Version 5.43A

SIC^2

Integrated simulation of canals and their control

UMR G-EAU · INRAE

2026-11-02

Table of contents

1 SIC² **1**

2 **3**

I SIC Documentation **5**

3 Preamble **7**
3.1 Documentation of SIC^2 : Simulation and Integration of Control for Canal (or Channels) 7

4 **9**

II User's Guide **11**

5 The project concept and the XML data file **13**
5.1 Network Properties 14

6 The three modes: geometry, hydraulic and result **15**
6.1 Editing Geometry 15
6.2 Editing hydraulic data 15
6.3 View results 16

7 The main window of EdiSIC **17**
7.1 Tree view part 18
7.2 **Search, extend and retract** 18
7.3 **Legend of the tree view** 19
7.4 Network graph part 19
7.5 **Menus and toolbars** 19
7.6 ***File** 19
7.7 ***Tools** 19
7.8 ***Display** 20
7.9 **Bottom of the window** 20

8 The protection key **21**

9 Menu Options > Options **23**
9.1 General 25
9.2 Automatic backup 25
9.3 **Retrieve an automatic backup** 25
9.4 Display 25
9.5 Limiting the number of points in a profile for recording 25
9.6 Link version of DDE Matlab or Scilab 26

10 Specifications of the different licenses of SIC software **27**

11 Installing the USB dongle **29**
11.1 Utilisation monoposte (Single use) 30
11.2 Serveur de licences (License server) 30
11.3 Client réseau (Network client) 31
11.4 **Emplacement du serveur prédéfini (Predefined location server) - Recommended Option** 31
11.5 **Mode Automatique (Automatic mode) - Not recommended** 32
11.6 **Further information** 32

12 Advanced settings of the USB dongle **33**
12.1 Setting the key server 34
12.2 Setting the network client 35

13 Installing parallel dongle **37**

14 Hardware specifications **39**

15 Installation procedure **41**

16 How to use the tools of automatic canal modeling **43**

17 Portion definition **45**

18 Nodes and reaches modification **47**

19 Sections modification **49**

20 Chainage and bottom elevation modification **51**
20.1 Adjust chainage to the network map 51
20.2 Scaling of the longitudinal abscissas 52
20.3 Define constant slope bottom elevation 52
20.4 Shift the Sections Elevations 52
20.5 Change the Interpolation step for the Sections in the Reach . . . 52

21 Section Profiles modification **53**

22 Interpolation of sections **55**

23 Section settings modification **57**

24 Network chart modification **59**

25 Geolocalisation **61**

26 Creating a network (nodes and reaches) **65**

27 Creation of section and description of cross sections **67**
27.1 Reach description 68
27.2 Sections description 69
27.3 **The tab "Profile Section"** 70
27.4 **The tab "Section description and picture"** 70
27.5 **The tab "Analysis"** 71
27.6 **The tab "Geolocation"** 72
27.7 Control input 73

28 Predefined profiles: managing a library of cross sections **75**

29 Import sections and cross profiles in text format **79**
29.1 Format accepted for tables Abscissa - Elevation 81
29.2 **Normal table** 81
29.3 **Transposed table** 81
29.4 Example of imported file 81
29.5 Import command line 84

30 Import geometry from Mascaret or HEC-RAS file **85**
30.1 *Macaret specifications 85
30.2 **Example of imported file** 86
30.3 *HEC-RAS specifications 86
30.4 **Example of imported file** 86

31 Import geometry from Mike 11 file **89**

32 Import geometry from a DTM or a geometer reading **91**
32.1 Format of the import file for points with coordinates (X, Y, Z) . 91
32.2 Trace of the central flow route 92
32.3 Extraction of cross profiles from a DTM 93
32.4 Extraction of cross profiles from a survey of cross profiles 94

33 Add/Change a background map **97**

34 Photos associated with network elements **99**

35 Description of a node and a pond **101**

35.1 Display of upstream and downstream reaches and definition of a pond . . . 102
35.2 Picture . . . 103
35.3 Geolocation . . . 104

36 Modification of network structure 105
36.1 Inserting a node : splitting a reach . . . 105
36.2 Deleting a node : merging two reaches . . . 106
36.3 Converting extrapolated sections into data sections . . . 107

37 Network inconsistencies management 109
37.1 Main errors . . . 110

38 Sections, branches and longitudinal profiles display. 111
38.1 Sections display . . . 111
38.2 Branches display . . . 112
38.3 Longitudinal profile display . . . 114

39 Mesh generation (Talweg) 115

40 Compatibility with older version (.TAL geometry files) 117

41 Export profiles of all sections with X, Y, Z format. 119

42 Change of the bathymetry with an external file 125

43 Parameters for calculating and writing results 127

44 Steady flow calculation parameters 129
44.1 Relaxation coefficient at diversions . . . 130
44.2 Max iterations meshed system . . . 130
44.3 Precision at nodes . . . 131
44.4 Small space step . . . 131
44.5 Debug mode . . . 132
44.6 Allow reversing the direction of the reaches . . . 132
44.7 Allow calculation of supercritical . . . 132
44.8 First time step computing only (in steady flow calculation) . . . 132
44.9 Write in all calculation sections (to allow to initiate an unsteady flow computation . . . 132

45 Unsteady flow calculation parameters 133
45.1 Discretization settings . . . 135
45.2 Numerical resolution method . . . 137
45.3 Write in all calculation sections (to allow a hot start unsteady flow computation) . . . 138

46 Time settings 139

46.1 Steady flow simulation . . . 141
46.2 Unsteady flow simulation . . . 141

47 Settings for results writing 143

48 General settings 145
48.1 Discharge calculation at offtakes . . . 146
48.2 Automatic head loss at expansions . . . 146
48.3 Writing simulation results . . . 146
48.4 How to write the XML file results . . . 147
48.5 Note on possible instabilities of the software . . . 147

49 Devices available in SIC 149

50 Discharge equations at devices 151

51 Overflow (Overtopping a gate) 153

52 GEC-Alsthom gates (AMIL, AVIS, AVIO, Mixtes) 155
52.1 Type . . . 155
52.2 Reference . . . 155
52.3 Decrement . . . 157
52.4 Axis elevation . . . 157
52.5 Sill elevation . . . 157
52.6 Radius . . . 157
52.7 Discharge equations . . . 157
52.8 Maximum opening . . . 157
52.9 Minimum Opening . . . 158
52.10Z upstream max . . . 158
52.11Filtrering S1/S2 . . . 158
52.12Tray offset . . . 159

53 Description of the offtakes 161
53.1 Characteristics of an offtake . . . 162
53.2 Regulate an offtake for obtaining a targeted discharge . . . 163
53.3 Initial discharge . . . 163
53.4 Reference discharge . . . 163
53.5 Sign convention for discharge . . . 163

54 Upstream boundary condition of a model 165

55 Downstream boundary condition of the model 169

56 Management of parameterized variables 175

57 Format of the .par file 177
57.1 Syntax for lines starting with X . . . 177

57.2 Syntax for lines starting with L 178
57.3 Example of .par file 180

58 The project explorer **181**

59 Definition of the roughness (Manning) **183**
59.1 Usual values of the Strickler coefficients n[[Extract from "Hydraulique pour le génie rural", J.-P. Baume, G. Belaud, P.-Y. Vion (Avril 2006)]] 185

60 Definition of lateral flow (infiltration) **187**

61 Definition of the calculation mode at nodes **189**

62 Initialisation of the calculation at diffluences **191**

63 The boundary conditions **193**

64 Cross structures description **195**
64.1 Regulation (steady flow only) : target elevation 196

65 Consistency of the hydraulic scenario **199**

66 Checking of initial discharges **201**

67 Time laws **203**

68 Management of variables (synthetic visualization) **205**
68.1 Entering variant data 206

69 Scenario and variant: operating instructions **209**

70 Running a steady state calculation (FLUVIA) **211**

71 Running an unsteady-state calculation (SIRENE) **213**

72 Overview of results **215**

73 Longitudinal profiles of water elevation and discharge **219**

74 Data available in the results **223**

75 The results tables **225**
75.1 Export data 225
75.2 Data sorting 225
75.3 Display graphics 225
75.4 Other possible actions 227

76 Results for objects of the same type **229**

77 Results time series for a selected object 231

78 Advanced settings of results tables 233

79 Color status used in results tables 235
79.1 Colors used for nodes 235
79.2 Colors used for the sections 235

80 Flags meaning 237
80.1 Meaning of the codes “Error flag” at the offtake 237
80.2 Meaning of the codes “Error Flag” at a section 237

81 Create a video file from a graphic animation 241
81.1 Save animation images in EdiSIC 241
81.2 Convert images to a video file with MakeAVI 241

82 The AMIL method 243
82.1 Controlleur 1 244

83 Exemple of calculation routine CAMIL (Amil gates) 245

84 The AVIS - AVIO method 249
84.1 Controlleur 1 249

85 Exemple of calculation routine CGEC (Avis and Avio gates) 251

86 The PID method 253

87 Example of LPID reading routine 255

88 Example of calculation routine CPID 257

89 Regulation modules 261
89.1 Choosing a regulation module 262
89.2 Editing a regulation module 263

90 General parameters for U 267

91 General parameters for Y 269

92 Targeted values for Y 271

93 General parameters for Z 273

94 Specific Parameters 275
94.1 Definition of specific parameters by text file 276
94.2 Example for a PID controller 276

95 Compilation 277

96 Remarks on regulation modules **279**

97 The ATV method **281**

98 The ATVPID method **285**

99 The BIVAL method **289**

100 The BOLST method **291**
100.1 Create a .txt command file in ASCII or ANSI or other text format 291
100.2 Configuring the BOLST module in SIC 293
100.3 Specific parameters of the BOLST module 293
100.4 **Scaling Parameter** 294
100.5 **Constant additive** 294
100.6 **Interpolation method** 294
100.7 **Write to LST file** 294
100.8 Restriction 294

101 The BOMAT method **299**

102 The BOSCIL method **301**
102.1 Create a command matrix with Scilab 301
102.2 Configuring the BOSCIL module in SIC 301
102.3 Specific parameters of the BOSCIL module 302
102.4 **Scaling Parameter** 302
102.5 **Interpolation method** 302
102.6 **Write to LST file** 303
102.7 Restriction 303

103 The DIGEST method **305**

104 The DSS method **307**

105 The FILEX method (FILe EXchange) **309**

106 The LP method **311**

107 The MATLAB method **313**

108 The PLOT method **319**

109 The POMPES method **321**

110 The PRINT method **327**

111 The QSCP method **329**

112 The SCADA method **331**
112.1 Global computing architecture . . . 331
112.2 Variables handled by the SIC Scada regulation module . . . 331
112.3 Files managed by the digester and Scada_SIC . . . 333
112.4 **Architecture of the subdirectories:** . . . 333
112.5 **Sub-directory Commande** . . . 333
112.6 **CdeF.txt:** . . . 333
112.7 **Config.txt:** . . . 333
112.8 **Sub-directory Csg** . . . 334
112.9 **Csg.txt:** . . . 334
112.10 **Sub-directory CsgRepli** . . . 334
112.11 **CsgRepli.txt:** . . . 335
112.12 **Sub-directory Data** . . . 335
112.13 **Data.txt:** . . . 335
112.14 **Sub-directory ParamDigesteur** . . . 339
112.15 **Param.txt:** . . . 339
112.16 **SaveCsg** . . . 340
112.17 **SaveData** . . . 340
112.18 Example of controller development . . . 341
112.19 **Development under SIC** . . . 341
112.20 **Craation of the Lieu.txt file** . . . 341
112.21 Activation of the module Scada_Sic . . . 342
112.22 Editing the control data for the SCADA module . . . 343
112.23 Interface of the Scada module of Sic . . . 343

113 The SCILAB method **345**

114 The STEPS method **351**

115 The STOP method **353**

116 The USER1 to USER9 methods **355**

117 The WDLANG method **357**

118 Presentation **363**

119 Adding an Quality module to an hydraulic scenario **365**

120 Definition of the quality classes **367**
120.1 Drift quality classes . . . 367
120.2 Fix quality classes . . . 368

121 Initial conditions for fix classes **371**

122 Exchange laws **373**

123 Boundary conditions **375**

124 Meteorological parameters **377**
124.1 Mask coefficients 378

125 Calculation settings **379**
125.1 Iteration parameters 380
125.2 Simulation settings 381
125.3 **Transient simulation** 381
125.4 **Text-dump of last time-step** 381
125.5 Time 381

126 How to make a network **383**

127 How to perform a steady state simulation (FLUVIA) **385**

128 How to perform a transient simulation (SIRENE) **387**
128.1 Creating a new scenario or variant 387
128.2 Import of initial condition 388
128.3 Edit Transitional Computing Events 388
128.4 Calculation Parameters 389
128.5 Start simulation 389

129 Simulating a check valve with a WDLANG module **391**
129.1 Creation of the module and connection to the devices 391
129.2 Writing the regulation script 401
129.3 Simulation parameters and results 402

130 Introduction **405**

131 Import of .TAL and .FLU files in batch mode **407**
131.1 Syntax 407
131.2 Parameter Description 407
131.3 Command line limitation: using EDISIC.MAC 408
131.4 Command Line Example 408

132 Import sections in text format **409**
132.1 Syntax 409
132.2 Parameter Description 409
132.3 Import modes 410
132.4 Command line limitation: using EDISIC.MAC 410
132.5 Command Line Example 410

133 EdiSIC macro mode : an alternative to the command line **411**
133.1 Syntax of EDISIC.MAC file 411

134 Importing an initial condition for a transient flow simulation **413**

134.1Command line syntax: 413
134.2Description of the command line parameters 413
134.3Usage example on the file C:/titi/toto.xml 414

135Running TALWEG, FLUVIA and SIRENE from the command line 415
135.1Meaning of the arguments: 415
135.2[**XML file name**] 415
135.3[**Interface mode**] 415
135.4[**Fluvia and Sirene options**] 416
135.5Usage example 416

136SicExport: text-format results export utility 417
136.1Usage syntax 417
136.2**Mandatory parameters** 418
136.3**Optional parameters** 418
136.4***Options on the file headers: the /quiet parameter** 418
136.5Example 419

137Appendix 421

138Supercritical discharge modelling 423

139Dry bed modelling 425

140Triangular weir modelling 427
140.1Weirs Laws 427
140.2Comparison of the results obtained by using the two laws on an example 428

141Equation and devices calibration (gates and weirs) 429
141.1Devices modelisation (gates and weirs) (Script to copy in a file Qouvrage.m) 429
141.2Intermediate function used for calibration (script to copy in a f.m file) 435
141.3Calibration of discharge coefficient (script to copy in a file calage_r.m) 436

142Modeling of an in-line pump 437

143Abacus of the Gec-Alsthom gates 439

144TAL file format (SIC version 4.33) 445
144.1Network description (1 line) 445
144.2Description of a node 446
144.3Description of a reach (2 lines + the sections of the reach) 446
144.41st line 446

144.52nd line . . . 446
144.6Description of a section (variable number of lines) . . . 447
144.71st line . . . 447
144.82nd line and following . . . 447
144.9Description of a branch . . . 448
144.10st line . . . 448

145FLU file format (SIC version 4.33) 449
145.1The comment line . . . 449
145.2Line for the calculation parameters . . . 449
145.3Weir line . . . 450
145.4Gate line . . . 450
145.5Gec-Alsthom gate line . . . 450
145.6Offtake line . . . 451
145.7Line for the Strickler coefficients . . . 451
145.8Line for the lateral flows . . . 451
145.9Line for the downstream conditions . . . 452
145.10ine for the diffluences . . . 452
145.11ine for calibration elevations . . . 452
145.12ine for the calculation mode at the nodes . . . 452

146SIR file format (SIC version 4.33) 453

147 461

III Theoretical concepts 463

148Description of the hydraulic network 465

149Choice of reaches 467

150Choice of branches 469

151Classification of reaches 471

152Identification of the upstream and downstream nodes of the model 475

153Description of data cross sections 477

154Abscissa-elevation 479

155Width-elevation 481

156Parametric form 483

157Siphons 485

158 How to order the reaches in the unknown vector to get an upper triangular matrix H **489**

159 Filling the H, B and A blocks **491**

160 Singular sections **493**

161 Computational sections **495**

162 Introduction **497**

163 Transformation of a data section into width-elevation format **499**

164 Interpolation of computational sections **501**

165 Interpolation of singular sections **503**

166 Sort algorithm for the reaches of the loop before the steady flow computation - Initial solution **505**

167 Correcting equations **507**
167.1 **For all the nodes of the loop that are not downstream nodes of the loop:** . . . 508
167.2* **Discharge continuity** . . . 508
167.3* **Water elevations equality** . . . 508
167.4 **For all the downstream nodes of the loop:** . . . 508

168 Resolution of the linear system of the correcting equations **509**

169 Cross structure equations **511**

170 CEM88(D) : Weir / Orifice (high sill elevation) **513**
170.1 Weir - Free flow ($h_1 < W$ and $h_2 \leq \frac{2}{3}h_1$) . . . 513
170.2 Weir - Submerged ($h_1 < W$ and $h_2 \geq \frac{2}{3}h_1$) . . . 513
170.3 Orifice - Free flow ($h_1 \geq W$ and $h_2 \leq \frac{2}{3}h_1$) . . . 514
170.4 Orifice - Submerged . . . 514
170.5 **Partially submerged flow ($h_1 \geq W$ and $\frac{2}{3}h_1 < h_2 < \frac{2}{3}h_1 + \frac{W}{3}$)** 514
170.6 **Totally submerged flow ($h_1 \geq W$ and $\frac{2}{3}h_1 + \frac{W}{3} < h_2$)** . . . 514

171 CEM88(V) : Weir / Undershot gate (small sill elevation) **517**
171.1 Weir - Free-flow . . . 517
171.2 Weir - Submerged . . . 517
171.3 Undershot gate - Free-flow . . . 518
171.4 Undershot gate - Submerged . . . 518
171.5 **Partially submerged flow** . . . 518
171.6 **Totally submerged flow** . . . 519

172 CEM02(V) et CEM02(D) : Trapezoidal sills and gates **521**

173 Equation at a singular section 523

174 Regulator 525

175 Offtake equations 527

176 Offtake downstream conditions 529

177 Numerical methods 531

178 Recapitulation of the Newton's method 533

179 Resolution of the water surface profile equation 535

180 Computation of the critical depth 537

181 Computation at a singular section 539

182 Computation of the offtake opening 541

183 Unit 2 - Steady flow 543

184 Introduction 545

185 Equation of gradually varied flow in a reach 547

186 Differential equation of the water surface profile 549

187 Saint-Venant's equations 553

188 Continuity equations 555

189 Dynamic equation 557

190 Initial and boundary conditions 561

191 Semi-implicit discretization 563

192 General equations of the scheme 565

193 Continuity equations 567

194 Dynamic equation 569

195 Performance indicators 573

196 Volume indicators 575

197 Time indicators 577

198 Introduction **579**

199 Double sweep method - General case **581**

200 Singularities **585**

201 Offtakes of the form Qp(t) **589**

202 Offtakes of the form w(t) **591**

203 List of symbols **593**

204 Exponential growth with fixed coefficient **595**
204.1 Definition . . . 595
204.2 Classic use . . . 595
204.3 Specifications . . . 595

205 Sediment transport **597**
205.1 Generalities . . . 597

206 Engelund-Hansen (1967) **599**
206.1 Engelund-Hansen formula . . . 599
206.2 Specifications . . . 599

207 Algal detachment in response to flushing **601**
207.1 Definition . . . 601
207.2 Characteristics . . . 601

208 Algal growth (PhD Thesis, O. Fovet, 2010, p.101) **603**
208.1 Definition . . . 603
208.2 Variables and parameters . . . 603
208.3 Limiting nutrient . . . 604
208.4 Specifications . . . 604

209 Algual detachment in response to flushing with reference shear stress equal to the first time step **605**
209.1 Definition . . . 605
209.2 Specifications . . . 605

210 Evolution of the temperature **607**
210.1 Definition . . . 607
210.2 Specifications . . . 607

211 Exchange laws **609**

212 Introduction **611**

213 Advection–Diffusion Transport Equation **613**
213.1 Quality classes . . . 613
213.2 Advection-Diffusion equation . . . 613
213.3 Calculation of the exchange rate . . . 614
213.4 Numerical resolution of the equation . . . 614

214 Concepts and equations at a node **615**
214.1 Inflow (positive discharge) . . . 616
214.2 Outflow (negative discharge) . . . 616
214.3 **Reaches** . . . 616
214.4 **Offtakes** . . . 616
214.5 **Infiltration at the node** . . . 616
214.6 Storage flow in the pond . . . 616
214.7 Mass balance at the node . . . 616
214.8 Distribution coefficient k . . . 617
214.9 **Outflows are homogeneous** . . . 617
214.10 **Outflows are heterogeneous and the node contains a pond** 617
214.11 **Outflows are heterogeneous and the node does not contain a pond** . . . 617

215 Network computation **619**

216 Steady state calculation **621**
216.1 Calculation at a section . . . 621
216.2 **Transport equation** . . . 621
216.3 **Integration using fourth-order Runge-Kutta méthod** . . . 621
216.4 Calculation at a node . . . 622
216.5 Continuity equation at a node . . . 622
216.6 **Newton's method** . . . 622
216.7 **Calculation for a node with a pond** . . . 622
216.8 **Calculation for a node without a pond** . . . 622

217 Transient calculation **625**
217.1 Calculation at a section . . . 625
217.2 **Discretization for Preissmann Scheme** . . . 625
217.3 **Resolution by conjugate gradient method** . . . 626
217.4 Discretization at a node . . . 626
217.5 **Mass balance at a node** . . . 626
217.6 **Newton's method for a node with a pond** . . . 626
217.7 Calculation at a node without a pond . . . 627

218 Transiant calculation for fix classes **629**

219 Temperature simulation **631**
219.1 Transport and exchange equation . . . 631
219.2 Calculation of the heat power term . . . 631

220 Theoretical concepts **635**

221 Introduction **637**

222 **639**

IV Commercial informations **641**

223 Commercial information **643**

224 **645**

V Hydraulic calculations with Cassiopée **647**

225 Hydraulic calculations with Cassiopée **649**

Chapter 1

SIC²

Integrated simulation of canals and their control

This manual is the complete documentation of the **SIC²** software (version 5.43A), developed and maintained by UMR G-EAU (INRAE). It is generated from the documentation website; the website remains the reference.

SIC² 5.43A — 2 November 2026

Chapter 2

Part I

SIC Documentation

Chapter 3

Preamble



UMR G-EAU Inrae Montpellier 361,
rue J.-F. Breton B.P. 5095 34196
Montpellier Cedex 5, France

3.1 Documentation of SIC^2 : Simulation and Integration of Control for Canal (or Channels)

- Guide Utilisateur SIC 5.x
- Guide Théorique

This documentation relates to : - Version 5.43 - February 2025 (validated)

The latest documentation is available at this URL: http://sic.g-eau.fr

The SIC^2 software has been designed and developed by the Operationnal Management Team of UMR G-EAU.

For all questions or comments, please contact: - David DORCHIES - Pierre-Olivier MALATERRE - Gilles BELAUD (sediment and quality module)

UMR G-EAU

Inrae Montpellier 361, rue J.-F. Breton B.P. 5095 34196 Montpellier Cedex 5, France

- Tél.: (33 ou 0) 467 04 63 56
- Tél.: (33 ou 0) 467 04 63 22
- Email: sic@g-eau.fr
- Web: http://sic.g-eau.fr

Chapter 4

Part II

User's Guide

Chapter 5

The project concept and the XML data file

An XML data file contains one project.

A project consists of: - A network also called geometry (Definition of nodes, reaches and sections) - One or more hydraulic scenarios (Definition of offtakes, cross structures, infiltration and Stricklers) - One or more regulation modules

Each hydraulic scenario may include: - One or more variants (Change management data from the data of the scenario) - A quality module

Each scenario, a variant of scenario or quality module of a scenario may include a simulation result.

5.1 Network Properties

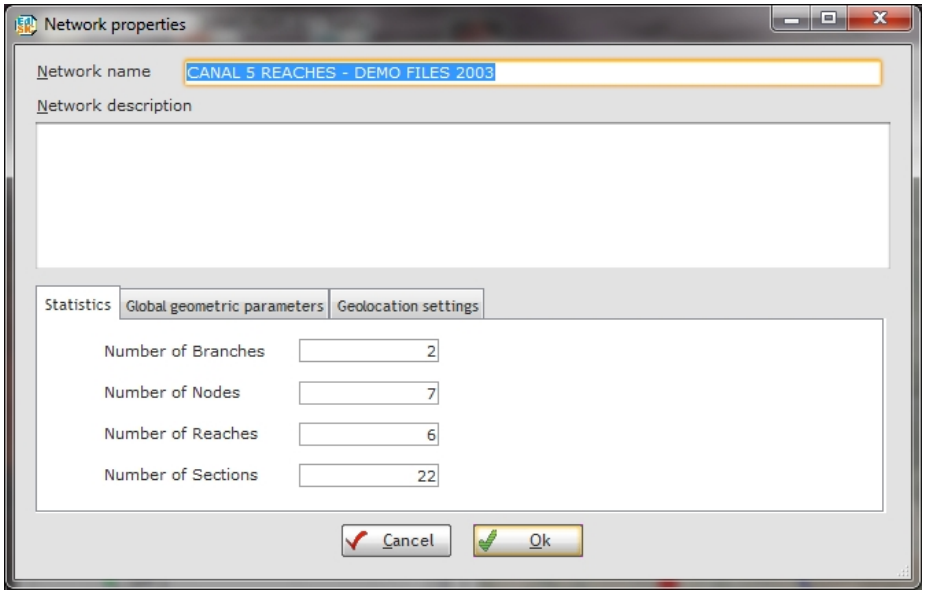


Figure 5.1: propreseau

The project name is defined in the "Network Name" field situed in the "Network Properties" window. You can add free descriptive text to archive information about this project.

We can define an homothety factor in Y and Z for cross profiles of a network. These homothety factors may also be defined reach by reach and section by section. This can be useful for recalibration of such networks.

A global space step can be defined here. It will be applied to all reaches of the network, except if this space step is redefined (different from 0) at the reach and/or section level.

This window also provides access to network statistics (number of nodes, sections and reaches) that check if it is within the operating limits of the SIC license. We advise to refer to the theorical manual to check the definition of these concepts which are different for each software.

Global geolocation settings can be set here. The units in decimal degrees or in metric units (UTM), as well as the angular reference for the geolocation of the cross sections (with the choices of Geographic North, the central line of the reach or the segment drawn within the reach).

Chapter 6

The three modes: geometry, hydraulic and result

The EdiSic interface operates in three successive modes: - Editing Geometry - Editing hydraulic data - Visualization of results

Navigation between these different modes is facilitated by the window “Project Explorer”.

6.1 Editing Geometry

In this mode, the user can : - import a .TAL file of older versions of SIC, - modify a network - or enter a new network.

Entering the network allows to define: - The reaches and their connecting nodes - The sections in the reaches - Among these sections, the singular sections that will host the cross structures in the hydraulic mode - The profiles of all sections (With definitions of minor / medium and major beds)

Before moving to hydraulic mode, the application will propose: - to check the completeness and consistency of the network entered - and launch the mesh network (TALWEG).

6.2 Editing hydraulic data

EdiSic allows to enter several hydraulic scenarios for one geometry.

In this mode, the user will define the following data: - Features of the offtakes in the nodes (Devices, and boundary conditions) - Features of cross structures located on the singular sections. - Roughness coefficients (Manning) in the reaches for minor and medium beds - Infiltrations - Regulation Modules - Simulation

parameters (including the definition of the results which should saved during the simulation for later viewing)

After verifying the completeness and consistency of data entered, the interface allows the launch of a hydraulic simulation in steady flow (FLUVIA).

If the simulation was successful, it is then possible to import water line obtained in steady flow in a new scenario to start a hydraulic unsteady flow simulation (SIRENE).

6.3 View results

As its name implies, this mode allows you to view the results generated by a hydraulic simulation in steady flow or unsteady flow. Results are represented as data tables and graphs.

Chapter 7

The main window of EdiSIC

The main screen shows the network in two ways : - on the left with a tree-like representation - on the major part of the right with a network graph

Selections can be made on either of two representations for most features.

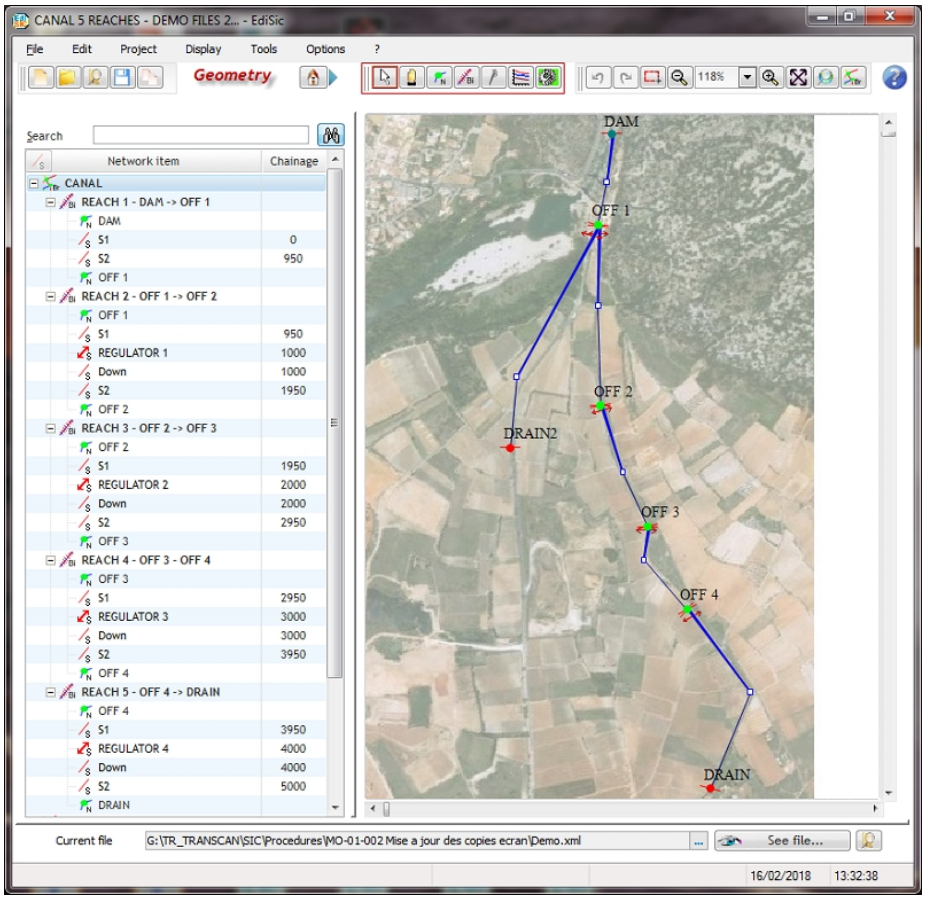


Figure 7.1: premiere_tal

7.1 Tree view part

7.2 Search, extend and retract

The “Search” allows you to search based on part of the name of an object, or the exact or approximated abscissa of it.

Button to the left of the title bar of the tree allows extend and retract the various levels of the network.



Figure 7.2: doc4

7.3 Legend of the tree view

Icon | *Name* | *Description* |



| Branche | Continuous series of reaches without diffluence or confluence |



| Reach | Continuous portion of the stream between two nodes |



| Node | Point of reach demarcation and of offtakes integration |



| Section | Description of a transverse profile |



| Singular Section| Description of a transverse profile with possibility to add devices (cross structure) |

7.4 Network graph part

7.5 Menus and toolbars

All modes are accompanied by three toolbars that combine features found in the menus of the same name:

7.6 *File



Figure 7.3: doc10

- New document
- Open an existing document
- Open Windows ® explorer in current directory
- Save the document
- Save the document in an other file

7.7 *Tools



Figure 7.4: doc11

The available tools vary depending on the mode (geometry, hydraulic, result). They will be described in the corresponding parts of the documentation.

7.8 *Display

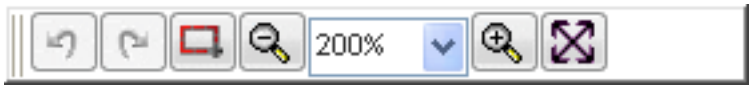


Figure 7.5: doc13

- Cancel previous operation
- Restore the canceled operation
- Select all objects in the network
- Zoom out
- Set the zoom
- Zoom in
- Adjust the zoom to the network (Network full page)

7.9 Bottom of the window

This part is used to display the current file with possibility of opening a new file, display the file in the default XML editor or open explorer in the current folder.



