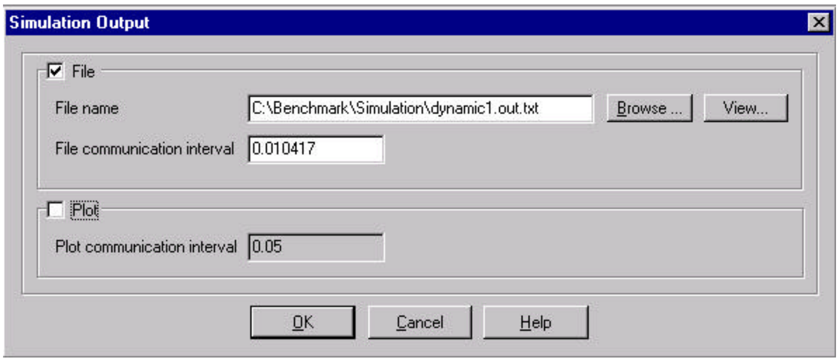
Figure 11.4: Integrator window showing the required parameters for the integration algorithm.

11.3.2 Input File

Four different input files should be constructed: one for the steady state simulation (100 days) and three for the dynamic simulations (*dry weather*, *storm* and *rain*). Each dynamic file should consist of 14 days of *dry weather* data followed by 14 days of *dry weather*, *rain* or *storm* data, depending on the simulation.

11.3.3 Output File

The '*simulation benchmark*' recommends analysing the dynamic output data at 15-minute intervals. This corresponds to a *File communication interval* of 0.010417 d in WEST, as seen in Figure 11.5.



Simulation Output

☑ File

File name C:\Benchmark\Simulation\dynamic1.out.txt

File communication interval 0.010417

□ Plot

Plot communication interval 0.05

Figure 11.5: The WEST Simulation Output window.

NOTE: The RK4ASC integration method induces a variable integration step size. Therefore, the dynamic output data will not be given exactly at 15-minute intervals. To minimise the effect of this drawback, the user can either choose another integration algorithm (e.g. RK4) with fixed time step specified as an integer fraction of the desired communication interval. However, this would greatly increase the calculation time. So, it is suggested that a variable step size integration algorithm (with settings as in Figure 11.4) be used. It can be shown that the results (effluent quality index, operating costs) obtained with such an algorithm are consistent with the *'simulation benchmark'* results.

11.3.4 Data Processing

A dedicated Excel spreadsheet has been built (available upon request) to calculate the *'simulation benchmark'* performance criteria from the data compiled in the output file. Also, a benchmark-specific sensor (*BenchmarkSensor*) has been created to calculate, as the simulation runs, the effluent quality index, the effluent quality variance and the effluent violations (number of violations and percentage of time the plant is in violation). These values are reported when the simulation run has finished, but it should be noted that the availability of this information within the experimentation environment means that these quantities can be used directly in WEST optimisation experiments. That is, these quantities can be used directly for controller tuning where the object is to maximise performance based on the benchmark criteria.

11.4 BASIC CONTROL STRATEGY - WEST

The implementation of the basic control strategy is quite straightforward. Two additional sensors need to be incorporated in the graphical editor of the modelling environment, i.e. the standard *DO* sensor model and the *NO3GaussDelayHold* model (Figure 11.6). The dissolved oxygen and nitrate PI-controllers are included by incorporating two *PI* control blocks.