Figure 7.6: doc14

This part is also used to display graphs of longitudinal profiles in the different modes.

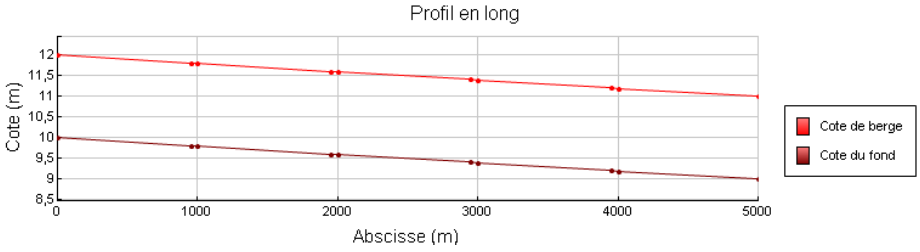


Figure 7.7: doc15

Chapter 8

The protection key

The parameters of the protection key are editable in the “Protection key” window (Options menu> protection key).



Figure 8.1: dungle

Mode without dongle, no normal use of EdiSIC in geometry mode (warning messages) or Talweg possible. But the rest of the application is usable.

When using the software, if the key becomes unavailable, the software will automatically switch without key.

If the key is available again, it will then go recheck in this interface the type of key used.

Chapter 9

Menu Options > Options

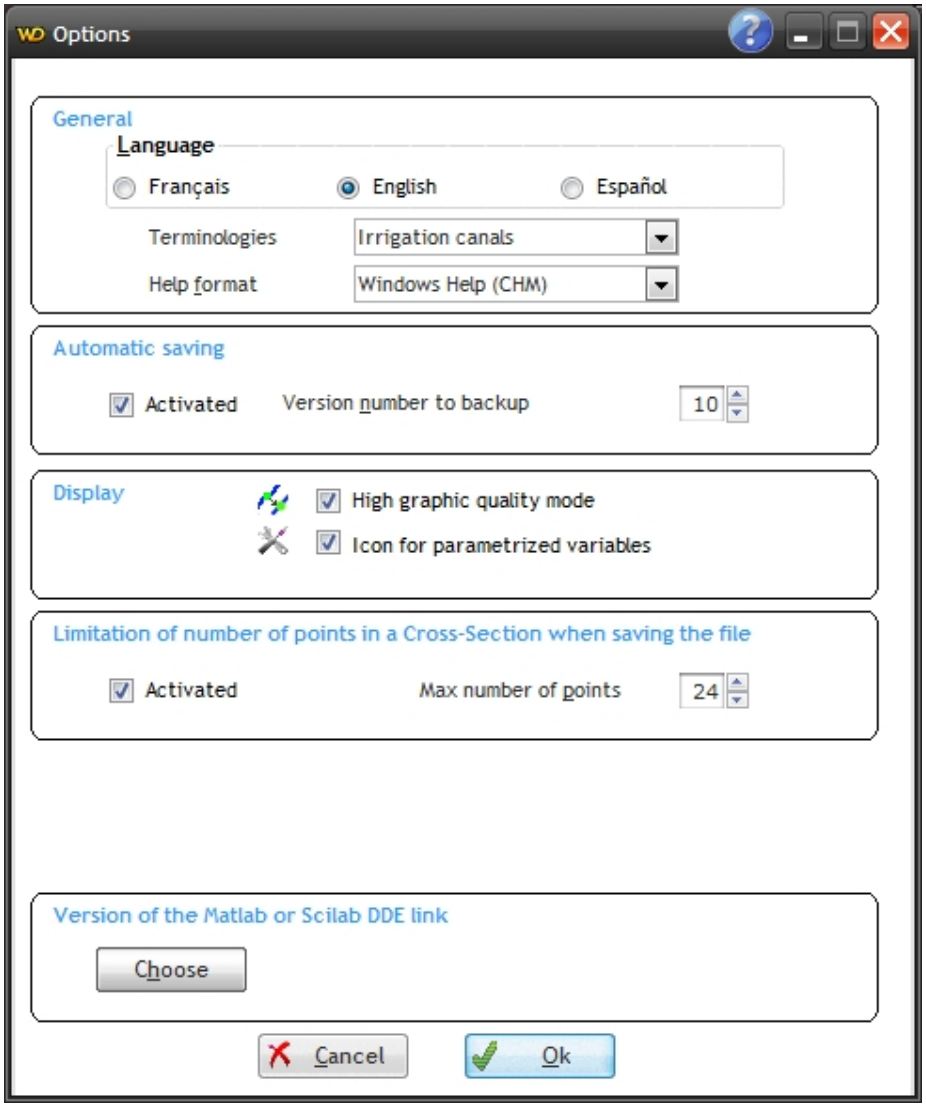


Figure 9.1: options

9.1 General

Language : The interface is translated in L'interface in French, English and Spanish. Portions of the software may not yet fully translated into English and Spanish, they then appear in French.

9.2 Automatic backup

- The button "*Activated*" can enable or disable the automatic backup of the file being input.

9.3 Retrieve an automatic backup

After a software crash, EdiSic proposes to recover the last backup file automatically. If the user agrees to open the recovered file, the software creates a subdirectory 'Sauv' in the file location and copy both file versions with the following names: - FileName_Récupéré.xml: copy of the file recorded by the automatic backup and that has been recovered. - FileName_Original.xml: copy the original file and that is the last recording made voluntarily by the user.

9.4 Display

- "High quality graphics mode": Improves the display of the network on the screen. Disable this option if the display of the network is too slow or choppy.
- "Icon for parametrized variables": Enables the display of the button "variable set" on the fields where this option is available.

9.5 Limiting the number of points in a profile for recording

When importing or entering profiles with more points than the limit set here, the software will automatically remove the necessary number of points by minimizing the information loss (minimizing height between the point and removed the segment connecting the previous paragraph with the following point). When saving the corresponding file, the removed points will be indicated in a trace window.

- "Activated": to control the registration of a geometry that the number of points of cross sections does not exceed the number defined below.
- "Max number of points": The default limit is 24 points and corresponds to the current limit of the mesh Talweg. Profile greater than 24 points will cause an error in the meshing program (Talweg).

9.6 Link version of DDE Matlab or Scilab

Tle Matlab and/or Scilab DDE links are not compatible with each other, and use dll libraries that have the same names but are not compatible. This is why we offer various versions of Fluvia and Sirene generated with various versions of the DDE link for Matlab or Scilab. You must choose the one that suits you, depending on the version that is installed on your computer. If you have multiple versions installed, it will probably even be necessary to disable versions you do not want to use, to avoid conflicts. It is not necessary to uninstall these versions, a simple temporary renaming of the directory where is located this version or these versions will be sufficient.

Chapter 10

Specifications of the different licenses of SIC software

There are several licenses for SIC software : - Professional version - Research and Educational version

The only technical difference between the Professional version and the Research and Educational version is the maximum size of the systems that can be modeled. Otherwise all options are available in both versions. The Professional version package also includes 3 day training for the user.

Since most of the calculation programs are in FORTRAN without automatic memory allocation there are still some limitations in the Professional version. But we can extend them as much as you want. We can model any system in the world, from the small scale model of the SupAgro engineering school in Montpellier, France, to the Amazone River in Brazil, Congo River or the Narmada canal in India. For your information the default limitations are:

Limitation on | Professional version | Research and Educational version |
number of branches (in case you have confluences, diversions, loops) IBR | 75 | 5 |
number of reaches (a reach is a portion of the canal or river between 2 offtakes, inflows or outflows) IB | 350 | 20 |
number of data cross sections (cross sections where you describe the geometry of the sections) NSD | 5000 | 200 |
number of calculation cross sections (cross sections where hydraulic values will be calculated ; they can be data sections you enter or additional automatically interpolated sections depending on the space step you specified for each reach) NSC | 20000 | 250 |

number of singular sections (where there is one or several weir, gate, Gec Alsthom gate, pump, etc.) NSS | 200 | 10 |
number of cross devices (weir, gate, Gec Alsthom gate, pump, etc.) NTO | 500 | 10 |
number of regulation modules (an algorithm that can move one or several control variables using one or several measured values, in order to control one or several controlled variables) NBCM | 100 | 20 |
number of control variables NBUM | 100 | 50 |
number of controlled variables NBYM | 100 | 50 |
number of measured variables NBZM | 500 | 500 |
number sediment classes (for the sediment transport module) NCL/NCLG | 6/11 | 6/11 |

Chapter 11

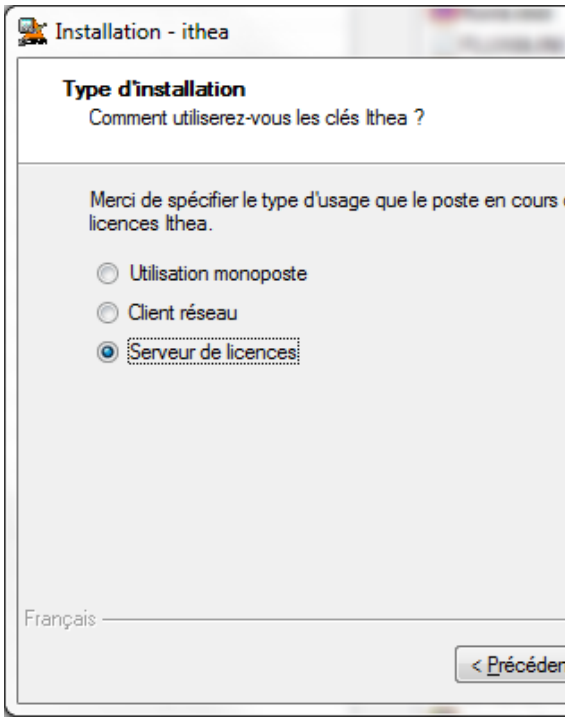
Installing the USB dongle

If your license of SIC uses a USB dongle, you must install on your computer drivers and services to access these dongles. This installation is performed by the program setup_ithea.exe.

At first use, SIC automatically offers to install the service dongle Ithea, since it detects it is not already installed on the computer. To manually install the service USB dongle, you have to launch the program setup_ithea.exe. This program is present in the exe subdirectory of your installation of SIC.



| Caution! The dongle driver installation must be performed before connecting the USB dongle to the computer! Otherwise, Microsoft Windows installs a default driver that prevents proper installation of Ithea driver. One will, in this case, manually remove the driver in the system settings of the computer. |



During installation, three types of facilities are proposed:

The type of installation is not fixed and can be changed without having to redo the installation (see advanced settings of the USB dongle).

11.1 Utilisation monoposte (Single use)

Use it if you want to connect the USB key directly on your computer. In this case, click once on the next, and the installation is complete. If you run SIC under a Wine session in a Linux computer this local connection will not work properly. You have to use the license server option described here after.

11.2 Serveur de licences (License server)

If you want to connect the USB key directly to the computer and you want that other computers on the same network can use this key. To install a license server, select the option and click Next, the installation is complete. A USB key is configured with a given number of tokens, specified in your contract. You can also access this server using a remote connection, for example through VPN.

11.3 Client réseau (Network client)

Use it if you use a key attached to another computer on the network. If you run SIC under a Wine session in a Linux computer the remote license server will also be available. To do this, select the option and then click Next. A configuration window appears:

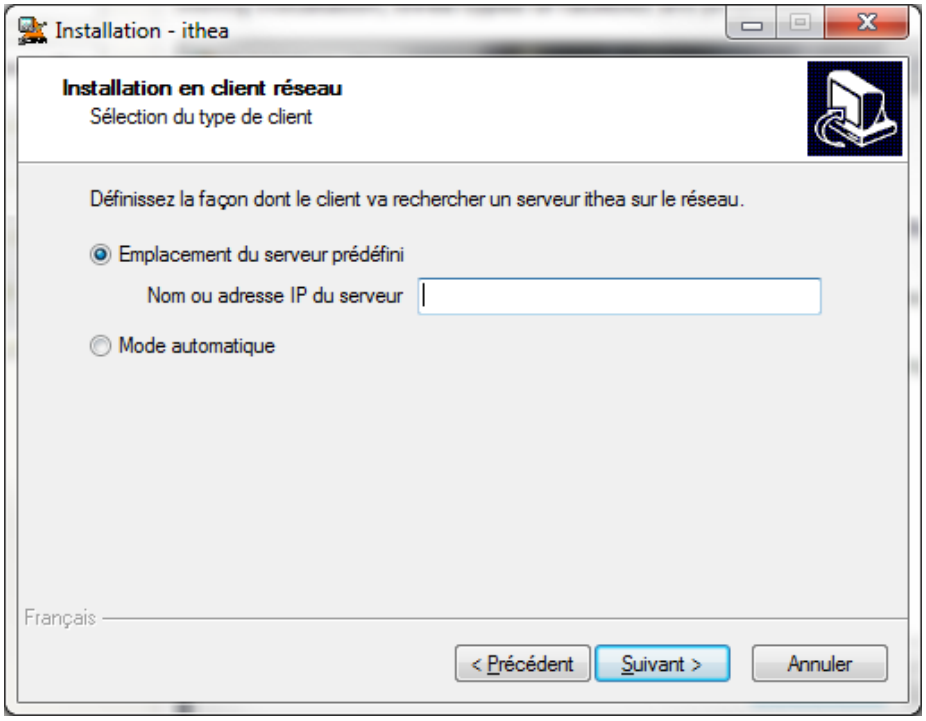


Figure 11.1: doc73

11.4 Emplacement du serveur prédéfini (Predefined location server) - Recommended Option

- If you choose this option, you must enter in “Nom ou Adresse IP du serveur”, the name or IP address of the license server.

11.5 Mode Automatique (Automatic mode) - Not recommended

- This mode allows you to search directly the computer station configured license server on the network. However, depending on the type of network access keys can be slowed or even not work.

11.6 Further information

- Once SIC is installed, additional information is available on this key, the drivers and their settings, in the file Actikey.pdf and Ithea-os_en.pdf under SIC/DOC. In particular, the setting of the Windows firewall is discussed, in order to solve possible problems of blocking the port 1331 used by this key. You have indeed to allow this port in incoming connections.

Chapter 12

Advanced settings of the USB dongle

Ithea provides a utility to change the configuration of the USB dongle service. It is available on `C:/Program Files/ithea/instsvc.exe` and must be run as administrator.

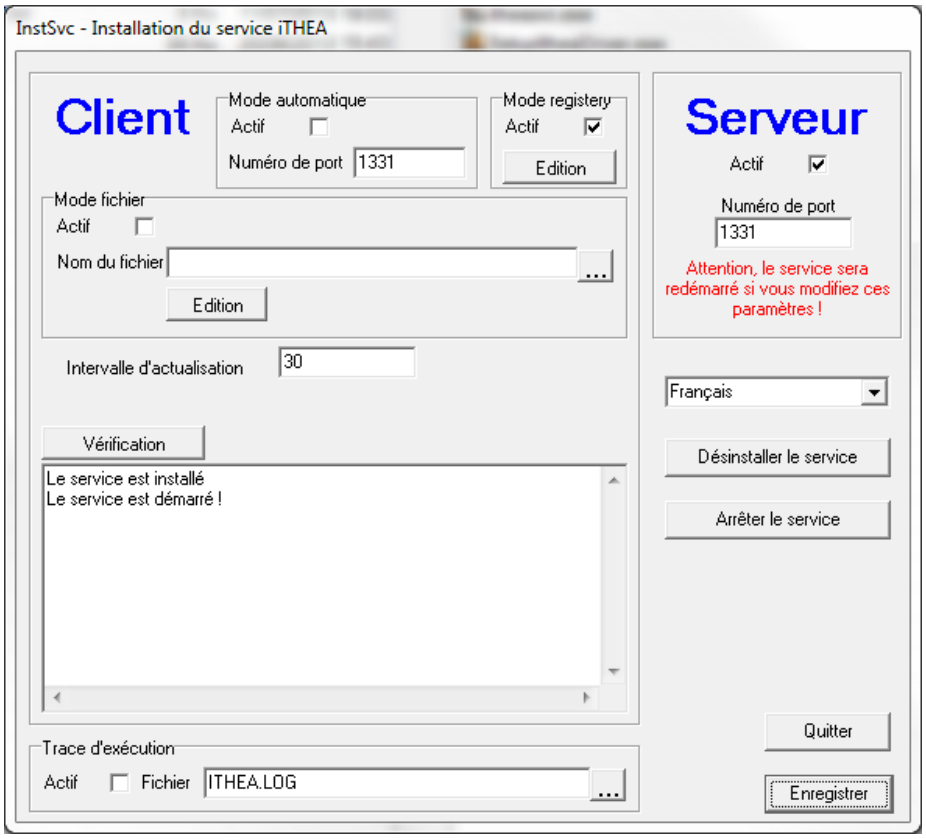


Figure 12.1: doc74

12.1 Setting the key server

To use the computer as a network server, you must check “Actif” in the box “Serveur” at the top right of the window.

The default listening port of the server is 1331. This field is to change if you have another application that may be using the same port.

The proper functioning of the server can be checked using the command `netstat -an`. The 1331 port is LISTENING should appear in the active connections:

```
Connexions actives
  Proto  Adresse locale         Adresse distante       État
   TCP    0.0.0.0:1331           0.0.0.0:0              LISTENING
```

If this line does not appear, check that the port is open for incoming connections in the firewall settings of the computer.

12.2 Setting the network client

If the dongle is on a server, it is necessary to specify the IP address of the server. In the “Client” part, you must check “Actif” in the box “Mode Registry” and then enter the IP address and port of the server key in the window accessible from the button “Edition”:

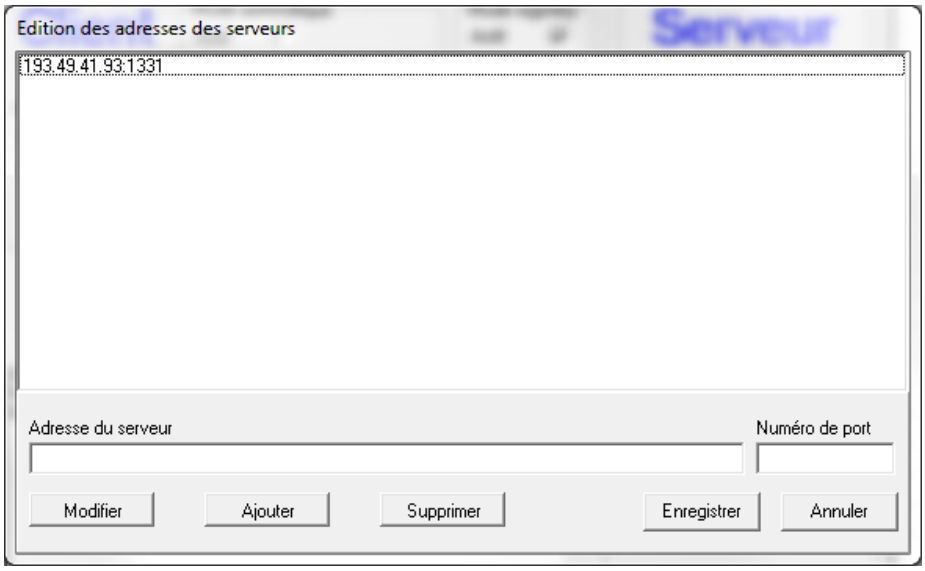


Figure 12.2: doc75

If the dongle server is likely to change, it is possible to enter multiple addresses.

Chapter 13

Installing parallel dongle

This notice applies to you only if your license includes the SIC dongle that plugs into the [?parallel port] of the computer. To use a USB dongle refer to the relevant documentation.

If SIC is installed on a computer run under Windows NT, 2000 or XP, you have to install the driver provided at the FAQ web page of the CANARI web server. You can download there a “Drivers.exe” file. By running the Drivers.exe file you will be able to copy the drivers files that you need depending on your computer (operating system). You can copy them in any permanent or temporary subdirectory:

This operation is just copying the drivers files, but not installing the drivers.

You must then install the drivers by running the install.exe file that you copied. The wizard for this lattest operation is in French only but you just have to decide to install (or des-install) and activate (or desactivate) the corresponding drivers.

If you want to use the dongle from a remote computer (on the local network) you must install a dongle server, also provided in the Drivers.exe file used above. In this case you can have one or several accesses to the dongle at the same time, depending on the configuration set by Cemagref in the dongle. By default, the dongles provided by Cemagref are configured to be only local dongles, but on special agreements such as for teaching for example we can configure remote dongles with several simultaneous connections.

Chapter 14

Hardware specifications

The SIC software is designed to run on an IBM PC-AT or on a compatible microcomputer IBM PC or PS/2.

The SIC 2.1 version only existed under the MS/DOS operating system.

The SIC 3.1 version was supplied both for DOS and Windows, in order to assure the progressive transition of all users toward Windows.

From version SIC 3.5 only the Windows version was supplied. This version was a 16 bits version and could thus run under Windows 3.1, 3.11, 95 or 98 (and Windows NT).

The present SIC version, is a full 32 bits version only able to be run under Windows 98, NT, 2000, XP and latter versions of Windows.

The 32 bits version is faster and more reliable, especially on computers equipped with a Pentium 2 microprocessor (and later version). In addition, this version allows multi-tasking, thus making possible to execute simultaneously Ediflu and Fluvia, for instance. It can also perform several SIC programs at the same time. However this kind of use is not advised and we cannot guarantee the way in which Windows manages the protection of files if used at the same time by different applications.

The Windows SIC versions can model larger systems than the old DOS versions could. Furthermore, the Windows version has a new topology and geometry editor (Edital program).

SIC requires a mathematical co-processor (automatically installed on the recent Pentium processors) and an EGA or VGA screen. RAM must be at least 128 MB (Mega-Bytes). About 80 MB is required on the hard disk for the three units of the software. A nice minimum configuration would be a Pentium IV, with 64 MB of RAM and a 1 Giga hard disk.

The software also has some graphical output possibilities since it generates DXF AutoCad, Wmf and Bmp files.

The topographic unit of the Software (Unit I) is protected by a dongle (electronic key) that must be connected to the parallel output port of the computer. If a printer (or any other peripheral) is also connected to this port, it must be switched on and the key must be placed directly on the computer output. The dongle can also be used through the network if it has been configured this way by Cemagref. Check the FAQ pages on the CANARI web server (http://www.canari.free.fr) for the procedure. The connections must be made when the computer and peripherals are switched off. In case of deterioration of this key, you must contact Cemagref to buy a new one (returning the old key). In case of lost, Cemagref does not assure any replacement.

In case your computer does not recognise the dongle, specially under Windows NT, 2000 or XP, please install the driver provided at the FAQ web page of the CANARI web server. The installation procedure is explained in the Installation Procedure page. **

Chapter 15

Installation procedure

All necessary files to install and start the SIC software are contained on a CD ROM, a USB key or can be downloaded on the CARANI web server (protected by a login and password). SIC software is installed from an exe file. To install the SIC model on the hard disk you must run this exe file.

- run this exe file (named sic???.exe where ??? contains the name and number of the version: example siccemagref410.exe)

- follow the instructions and answer the questions of this installation utility (in French or English). - the Setup wizard creates a " SIC " program group and the necessary icons.



| With Windows Vista and above, you must ensure that the user will have write permissions in the SIC installation folder. Software configuration is defined in ini files located in the "exe" subfolder of the installation. These files will need to be modified by the program. |

Whatever language you choose for the installation procedure, the SIC software libraries will be installed and therefore available in French, English and Spanish. To shift from one language to another one you must go into the "Options" menu.

To install the USB protection key, depending on the type of key:

- Installing USB protection key
- Installing parallel protection key

Chapter 16

How to use the tools of automatic canal modeling

The tools of automatic canal modeling allow to define in some mouse clicks the parameters of a network part (nodes, reaches, sections, abscissas, slopes, profiles, parameters).

For using them, one should have drown some reaches on the graph (for example, from a map, and then select the part on which work and last, click on  "Automatic building". The complete network can be selected with Ctrl+A. If several possibilities of linear series of reaches do exist in the selection, they will be suggested for selection with the additional indication of its total length. It will then still be possible to change the upstream and downstream reach, and then the upstream and downstream sections within theses reaches. The actions will then be applied on this selection.

It will be possible for most actions (indicated in the help note bullet over the corresponding action button) to propagate these actions on the complete network, by clicking on Shift+Click or Ctrl+Click keys when clicking on the action button.

Several tools are available through the tabs:

- Portion definition
- Nodes and reaches modification
- Sections modification
- Chainage and bottom slopes modification
- Cross profiles modification
- Cross profiles interpolation
- Parameters modification
- Graph reaches size and shape modification

Chapter 17

Portion definition

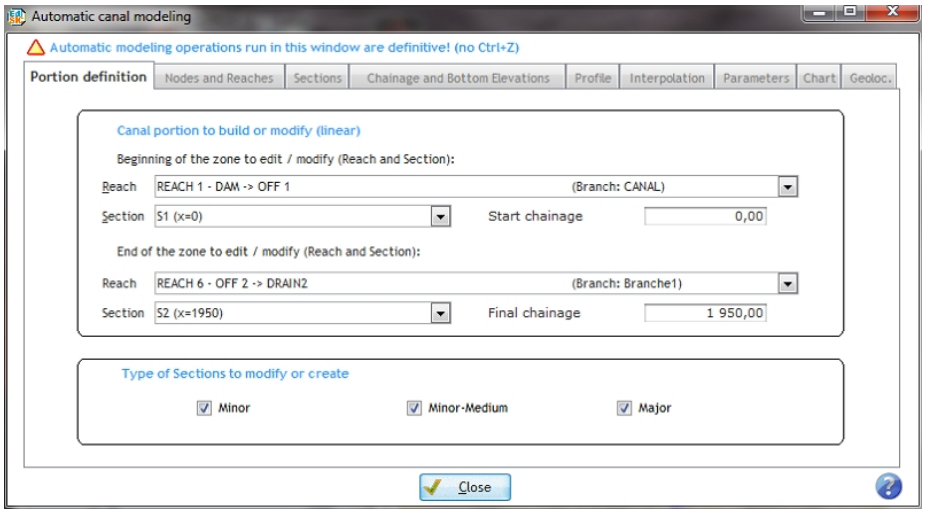


Figure 17.1: extrapolationonglet1

The first tab is used to refine the selection done with the mouse on the graph or the treeview of the network by allowing to:

- Selection of the beginning and ending Reaches within respectively the Upstream Reach and the Downstream Reach of the selection, to focus on only part of it.
- Selection of the beginning and ending sections within respectively the Upstream Reach and the Downstream Reach of the selection.
- The section type which will be impacted by the actions of automatic modeling (minor bed, minor-medium bed, major bed).

Chapter 18

Nodes and reaches modification

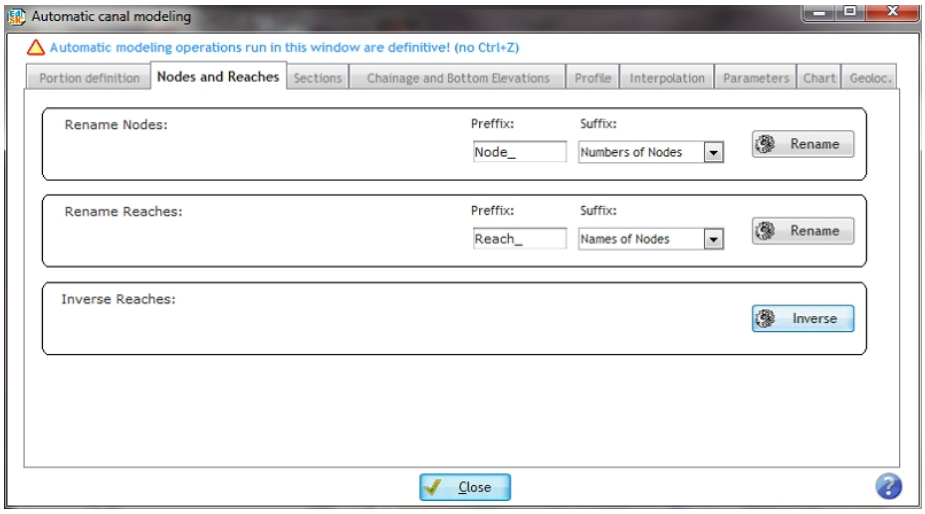


Figure 18.1: extrapolationonglet2

In this part, several operations are available:

- Automatic rename of nodes with an order number or also with the branch name.
- Automatic rename of reaches with names or numbers of upstream and downstream nodes, or also with the branch name and reach number.
- Reversing the default flow direction of the reaches (the flow reference Q>0, knowing that there will always be a possible flow reversal and therefore

Q<0, if the boundary conditions allow it).

Chapter 19

Sections modification

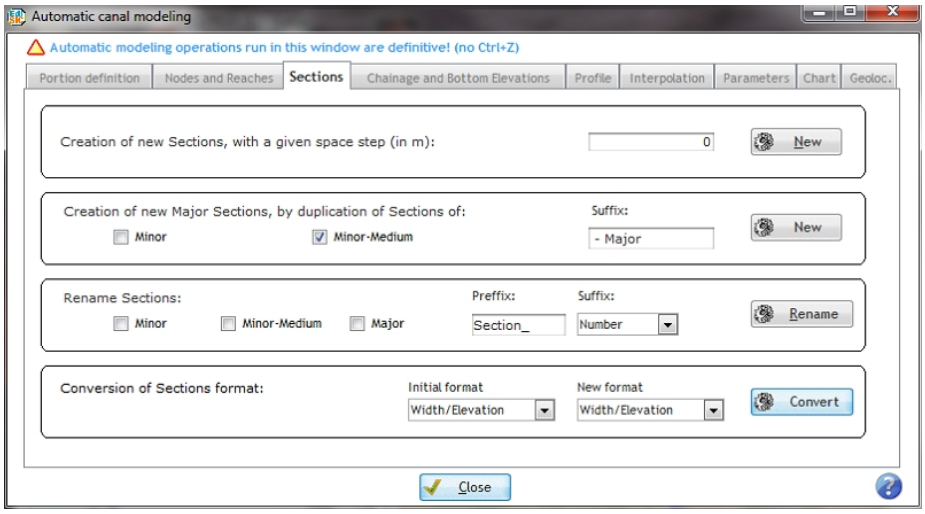


Figure 19.1: extrapolationonglet3

In this part, several operations are available:

- Design of sections equidistant of a defined step. At this stage there is no interpolation, neither on the shape of the section, nor on the bottom elevation. To do this, use the "Interpolation" tab to interpolate the sections in shape and possibly in bottom elevation, or just the "Chainage and Bottom Elevations" tab to change the bottom elevations;
- Design of major bed sections from minor or minor-medium bed sections;
- Automatic rename of sections with an order number or function of the chainage.

- Changing the format of the sections, for example from Width/Elevation or Abscissa/Elevation to Trapezoidal, or vice versa, from Trapezoidal to Culvert (thus closing the section to its ceiling), etc. Currently all option combinations are not supported, but the list will be expanded, and the more traditional options are implemented. Do not hesitate to contact the developers for a particular option if necessary.
- Change of type of sections, from Minor to Minor/Medium or vice versa. In the case of passage from Minor to Minor/Medium, the overflow elevation between the 2 beds is proposed as being (1) the average of the bottom and bank elevations, (2) the bank elevation. Then you have to modify it if necessary. We can imagine later other choices. Changes to or from Major Beds are not currently allowed, as a Major bed exists only as a complement to a Minor or Minor/Medium bed. But if an interest appears for this option we can do it

Note: If in Width/Elevation format there are more than 2 pairs of values, only the 2 first ones will be used for a passage in Trapezoidal, Culvert or Rectangular format. For a passage in Rectangular format, the second pair will be just used for the bank elevation, the width will be ignored. Same for an Abscissa/Elevation format if there are more than 4 points, only the first 4 will be used.

Chapter 20

Chainage and bottom elevation modification

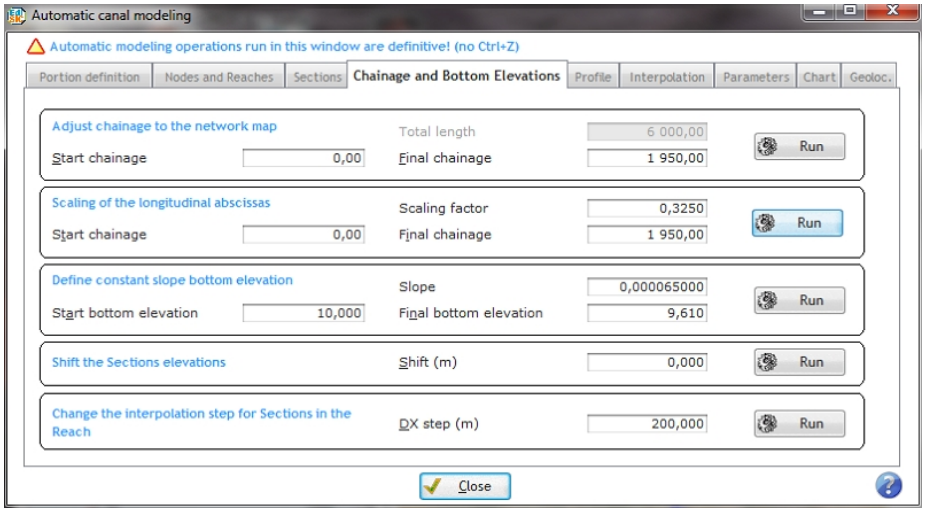


Figure 20.1: extrapolationonglet4

In this part, several operations are available:

20.1 Adjust chainage to the network map

The chainage of existing sections will be adjusted considering the begin and end chainage defined here and proportionally to their position on the network graph.

This functionality is useful to roughly define the sections chainage (even more the nodes) of a network drawn from a plan or a background map.

20.2 Scaling of the longitudinal abscissas

Allow to modify on the fly sections abcissas with keeping the network proportions.

20.3 Define constant slope bottom elevation

This tool allows to define a constant slope from the elevation of the Upstream Section, the elevation of the Downstream Section of the selected portion and/or the slope.

20.4 Shift the Sections Elevations

Allow to raise or to pull down the whole selected network with a chosen height.

20.5 Change the Interpolation step for the Sections in the Reach

Allow to modify the Calculation Sections Interpolation Step for several reaches.

Chapter 21

Section Profiles modification

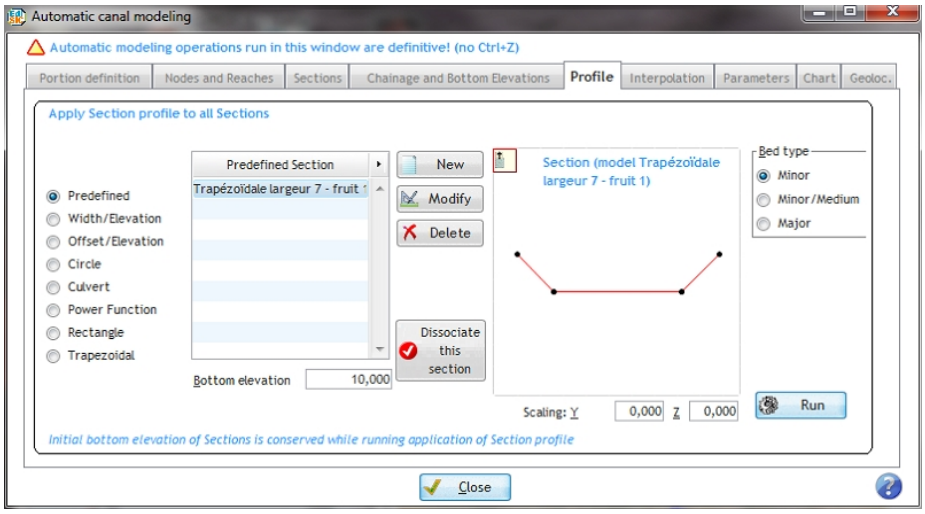


Figure 21.1: extrapolationonglet5

This tool allows to apply a cross profile to all the selected sections. The profiles get into position in altitude from the bottom elevation of the sections.

Predefined profiles are particularly useful in order to easily modify the profil of all the Sections having the same template afterwards.

Chapter 22

Interpolation of sections

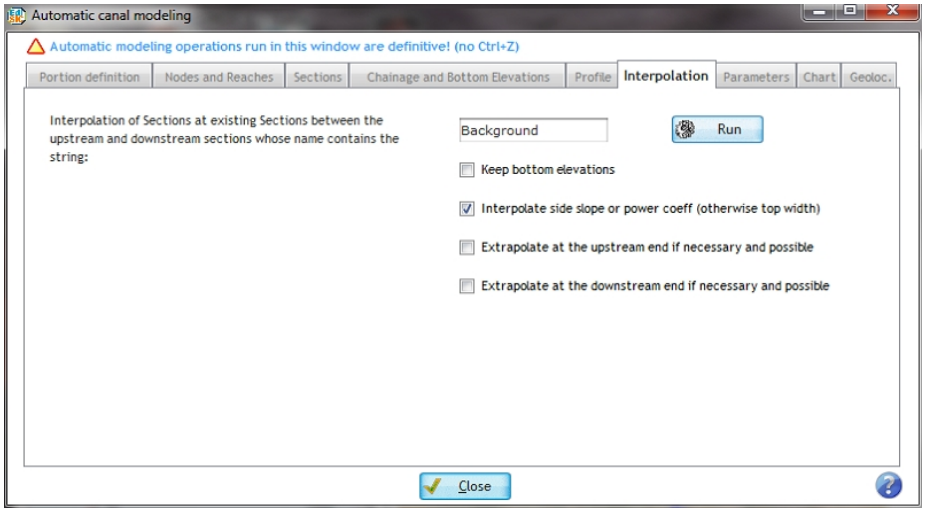


Figure 22.1: extrapolationonglet6

This tool linearly interpolate Sections between special Sections known as background or template or reference Sections. The profiles are positioned at the same abscissa as the existing sections. The bottom elevations can be maintained or changed (and then in this case also linearly interpolated between the upstream and downstream template Sections).

Currently the upstream and downstream template sections must be of the same type, and of parameterized types: rectangular, trapezium, circular, culvert, power (sections in width - elevation or abscissa - elevation formats are not yet taken into account). On the other hand, as with the option to change the formats

of the sections, you can change the format from Width - Elevation or Abscissa - Elevation to Trapezoidal (using the first 2 pairs W-E, or the fours points A-E starting from the bed allowing to describe such trapezium), this limitation can therefore be raised indirectly.

If no template Section is found, then the upstream and downstream sections of the selected portion are selected as the template Sections, and will be used as such for interpolations.

If only one template Section is found, then for the moment it is kept and the upstream and downstream Sections are added as in the previous case where there is no template Section at all. So we end up with 3 template Sections.

If we have at least 2 template Sections, they will be used to interpolate the Sections between them and according to their characteristics. If the options "Extrapolate at the upstream …" and / or "Extrapolate at the downstream …" are activated, and if the selected zone exceeds upstream and/or downstream the zone framed by the first and the last template Section, then these extreme zones will also be modified but not by interpolation between the template Sections, but by extrapolation of the 2 closest template Sections. If the options "Extrapolate at the upstream …" and/or "Extrapolate at the downstream …" are not enabled, the interpolations will start at the first template Section and will end at the last template Section, the others extreme upstream and downstream sections out of this template-template area will not be altered, even if they are part of the selection.

The interpolation method is not unique. For example, for a trapezoidal section, we can choose to linearly interpolate the banks slope, which is the most traditional method and easy to implement since this slope is one of the parameters of the section. Ditto for the sections of power form, where we can interpolate the coefficient power. But we can also choose another method which consists of interpolating linearly the top width, which is not a parameter of the section, but a consequence of the other parameters (width at the bottom, banks slope and depth for a trapezoidal section). The two methods are not equivalent in the case where the depth (bank elevation - bed elevation) is not identical in the two sections to be interpolated. An option is therefore available to choose one method or another (checkbox "Interpolate side slope or power coefficient (otherwise top-width)"). For now this option is implemented for trapezoidal sections, but not yet for sections power for which only the power interpolation option works.

Chapter 23

Section settings modification

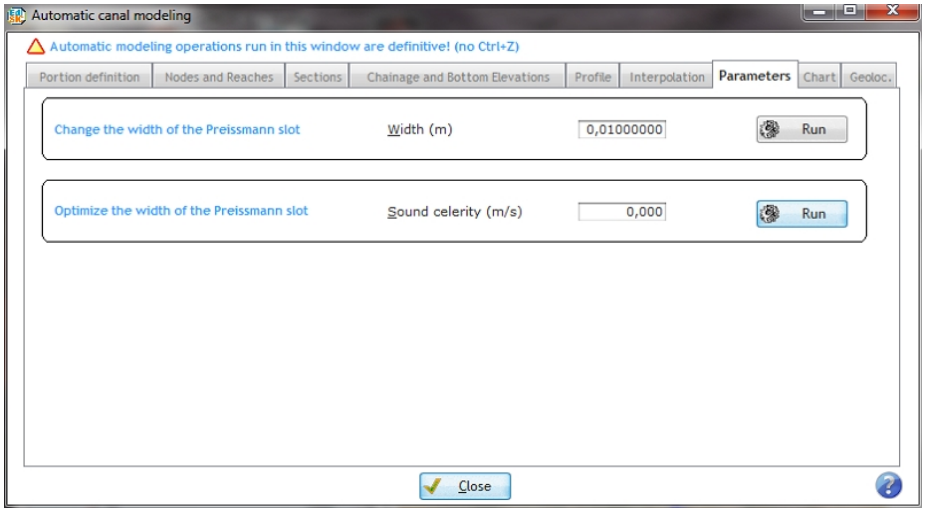


Figure 23.1: extrapolationonglet7

This tab allows to modify the settings related to the Preissmann slot used in case of closed-conduit flow.

For the selected Sections, one can define:

- a fixed slot width;
- a width calculated for each Section from a fixed sound celerity.

Chapter 24

Network chart modification

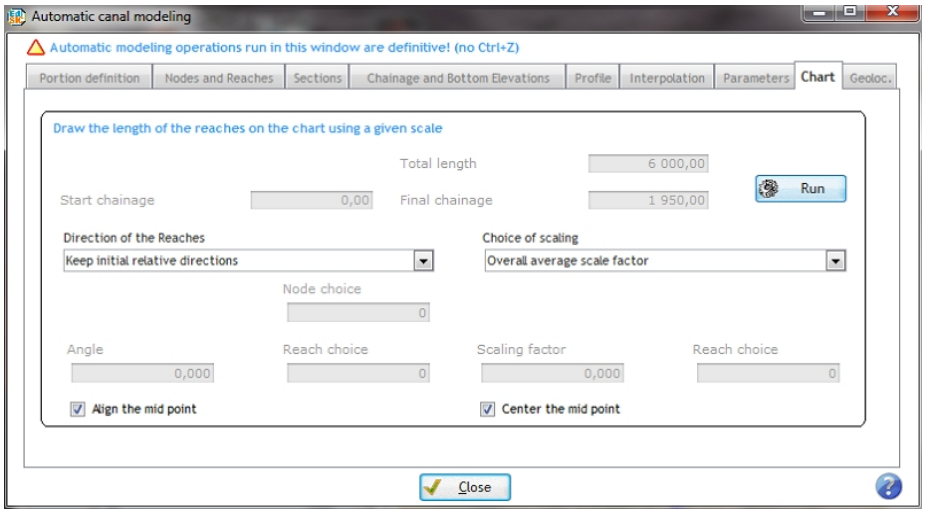


Figure 24.1: extrapolationonglet8

This tool allows to draw again the network chart keeping the proportions of the Sections Chainage in the reaches.

This option does work completely satisfactorily for meshed and / or looped networks

Chapter 25

Geolocalisation

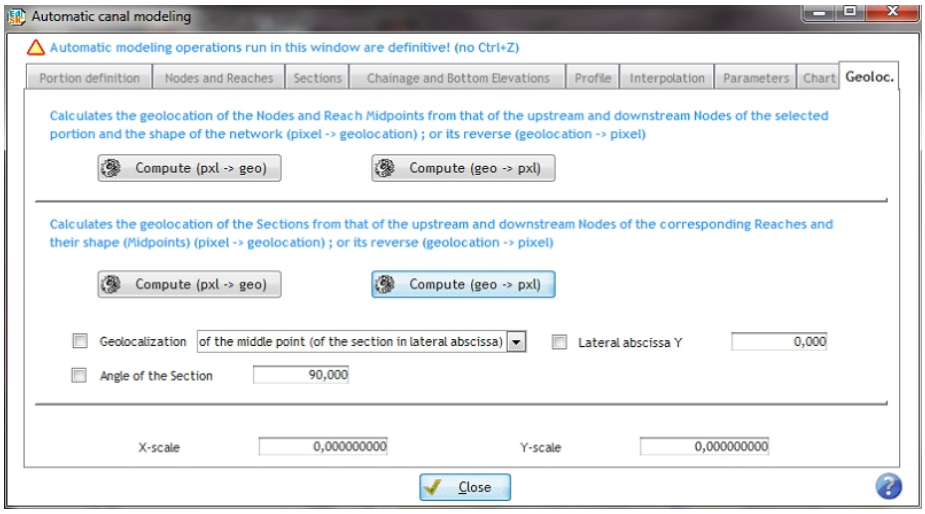


Figure 25.1: extrapolationonglet9

This tool allows to manage the geolocation of the objects of the network (nodes, points of preemption of the reaches also called midpoints, cross sections).

The upper part of the “Geolocation” tab allows you to process the nodes and midpoints of the reaches (also called preemption points since they allow you to select a reach, to edit its parameters or view the results).

The “Compute pxl -> geo” option will use the pixel data (x-y position of the graphic objects on the 2D plan drawing) to calculate the geolocation (lat-lon) of the processed objects (nodes and preemption points here). This will be done using the geolocation of the upstream and downstream nodes of the selected

area. You must therefore enter this before using this option. This is done by simple proportionality relationships between x-y and lat-lon (known as the rule of 3). This geolocation data will then be stored in the corresponding fields of the corresponding objects.

The "Compute geo -> pxl" option will do the opposite, that is to say position (x-y) the objects on the 2D plan drawing taking into account the geolocation data (lat-lon), while retaining the upstream nodes and downstream as references (x1-y1 and x2-y2) and using simple rules of 3 to calculate their positions. You must therefore position these upstream and downstream points where you want them, for example on a previously imported base map (see the dedicated page). This data will be stored and the node and midpoint objects will then be drawn at these locations on the 2D plane.

The lower part of the "Geolocation" tab allows you to process cross sections. This option is close to the previous one in the ideas behind "Compute pxl -> geo" and "Compute geo -> pxl". However, the reference points will be taken for each reach with its upstream node and its downstream node (and not, as previously, the upstream node of the selection and the downstream node of the selection). You must therefore have previously processed them with the options in the upper part of the tab, detailed above. In addition the positions (x-y) of the cross sections will be calculated with the "Compute geo -> pxl" option, and drawn thus just after launching this calculation (clicking on the corresponding button). But since the reaches are drawn on the 2D plane as 2 line segments between the upstream node and the midpoint then the midpoint and the downstream point, and that the cross sections are drawn and distributed on these segments according to their longitudinal abscissa within their reach (the coordinates of the x-y pixels will therefore be recalculated on each display), this detailed plan drawing of the sections will not be drawn like this on subsequent displays.

Further evolution of SIC allowing more segments and potentially as many as cross sections would handle this more aesthetically.

In this management of the geolocation of cross sections, we must also provide other information, since the entire cross section is geolocated (only 1 lat-lon data for each section) and not the various points of the section. However, a section seen up close is not a local object, but made up of a set of points (and can be several hundreds of m or km long). It is therefore necessary to indicate to what point the given geolocation corresponds, as well as how the section is located in relation to this geolocation point, laterally and angularly. The angle is given in relation to a reference indicated in the global geolocation options (Edit/Network Properties/Geolocation Settings), with the choices of Geographic North, the central line of the reach or the segment drawn within the reach . The "lateral abscissa Y" is given in the case of the choice "for a given lateral abscissa"). If we check the boxes for these values they will be filled in for the corresponding objects when launching the "Compute pxl -> geo" calculation. Currently this data is stored but is not taken into account when drawing cross sections (to be done in a later version). For the moment the angle used for the drawing is 90°

relative to the segment, and the assumed geolocation point is the central one of the section.

At the bottom of the tab we indicate a scale in X and Y, making the correspondence between lat-lon and x-y. These fields allow you to visualize these scales after using one of the 2 modification options above, but not to force a drawing to a given scale (these are output variables, not calculation inputs). As indicated by the tooltip when positioning the mouse on the corresponding fields, the values indicated are the number of pixels for 1 degree of angle.

An example of geolocalized project is given at: dat/ex30_garonne/Garonne_benchmark_MinMoy_v4.xml



Figure 25.2: Garonne_Geoloc.png — image absente de la source

Chapter 26

Creating a network (nodes and reaches)

The hydraulic network in SIC consists of nodes and reaches. Each node can be connected to one or more reaches and each reach is necessarily located between two nodes.

Nodes will be located at each location: - We define a boundary condition (entry or exit point of flow, definition of a fixed water elevation or a relationship between water elevation and flow); - We want to define a local head loss.

The creation of nodes and reaches is done using the following tools: - Tools Menu> Create node or  button in the toolbar - Tools Menu > Create reach or  button in the toolbar

To draw a continuous network, it is possible to draw at once the nodes and reaches connected by checking the option “Automatic creation of the reaches” of the Options menu.

|The order of creation of the network is important! It must be drawn from upstream to downstream in the direction in which the flow will go in steady flow simulation. If the automatic creation of reaches is enabled, nodes must be created in this order and the creation of reach must be done by connecting the upstream node to downstream node in that order.|

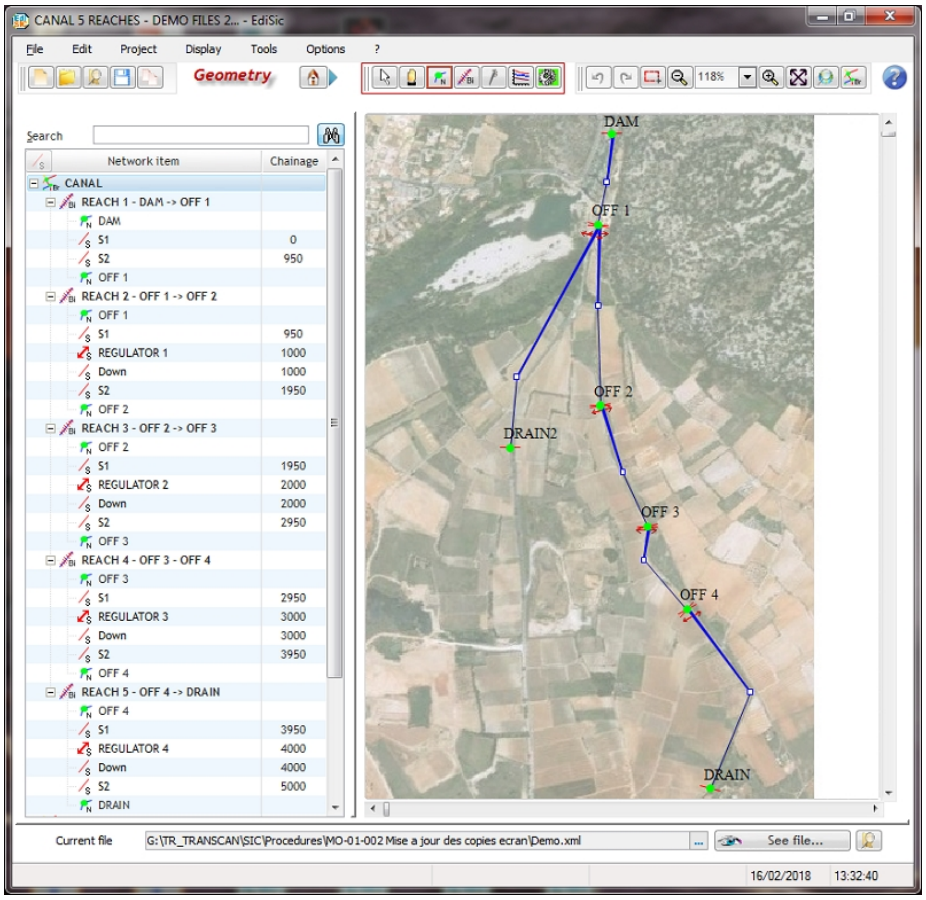


Figure 26.1: premiere_talnoeud

Chapter 27

Creation of section and description of cross sections

There are several ways to access the edit window of a reach: - Right-click on a reach in the tree or in the graph and choose “Description of the reach XXX” - Double clicking on a reach in the tree or in the graph if the “Select” tool is enabled (click  or press [Esc])

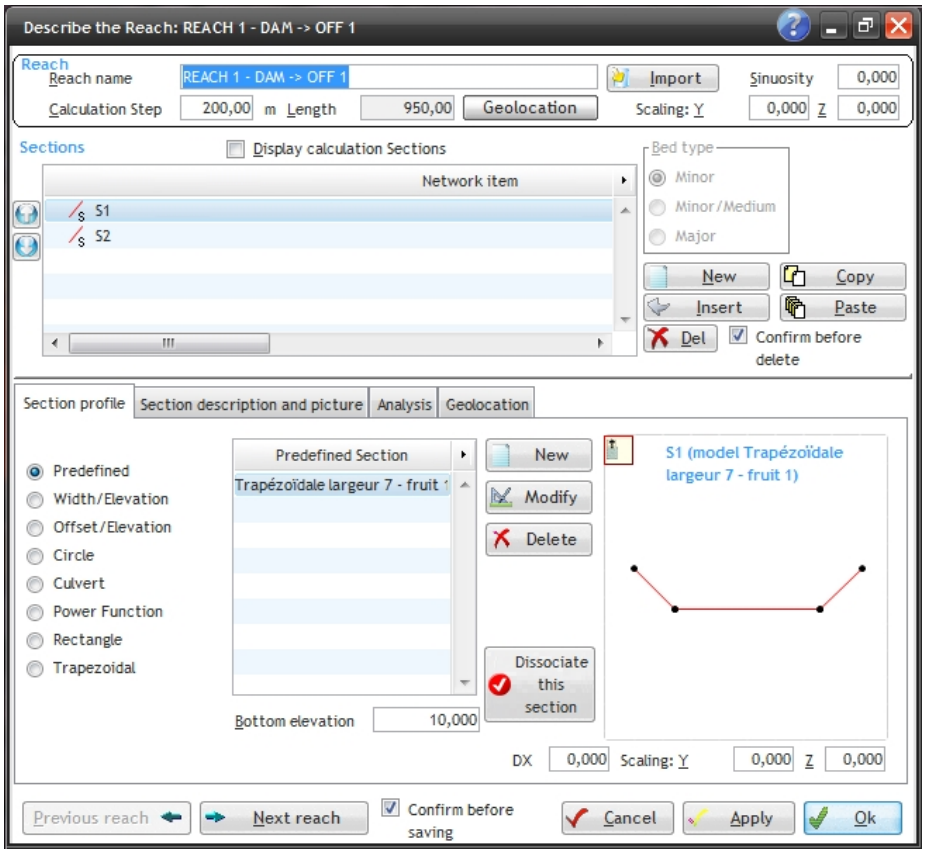


Figure 27.1: descriptionbiefonglet1

27.1 Reach description

The description of the reach allows entry of: - Name of the reach - Calculation step that defines the maximum space between each calculation section that will be generated in this reach by TALWEG. This notion is explained in the theoretical documentation. This space step can be defined at 3 levels: at the level of the complete network (Network Properties), at the level of the reach (screen above, on the top left), at the level of the sections (screen above at the bottom, in this case the space step will be applied between this section and the next one). If a level has a value of zero, the value of the upper level is used. If all values are null, a default value (100 m) will be used. A message will be written in the .log file as well as in the message window. - Sinuosity coefficient (coefficient, between 0 and 1 which is used to change the volume of the medium bed (if any) to reflect the sinuosity of flow channel inside the medium bed). If 0 or 1 is entered, the volume is not changed, considering that the minor bed and the

medium bed are aligned. For example, if we put 0.5 as the sinuosity coefficient, we assume that the curvilinear abscissa in the medium bed is half that in the minor bed (the minor bed meanders in the medium bed). - Homothety factors Y and Z for cross sections of the reach (which will override for this reach those defined in Network Properties) - Geolocation of the middle point of the reach (preemption point) can be visualized and modified. This geolocation is given in decimal degrees or in metric variables depending on the chosen option. - Cross sections can also be imported within the current reach at different format (text, Mascaret)

- New sections by clicking the button “New” or “Insert”
- Order of sections by clicking on the two arrows at the left of the listing sections table

We will create sections in the following locations: - At the beginning and end of each reach. The abscissas of these two sections will determine the length of the reach. Understand that these sections are physically present at the location of the node at the ends of the reach. - Every time change of profile geometry of the section (change in slope …). - At each location of a cross structure formed by one or more devices (weir, gate…) and immediately downstream (same abscissa) of these structures.

Coordinate values between different reaches have no special meaning for the software: they can be continuous or discontinuous between the reaches, increasing or decreasing within the reaches.

27.2 Sections description

For each selected section in the table section, we will have: - The abscissa in meters (in a reach, the abscissas must be monotonically increasing or decreasing). - The bed mode (Minor, Minor-Medium or Major) - The type of profile - The singularity (a singular section is a section in which a cross structure can be defined) - The elevation of separation between the minor and medium bed if the minor-medium bed mode has been activated - Factors Y and Z scale factor for the cross sections of the section (which will overwrite for the selected section those defined for the network and the reach selected) - the space step DX between this section and the following, until new definition of this DX, for the possible interpolation of new calculation sections. This DX space step can be defined at the level of the complete network, at the level of the reach (see above), and here at the level of this section. If a value is at 0 Talweg will take the value of the higher level - the width of the Preissmann slot if this section is closed at the top

Singular sections are the only sections where we can add, in hydraulic mode, a cross structure with one or more devices. It is advisable to create a section downstream at the same abscissa of each singular section for ease of reading in graphics results.

27.3 The tab “Profile Section”

Different types of section profiles are available: - Predefined - Width / Elevation - Offset / Elevation - Circle - Culvert - Power function - Rectangular - Trapezoidal

Predefined profiles enable you to not have to reenter profile that is repeated throughout the network (see Preset profiles: managing a library of cross sections).

27.4 The tab “Section description and picture”

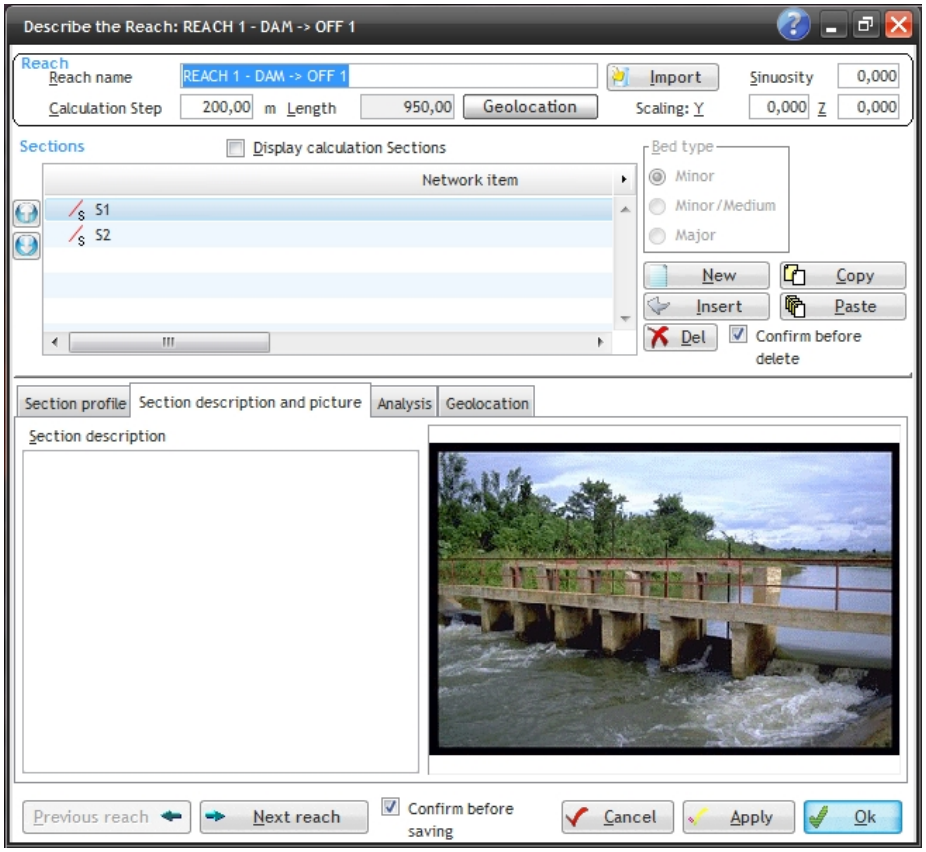


Figure 27.2: descriptionbiefonglet2

Click on the picture to set the picture associated with the section.You can also add free text to indicate and store information if necessary.

For more details on image management, see the dedicated chapter.

27.5 The tab "Analysis"

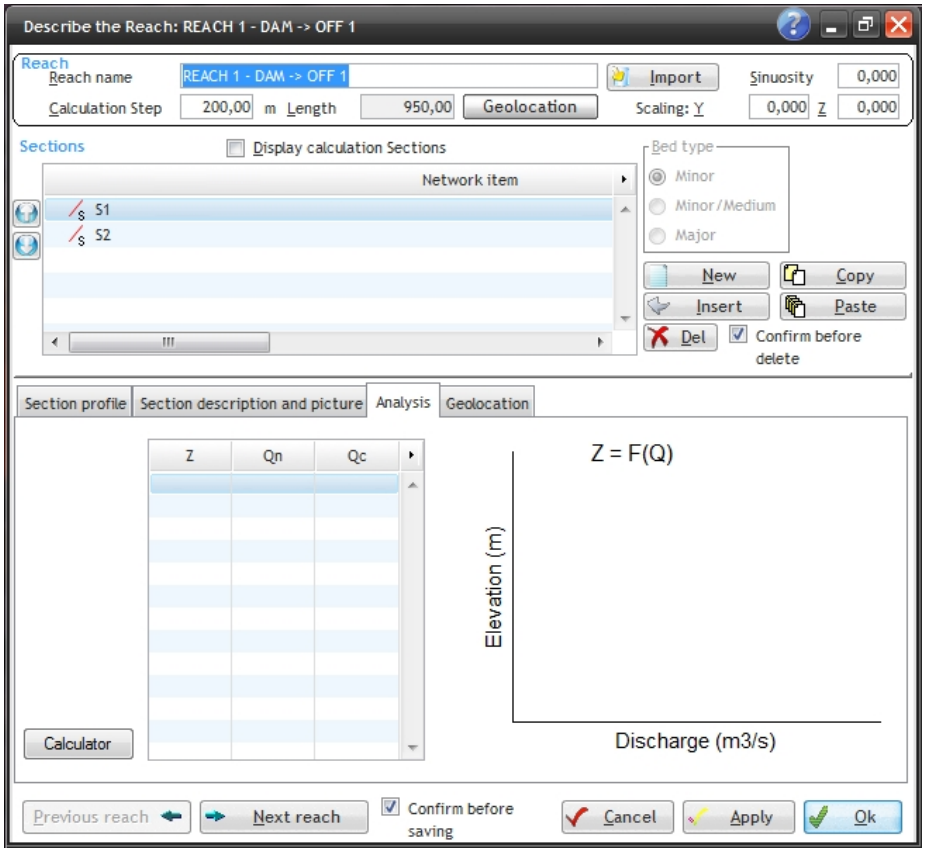


Figure 27.3: descriptionbiefonglet3

This tab makes it possible to analyze the hydraulic characteristics of the section, and in particular to calculate the critical and uniform flow rates. The "calculator" tool works the same way as in the Q(Z) offtake or boundary condition mode (except the option to select the upstream reach, since here we already are in a given reach).