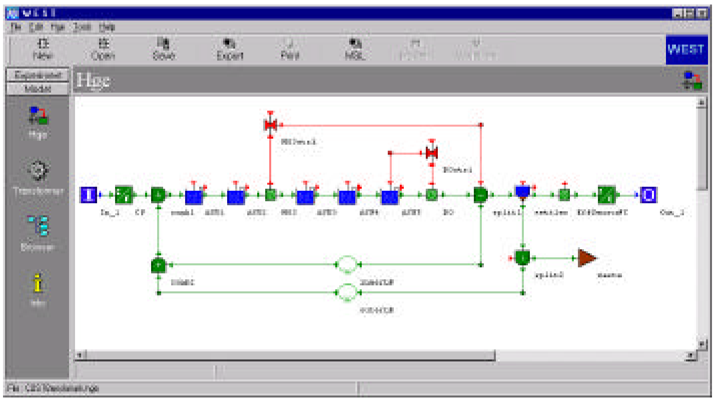
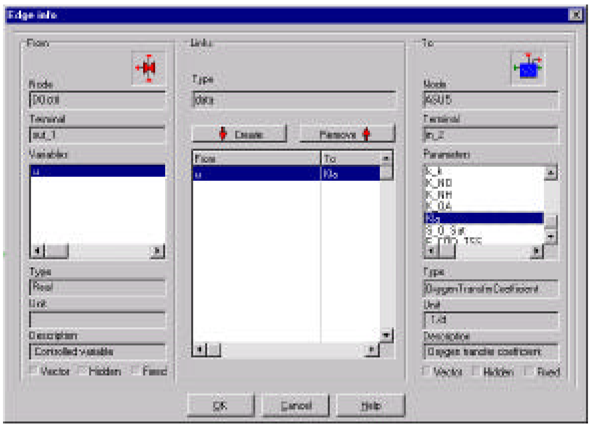


Figure 11.6: Interface layout of the closed loop *'simulation benchmark'* in WEST.

The sensor outputs are linked to the control inputs and the controller outputs are linked to the K_La of the 5th ASU (Figure 11.7) and the internal recycle flow rate, Q_a, respectively. Note that control loops are explicitly modelled in WEST (using sensor and controller blocks) and that information flows and material flows use different line colours (to avoid incorrect flow connections).



Edge info

From
Node: DOctrl
Terminal: out_1
Variables: u
Type: Real
Unit:
Description: Controlled variable
□ Vector □ Hidden □ Fixed

Links
Type: data

From	To
u	Kla

To
Node: ASU5
Terminal: in_2
Parameters: K_X, K_ND, K_NH, K_OA, Kla, S_O_Sat, F_COD_TSS
Type: OxygenTransferCoefficient
Unit: 1/d
Description: Oxygen transfer coefficient
□ Vector □ Hidden □ Fixed

Figure 11.7: Connection specification window showing the linkage of the *DOctrl* PI-controller's output to *ASU5*'s K_La parameter.

After compiling and linking, a new experiment can be built with the new WEST Model Library. In addition to the parameters already specified in the openloop case, the parameters of the nitrate sensor

- delay time
- hold time
- standard deviation of the Gaussian white noise on the measurement
- minimum and maximum range of sensor output to avoid nonsensical negative nitrate concentrations due to the infinite noise tails

and the two PI-controllers

- DO and nitrate setpoints, *y_S*
- no error action, *u_0*
- proportional gain, *K_P*
- integrator time constant, *T_I*
- minimum and maximum control action, *u_min* and *u_max*

need to be specified. Note that for the DO sensor, no parameters need to be specified as the sensor is assumed to be ideal, reporting the actual dissolved oxygen concentration.

As the tight control of the DO and NO_3 controllers significantly affects the numerical properties of the model (the system becomes more stiff), it is necessary to change the accuracy of the *RK4ASC* algorithm to 1.E-9 and to change the minimal step size to 0.0001 d. These changes are necessary to achieve sufficient simulation accuracy.

11.5 CONCLUSION

Setting up the *'simulation benchmark'* in WEST is relatively simple and the compiled code that is generated ensures high calculation performance. Users should be aware of several WEST-specific features including the pros and cons of using a variable step integrator. Use of the suggested integrator results in a deviation from the *'simulation benchmark'* definition, but the use of this integrator ensures that calculation times are minimised, with minimal variation in the output data communication interval and performance data. A *BenchmarkSensor* model was written for this study (and is now included in the standard WEST modelbase) and is used to perform standard benchmark data calculations. An important feature of this specific model is that the generated data can be used for the automatic tuning of user-defined controllers for optimal performance based on the benchmark criteria. However, it should be clear that users are not limited to the benchmark criteria only as the raw simulation data can still be exported from WEST as a standard text file for detailed data interpretation.

11.6 ACKNOWLEDGEMENTS

The work on the Benchmark in WEST is a joint effort by a team of Hemmis employees and researchers at the BIOMATH department of Ghent University (Belgium) that each contributed in different parts of its development. Especially Youri Amerlinck, Diedert Debusscher, Stefan De Grande, Matty Janssen, Jurgen Meirlaen, Dirk Stevens and Henk Vanhooren should be acknowledged for their contributions.