27.6 The tab “Geolocation”

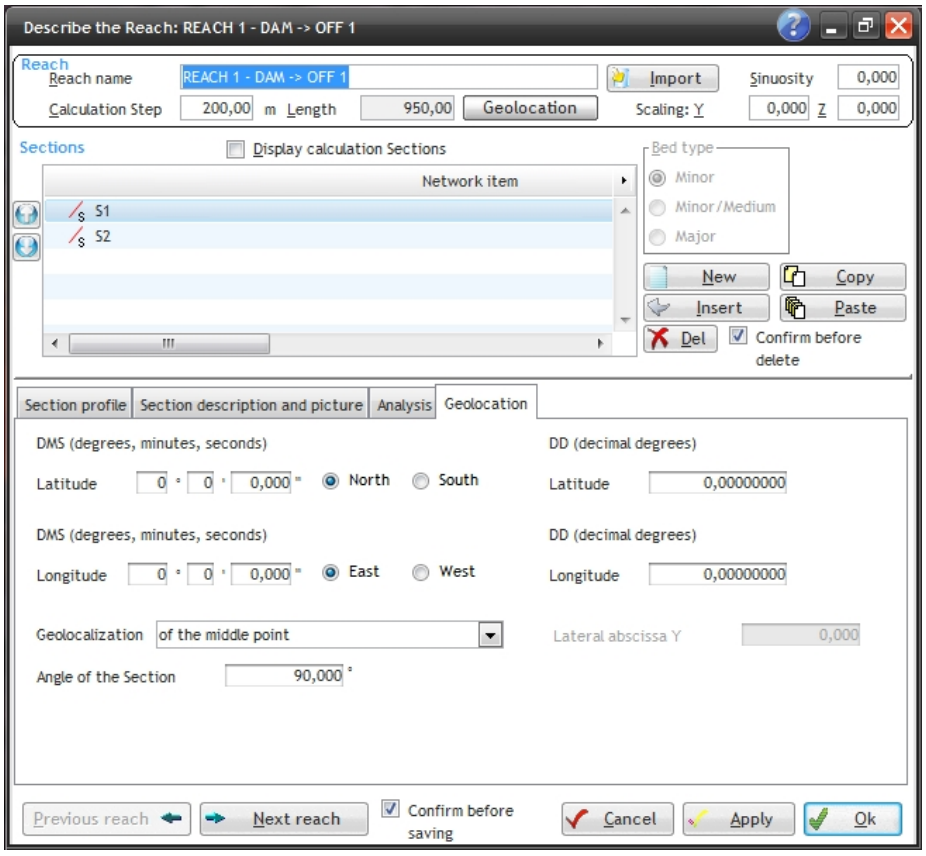


Figure 27.4: descriptionbiefonglet4

This tab is used to define and visualize the geolocation variables of the section. If the geometry has been imported globally from a .CSV file (HEC-RAS style) or inside reaches from TXT files with geolocation data, this information will appear here. Ditto if the “Automatic construction” mode was used to fill in these data automatically from the graphical layout of the node and midpoint objects on the screen.

For a section, which actually has a spatial extent, assumed in a vertical plane, the geolocation variables are given for a point of the section with various options (midpoint of the section, right bank, left bank, lower point, etc.), as well as an angle of the section with respect to a reference. The options are:

-1: of the middle point -2: of the lowest point -3: of the highest point -4: of the left bank -5: of the right bank -6: for a given lateral abscissa

Currently this reference is the geographical North. In the future other options can be managed (example other fixed orientation, orientation of the central line of the reach, orientation of the drawing line on the screen of the reach, etc.). This choice can be made in the Network Properties window, Geolocation Parameters tab.

Geolocation of the chosen point is given in decimal degrees, or in metric variables depending on the chosen option.

27.7 Control input

The reaches shown in fuchsia on the network diagram have an incomplete entry. Those considered valid are drawn in blue.

To verify that the input geometry is consistent, use the tools: - “Views of sections, branches and longitudinal profiles” - “Managing inconsistencies network.”

Chapter 28

Predefined profiles: managing a library of cross sections

Creating a predefined profile is done by clicking on the button 

present in the description of the cross section of an existing section.

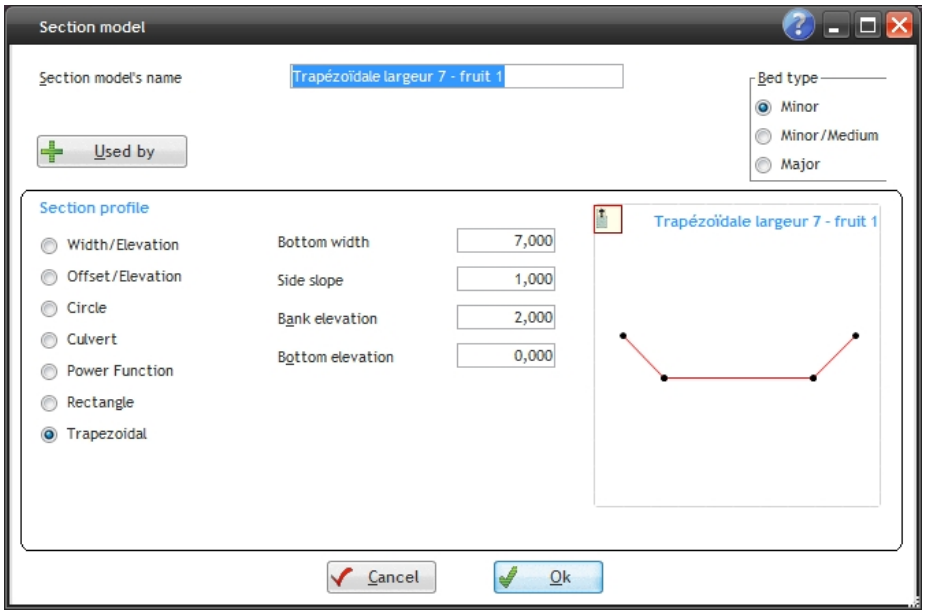


Figure 28.1: sectionpredefinie

The bottom elevation will then be the only parameter changed in the section using a predefined profile. The other parameters of the vertical profile will be redefined in relation to the bottom elevation in this predefined profile.

The button “Used by” displays the list of sections using the predefined profile in progress and lets you change the bottom elevation of these sections.

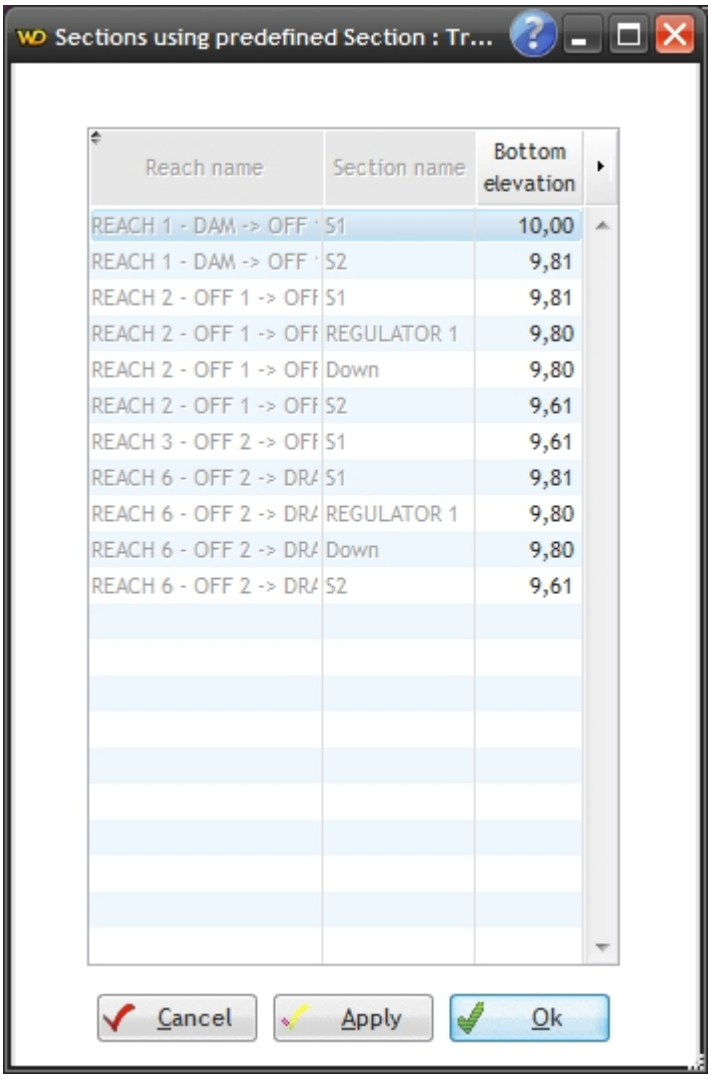


Figure 28.2: sectionsutilisant

Chapter 29

Import sections and cross profiles in text format

It is possible to import sections from text files containing tables with values separated by tabs, on several lines. This file can be at the formats described below, be a Mascaret geo file or be a HEC-RAS g* file .

In the case of a text file, all formats managed by SIC are available:

-* L or 1 = Section of type Width - Elevation -* A or 2 = Section of type Abscissa - Elevation -* C or 3 = Section of type Circular -* D or 4 = Section of type Culvert -* P or 5 = Section of type Power -* R or 6 = Section of type Rectangular -* T or 7 = Section of type Trapezium

If behind this letter or this number is added the letter s or S (attached, without space), this section will be considered as singular, to insert one or more device (weir, gate, etc.) in the Hydraulic mode.

Example :

```
section_234 $145782.9$ $0$ A $45.089584$   9.902083  $2$ 0.0 $22.0918
  -174.223       39.000
  -173.065       38.312
   173.065       38.312
   174.223       39.000
ld_Isola-Serafini$   146338.4$$ 0 $As$  45.092918 $9.900417$ 2 $0.0$ -
2.8141
   -57.613       38.000
   -55.546       36.403
    55.546       36.403
    57.613       38.000
```

```
section_236 $147124.1$ $0$ A $45.099583$   9.897083  $2$ 0.0$-
22.3343
  -125.365       32.880
  -124.500       32.486
   124.500       32.486
   125.365       32.880
```

To access the Import window sections of a reach, there are two possibilities: - Select a reach, make a right-click and choose “Import sections” in the context menu; - In the description window of a reach, click the “Import” button at the top of the window.

These tables can be quite varied in format (see below).

Before each Abscissa - Elevation table, a line allows to import the general data of the section, separated by the characters$ (dollar). 1. Section name 2. Abscissa (in m) 3. Minor-medium elevation overflow (for minor-medium bed section) or distance for the major bed (for a major bed section). Just leave space character if the section is just a minor bed section. 4. Section type : 1 for major bed section, 0 for minor bed section or minor-medium bed section if the minor-medium elevation overflow is filled. You can also just put a space character if the section is just a minor bed section 5. Format of the section with a letter of number code (L or 1, A or 2, C or 3, D or 4, P or 5, R or 6, T or 7). Additional letter s or S (attached, without space) if the section is singular (for a cross device such as weir, gate, etc) (Ls, As, Cs, etc) 6. Latitude in decimal degrees or in metric unit (option) 7. Longitude in decimal degrees or in metric unit (option) 8. Geolocation reference of the section (option, soon)

-* 1: of the middle point -* 2: of the lowest point -* 3: of the highest point -* 4: of the left bank -* 5: of the right bank -* 6: for a given lateral abscissa

9. Lateral offset of the geolocation reference (option, soon)

in meters when previous option is set to 6 (“for a given lateral abscissa”)

10. Angle of the section with respect to a reference (option, soon)

in degrees, with reference to the North direction for example. This reference can be changed in the Network properties, Geolocation settings tab. This can be Geographic North, Centerline, or Draw segment.

These geolocation data will appear in the Geolocation tab of description of sections

The abscissa and the overflow elevation are optional. In the absence of values (or just one or several space characters), the abscissa default will be 0 and the section will be considered as a minor bed section (without overflow minor-medium bed elevation). If the section format is not given the Abscissa - Elevation format will be supposed (therefore corresponding to the A or 2 code).

29.1 Format accepted for tables Abscissa - Elevation

29.2 Normal table

On one column:

```
Abs1 <tab> Cote1
Abs2 <tab> Cote2
Abs3 <tab> Cote3
Abs4 <tab> Cote4
```

On several columns:

```
Abs1 <tab> Cote1<tab>Abs2 <tab> Cote2
Abs3 <tab> Cote3<Tab>Abs4 <tab> Cote4
```

29.3 Transposed table

On a row:

```
Abs1 <tab> Abs2 <tab> Abs3 <tab> <Abs4>
Cote1 <tab> Cote 2 <tab> Cote 3 <tab> < Cote 4>
```

On several rows:

```
Abs1 <tab> Abs2
Cote1 <tab> Cote 2
Abs3 <tab> <Abs4>
Cote 3 <tab> < Cote 4>
```

29.4 Example of imported file

```
DX=00000250.0;
section_1  $0.0$ $0$ A $45.087082$    9.034584  $2$ 0.0 $22.0918
   -92.213       60.849
   -82.990       52.889
    82.990       52.889
    92.213       60.849
section_2$       555.6  $$ 0 $A$   45.087917  $9.039583$ 2 $0.0$ -
2.8141
  -131.488       60.778
  -122.729       53.538
   122.729       53.538
   131.488       60.778
section_3  $1111.1$ $0$ A $45.087082$    9.044583  $2$ 0.0$-
22.3343
```

```
   -92.213       60.707
   -82.990       52.747
    82.990       52.747
    92.213       60.707
DX=00000100.0;
ld_Pancarana$      1666.7  $$ 0 $A$   45.086250  $9.048750$ 2 $0.0$ -
2.1906
   -92.213       60.636
   -82.990       52.677
    82.990       52.677
    92.213       60.636
section_5  $2452.3$ $0$ A $45.082916$    9.054584  $2$ 0.0 $3.2219
   -70.381       60.566
   -60.195       51.416
    60.195       51.416
    70.381       60.566
DX=00000400.0;
section_6$      3007.9  $$ 0 $A$   45.083748  $9.058750$ 2 $0.0$ -
41.0541
   -43.210       60.495
   -43.210       50.193
    43.210       50.193
    43.210       60.495
section_7  $3793.6$ $0$ A $45.087082$    9.064583  $2$ 0.0$-
49.3008
   -92.122       60.424
   -82.602       52.077
    82.602       52.077
    92.122       60.424
```

For the format Width - Elevation (code L or 1) we also have access to 2 possibilities : normal or transposed, with the same idea as above but with Widths instead of Abscissas.

For the parametered formats the transposed mode does not exist. Data must be given under the following format:

C or 3 = Section of type Circular:

```
Radius
Bank Elevation <tab> Bed Elevation
```

D or 4 = Section of type Culvert:

```
Width<tab> Side slope
Bank Elevation <tab> Bed Elevation
```

P or 5 = Section of type Power:

```
Width<tab> Coeff Power
```

```
Bank Elevation <tab> Bed Elevation
```

R or 6 = Section of type Rectangular:

```
Width
Bank Elevation <tab> Bed Elevation
```

T or 7 = Section of type Trapezium:

```
Width<tab> Side slope
Bank Elevation <tab> Bed Elevation
```

Example with different formats:

```
section1$         0.0  $$ $A$   45.047122  $10.768145$ 2 $0.0$ 22.0918
   -99.547       25.486
   -84.567       16.489
    84.567       16.489
    99.547       25.486
section2  $800.0$ $$L $45.039195$   10.793122  $2$ 0.0$-2.8141
   149.536       15.432
   189.072       25.419
section3$       875.0  $$ $C$   45.042937  $10.806417$ 2 $0.0$ -
22.3343
   111.111
    25.409       15.603
section4  $1185.0$ $$D $45.042849$   10.822073  $2$ 0.0$-2.1906
   222.222        0.5
    25.379       15.661
section5$      1525.0  $$ $P$   45.040071  $10.833905$ 2 $0.0$ 3.2219
   127.411        0.536
    25.333       15.274
section6  $2050.0$ $$R $45.046153$   10.844424  $2$ 0.0$-41.0541
   171.086
    25.226       17.043
section7$      2725.0  $$ $T$   45.049515  $10.848664$ 2 $0.0$ -
49.3008
   176.586        0.536
    25.096       17.318
```

The interpolation space step DX can also be specified in the file. The format is:

```
DX=00000500.0;
```

If this file is created by the software Matlab, the write format is for example:

```
fprintf(fidout,'%s%010.1f%s\\r\\n','DX=',PasDX,';');
```

The first "DX=" line found will be attributed to the whole reach. The next ones, if any, will be attributed to the following sections until a new one is specified.

The “DX= lines” must always be written before a section data line (cf example above).

29.5 Import command line

If large numbers of reaches must be created, it is possible to automate this import by calling the software from the command line with multiple profile files as parameters, either in the options of the command line, or in a edisic.mac file. For more information, see dedicated chapter.

Chapter 30

Import geometry from Mascaret or HEC-RAS file

Mascaret Software (http://www.openmascaret.org) is a 1D free-flow hydraulic modelisation code based on Saint-Venant equations and developped by EDF R&D and the Centre d'Etudes Techniques Maritimes et Fluviales (CETMEF).

HEC-RAS software (http://www.hec.usace.army.mil/software/hec-ras/) is a one and two-dimensional hydraulic modelisation code developped by the US Army Corps of Engineers.

Importation of geometry file (.geo file for Mascaret, .g01, .g02… for HEC-RAS) is available in the Import sections and cross profiles in text format tool.

This tool allows to import cross sections of and into only one reach. In case of having several reaches in the file, only the first reach will be imported, into the current reach selected on the SIC² interface. To import the next reaches, you must delete the lines concerning the reach already imported in the file and then do a new importation, into another reach.

30.1 *Macaret specifications

In Mascaret software, description of cross profiles contains a topology and a bathymetry parts which are converted in SIC² into minor and medium river bed. Also, separation between minor and medium bed in Mascaret depends on abscissa in the profil when it depends on minor-medium elevation overflow in SIC² (cf. Sections description). This separation in Mascaret can occur for two different elevation on the left and right banks, SIC² will store the mean value by default, but in case of several beds, a screen will be displayed allowing you to select the value you want among different options.

30.2 Example of imported file

```
PROFIL Bief_1 P_29 13150.0
0.0 31.0 T
51.05 28.0 T
3535.11 27.1 T
3609.48 27.1 T
3619.07 26.63 B
3627.03 22.7 B
3688.68 14.6 B
3700.09 15.46 B
3822.77 23.98 B
3827.4 24.23 B
3835.92 24.15 T
3850.62 38.0 T
PROFIL Bief_1 DUPL_P_30_A 13950.0
0.0 31.0 T
44.74 27.75 T
2974.32 27.3 T
2987.55 27.17 B
3087.12 14.18 B
3184.8 25.09 B
3219.98 25.0 T
3693.8 35.0 T
```

30.3 *HEC-RAS specifications

HEC-RAS allows the use of multiple beds more complex than the concept of minor / medium bed developed in SIC and Mascaret. Einstein's formula (Hans Albert, Albert's eldest son) is then applied. It is not especially better than Debord's formula (on the contrary according to various studies, see the work of Proust et al.). But it allows more than 2 beds. The import into SIC does not take into account the different beds detailed in HEC-RAS. But if more than one bed is present, the import will place a Minor - Medium overflow elevation on the lowest one of the various beds found. In addition, HEC-RAS defines its multiple beds by transverse abscissas, while SIC does so by an overflow elevation. Check this point during import, we do not guarantee an ideal recovery of this notion. SIC also recovers the geolocation of the sections, as well as the side dikes (levees) if present.

30.4 Example of imported file

```
Geom Title=Base Geometry Data
Program Version=4.00
Viewing Rectangle= 0.0614 , 0.8779 , 1.0165 , 0.0426
```

```
River Reach=Butte Cr.       ,Tributary
Reach XY= 15
           0.756       0.9830827           0.732       0.9755639
             0.7       0.9567669       0.6746667       0.9172932
           0.644       0.8721805       0.6306667       0.8289474
       0.5986667       0.7932331       0.5718829       0.7725555
       0.5426667            0.75       0.5146667       0.7142857
       0.4893333       0.6578947            0.48       0.6109023
       0.4746667       0.5601504       0.4600303       0.5263027
       0.4452555       0.4921348
Rch Text X Y=0.6506667,0.8590226
Reverse River Text=-1

Type RM Length L Ch R = 1 ,0.2     ,500,500,500
BEGIN DESCRIPTION:
Upstream Boundary of Butte Cr.
END DESCRIPTION:
XS GIS Cut Line=5
0.76654992940443 0.95100091453960.757316930017350.97887906106609
           0.756       0.98308270.754683069982660.98728633893391
0.748537396568381.00690332062549
Node Last Edited Time=Jul/08/1997 13:59:46
#Sta/Elev= 8
     210      90     220      82     260      80     265      70     270      71
     275      81     300      83     310      91
#Mann= 3 , 0 , 0
     210     .07       0     260     .04       0     275     .07       0
Bank Sta=260,275
XS Rating Curve= 0 ,0
Exp/Cntr=0.3,0.1

Type RM Length L Ch R = 1 ,0.1     ,500,500,500
BEGIN DESCRIPTION:
Upstream Boundary of Butte Cr.
END DESCRIPTION:
XS GIS Cut Line=5
0.59020551178268 0.74403505399820.572259239672030.76728092779905
0.569567297985670.770767808197710.566875356299310.77425468859636
0.554312965514880.79052680001996
Node Last Edited Time=Jul/08/1997 13:59:46
#Sta/Elev= 8
     210    89.4     220    81.4     260    79.4     265    69.4     270    70.4
     275    80.4     300    82.4     310    90.4
#Mann= 3 , 0 , 0
     210     .07       0     260     .04       0     275     .07       0
```

```
Bank Sta=260,275
XS Rating Curve= 0 ,0
Exp/Cntr=0.3,0.1

Type RM Length L Ch R = 1 ,0.0     ,0,0,0
BEGIN DESCRIPTION:
Upstream Boundary of Butte Cr.
END DESCRIPTION:
XS GIS Cut Line=5
0.484413027535690.497599132986690.457457886370460.50925501974819
0.453414614630170.511003401454660.449371342889890.51275178316112
 0.43050274258970.52091090046108
Node Last Edited Time=Jul/08/1997 13:59:46
#Sta/Elev= 8
     210    89.1     220    81.1     260    79.1     265    69.1     270    70.1
     275    80.1     300    82.1     310    90.1
#Mann= 3 , 0 , 0
     210     .07       0     260     .04       0     275     .07       0
Bank Sta=260,275
XS Rating Curve= 0 ,0
Exp/Cntr=0.3,0.1
```

Chapter 31

Import geometry from Mike 11 file

The Mike 11 software (https://www.mikepoweredbydhi.com/products/mike-11) is a monodimensional free-form hydraulic modeling code based on the Saint-Venant equations developed by DHI (Dansk Hydraulisk Institut, or Danish Hydraulic Institute).

To import a file in Mike 11 format, use the online file conversion tool in Mike 11 format to Import sections and cross profiles in SIC text format : http://sic.g-eau.fr/sic_tools/import/_mike11.html

Another option is, under Mike 11, to export the sections in a txt file in raw data mode, then to launch a routine under Matlab (script mike2sic.m) which will convert this file into another txt file in the format of import from SIC, and finally to import the data from this file into a reach. As an option, the section format can be converted from abscissa-elevation to width-elevation (routine ac2lc), as well as reducing the number of description points (routine F_Approx_Sections_v6). These routines are available upon request.

Cross sections from MIKE11 can be exported by following the steps below: -* Open MIKE Zero -* Open the .xns11 through Open (or drag and drop) -* Select File » Export All Sections » Export Raw Data… -* Save the cross sections out as a .txt file

Chapter 32

Import geometry from a DTM or a geometer reading

We offer an online tool to extract cross profiles and create a SIC text import file from a DTM (Digital Terrain Model) or a geometer reading:

http://sic.g-eau.fr/sic_tools/import/_xyz.html

Two methods of extracting cross profiles are proposed: one is more suitable for homogeneous point measurements such as DTMs, the other is more suitable for surveyor measurements where cross profiles have been recorded.

32.1 Format of the import file for points with coordinates (X, Y, Z)

The import file is a text file containing 3 columns representing the abscissa, ordinate and altitude in an orthonormal and metric reference frame of the geometry points recorded in the field (For example, points referenced in the Lambert coordinate system for France). The accepted separators for the columns are tabulation, semicolon and comma. The character used for decimals is the point.

Example of a file in Lambert coordinates with tab separator: <doc115|center>

It is recommended to put the tool directly in full screen before importing the file. The import consists in choosing the file on the hard disk using the “Browse…” button and clicking on the “Process file” button.

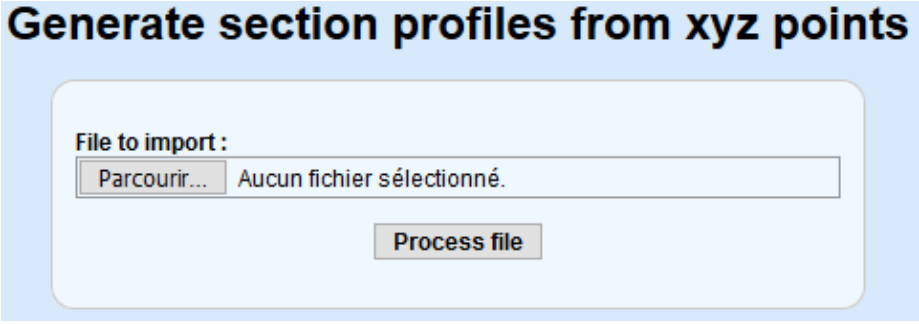


Figure 32.1: doc116

32.2 Trace of the central flow route

Once the file has been imported, you must indicate where the stream to be digitized is located by drawing the central flow route with the mouse. This is done by placing, with a mouse click, nodes that will form a set of segments.

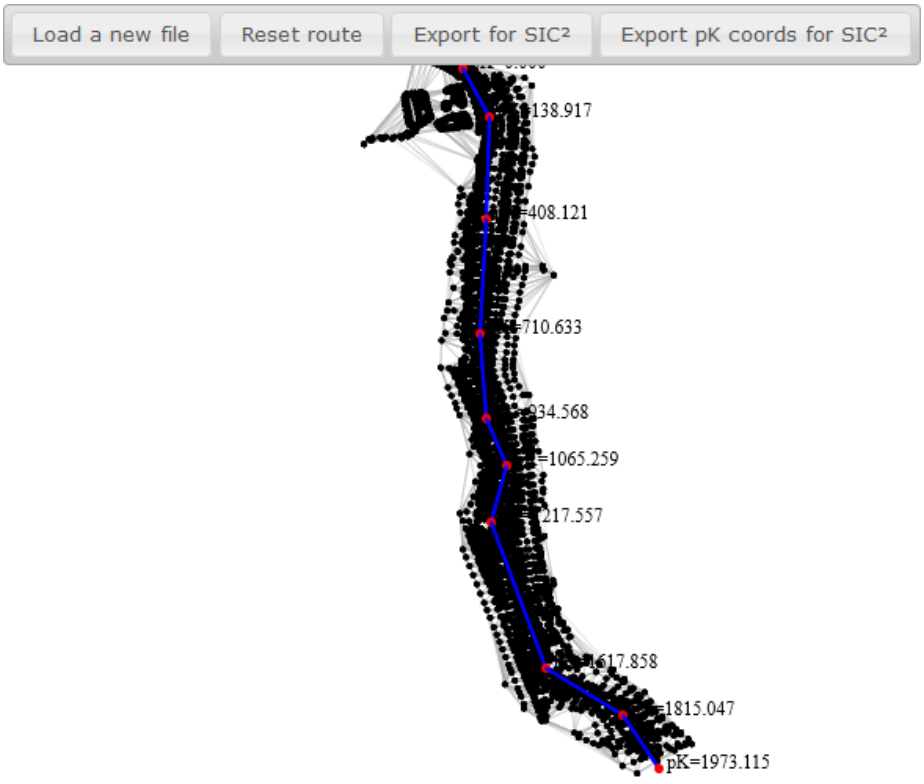


Figure 32.2: doc117

The kilometric point (pK) is indicated at each plotted node. It is possible to restart your route by clicking on the “Reset route” button.

32.3 Extraction of cross profiles from a DTM

This tool allows cross profiles to be extracted at regular intervals by transforming the point reading into a digital terrain model using the Delaunay triangulation method.

Access to this tool is via the “Export for SIC²” button. The tool parameters are as follows: - Space step (m): the desired space step between each cross section (in meters); - Section width (m): the maximum width of the section (in meters). Beyond this distance from the center line, the intersections of the section with the edges of the Delaunay triangles will be ignored; - Max length of triangle edges (m): Maximum length of the edges of the Delaunay triangles.

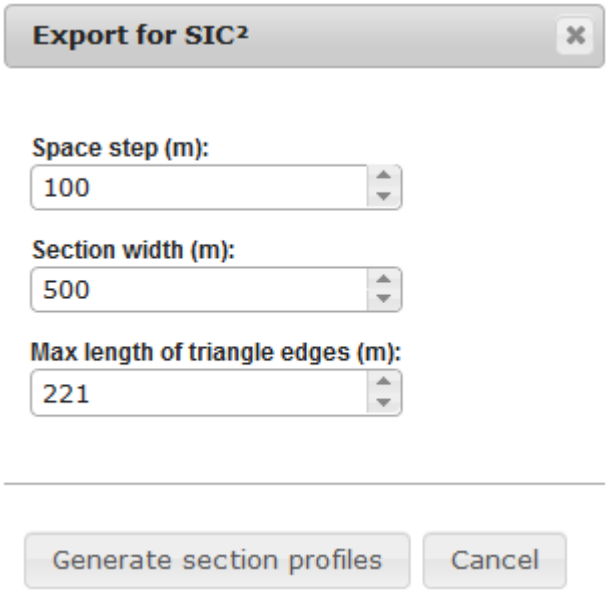
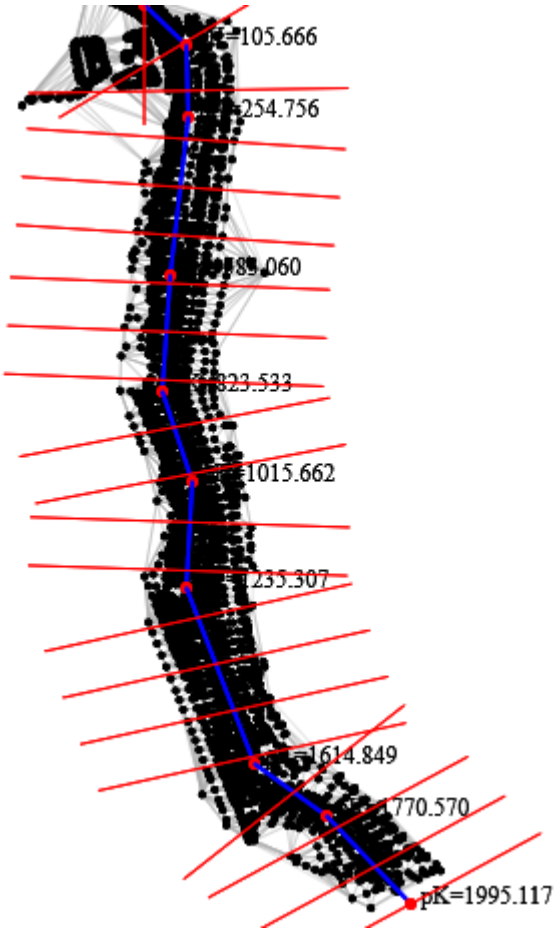
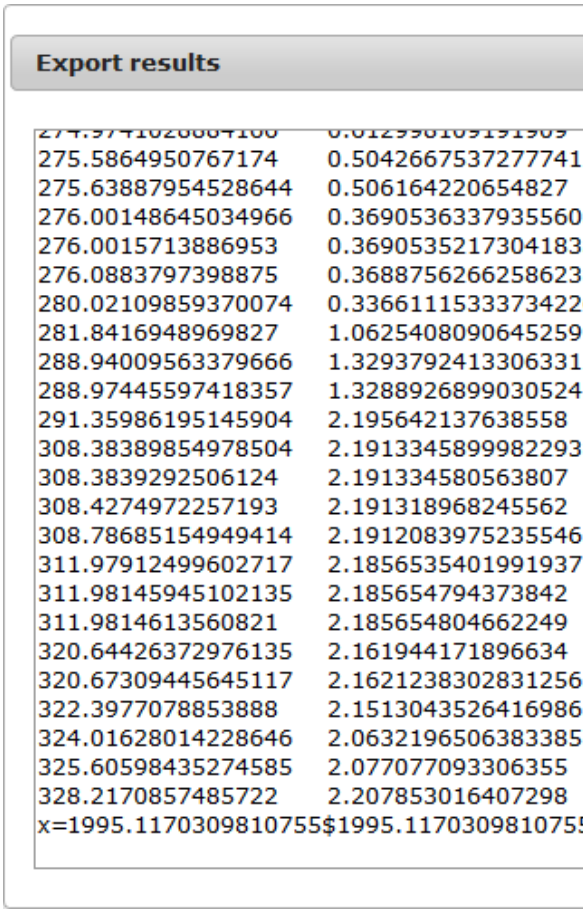


Figure 32.3: doc118

Pressing the “Generate section profiles” button starts the tool with the selected parameters. The result is displayed graphically and produces a text to copy and paste into the EdiSIC text format cross section and profile import tool. The operation may take time and you must wait even if the browser indicates that a script is blocking the operation of the application.





32.4 Extraction of cross profiles from a survey of cross profiles

This tool automatically creates sections for SIC² at the location where cross-sectional profile surveys were performed. The algorithm converts the coordinates of the points into a reference point corresponding to the line of the central flow line (pK axis). It then groups the related points by section.

Access to the tool is via the "Export pK coords for SIC²" button. The tool parameters are as follows: - Max length between points (m): maximum distance along the pK axis between two points for a grouping in the same section (in meters); - Max width of cross profiles (m): maximum width of cross sections (in meters). Beyond this width, the points are ignored. - Min number of points in section: minimum number of points to constitute a cross section. If a grouping

of points does not reach this minimum limit, the section is ignored.

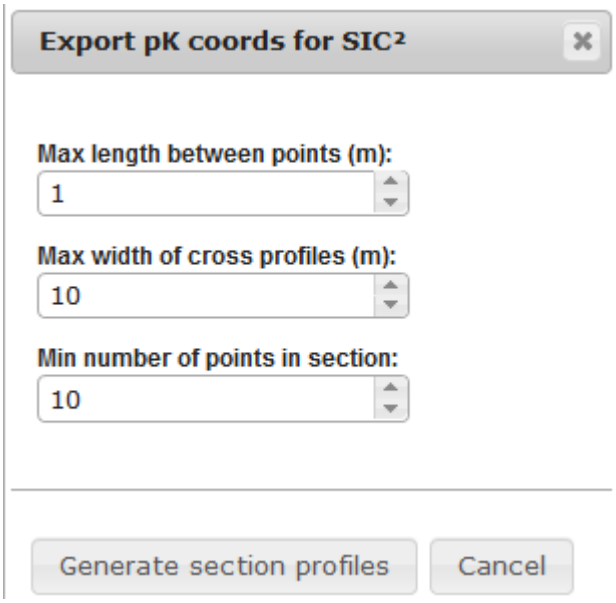


Figure 32.4: doc121

Pressing the “Generate section profiles” button starts the tool with the selected parameters. The result is displayed graphically and produces a text to copy and paste into the EdiSIC text format cross section and profile import tool. The operation may take time and you must wait even if the browser indicates that a script is blocking the operation of the application.

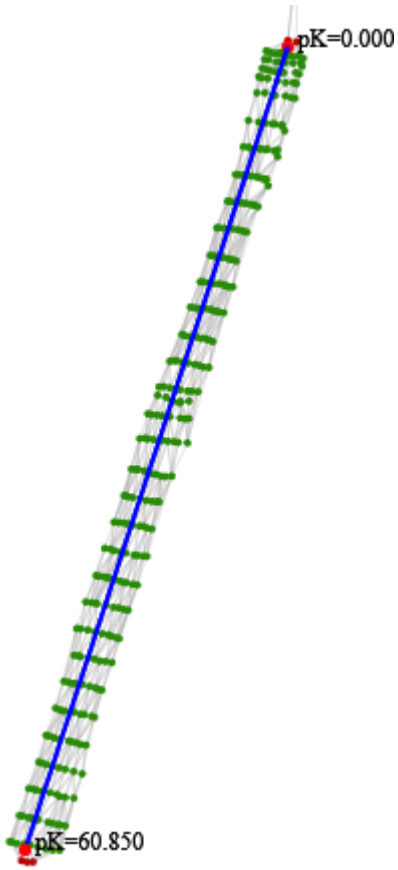


Figure 32.5: doc122

Chapter 33

Add/Change a background map

It is possible to add a background image under the the network. The image can be in modified in the application (brightness, opacity ...) for a better readability.

The background map can be changed in Tools > Add/Change a background map or by clicking on the  button.

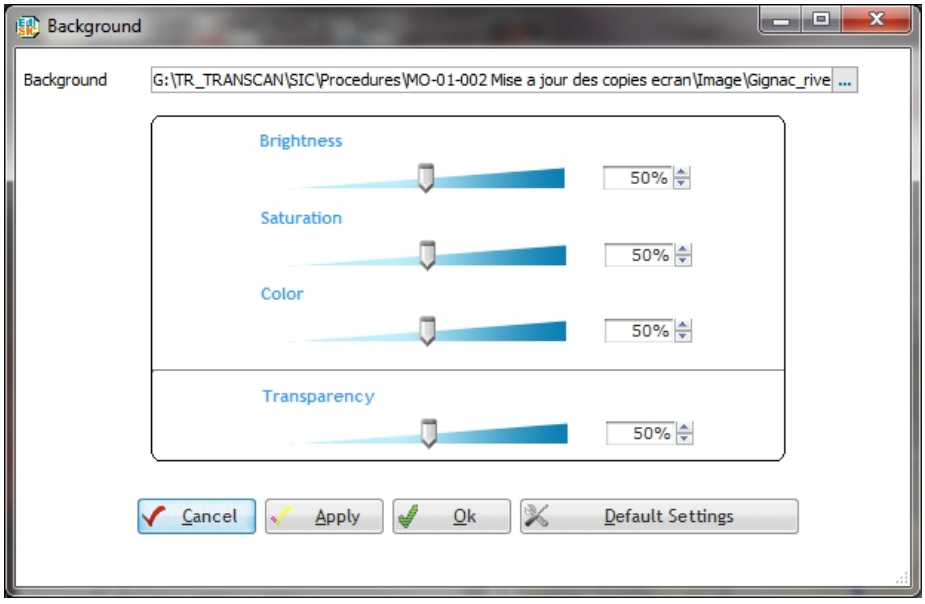


Figure 33.1: fond

More information on image manipulation is available on this page.

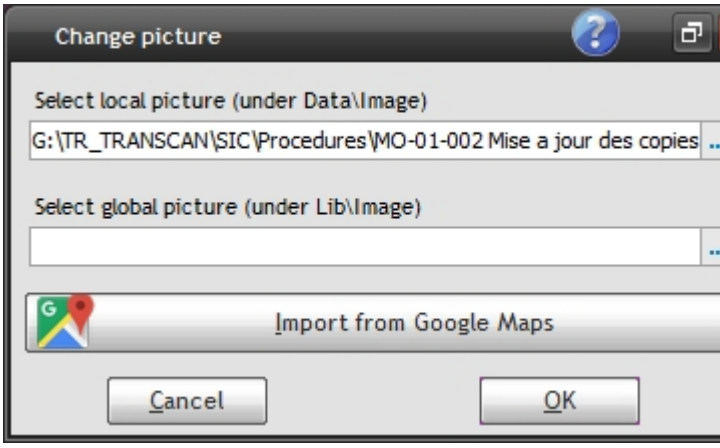
Maximum dimensions for the background map are 2000 x 2000 pixels. Beyond that, the image will be trimmed.

When creating a network, the zoom level is 100%. The background map's dimensions should be adjusted to facilitate data entry with this zoom level.

Chapter 34

Photos associated with network elements

Several network elements can be associated with an image : - background map for the network graph - section - node - offtake (in hydraulic mode)



This association is made through the following window :

The image file associated with an element can be selected from the SIC library (global image) or chosen by the user (local image). Local images are stored in a subdirectory “Image” of the current xml project file’s directory. If the selected file is located elsewhere, it is copied in this directory.

All classic file formats are supported (BMP, JPEG, GIF, PNG, TIF, WMF…)

The image is displayed when the cursor goes over the element associated with it.

Chapter 35

Description of a node and a pond

There are several ways to enter the edit window of a node: - Right-click on a node in the tree or in the graph and choose “Description of XXX” - If the “Select” tool is enabled (click  or press [Esc]), access is via a double-click on a node in the tree or in the graph

35.1 Display of upstream and downstream reaches and definition of a pond

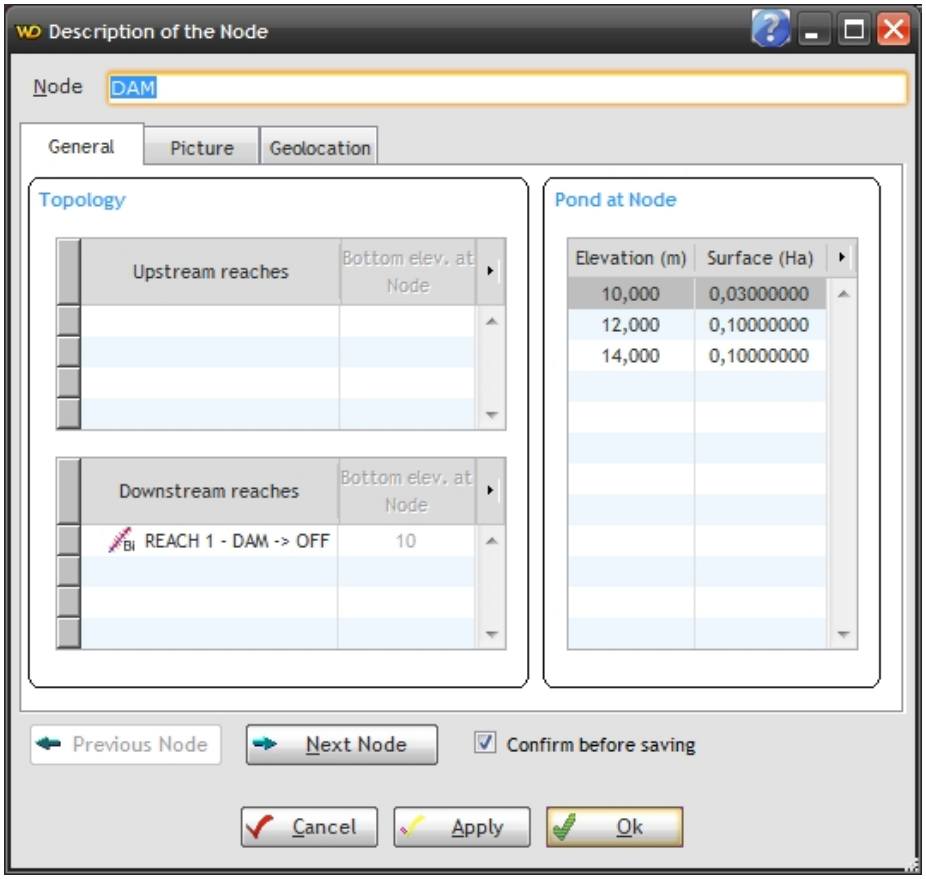


Figure 35.1: descriptionnoeudonglet1

In the "General" tab, there is a reminder of the topology with a list of the upstream and downstream reaches connected to the node as well as a table allowing the edition of a pond at the node by entering a series of elevation/surface values.

Note: In previous versions of SIC (version 4.32 and before) the edition of a pond was located in EDISIR, meaning just used in the unsteady flow calculation module.

35.2 Picture

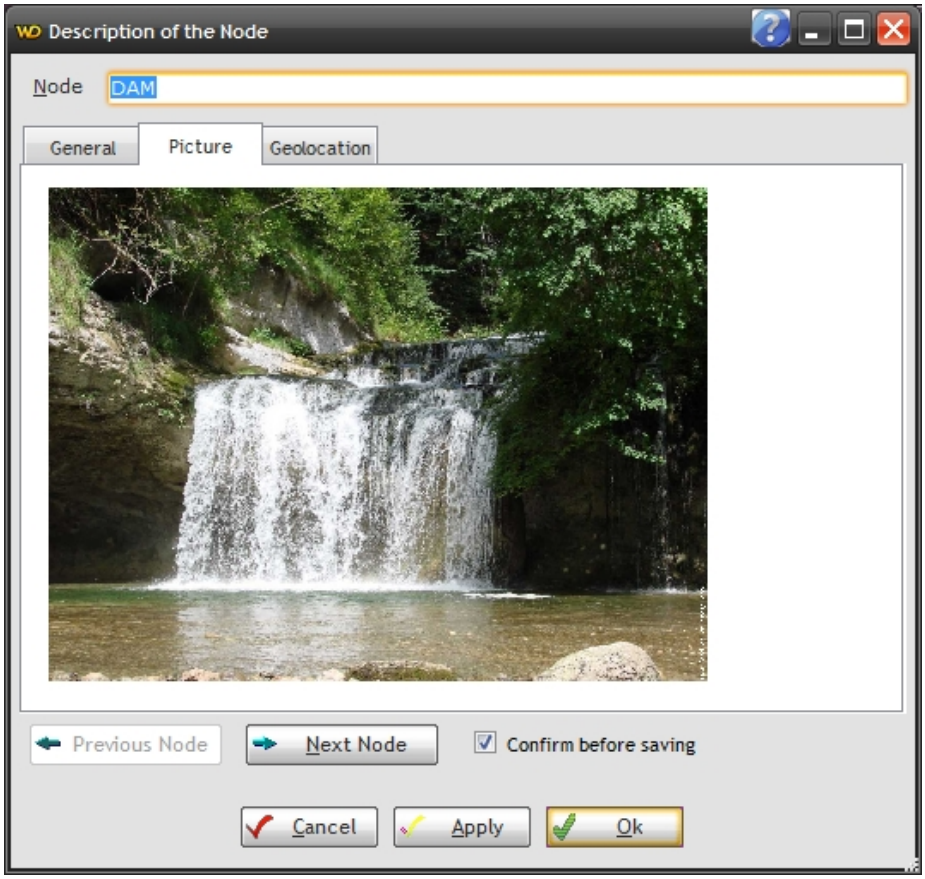


Figure 35.2: descriptionnoeudonglet2

As well as sections, nodes can be associated with a picture by clicking the image in the “Picture” tab. For more information on picture management, see the dedicated page.

35.3 Geolocation

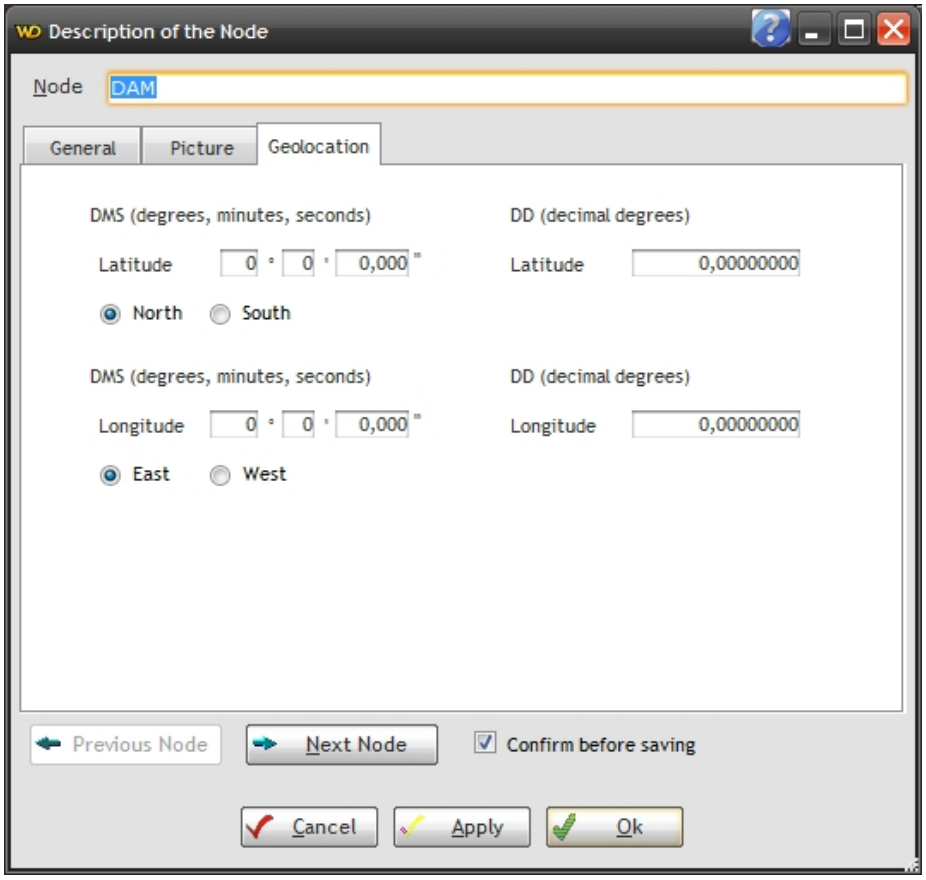


Figure 35.3: descriptionnoeudonglet3

The geolocation of the node (latitude and longitude) can be defined in this window (third tab). This information can then be useful to display the network according to these geolocations (using the automatic definition option), and also used in the import & export of cross section data. This geolocation can be given in decimal degrees or in metric units depending on the chosen option.

Chapter 36

Modification of network structure

36.1 Inserting a node : splitting a reach

A reach can be split into two reaches by selecting it or a section in it with a right-click. In the contextual menu that appears, choose “Split reach.”

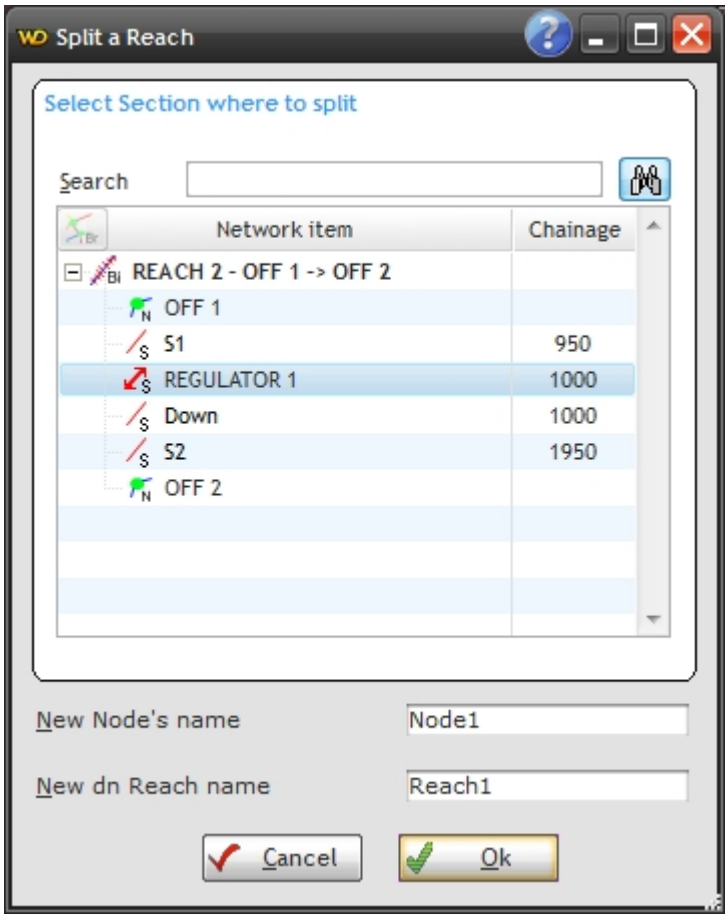


Figure 36.1: scinderbief

The “Split a Reach” window then allows the user to choose on which section the new node should be created and the names of the new node and the new reach. The node thus created inherits the geolocation of the section, if it exists. The newly created reach appears at the bottom of the tree display (left column), but will be correctly classified the next time the network is saved, or the geometry mode display is refreshed.

36.2 Deleting a node : merging two reaches

A node can be deleted by right clicking it and choosing “Merge Reaches” in the contextual menu.

36.3 Converting extrapolated sections into data sections

For a better display of calculated data, it can be preferable to add sections. This can be done by converting some interpolated sections generated by TALWEG into data sections where results can be displayed.

In geometry mode, this feature is available via Tools > Conversion of interpolated sections in data sections.

Chapter 37

Network inconsistencies management

When saving the network before running the mesh generator TALWEG, the application performs consistency checks on the network. The errors are of two distinct levels: - Warning: the inconsistencies found does not prevent the mesh generator and the simulation from running, but the network entered does not match the requirements for using the software in good conditions. - Error: the inconsistencies found prevents the generation of a mesh by TALWEG.

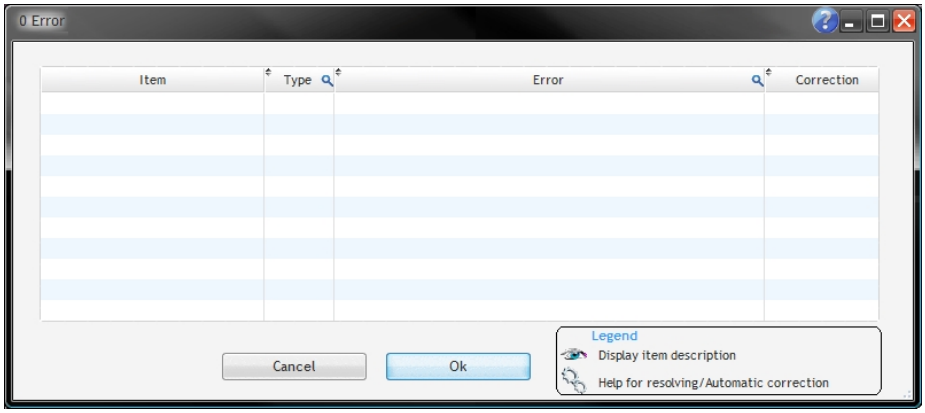


Figure 37.1: erreurs_tal

A click in the “Correction” column opens the window where the error can be fixed.

37.1 Main errors

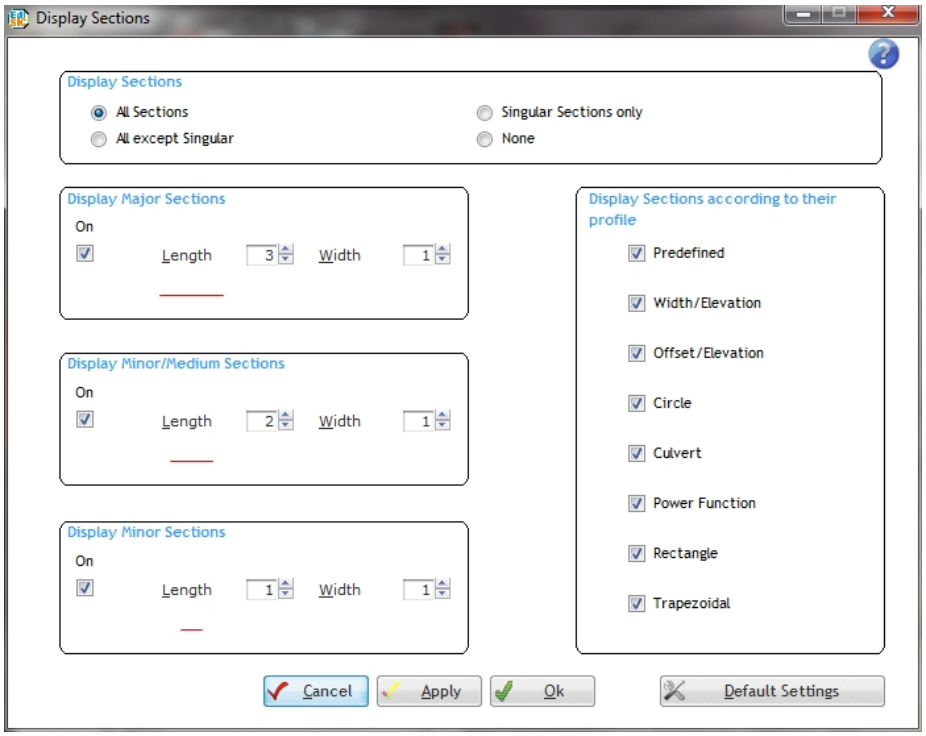
Here are the most frequent error messages : - Locations should be monotonic within the reach. - The space step cannot be nil (otherwise a default value 100 m will be chosen). - Major Section offset must be between min and max offset of Minor or Minor/Medium Sections. - Distance of first Major Section (in order of increasing offsets) must be null. - At least two Major Sections. - Major bed should hold at least two Major Sections with distinct offsets. - Distance of last Major Section (in order of increasing offsets) can't be null, or another Major Section should follow with a distance >0. - Beginning and end abscissas of the reach are similar. Length of Reach can't be null. - There is at least one loop in the network. - Network is splitted. - There is a loop in the branch. - Branch is splitted or is not linear. - Pond elevations at Node must be monotonic.

Chapter 38

Sections, branches and longitudinal profiles display.

38.1 Sections display

Section display on the graph is entirely customizable. In the "Display" menu, it is possible to choose whether to display sections and singular sections. In the same menu, "Section Display Settings" allows for fine adjustments on the appearance and the type of section to display.



38.2 Branches display

This display option is available through the Tools > Display Branches menu.

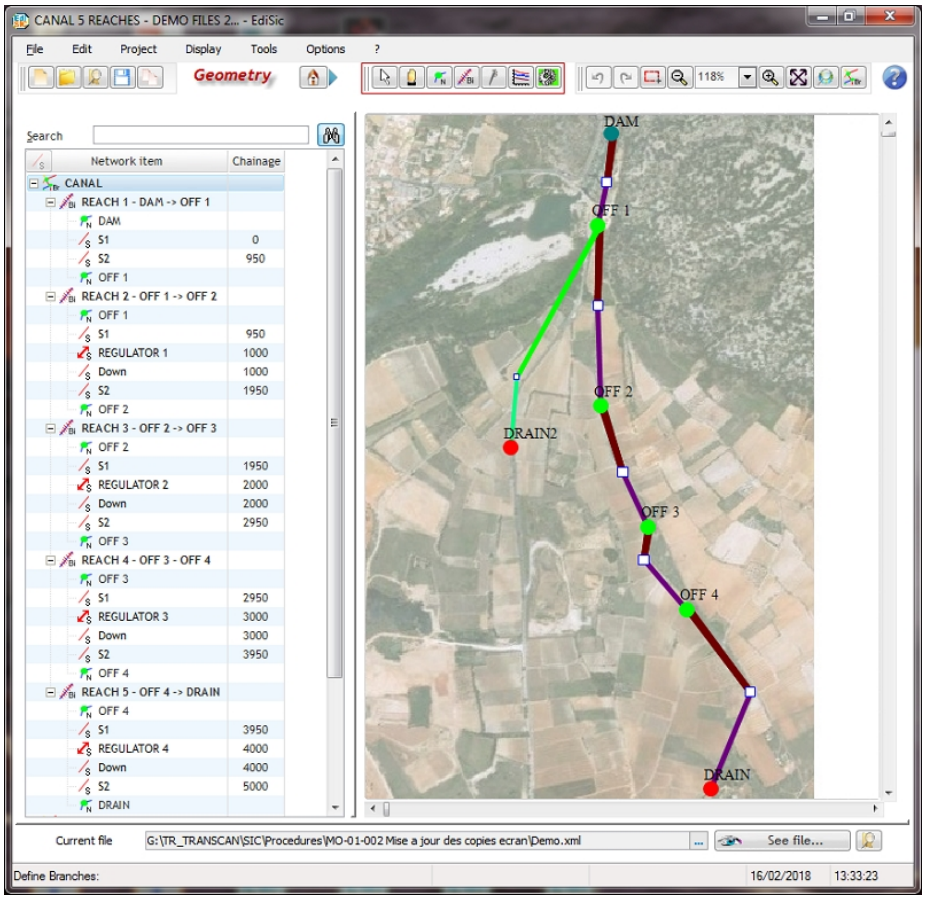


Figure 38.1: premiere_branche_tal

Branches, contiguous set of reaches without diffluence or confluence, are handled automatically by the application and there should be no reason to modify them. However, it is possible to assign one or more reaches to a branch right-clicking the reach and selecting “Change branch> …” from the popup menu.

38.3 Longitudinal profile display

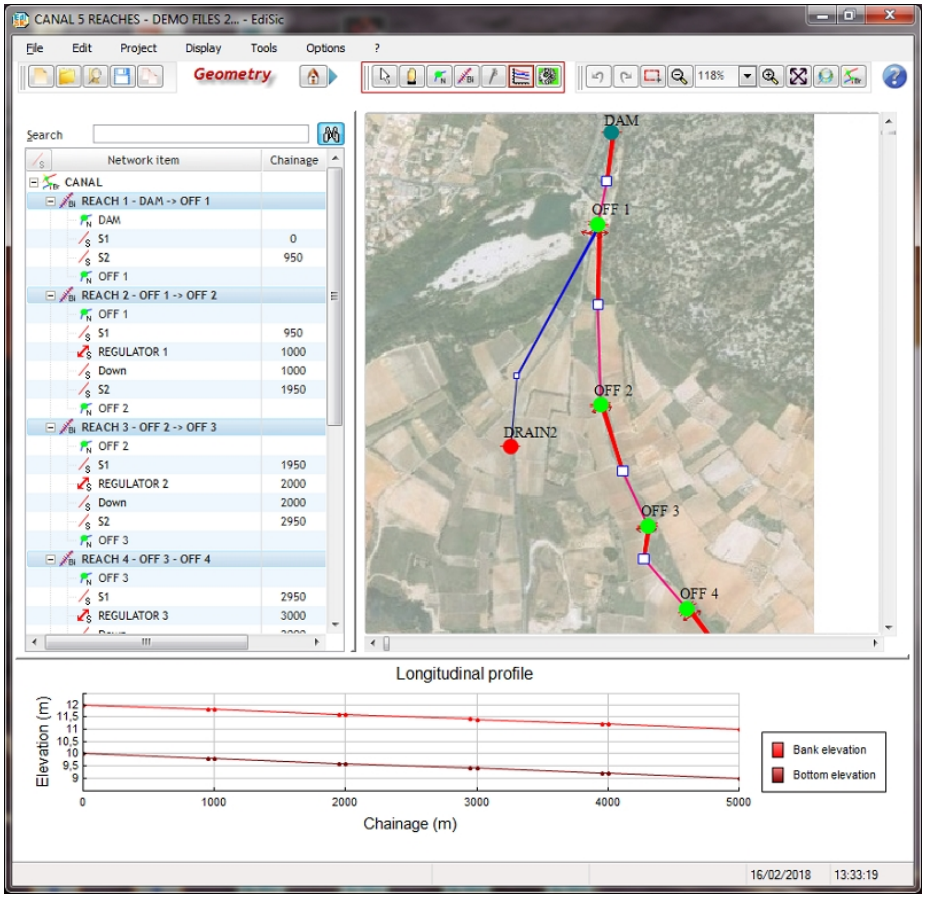


Figure 38.2: premiere_profil_tal

This option which can be very useful for checking the consistency of the network topography displays the bottom elevation of a portion of reach of the network.

Once the reaches are selected, this option is available by clicking the Tools > Longitudinal profile menu or the  button.

Chapter 39

Mesh generation (Talweg)

Before you can enter the hydraulic data and run simulations, it is necessary to run the mesh generator Talweg.

The mesh generator is launched from the Tools > Start network meshing (Talweg module) or by clicking the button .

If the network contains errors, the software displays them and enables the user to fix the inconsistencies, as explained in this page).

If the mesh is correctly executed, the “Start network meshing” button appears in green .

The software creates interpolated sections for calculations, that can be transformed into data section for a better visualization of the results.

More information on the mesh generator are available in the theoretical documentation.

Chapter 40

Compatibility with older version (.TAL geometry files)

The geometry files of previous version of the SIC software can be imported via the File > Open menu, and selecting the .TAL file.

The formats of the .TAL files included are the formats of versions 4. *, as well as the versions generated by the PamHyr editor (Inrae Lyon).

Then, the software automatically suggests to save as a .xml project file.

 | Even though projects can still be saved as .TAL files, data regarding new functionalities of SIC can only be saved in a .xml project file. |

Chapter 41

Export profiles of all sections with X, Y, Z format.

This feature allows you to export in a text file with$separators for section information and tab separators for numeric values all the profiles of the sections of the minor/medium bed and/or of the major bed that are entered as width-elevation, abscissa-elevation, circular, curvert, power, rectangular, or trapezoidal format.

Comments (written after the special character% in the first column) can also be written.

To run the export, you must be in geometry mode and choose "Export profiles of all sections with X, Y, Z format". The software then offers:

- to choose between sections of minor/medium or major bed,
- choose the file format: -* linear text (longitudinal abscissa, Section) -* 3D text (x, y, z, idminmoy) -* 3D kml format (not yet managed, but in progress ...) -* 3D cvs format HEC-RAS style: river, reach, station, x, y, z (not yet managed, but in progress ...)
- choose the format of the sections (original or systematically converted to abscissa-dimension)
- choose to add comments at the head of the file (type of bed) as well as between each section (after the character%, its longitudinal abscissa, its type of format)
- choose to add comments on the nodes separating the reaches comprising the sections (after the % character, the graphic coordinates and geolocation of the midpoint, the names of the upstream and downstream nodes, as well as their graphic position and their geolocation)

- choose to write the geolocation (latitude, longitude in degrees and its decimal fractions, or in metric values depending on the chosen option)
- choose to have a 3D visualization of the exported geometry

Then a file name can be chosen for this export, with txt extension.

1. In "linear text" format, the generated file is in the same format as the one allowing the import of section data into a reach. It contains a line describing the section, with fields separated by the code$, then lines describing the numeric parameters of the section. It can contain optional comments starting with the character % in the first line.

The descriptive line of the section contains its general information, separated by the character$ (see for details).

The following lines (2 or more depending on the format) contain the geometric parameters of the section, according to its format (see for details).

2. In "3D text (x, y, z, idminmoy)" format, the file can contain optional comment lines as above, then the points x, y, z, idminmoy in four columns separated by tabs:

- Column 1: longitudinal abscissa of the section (x)
- Column 2: coordinate of the current point in the profile section (y)
- Column 3: elevation of the current point (z)
- Column 4: flag idminmoy (0 or 1), with a value of 1 if this (these) point(s) is the delimitation point of the minor-mdium beds, and 0 otherwise. For the moment, if the minor-medium overflow elevation does not correspond to a point indicated, this flag will therefore not be entered. In the future we will then create an additional point for this particular point. Note that in the linear format preceding this is well managed because the minor-medium elevation is indicated in a specific line before the description of the points constituting the section.

This file (in its form with comments) can be imported into a 3D visualization tool linked to below.

If the 3D visualization option is selected (see above), we will also have a direct visualization in an interface window. But for the moment this visualization is limited in terms of functionality and maximum points that can be managed (limitation related to the possibilities offered to date by Windev 25, but which should evolve in the future: modifiable angle of view, transparency, etc.) . Other functions make it possible to make 3D visualizations in a Matlab script provided as an example under ex45_3Dvisu under sic / dat, from the 2 file formats treated above. If the comment option has been chosen, then the original format of the section is indicated in this comment, in addition to the longitudinal abscissa of the section.

Soon the 2D positions x,y will be based on the geolocalisation of the nodes and sections (in option), and not only on a linear longitudinal reference as now.

3. In “kml 3D” format the coordinates of the points of the sections will be in geolocation (latitude, longitude) and altitude, and in kml format that can be imported into Google Earth (soon … in progress).
4. In “3D cvs” format, the coordinates of the points of the sections will be written in the csv import format in HEC-RAS style, with 6 columns: river, reach, station, x, y, z. As we have this information in SIC we will also add 2 additional columns with the name of the section as well as the longitudinal abscissa (not yet managed, but in progress …).

The files generated in these 2 formats can either contain the information in the original mode of the section (configured as rectangular, box culvert, trapezoidal, etc., width-dimension, abscissa-dimension), or transformed into abscissa-dimension.

Example in “3D text (x, y, z, idminmoy)” format:

```
%
% DEM of the Minor-Medium bed
%
%
%   0   Abscissa/Elevation
0   -3.18   48.08     0
0   -3.18   44.3     0
0   0   44.3     0
0   3.18    44.27     0
0   3.18    49.05     0
%
%   100 Abscissa/Elevation
100 -8.3    48.85     0
100 -5.4    46.57     0
100 -4.85   46.57     0
100 -3.2    45.01     0
100 0   45     0
100 3.2 45.02     0
100 4.7 46.72     0
100 6.85    48.75     0
%
%   200 Abscissa/Elevation
200 -7.1    49.18     0
200 -4.4    46.56     1
200 -3.3    45.09     0
200 0   45.13     0
200 3.6 45.16     0
200 5.15    46.62     0
200 7.65    48.96     0
%
%   300 Abscissa/Elevation
```

```
300 -7.75   49.34     0
300 -4.86   46.58     0
300 -3.25   45.04     0
300 0   45.04     0
300 3   45.09     0
300 4.55    46.67     0
300 7.5 49.33     0
%
%   400 Abscissa/Elevation
400 -6.7    49.58     0
400 -3.75   47.44     0
400 -2.7    45.29     0
400 0   45.23     0
400 3.3 45.46     0
400 5   47.7     0
400 6.8 49.87     0
%
%   500 Abscissa/Elevation
500 -4.95   48     0
500 -2.3    45.45     0
500 0   45.21     0
500 2.35    45.42     0
500 4.4 47.92     0
%
%   550 Width/Elevation
550 0   45.28     0
550 -1.96   45.52     0
550 1.96    45.52     0
550 -3.03   45.84     0
550 3.03    45.84     0
550 -4.665  48.07     0
550 4.665   48.07     0
550 -4.885  49.31     0
550 4.885   49.31     0
```

The export inserts a line break between each new section.

All sections are managed regardless of their format (abscissa-dimension, width-dimension and parameterized), as well as model sections. The homothety in y is also managed for width-dimension and parameterized formats, but not yet for abscissa-dimension, nor the homothety in z (soon …).

A 3D visualization tool of the geometry is available with your internet browser at the following address: http://sic.g-eau.fr/sic_tools/geometrie.html, or with a Matlab script provided as an example under ex45_3Dvisu under sic/dat.

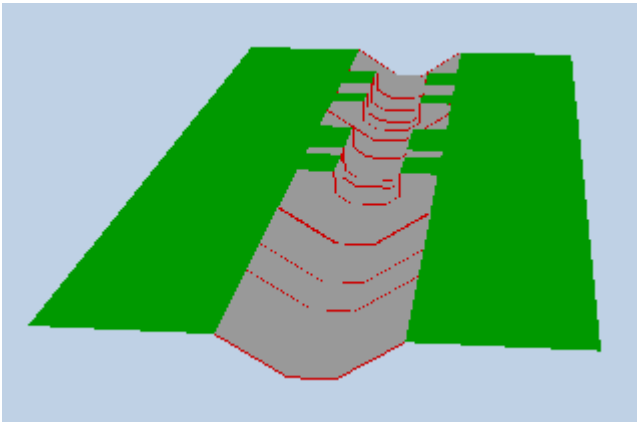


Figure 41.1: doc112

Chapter 42

Change of the bathymetry with an external file

We sometimes want to change the bathymetry of a canal, a channel or a river from an external program (eg Matlab script, Scilab, R, etc) to do the optimization of profiles for example, or to calculate the sensitivity of certain variables to bathymetry parameters.

This can be done with a file named changebathy.txt, which must be located with the xml project file. The example dat/ex32_lacguiers/LacGuiers_v1.xml uses this feature.

This file has the following structure:

```
27 2 0.00000 0.00000 0.10000
28 2 0.00000 0.00000 0.10000
28 3 0.00000 0.00000 0.15000
```

We can have as many lines as we want. The format is free ('*' in Fortran format), that is to say that the numbers can be separated by as many spaces as desired, or a priori also tabs.

On each line:

-* the first number is the number of the reach -* the second number is the number of the section in that reach -* the third is the scaling factor in Y (lateral) that we want to apply -* the fourth is the scaling factor in Z (vertical) that we want to apply -* the fifth is the vertical offset of the bottom elevation, and therefore the entire section (in m)

When executing the calculation program (Fluvia or Sirene) the sections thus modified will be indicated in the control window.

Chapter 43

Parameters for calculating and writing results

The parameters for calculating and writing results are applied in steady flow calculations (FLUVIA) and unsteady flow calculations (SIRENE). These parameters are defined by scenario and will be used for all the variants of the scenario.

Access to the window can be made from: - The Project Explorer window by selecting a hydraulic scenario or variant and pressing the “Calculation settings” button - The main window in hydraulic mode by pressing the  - The main window in hydraulic mode via the Tools menu > Settings.

The window consists of five tabs: - Time: definition of temporal parameters of the simulation (start time, end time, calculation time step, beginning time for writing results and write frequency of results) - Settings for results writing (definition of the data that will be usable in “Result” mode) - Steady flow calculation settings - Unsteady flow calculation settings - General settings

Chapter 44

Steady flow calculation parameters

This tab allows you to change the parameters influencing the calculation algorithm in steady flow, especially when the iterations in case of mesh network, and/or with offtakes in flow calculation mode.

The default settings allow almost all the time to compute a solution with very good accuracy. In some cases one can help, accelerate or make possible the convergence of the algorithm.

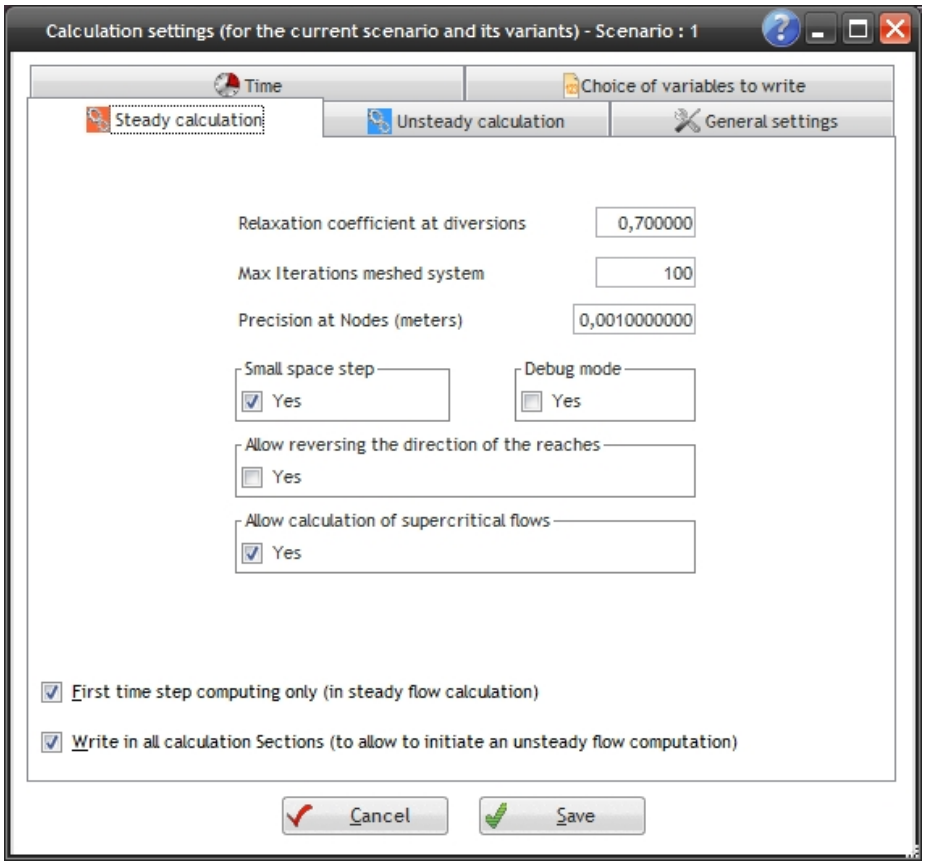


Figure 44.1: paraflu_flu

44.1 Relaxation coefficient at diversions

This option allows you to speed up or slow down how the flow distribution is corrected at diversions during iterations of the calculation mesh. It may be necessary to decrease it when the mesh algorithm stops because of negative flow in a reach.

44.2 Max iterations meshed system

One can increase the number of iterations if the number provided by default is not sufficient to ensure the convergence of the mesh calculation.

44.3 Precision at nodes

This is the maximum permitted deviation of the water elevation at offtakes and diversions (in meters) between the assumptions made in the current iteration and real dimensions calculated considering the hydraulic conditions (elevations and flow rates) calculated in the system.



| In case of difficulty to calculate in steady flow, especially in highly meshed networks where it is difficult to know the direction of flow a priori, two possibilities are offered:

- It is possible to find the hydraulic solution by greatly increasing the value of "precision at nodes" (eg by setting it to 1 or 2 meters and run the calculation in steady flow, and then start the calculation in unsteady flow which is able to reverse the direction of flow in the reaches if necessary, and find the hydraulic solution.
- You can activate the option "Allow reversing the direction of the reaches" |

44.4 Small space step

The "Small space step" mode is used to locally introduce new calculation sections (dividing the local space step by 10) if high water line slopes are detected (slope superior to 5/1000 including reverse slope). Please note that this mode is not supported in unsteady flow calculation, which is not a big problem, but this can introduce "waves" at the beginning of unsteady flow simulation.

44.5 Debug mode

You can activate the “Debug” mode that will record more intermediate variables on the log file during steady flow calculation. These informations allow to follow in more detail the computation iterations.

44.6 Allow reversing the direction of the reaches

As the name suggests, this option allows the mesh resolution algorithm to reverse the flow direction of the reaches during the calculation. Only “intermediate” reaches (ie not adjacent to a node located upstream or downstream of the system) may be reversed.

44.7 Allow calculation of supercritical

When this option is checked, FLUVIA calculates supercritical water lines and the location of hydraulic jumps. Otherwise, the supercritical flows are represented by a critical flow. To perform an unsteady flow calculation from a water line including supercritical flows, it is recommended not to check this option so that the calculation can be done with the “Simple supercritical” of SIRENE.

44.8 First time step computing only (in steady flow calculation)

This is useful for running only one time step even if the time settings specify more time steps to calculate. Indeed, the conventional use of the software to perform an unsteady flow calculation is to create a scenario specifying the simulation time, perform a calculation on the first time step in steady flow, then create a variant to perform calculation in unsteady flow with the same parameters.

This option will be unchecked only if it is desired to carry out a simulation with successive steady flow simulations.

44.9 Write in all calculation sections (to allow to initiate an unsteady flow computation

A calculation made with this option can then be selected for importing an initial condition for an unsteady flow calculation.

Chapter 45

Unsteady flow calculation parameters

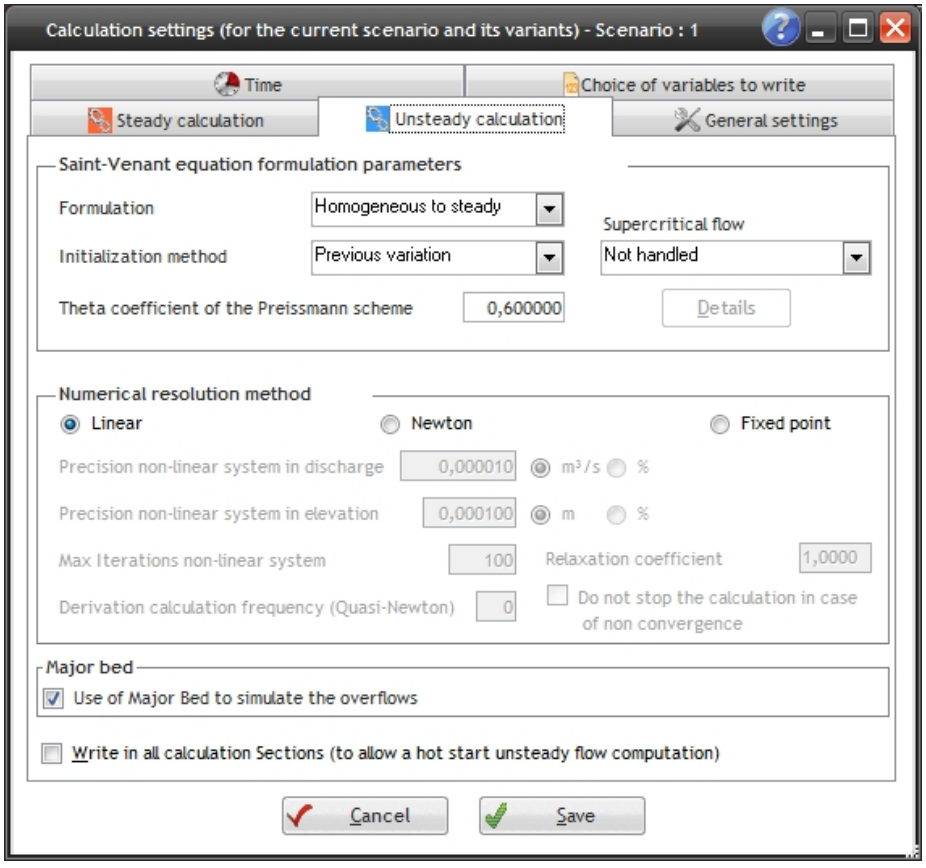


Figure 45.1: paraflu_sir

45.1 Discretization settings

Discretization method:

-* The method called “Classic” is the one described in the theoretically documentation. -* The “homogeneous permanent” method limits the “waves” that can appear when starting a unsteady flow calculation from a water line computed by the steady flow calculation (Fluvia). Indeed, in this case, the discretization of the transient equations gives exactly the equations of the steady state when one removes the derivative terms with respect to time. It is less conventional because it can lead to not very normal hydraulic phenomena in the case of change of geometry (cf Cunge book 1980). The small “waves” mentioned below can be explained by the fact that the stabilization regime of the classical scheme does not lead exactly to the initial permanent water line. The new equilibrium will not be very far, but to achieve this, a balancing of levels and therefore mass transfers will be realized.

Initialisation method:

-* Zero Variation: for a given time step, the calculation of the new water line is initialized from the water line calculated at the previous time. -* Previous Variation: The calculation of the new water line is initialized from the water line calculated at the previous time at which we apply the flow and water elevation variations observed in the previous time step. This option can accelerate the convergence if hydraulic variables are often progressive. However, our experience indicates that convergence is very fast, and this option is not essential.

Theta coefficient of the Preissmann scheme:

This coefficient sets the implicit elements of the numerical scheme. It should be set between 0.5 (stability limit) and 1 (total implicitation). If this condition is met, the Preissmann implicit scheme is unconditionally stable. It is set to 0.6 by default. It is possible to modulate it automatically according to the Froude (cf below). This can be useful because a high Theta coefficient will generate more numerical dissipation and thus limit the numerical oscillations that can appear with high Froude. Putting it large (close to 1) all the time is not very good numerically, so the option to set it variable, just according to the Froude, is interesting.

Note: by default HEC-RAS advises to start modeling with Theta=1, which is quite violent anyway, because it generates a lot of numerical dissipation.

Simple super critical

This option allows, in the case of classical discretization as well as the homogeneous to steady flow discretization, to gradually eliminate the inertia terms of the dynamic equation of Saint-Venant when the Froude number exceeds a certain value and for example becomes close to 1. This is a classical method to better manage high Froude numbers, as well as supercritical flows in some cases. This stems from the fact that it is these terms that generate numerical

problems in the Preissmann scheme at high Froude numbers. We delete these terms progressively and linearly: 0% for Fr = FrMin (default 0.6) up to 100% for Fr >= FrMax (default 0.9). The default values of FrMin and FrMax can be changed in the advanced settings of this option ("Detail" button). This option is to be used with moderation because it presents certain risks of numerical stability with Froude high and especially with low water depths.

-* No: The option is not applied and the calculation is stopped as soon as the Froude is greater than or equal to 1 on one of the sections of the model. -* Yes local: The option is only applied to sections with high Froude, at the moment it happens and gradually as explained above. -* Global Yes: The option is applied to all sections of the model and removing 100% of the selected terms during all the simulation time. -* Sart-Preissmann: more advanced method with modification of the Preissmann scheme cf article Sart et al 2010 (http://hal.ird.fr/hal-00632009/document).

One can also, in the advanced options ("Detail" button), choose the term(s) to be deleted: the local acceleration termdQ/dt, and/or the term d convective acceleration$\frac{d\frac{Q^2}{S}}{dx}$. If we remove these two terms, we have the model of the diffusive wave. Older versions of SIC (prior to 5.37b) allowed to remove only the second term, convective acceleration which is the one that poses the problems of numerical stability at high Froude numbers. These terms are progressively removed locally or globally (depending on the option chosen above). For the local option, this or these terms are removed for Froude values between FrMin (0.6 by default) and FrMax (0.9 by default), but nothing prevents to put FrMin=0 and FrMax=0, which implies that these terms will always be removed whatever the Froude values. In this case we have an implementation of the diffusive wave model. The global option will give the same thing. Attention in case of variable Theta, this reduction will also apply on this coefficient Theta, in this case one would have then Theta=ThetaMax.

We currently do not allow to remove the term$\frac{dy}{dx}$, which would then lead to the kinematic wave model. See for example http://www.agroparistech.fr/coursenligne/hydraulique/degoutte1.pdf for information on these simplified variants of the Saint-Venant equations.

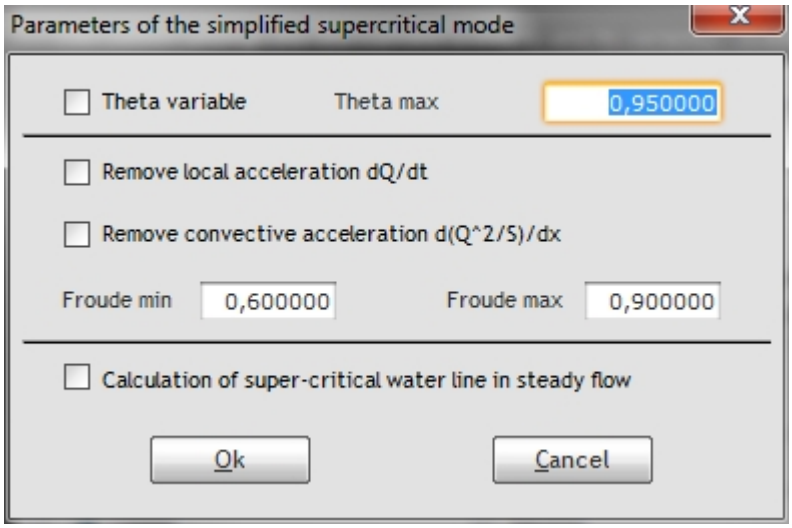


Figure 45.2: paraflu_torrentiel_details_sir

Another option is to propose a variable Preissmann Theta implicitation coefficient, increasing when the Froudes are high. It is varied with the same logic as for the terms of inertia, locally, and linearly between Theta and ThetaMax depending on the Froude for the local option. For the global option one has in this case systematically Theta = ThetaMax, everywhere and all the time.

Another activatable option is to recalculate the torrential water line in steady state. Flow rates and water elevations are first calculated with the conventional method, and then the water elevations are recalculated with the steady state algorithm. This option is to be used without reserve and without guarantee, especially in the case of meshed and/or branched networks, and is no longer conservative in volume, since the dimensions of the water line are recalculated.

45.2 Numerical resolution method

The calculation can be done in linear (without iterations) or non-linear with the Newton or quasi-Newton method (the derivative is not calculated at each iteration). In the case of non-linear calculation, the iterations convergence criteria should be defined:

Precision of the non linear system

The convergence of the linear system is tested on flows of structures and boundary conditions. The iterations are also made on the Saint-Venant equations.

The calculation succeeds when the deviations of flow and water elevation are below the requested precisions. The requsted precisions can be defined in absolute

terms (in m3/s for flows and in m for water elevations) or in relative values (in proportion of the flow rate or in proportion of the water elevation with reference to the lowest elevation in the studied model).

This test can be very demanding if the precision is set to a small value (eg.: 0.0001 m3/s). On systems with significant flow rates, it is desirable, even mandatory, to increase it. For example, the Rhône, which has flows of about 2000 m3/s the system will fail to converge to an accuracy of 0.0001 m3/s. However with a value of 0.01 m3/s the calculation is done smoothly and with a largely sufficient accuracy. If you have any messages indicating that convergence is not possible, decrease the coefficient within reasonable limits (ex .: 0.1% of nominal capacity).

Max iterations of the non-linear system

Maximum number of iterations to converge the system. If a significant accuracy is required on a system having difficulties to converge, it may be necessary to increase it.

Note that if the system fails to converge, SIRENE restarts the calculation again by halving the time step of calculation. This is repeated if necessary until divide by 8 the time step calculation chosen by the user.

Derivative calculation frequency (Quasi-Newton)

- 0: the calculation is linear (without iterations).
- 1: the calculation is performed by non-linear with Newton's method
- 1: the calculation is performed by quasi-Newton (the calculation of the derivative is not performed at each iteration)

Do not stop the calculation in case of non convergence

By default, if the calculation did not reach the precision required after the maximum number of iterations and after the divisions of the time step of calculation, the simulation is stopped. Enable this option allows in this case to only display a warning message and switch to the calculation of the next time step.

45.3 Write in all calculation sections (to allow a hot start unsteady flow computation)

If selected, we can use the results of the simulation as the initial condition for a new unsteady flow simulation. For networks with many sections of calculation, it may be appropriate to increase the write frequency results in order not to burden the XML file and associated treatments (see Time Settings).

Chapter 46

Time settings

This tab is used to define the simulation time parameters and result writing time parameters for a scenario and its variants.

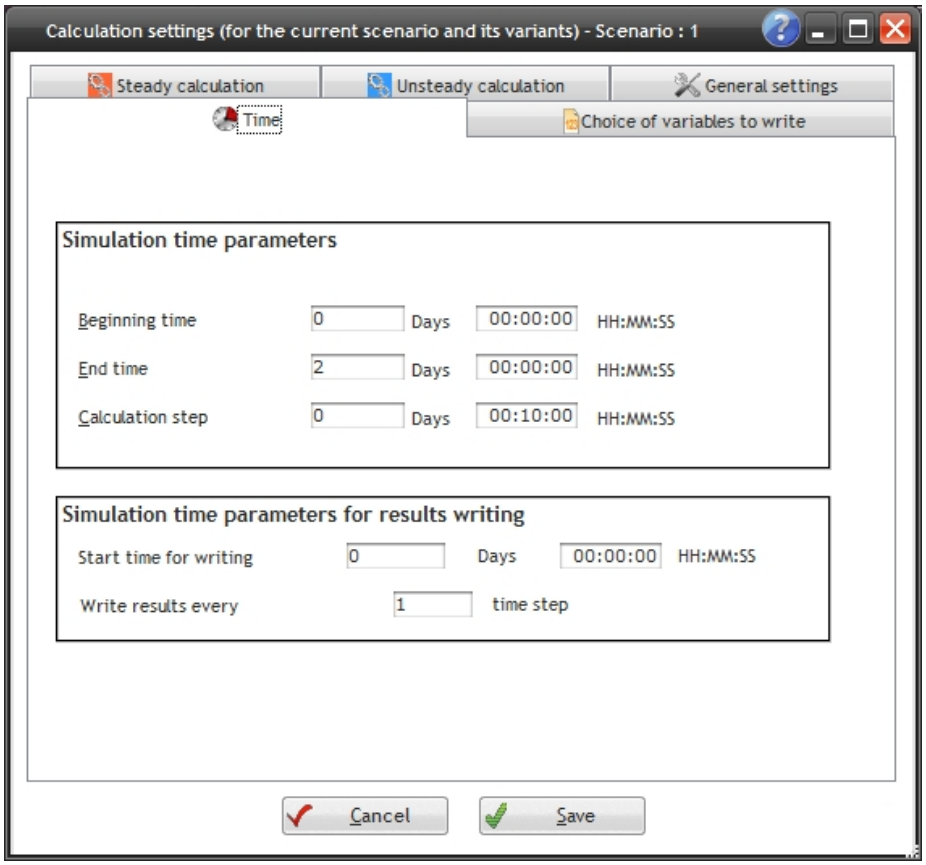


Figure 46.1: paraflu_temps

Simulation time parameters

- Beginning time (Day: 0 Time: 00:00:00 by default)
- End time (Day: 0 Time: 00:00:00 by default)
- Calculation step (10 minutes by default)

Simulation time parameters for result writing

These parameters are used to reduce the size of written results during simulation. This allows to get simulations and operations on results faster but also to solve memory overflow problems that may occur with very long simulations on large systems.

- Start time for writing (day: 0 Time: 0:00:00 by default): useful if one is interested in a phenomenon having place only after a certain time step.
- Write results frequency (1 time step by default): useful if the results are not essential at each calculation time.

46.1 Steady flow simulation

By default, for a steady flow simulation (FLUVIA), only the first time step is executed. To run several successive steady flow regimes over time, the option "First time step computing only" in the "Steady calculation" tab should be unselected.

46.2 Unsteady flow simulation

The default value of the calculation time step is set at 10 minutes. One must be careful when changing this value since it has an influence on the numerical scheme and consequently the accuracy of the calculation. However, it has no influence on the stability of the numerical scheme since the scheme used is the implicit scheme Preissmann which is unconditionally stable (see Theoretical documentation).

We can choose local spaces step and the time step as desired according to the study carried out. What is known (cf. Cunge 1983, Malaterre 1994) when the [?Courant–Friedrichs–Lewy condition] is very small (eg. <0.1) or very large (eg.> 10) then the scheme introduce numerical diffusion and phase difference. It is therefore desirable to keep the Courant–Friedrichs–Lewy condition in the range [0.5, 10].

Chapter 47

Settings for results writing

Results recorded can be set for each scenario of the project.

The user can choose which data that will be available in the results for each network object (section, node, offtake, device). To do this, open the calculation settings window and check the data on the tab “Choice of variable to write” (see below).

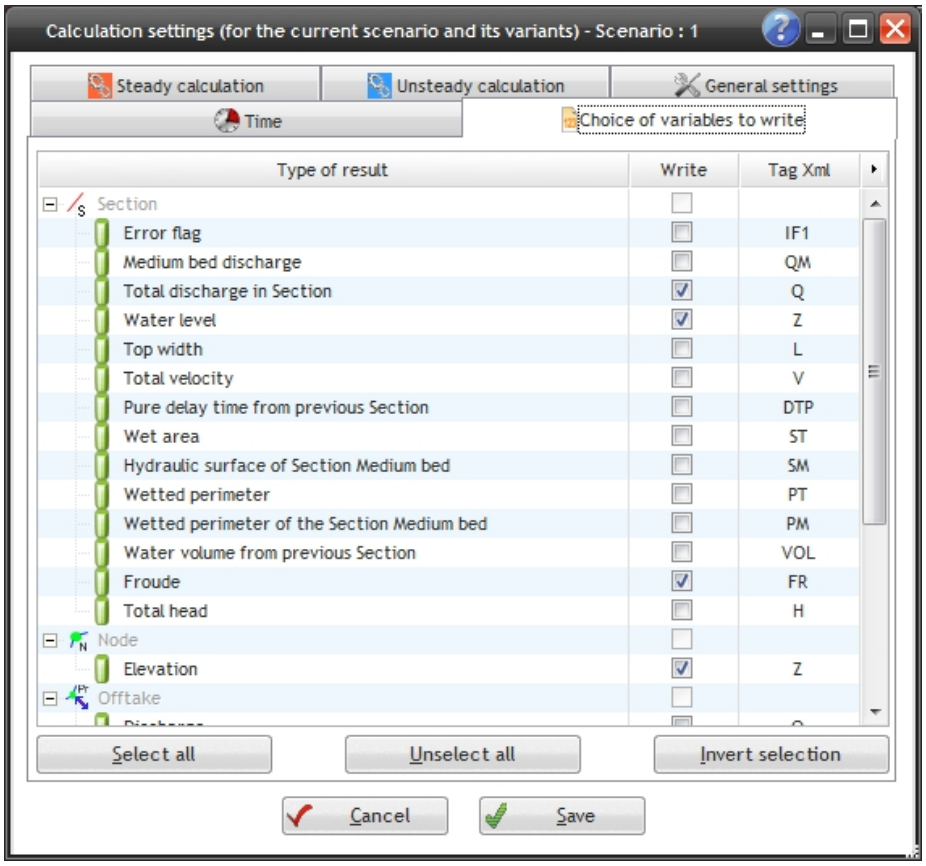


Figure 47.1: paraflu_result

In addition to the variables mentioned in the table above, every regulated variable (by law function of time, by objective regulation at an offtake or at a cross-structure, or variable regulated by a regulation module) will by add to the results.

Chapter 48

General settings

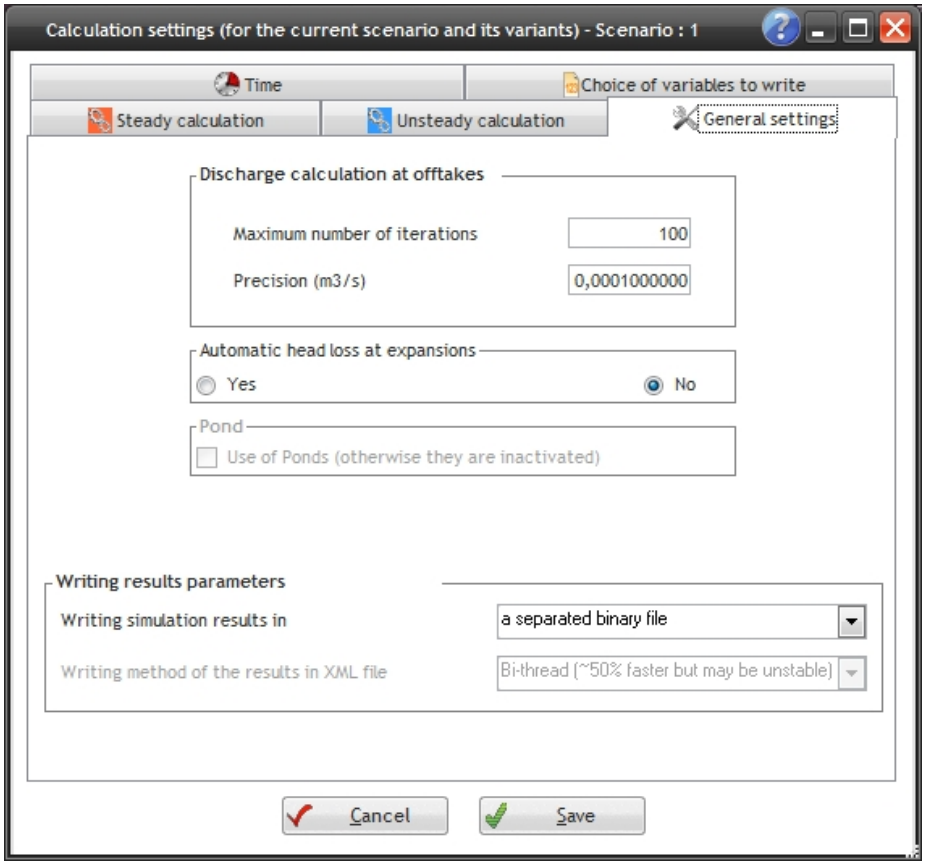


Figure 48.1: paraflu_gen

The tab “General settings” contains three blocks:

48.1 Discharge calculation at offtakes

This block rules the criteria of the iterative calculation at offtakes. The parameters are the maximum number of iterations and the precision to reach in m3/s. The default values are suitable for most cases. It could be necessary to refine theses criteria if one ask a greater precision inthe precision at nodes in steady calculation parameters or in the precision of the non linear system in unsteady calculation.

Ce bloc règle les critères du calcul itératif aux prises. Les paramètres sont le nombre maximum d'itération et la précision à atteindre en m3/s. Leur valeur par défaut convient pour la plupart des cas. Il peut être nécessaire d'affiner ces critères si on demande un précision importante dans la précision aux noeuds en calcul permanent ou dans la précision du système non linéaire en calcul transitoire.

48.2 Automatic head loss at expansions

An automatic head loss at expansions may be added to take into account the loss of hydraulic head at locations where the cross section is getting suddenly larger (width increasing and/or bed elevation decreasing). In this case, a head loss is added using the Borda formula$\Delta H = c(V_{upstream} - V_{downstream})^2/2g$ where$c$ is a parameter dependent on the angleα of the enlargement: - $c = 0.525ln(tan(\alpha))+1$ (Neperian logarithm) - $c = 0$ for $tan(\alpha) < 0.15$ - $c = 1$ for $tan(\alpha) \geq 1$

48.3 Writing simulation results

There are two options: - into the XML file project - into a separated binary file

This option, when selected, is valid for the current scenario and all its variants. But you can change this option at any time, and thus have, if desired, disparate results in the format that you want for the various calculations, either for a scenario or for one of its variants, whether for a steady flow or an unsteady flow calculation. In the case of choosing the XML file, all the results of calculations performed for this branch of the project (scenario and variants) after this choice will be written in the XML project file. The advantage is to have all the input and output data in a single file. But the downside is that the XML file can become very large, and depending on the power of the computer (memory in particular), it can sometimes cause problems with losses of pieces of the XML file. We have indeed noticed that when the XML file is over 60 or 70 MB (megabytes), these problems can appear. In addition to the risks related to the size of the XML file, the time of writing and reading the results of this XML file can become very long. It is for these reasons that we have added the option of writing the

results on binary files. These files have the extension .res for the results in steady or unsteady flow, and .rci for the initial conditions. The filenames with the .res and .rci extensions are constructed from the name of the XML file with an additional _ns_nv, where ns is the number of the scenario and nv the number of the variant (or 0 if it is not a variant, but directly a scenario). All these files are written into the subdirectory where the XML file is located.

48.4 How to write the XML file results

There are two options: - Mono-thread - Bi-thread

The bi-thread option on computers with multiple cores, significantly increase the speed of writing the results of the XML file in a simulation with Fluvia or Sirene. It has the disadvantage of not being completely stable and, in rare cases, crashes the simulation program at the time of writing the results.

48.5 Note on possible instabilities of the software

If you experience random crashes of Fluvia or Sirene calculation programs at the time of writing the results, choose the mono-threaded option.

If you notice any problem with the instability of the software and possible data loss, or even a complete loss of the XML file, then use the option with the results on binary files. Automatic backup .bak files are also created in the subdirectory of the project, which generally allows to recover the old XML file. It may also be safer to work directly on a subdirectory with automatic archiving (eg Dropbox), which will allow to retrieve all the intermediate versions of the files in case of problems.

Chapter 49

Devices available in SIC

The devices can be defined in cross-structures at a singular section or in 1st or 2nd level of device structure in an offtake.

In a structure, several devices can be set in parallel.

Several type of devices can be combined in a same structure. For each of them, SIC proposes a choice of discharge equations (see below): - Rectangular sill (CEM 88 (D), FreeFlow, CUN 80, CEM 88 (V)); - Rectangular gate(CEM 88 (V), Submerged, CUN 80, CEM 88 (D)); - Circular gate (CEM 88 (V), Submerged, CUN 80, CEM 88 (D)); - GEC-Alsthom auto-regulated gates: AMIL, AVIO-AVIS et MIXTES (GOU 93, CEM 02 (V)); - Trapezoidal sill (CEM 02 (V), CEM 02 (D)); - Trapezoidal gate (CEM 02 (V), CEM 02 (D)); - Pipe - APM

Chapter 50

Discharge equations at devices

- "CEM 88(V)" (weir-undershoot gate) is the default equation for rectangular or circular gates. It takes into account the different flow conditions (submerged, freeflow, open channel flow, and flow in pressure). Moreover, it assures discharge continuity between these differents flow conditions.
- "CEM 88(D)" (weir-orifice) is default equation for rectangular sill. As "CEM 88(V)" it takes into account all flow conditions with discharge continuity.
- "CUN 80" are equations described by Cunge in his book [[Cunge, Holly, Verwey, 1980, "Practical aspects of computational to river hydraulics", Pitman, p. 169 for weir and p. 266 for gates]], or in more detail in Mahmood and Yevjevich article [[Mahmood K., Yevjevich V., 1975, "Unsteady flow in open channels, Volume 1 and 2", Water resources publications, Strong Collins, USA, 923 p]]. "Cun80" 's equations also take into consideration the different flow conditions: submerged, freeflow, open channel flow, and flow in pressure pipes (as Cemagref equations "CEM 88"), but there is no continuity between conditions of open channel flow and pressure pipes flow. Therefore calculation problems can arise close to this transition.
- "Submerged" equation corresponds to the classical submerged gate equation.
- "Freeflow" equation corresponds to the classical freeflow weir equation.
- "GOU 93 Amil" et "GOU 93 Avi" are the equation proposed by default for GEC-Alsthom AVIS/AVIO and AMIL gates. The latter come from constructor documents and permit to reproduice the characteristics observed in GEC-Alsthom constructor's abacus.
- "CEM 02(V)" is default equation for trapezoidal gates. It is the sum of "CEM 88(V)" equation for the rectangular part and Gourley equation for the triangular part.

- "CEM 02(V)" is default equation for trapezoidal sills. It is the sum of "CEM 88(D)" equation for the rectangular part and Gourley equation for the triangular part.

Chapter 51

Overflow (Overtopping a gate)

One takes into account the fact that the undershot gate has a certain height and if the water level rises upstream of the gate, water can flow over the gate. The flow overtopping the gate is then added to the flow resulting from the previous pipe flow computations

To activate overflow on a gate: - check the box “Overflow” at the bottom of the screen, - define height of gate - choose a sill equation (from rectangular sill equations) - define discharge coefficient

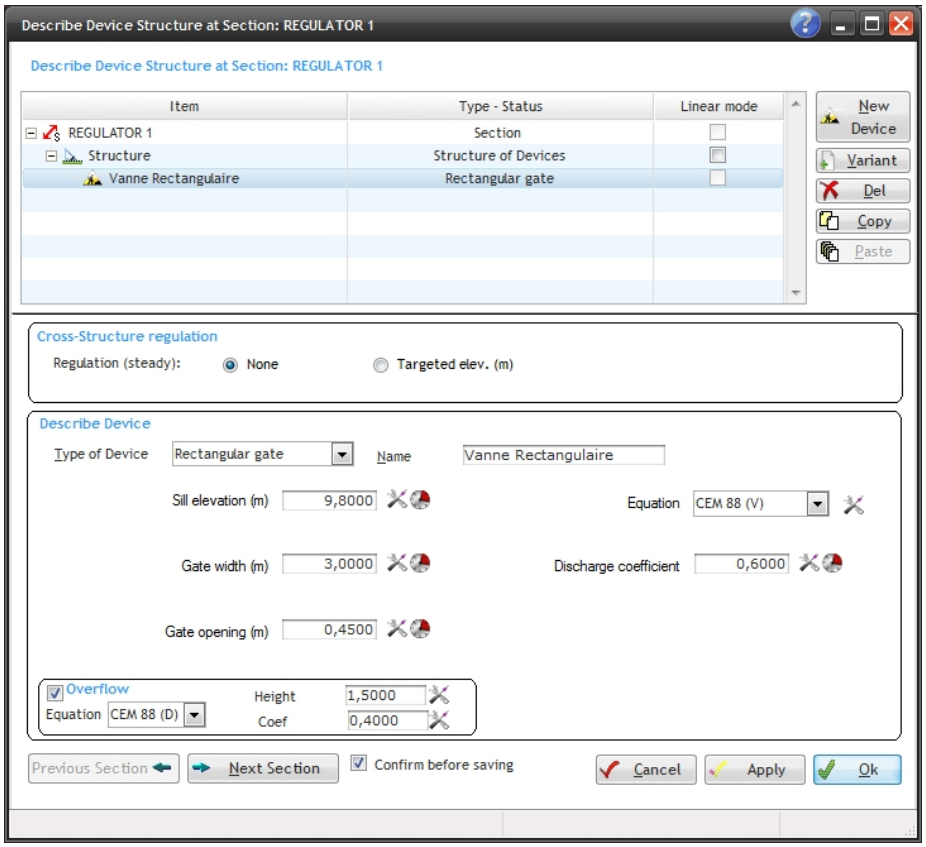


Figure 51.1: descriptionouvragetravers

Overflow starts when upstream water elevation exceeds the sum of the sill elevation, the opening of the gate, and the gate height.

The width taken into account for overflow is: - the width of a rectangular gate; - the width of a trapezoidal gate at bank elevation.

If Overflow is not checked, gate's height is considered as infinite and no overflow is possible.

Chapter 52

GEC-Alsthom gates (AMIL, AVIS, AVIO, Mixtes)

The Gec-Alsthom gates are also available: AMIL, AVIS, AVIO and Mixte gates.

When you select a type and reference, the default values (as suggested in the Gec-Alsthom documentation) are proposed for the different parameters. You can modify them if you want. But be aware that the correct functioning of these gates depend on the correct selection of these parameters. When you select a new type or reference, or when you change the radius of the gate, the maximum allowed head loss Jmax, the corresponding discharge Q for Jmax, the maximum allowed discharge Qmax, and the corresponding head loss J for Qmax are indicated in order to help you to select the proper reference linked to your local hydraulic conditions.

The abacus recalculated from the equations we programmed are displayed in Annex. We can verify they fit the original ones of Gec-Alsthom.

52.1 Type

The "type" corresponds to the range of the selected valves (Amil, Avio High head, Avio Low head, Avis High head, Avis Low head, Mixed without mask max. opening 45°, Mixed without mask max. opening 55°, mixed with mask).

52.2 Reference

The "reference" corresponds to the manufacturer catalog Gec Alsthom (now Hydrostec). If you want to model a different gate you can select "other." In this case we use some dimensions required for the calculation determined from the radius R indicated by drawing values close to conventional gates:

```
-* Amil :
-** D=1.6*R
-** E=0.95*R
-** b=0.91*R (Width at bottom)
-* Avis high head:
-** E=0.81*R
-** b=1.19*R
-* Avis low head:
-** E=0.95*R
-** b=1.19*R
-* Avio high head:
-** L=0.5*R
- Avio low head:
-** L=R
```

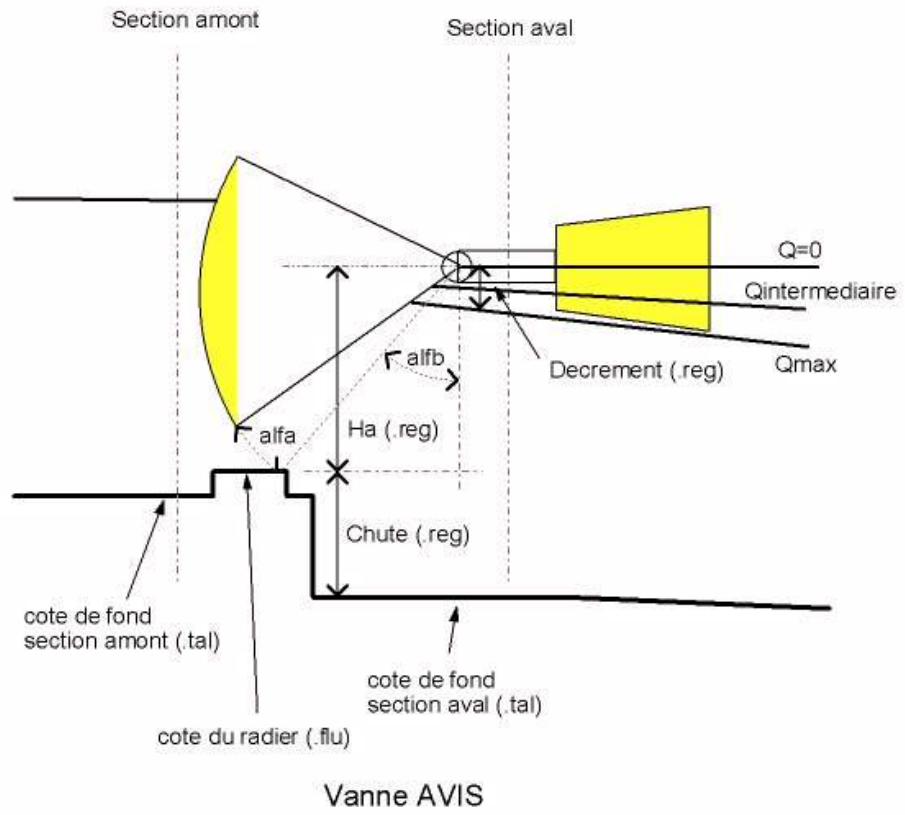


Figure 52.1: doc67

52.3 Decrement

The "decrement" is the difference between the minimum and maximum regulated levels. For stability reasons this decrement can not be null. For an AMIL gate the default value suggested by the manufacturer (and the SIC software) is d=0.02*D. For an AVIS gate, the default value is d=0.028*R (low head) and d=0.032*R (high head). For an AVIO gate, the default is d=0.028*R (low head and high head).

52.4 Axis elevation

The "axis elevation" is the elevation of the axis of rotation of the gate. It corresponds to the set point for Q=0 for the AVIS, AVIO and the AMIL gates in alternate setting (or for Q=Qmax for an AMIL gate in conventional setting). The default axis elevation proposed by the manufacturer is calculated and proposed by the SIC software.

52.5 Sill elevation

The "sill elevation" is the elevation of the bottom of the civil engineering work at the point where the gate stops in full closure.

52.6 Radius

The "radius" is the radius of the radial gate. It is proposed automatically according to the reference of the chosen gate, but it can be modified. It is an essential parameter for calculating the operation of the gate.

52.7 Discharge equations

The equation "GOU 93" corresponds to the equation deduced from a document provided by Goussard, former engineer of Gec-Alsthom, and written in 1993.

The equation "CEM 02" corresponds to an equation developed by Cemagref in 2002.

52.8 Maximum opening

The variable "maximum opening" corresponds to the actual blocked opening of the gate if the equation chosen is "CEM 02 (V)" and corresponds to the maximum opening reached for the adjustment related to the decrement of the gate if the equation chosen is "GOU 93". A default value corresponding to the mechanical stop is offered by default depending on the type and reference of

the gate. This value can be changed if a different stop limits or maximizes the maximum reachable opening.

For an Amil gate in conventional setting the gate will be at this max opening as soon as the level of the upstream water will be >= the elevation of the axis. For an Amil gate in an alternative setting, the gate will be at this max opening as soon as the level of the upstream water is >= the axis elevation + decrement. For an Amil gate in intermediate setting the gate will be at this max opening as soon as the upstream water level is >= the axis elevation + Z upstream max (see below for the definition of this variable).

For the Avis and Avio gates, the gate will be at this max opening as soon as the level of the downstream water is <= the elevation of the axis - decrement. Changing this value amounts to modifying the stop controlling the maximum opening of the gate and to modify the balancing of the gate.

52.9 Minimum Opening

By default, GEC gates in SIC have a minimum closing opening of 0.5% of the gate radius. This makes it possible to mimic the fact that these gates are not perfectly sealed and facilitates the computation in steady state for the AVIS and AVIO gates because a gate completely closed would generate a very important head which will propagate from downstream upstream.

52.10 Z upstream max

The variable “Z upstream max” corresponds to the maximum water height in relation to the axis elevation (corresponds to the setting in the case of the AMIL gate): 0 <= Zmax <= d. Zmax = 0 for the standard setting and Zmax = d for the alternate setting. Intermediate settings are possible. For the conventional adjustment, the upstream levels will therefore vary between Z = elevation of the axis - d for Q=0 and Z = axis elevation for Q=Qmax. For the alternative setting, the upstream levels will therefore vary between Z = axis elevation for Q=0 and Z = axis elevation + d for Q=Qmax. This variable is not used for gates other than Amil gates.

52.11 Filtrering S1/S2

The variable “Filtering S1/S2” makes it possible to filter the water elevation influencing the displacement of the float of the gate. We do not currently use this variable for Amil gates. One could imagine using it to mimic the damping related to the piston damper installed on these gates. For the Avis and Avio gates, this value mimics the filtering linked to the tank in which the float is located. Theoretically, the coefficient S1/S2 to be entered here corresponds to the ratio between the free surface of the water in the tank around the float

divided by the surface of the orifice which causes the tank to communicate with the channel downstream. In practice, however, this coefficient must be increased to take account of the additional head losses associated with the possible piping of this water intake.

52.12 Tray offset

The variable "Tray offset" corresponds to the deport of the water intake of the tank towards the downstream reach. In practice, in particular when there are several gates in parallel, the water intake of the tank(s) is connected to a pipe communicating with the downstream reach several meters downstream of the gate in a tranquilized zone. This offset is given in meters (SIC then searches for the water elevation value in the nearest calculation section).

Chapter 53

Description of the offtakes

Access to the description of an offtake by double-clicking a node in the graph or in the tree view.

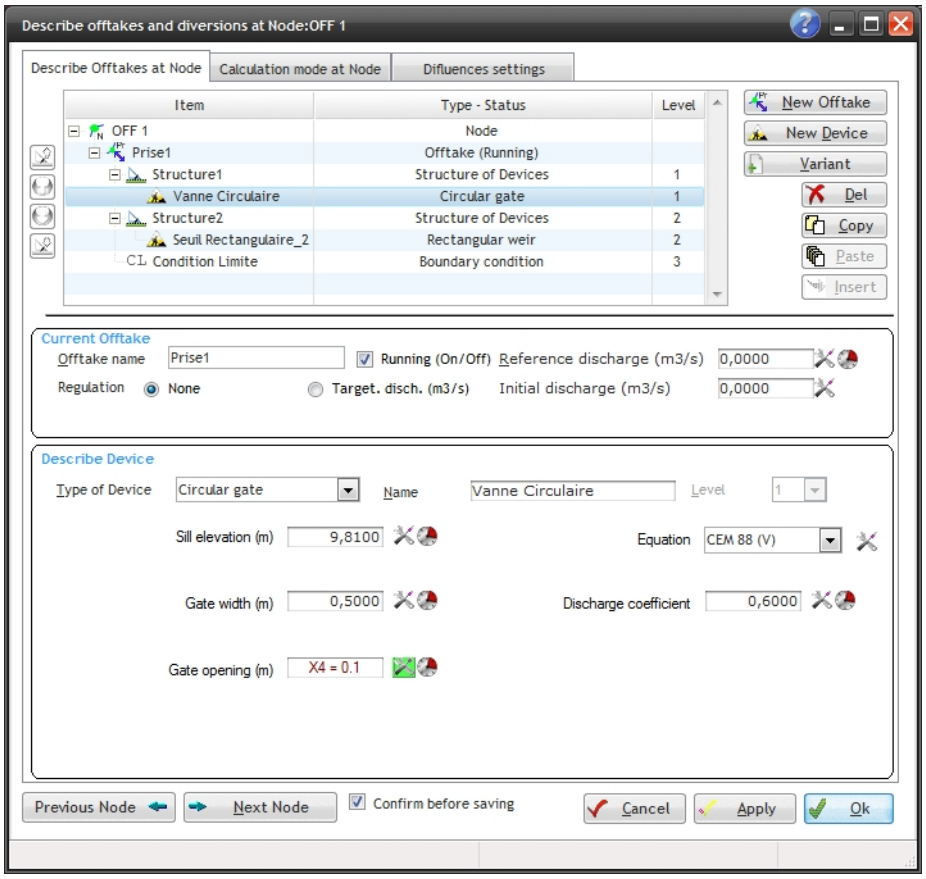


Figure 53.1: descriptionprise

From SIC 5.20, intermediate nodes can have multiple offtakes.

53.1 Characteristics of an offtake

Each offtake has the following characteristics: - A downstream boundary condition which can be an imposed flow $Q(t)$, a fixed elevation $Z(t)$, a rating curve $Q(Z)$ (or rather $Z(Q)$), or a law function $Q = Z^{\alpha}$ (or rather $Q(t) = Q_{ref}[(Z(t) - Z_0)/(Z_{ref} - Z_0)]^{\alpha}$ to be more precise)

- Optionally a first level of structure with one or more devices including one may be regulated in order to obtain a discharge goal.
- Possibly a second level of structure downstream of the first containing one or more fixed devices.

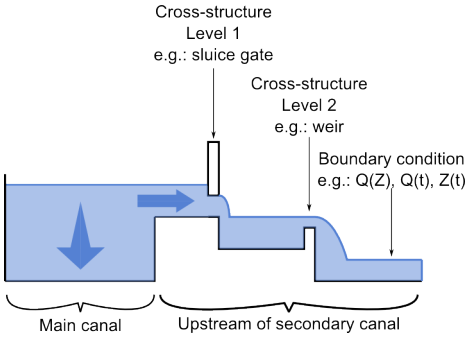


Figure 53.2: doc114

53.2 Regulate an offtake for obtaining a targeted discharge

In steady (FLUVIA) and unsteady (SIRENE) calculation, it is possible to calculate a characteristic of a device (opening of a gate, sill of a weir…) situated in the first level of structure to obtain a defined discharge (called “targeted discharge” or maybe sometimes also “objective discharge”) at the offtake.

53.3 Initial discharge

The initial discharges serves two purposes : - The initial discharge at the offtakes is used to calculate the distribution of the discharge in the reaches of the network for the steady calculation (See Verification of initial rates). - When an offtake is not in regulation “targeted discharge” mode, the discharge calculation is performed by dichotomy and requires an initial discharge to start the calculation.

53.4 Reference discharge

The reference discharge is not directly used in hydraulic calculations. In EdiSIC result mode, it is used to calculate indices of performance by comparing the distribution (calculated discharge) and the request (reference discharge).

The reference discharge can also be used as an input by the regulation modules.

53.5 Sign convention for discharge

Since the discharges can enter or leave the hydraulic system (at nodes or at sections through lateral discharges or seepage), a sign convention had to be choosen. All discharges entering into the hydraulic system are positive. All discharges leaving the hydraulic system are negative. This is true at all nodes

and cross sections for lateral discharges at the data input stage and also at the result display stage. For cross structures (gates, weirs, etc) the sign convention is positive for flows from upstream to downstream and negative for downstream to upstream flow directions.

Chapter 54

Upstream boundary condition of a model

One can directly list and edit the Upstream Nodes (in view of the network topology) by going in the Tools Menu > Upstream Nodes.

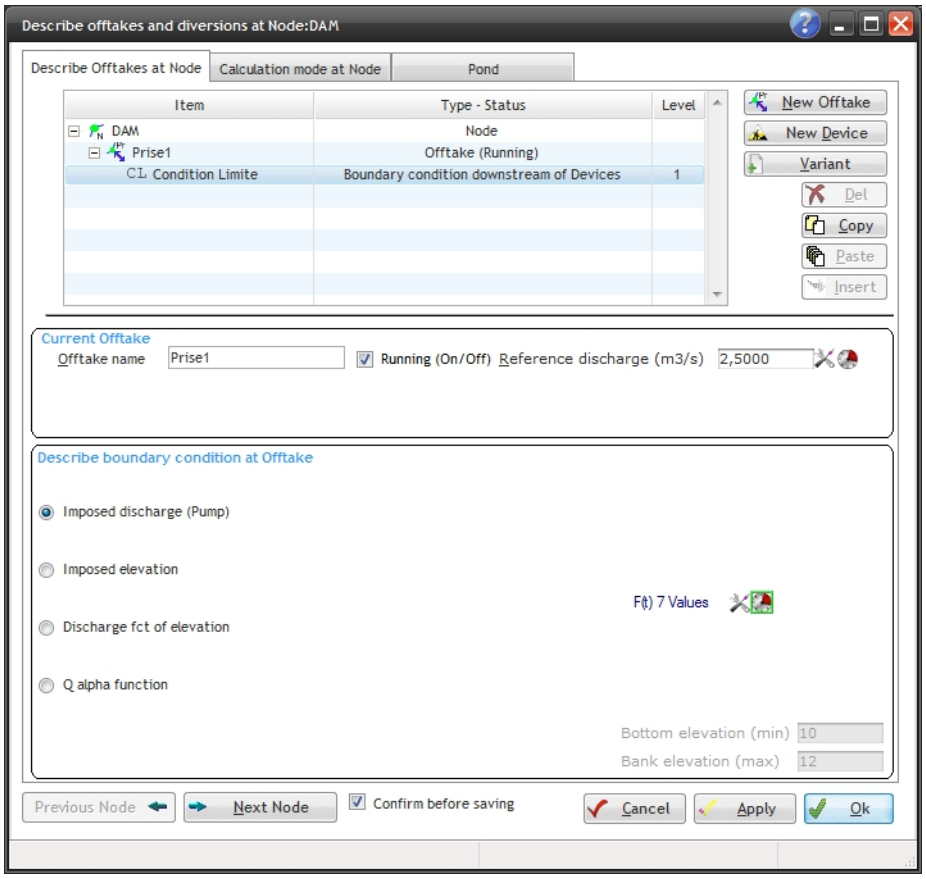


Figure 54.1: descriptionpriseconditionamont

The Upstream-Downstream direction defined in Geometry Mode determines the steady flow calculation. In unsteady flow, the topological direction of the reaches doesn't matter.

Upstream Boundary Conditions will be the ones available to bring flow into the network. These are the Boundary Conditions types: - "imposed discharge (pump)" ($Q(t)$) with a positive or nul discharge - "fixed elevation" ($Z(t)$) with an elevation requiring a positive discharge considering topographic and hydraulic conditions of the network.

The Upstream Boundary Condition of type "fixed elevation" ($Z(t)$) is available in steady flow calculation since version 5.27 of SIC. To the best of our knowledge, it is the only software of this type having this possibility.

N.B. : It is quite possible to mix offtakes with Upstream Boundary Conditions and offtakes with Downstream Boundary Conditions in an Upstream Node. This

only necessary condition in steady flow will be that the balance of incoming and outgoing flows should be positive.

Chapter 55

Downstream boundary condition of the model

Entering the downstream boundary condition(s) of the network can be done in different ways, by: - Menu Tools > Downstream node - Double-clicking on the downstream node on the graph or on the tree view

A hydraulic model necessarily has boundary conditions upstream and downstream, which indicate how the network that is modeled is connected to the rest of the world at its borders. In the absence of such explicit precisions (ie if we do not enter "offtake" at these nodes), SIC will assume that the boundary condition(s) at the upstream nodes (as well as at the nodes possible intermediates) is $Q(t) = 0$, and that the downstream boundary condition(s) is the critical depth $Y_c(Q)$, which is a function of the rate Q. Why the critical depth $Y_c(Q)$ and not the normal depth $Y_n(Q)$? Especially since the critical depth is more difficult to manage in a hydraulic model, especially under transient conditions. In fact, by definition, the critical depth corresponds to a Froude number $F_r = 1$, with very strong backwater slopes, which SIC can process in a very good way in steady state and with sometimes more difficulties in unsteady state, or in any case with simplifications (ex: partial or total suppression of the terms of inertia, option Preissmann-Sart, etc.). The reason is simple, it is that the uniform regime, which would be another good candidate, does not always exist, contrary to the critical regime (in the event of a null slope or of slope). In spite of these choices of upstream and downstream limit conditions by default, it is always better to define them explicitly, to better control what the model does.

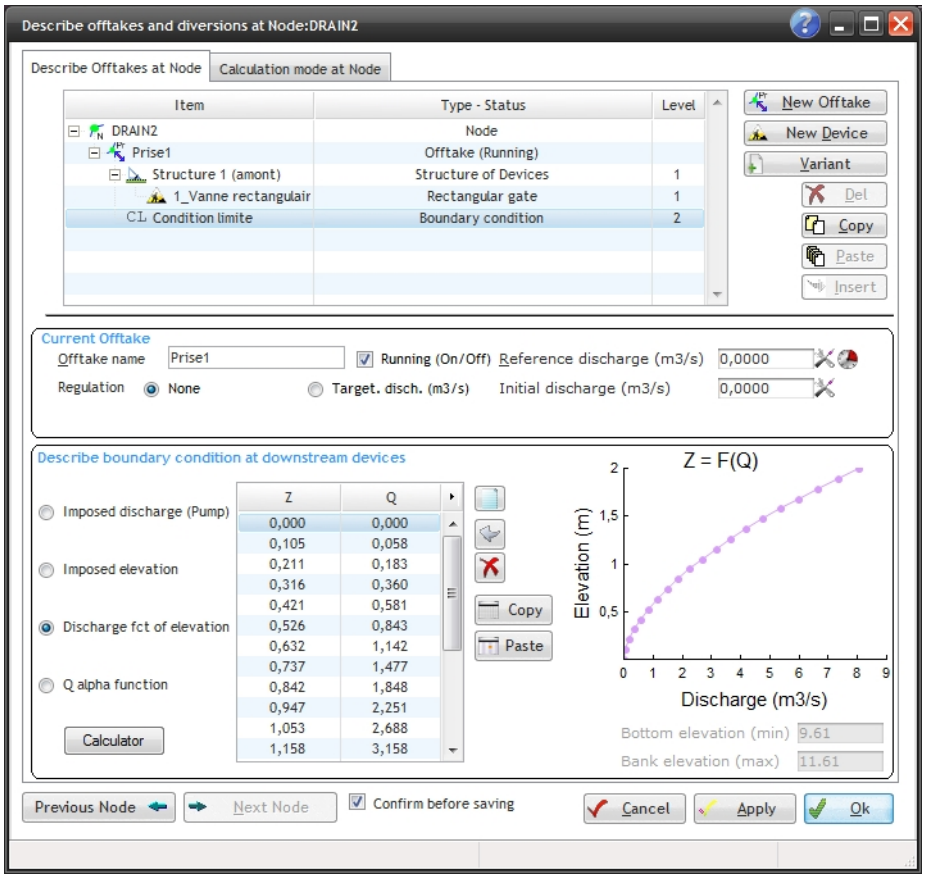


Figure 55.1: descriptionpriseconditionav

If the network is meshed, there may be more than one node of this type (downstream node). In addition, it is also possible to enter one or more "offtake" at each node, whether upstream, intermediate or downstream, and each of these offtake must contain a downstream condition. The details and information given here are therefore applicable to other cases than to the conditions at the downstream node stricto-sensu.

The calculation of the backwater curve in steady state requires fixing or being able to calculate the water elevation at the downstream nodes of the model. This elevation is set by the user (fixed elevation $Z(t)$), or is calculated by a relationship between discharge and elevation ($Z(Q)$ or $Q = Z^{\alpha}$).

For unsteady calculation, all boundary conditions are possible ($Z(t)$, $Q(t)$, $Q(Z)$).

If you want to get a $Q(Z)$ (or rather $Z(Q)$) downstream law of type uniform flow or critically flow, you can use the calculator available by pressing the button

“Calculator”.

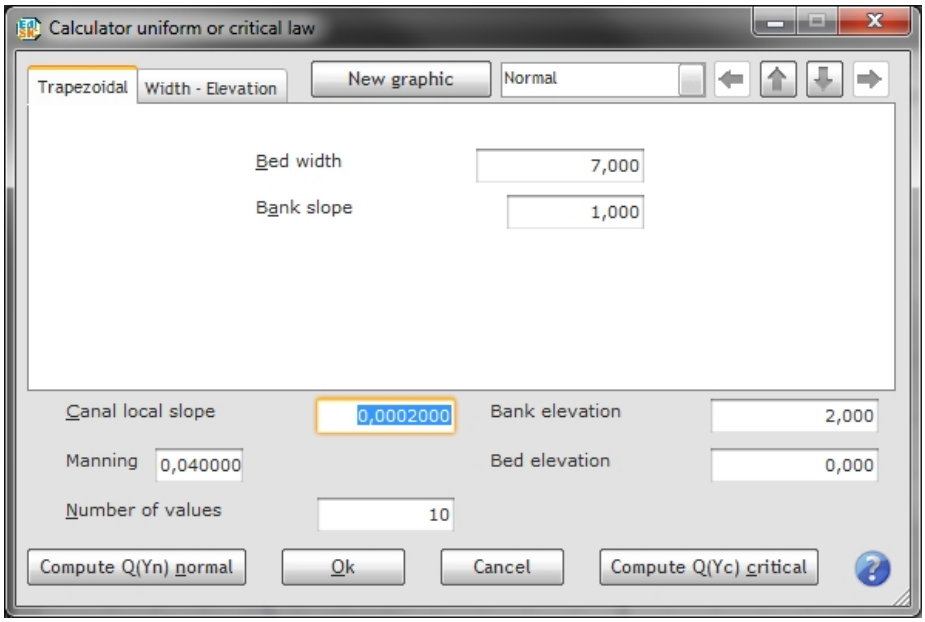


Figure 55.2: calculettetrapez

This calculator allows to recover the local cross section (upstream of the node) or any other section by moving upstream or downstream along the system, using the arrows on the upper part of the windows, and to generate the corresponding law.

If several reaches reach this node, a question will be asked to choose the desired upstream reach, to retrieve the section and its geometric description. The left and right arrows in the upper part of the window make it possible to change this reach if necessary.

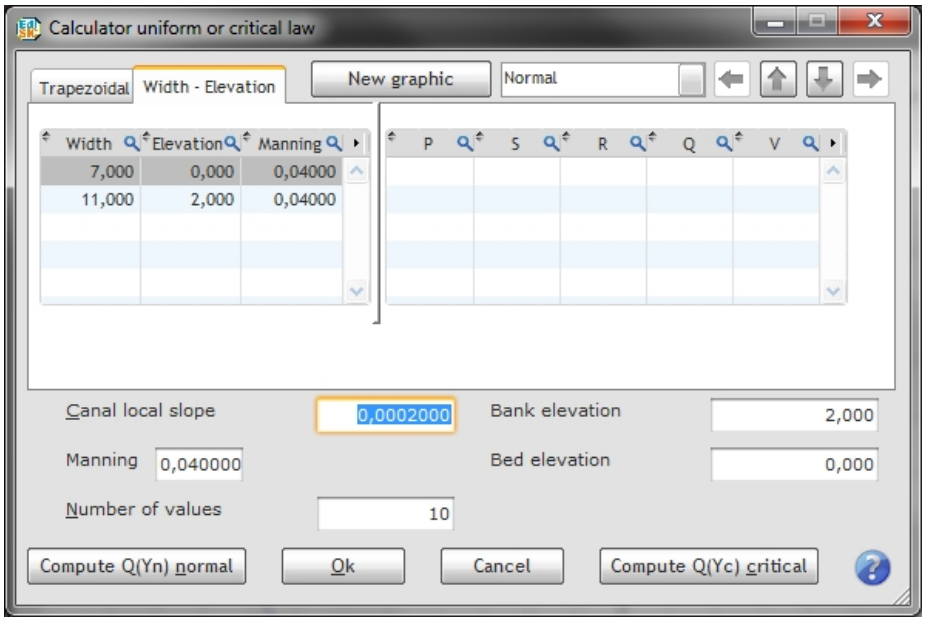


Figure 55.3: calculettelargeurcote

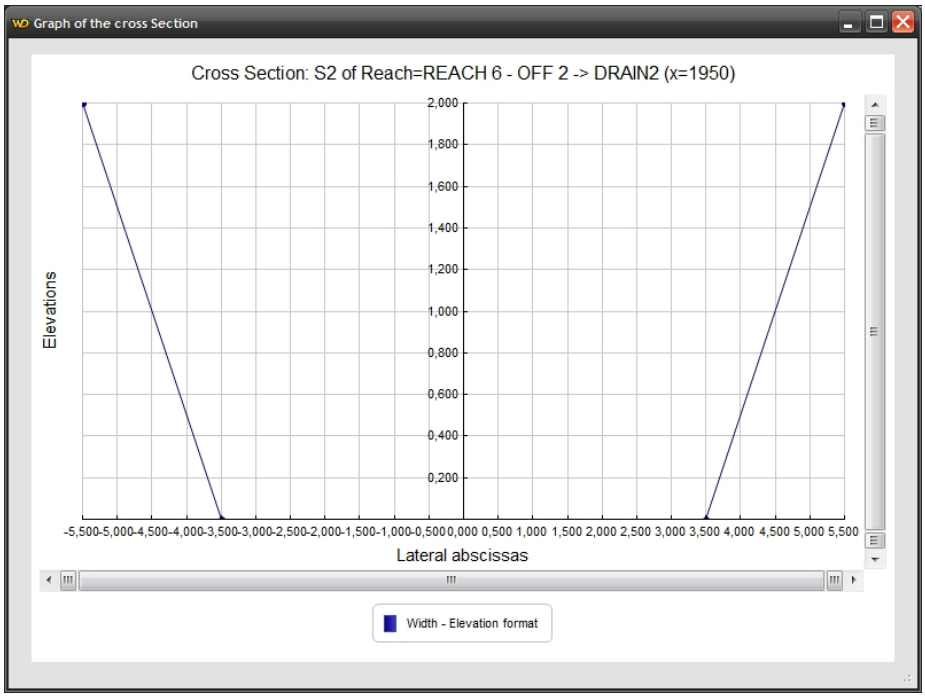


Figure 55.4: dessinsection

The calculator makes it possible to calculate a law $Q(Z)$ corresponding to the uniform regime (Y_n), as well as to the critical regime (Y_c).

For the uniform regime the equation used is the classical Manning-Strickler equation:

$Q_n = K_s S R^{2/3} \sqrt{I}$, where K_s is the Strickler coefficient, and I is the bottom slope, S the wet surface, R the hydraulic radius ($R = \frac{S}{P}$, where P is the wetted perimeter).

For the critical regime, the following equation is used:

$Q_c = \sqrt{gS^3/L}$, where L is the top width, $g = 9.81 ms^{-2}$.

By clicking on the button "Normal $Q(Y_n)$ Calculation" or "Critical $Q(Y_c)$ Calculation", EdiSIC will calculate the corresponding Q values for different elevations: - firstly for the elevations corresponding to the description of the section in width - elevation format. These are the values that appear on the right pane in the window. Intermediate or auxiliary variables P, S, R and V are calculated. - in order to have a finer definition of these laws, one can specify the number of calculated values, as well as the min (bottom elevation) and max (bank elevation). These min and max dimensions are pre-filled (at the bed and bank elevations of the studied section) but they can be modified. The number of default values

entered is 20 but can also be changed. The automatically generated elevations, for which the flows will be calculated, are a priori equidistributed. But in order to preserve the points of the width-elevation description, these particular points replace the points closest to the initial equi-distributed generation.

This calculator also handles sections with a medium bed, if any. In this case, the overflow elevation between the minor bed and the medium bed must be entered (it is pre-filled, with the correct value, but can possibly be modified). As well as 2 coefficients of friction, one for the minor bed, the other for the medium bed. The exchange variables between the minor bed and the medium bed are also calculated and graphed.

If the selected section has no medium bed, it is still able to do calculations in this configuration by increasing the maximum elevation. In this case, the old maximum elevation becomes the minor-medium bed elevation, and a new Strickler coefficient can be entered for that medium bed.

Chapter 56

Management of parameterized variables

Some variables can be "parameterized". This means that instead of assigning them a fixed value, they are assigned a variable name (ex .: X1, X2, etc.), and the value taken by this variable will be read in a .par file containing these variables and their values. The generation of names of variables (X1, X2, etc.) takes place automatically, with the letter X, and a number which is incremented automatically.

This file has a format similar to:

```
//Nd : Lezlir > Pr : Prise1 > CL Débit
X1  4.12
//Nd : Deversoir Latéral > Pr : Structure1 > Ouv : Seuil évacuateur de crues Cote de seuil (m) Va
X2  8.8
```

This file is created and saved automatically when saving the xml file. But it is also possible not to rewrite this file at each record by unchecking the appropriate box (bottom left on the screen below).

The advantage of this option is to generate or modify this .par file manually or by an external program (Notepad, Matlab, Scilab, R, etc.) and thus to control simulations with easily modified variables. Possible uses are for example for automatic calculation sequences for sensitivity analysis, optimization, calibration, etc.

The "Parameterized Variable" menu in the Hydraulic mode allows a synthetic view of these variables and their default values. At the time of execution of the hydraulic calculation of the scenario (and possibly the corresponding variant), if the .par file is found and if the value of the variable is found, then that value will be used. Otherwise it will be the default.

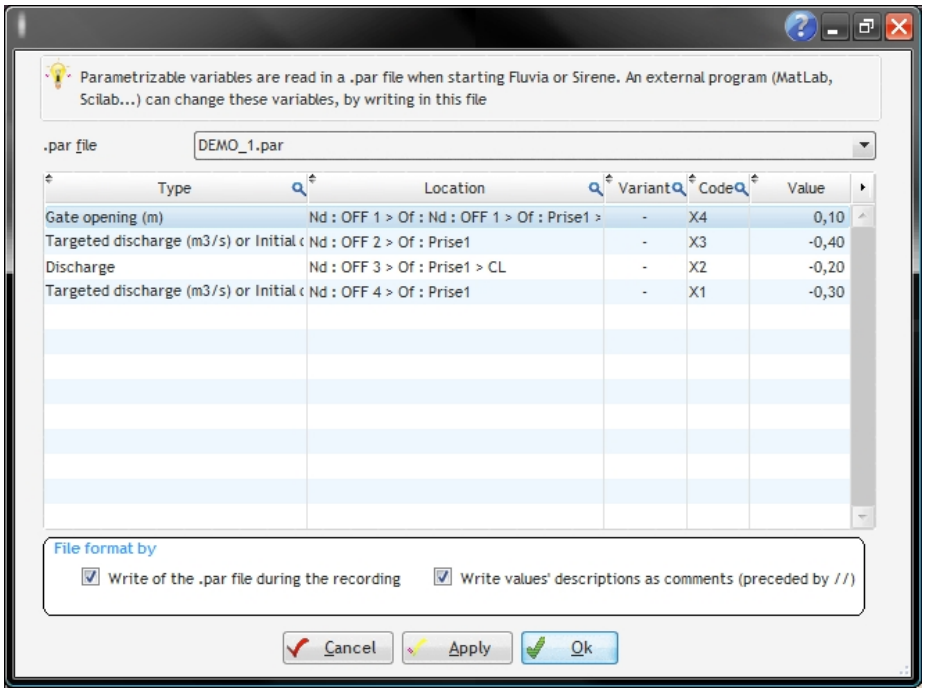


Figure 56.1: variableparametrees

On the same window you can see the name of the .par file in which these variables will be written and required. This name is automatically created from the name of the xml data file (file name when first creating the file) to which is added the code "_" followed by the scenario number. This file management is indeed made at the scenario. In the current version of SIC that name can not be changed (unless you go edit it directly in the xml file) and is stored in the xml file and kept in memory, even if all the variables are parameterized rebates normal variables. Thus, if the variables are switched again in parameterized variables the same name will be used again. It is therefore possible to rename the xml file, without having to change the .par file used.

Warning, if .par file is created or modified from an external program, it is important to maintain its format. Especially in the current version of Edisic, the separator between the variable (ex .: X1) and its value is a tab. However, comment lines (beginning with //) are optional.

Chapter 57

Format of the .par file

A file of parameterized variables is created per scenario and is automatically updated when the scenario is saved in EdiSic. The file is named `[XML Project Name] _ [Scenario #].par`.

In the above file name, we have chosen to keep the name of the initial project [XML Project Name] in the name, even if the name of the project is then changed, either by the command “Save as” from EdiSic, or by hand under Windows or under Dos. If you have a routine generating this file for automatic executions, you will not have to modify it even if the name of the project file is then modified.

It is a text file containing rows and columns. The accepted separators for columns are spaces, tab, semicolon, and comma. The format is as follows: - Lines not starting with “L” or “X” are ignored. By convention, the comment lines begin with the double symbol “//”. - The lines beginning with X1, X2, X3, … Xn are used to define the value of the parameterized variable, - Lines beginning with L1, L2, L3, … Ln are used to define the location of a parameterized variable in the network.

EdiSic is used to define the location of the variables parameterized from the input interfaces of the hydraulic mode (see the management of parameterized variables). But it is also possible to define new variables parameterized directly in the .par file by defining their location in the network using lines starting with L.

57.1 Syntax for lines starting with X

For a fixed value, the syntax will be: `X[n] [Value]` Example: `X1 0.25`

This means that when running the simulation of the corresponding scenario and/or variant, either in Steady (Fluvia) or Unsteady (Sirene) flow, the parameter

or variable named X1 will be given the value 0.25 instead of the value indicated in the EdiSic interface and stored in the xml file.

For a time-varying value (time-dependent law), the syntax will be:

```
X[n]    LOI    [E or R]
X[n]    [Time in seconds]    [Value]
X[n]    [Time in seconds]    [Value]
...
```

The [E or R] parameter allows you to choose between step mode [E] or ramp mode [R].

Example for a water rotation schedule evolving at 0, 1 and 2 hours:

```
// Function law of time in "E" mode (E stands for Echelon meaning step in French)
X1  LOI E
X1  0   -0.1
X1  3600    -0.2
X1  7200    -0.1
```

57.2 Syntax for lines starting with L

Lines beginning with L contain a set of columns with pairs [Object Type] = [Value]. The meaning of the object types is as follows: - BF: n° of the Reach - ND: n° of the Node - PR: n° of the Offtake - ST: n° of the Structure - OUV: n° of the Device - SN: n° of the Section - PBF: n° of the Reach zone (for Manning-Strickler and Infiltration) - CAR: Variable to be parameterized on the object (see list of variables below)



| The number of Reaches, Nodes and Sections is

visible in the tooltip that appears when you mouse over the various network objects in the EdiSic tree view. The .par file written by EdiSic has the comment line "L" for each variable defined as an example of syntax|

It is possible to define several locations for a single variable. A no-consequence warning will be displayed during the simulation and the same value will be applied to the different locations requested.

Here are different localization examples for different network objects:

Type of object | Syntax of the line |
Discharge of offtake n°1 of node n°1 | `L1 ND=1 PR=1 CAR=Q` |
Opening of the gate of the first structure level of offtake n°1 of node n°2 | `L2 ND=2 PR=1 ST=1 OUV=1 CAR=W` |
Discharge coefficient of the gate of structure n°1 of the cross-structure located in section n°4 of reach n°1 | `L3 BF=1 SN=4 OUV=1 CAR=CoefQR` |
Strickler of the first zone of Strickler of reach n°1 | `L4 BF=1 PBF=1 CAR=KMin` |

At the EdiSic interfaces the frictions are indicated in Strickler in French and Manning in English and Spanish. But internally, these frictions are written in the .xml files as well as in the .par files in Strickler. If you are Anglo-Saxon and used to manipulating Manning, you will have to think of converting them (Ks = 1/n) into Strickler if you write them in a .par file.

The list of variables available by the CAR parameter is as follows (note that the upper and lower case are important):

Variable | *Signification* | *Possible Localization* |
Q | Discharge or Targeted discharge | Offtake with a boundary condition in discharge or with a targeted discharge controlling a device (`ND=... PR=... CAR=Q`) |
Z | Water elevation | Offtake with a boundary condition in elevation (`ND=... PR=... CAR=Z1`) |
KMin | Strickler of the minor bed | Zone of a reach (`BF=... PBF=... CAR=KMin`) |
KMoy | Strickler of the medium bed | Zone of a reach (`BF=... PBF=... CAR=KMoy`) |
Inf | Seepage | Zone of a reach (`BF=... PBF=... CAR=Inf`) |

It will also be possible to control parameters of cross-devices (`BF=... SN=... OUV=...`) and offtake devices (`ND=... PR=... ST=... OUV=...`).

The available variables are:

Variable | *Signification* | *Possible type of device* |
CoteRadier | Sill elevation | All devices |
Largeur | Width | All gates and weirs except circular gates and tubes |
Ouverture | Gate opening | All gates except GEC-Alsthom |
CoefQR | Discharge coefficient for rectangular device | All gates and weir equations except Goussard equation for GEC-Alsthom gates |

SurverseHauteur | Height of the gate | All gates |
CoefQSurverse | Discharge coefficient for overtopping flow | All gates with a possible overtopping over the gate |
TanAl | Side slope | Trapezoidal gates and weirs, GEC-Alsthom gates |
CoefQT | Discharge coefficient for trapezium device | Trapezoidal gates and weirs and GEC-Alsthom gates using CEM02 equation |
CoteAxe | Axis elevation | GEC-Alsthom gates |
Rayon | Radius of the gate | GEC-Alsthom gates |
D | Caracteristic "D" of the Goussard equations | AMIL gates |
JMax | Maximum head-loss | GEC-Alsthom gates |
S1S2 | Ratio of the tank surfaces for filtering | AVIS, AVIO, Mixtes gates |
Decal | Shift of the tank inlet location | AVIS, AVIO, Mixtes gates |
Decrement | Decrement | AMIL, AVIS, AVIO gates |
CoteAmont | Maximum upstream elevation | AMIL gates |

57.3 Example of .par file

```
// Localization defined by EdiSic
//Nd : OFF 3 > Pr : Prise1 > CL Débit
// L1   Nd=3    Pr=1    Car=Q
// Law function of time in mode "E" Echelon (Step)
X1  LOI E
X1  0   -0.1
X1  3600    -0.2
X1  7200    -0.1
// Added by the user of Nd : DAM > Pr : Offtake 1 > Discharge boundary condition
L2  Nd=1    pr=1    car=Q
// Law function of time in mode "R" Rampe (Ramp)
X2  LOI R
X2  0   2.5
X2  3600    3
X2  7200    2.5
// Added by the user of Nd 3 and 5 Offtake 1 with a constant discharge of 0.4 m3/s
L3  Nd=3    Pr=1    Car=Q
L3  Nd=5    Pr=1    Car=Q
X3  -0.4
```

Chapter 58

The project explorer

The Project Explorer is the window to navigate between different modes of EdiSic (geometry, hydraulic and result) but also to manage the hydraulic scenarios and variants to be discussed in this part (see The three modes: geometry , hydraulic and result).

To access it: - Menu Project> Project Explorer - The button 

Note: it is also possible to quickly navigate between different modes (Geometry - Hydraulics - Result) using the arrows to the right and left of the button 

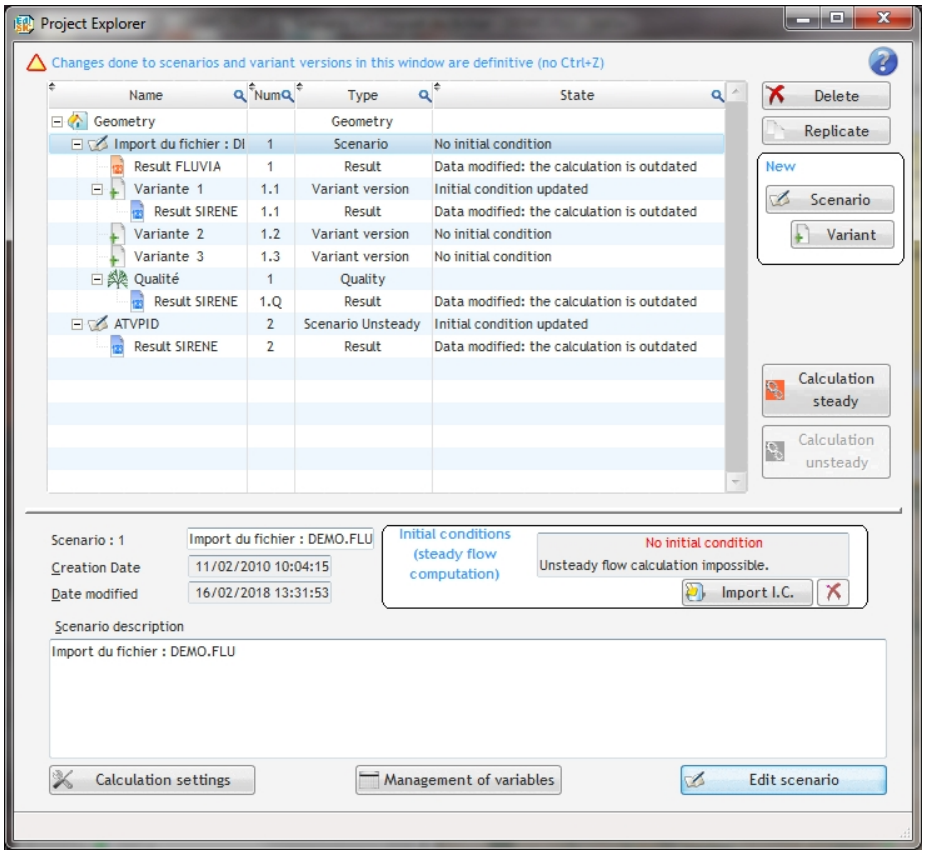


Figure 58.1: explorateurprojet

In this window, you can create new hydraulic scenarios, duplicate or delete existing scenarios.

To edit a scenario: - Select the scenario to edit then click on the button “Edit Scenario” - Double-click on the scenario to edit

Chapter 59

Definition of the roughness (Manning)

In English and Spanish interfaces, the roughness coefficients are input in terms of Manning. In interfaces in French the roughness coefficients are input in terms of Strickler which is the inverse of Manning: $n = \frac{1}{K_S}$

To access the input interface of the roughness coefficients: - Tools menu> Setting Manning - Button 

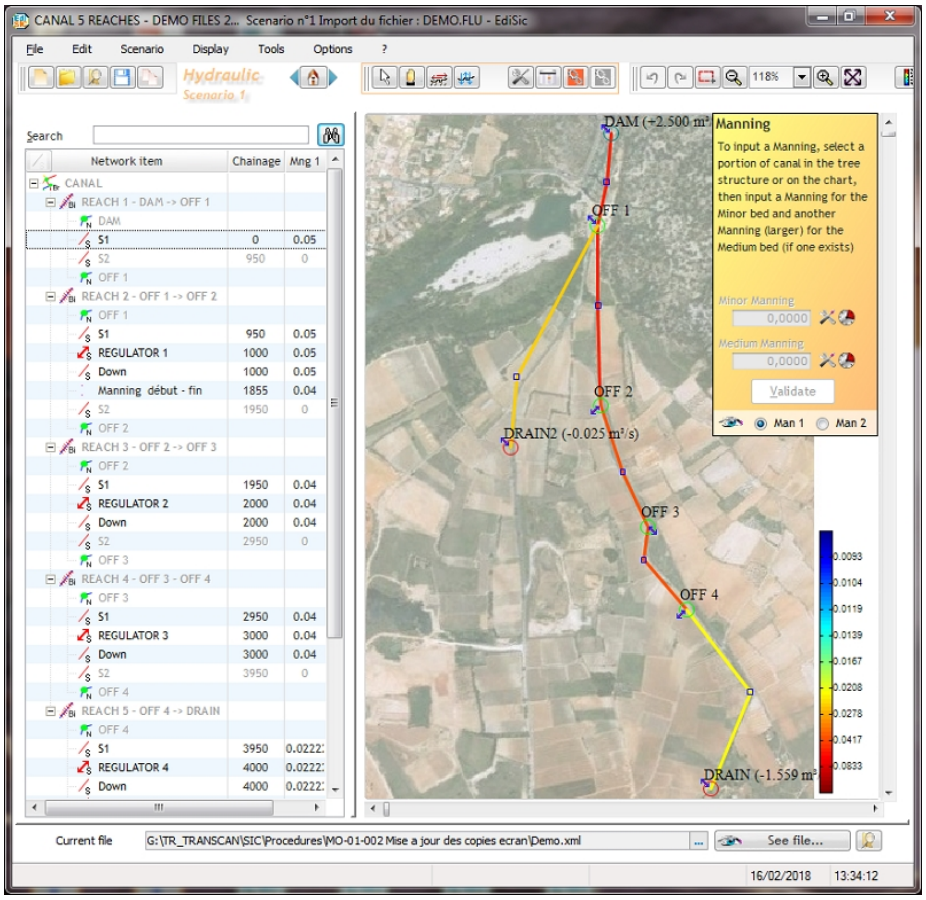
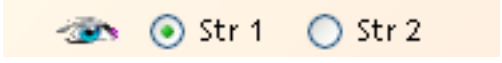


Figure 59.1: premiere_flustrickler

The sections are selected directly on the graph or the tree view. It is also possible to refine the selection and capture coefficients between two abscissas even though there is no data sections to these abscissae [[The coefficients will be applied to the closest calculated sections]]. The abscissa and the coefficients are entered in the right side of the screen: - Manning min for the minor bed - Manning med for the medium bed if exists

The coefficient is represented on the graph by a color scale: - The 

radio buttons let you choose the Manning plotted (minor or medium bed) - In the 'view' toolbar , the button  allow to display or not the legend of the coloring - In the menu "Display> Change the color chart," you can also change the settings of the coloration

59.1 Usual values of the Strickler coefficients n[[Extract from "Hydraulique pour le génie rural", J.-P. Baume, G. Belaud, P.-Y. Vion (Avril 2006)]]

Type of stream | {{Usual values of n (s/m1/3)}} |
Concrete channel, very smooth | 0.010 - 0.013 |
Concrete channel, average state | 0.013 - 0.020 |
Earthen canal | 0.020 - 0.033 |
Regular stream, well maintained | 0.020 - 0.025 |
Ordinary stream | 0.025 - 0.033 |
Stream with jams | 0.033 - 0.050 |

Chapter 60

Definition of lateral flow (infiltration)

To access the input interface of infiltration: - Tools menu> Setting infiltration - Button 

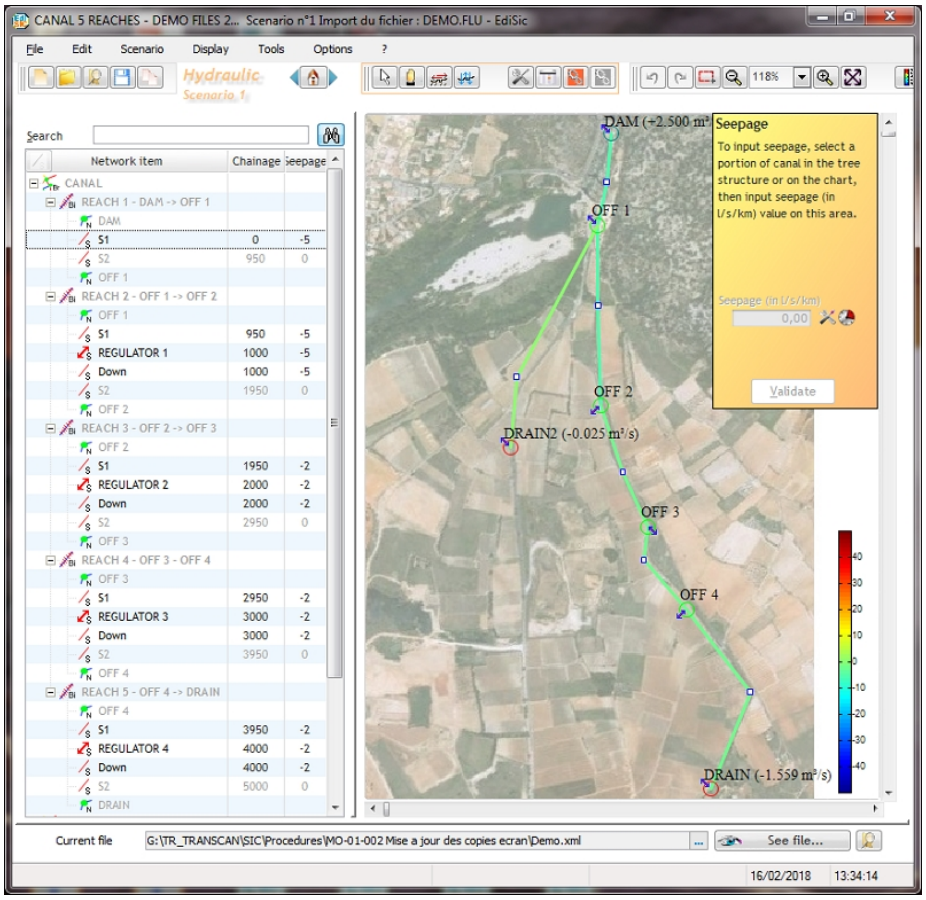


Figure 60.1: premiere_fluinfiltration

Entries are made in the same way that the input of roughness coefficients.

This screen displays the value of the current infiltration (negative) or lateral flow of contributions (positive). Flow is expressed in l / s / km.

Chapter 61

Definition of the calculation mode at nodes

The resolution of the calculation of the hydraulic variables at the node can be done in two ways:

- equality of the elevations (default mode)
- equality of the hydraulic heads

In the case of equality of heads, it is possible to define a head loss at the entrance of the node (for upstreams reaches arriving at this node) and at the exit of the node (for the diversion reaches downstream). It is possible to have for the same node reaches arriving at or leaving this node with different modes (elevation for some and head for others). In this case of a head loss, a Borda type equation (proportional to the speed squared) is used.

Remark: In the case where the flow at a node is in supercritical regime on one or more of the reaches arriving or leaving this node, the calculation mode will be automatically switched to "equality of head" mode, even if the "equality of elevations" mode had been chosen here. This is indeed more physically realistic. For calculations in torrential regime to be authorized, the corresponding option must be activated in the permanent calculation parameters.

Chapter 62

Initialisation of the calculation at diffluences

On a network with diffluences (classical on irrigation canals whereas rare for rivers), the calculation of the steady-state water line performs iterations to distribute the flow arriving from upstream into each reach downstream of the diffluence point.

By default, the distribution is initialized from the relative surface of reaches located downstream from the diffluence. When this default distribution would cause negative initial flow rates in some branches of the network, the user can change these settings by double-clicking the node causing the problem and change the distribution of the flow between the diversion reaches leaving from this node.

Chapter 63

The boundary conditions

The offtakes at nodes must have a boundary condition. The available boundary conditions are: - Flow rate (Pump): Q(t) - Fixed elevation: Z(t) - Rating curve function (tabulated): Q(Z) (or rather Z(Q)) - Function Q power alpha: $Q(t) = Q_{ref}[(Z(t) - Z_0)/(Z_{ref} - Z_0)]^{\alpha}$

The upstream and downstream nodes are also considered as locations where "offtakes" must be defined in order to provide the boundary conditions for the hydraulic calculation algorithms. The terminology may seem abusive but it is necessary to choose a generic term. This is defensible in the sense that it is the points that allow injecting or withdrawing from the flow in the hydraulic network, at the upstream and downstream nodes as well as at the intermediate nodes.

The conditions at the upstream and downstream nodes can be chosen from this list. The details given for downstream nodes are also valid for upstream and intermediate nodes.

All combinations are possible in transient mode.

Some combinations are impossible in steady state: Q(t) upstream and downstream. Indeed in this case either there is no solution, or there is an infinity of solutions. The most traditional configuration is Q(t) upstream and Q(Z) downstream. Other options that do not explicitly specify upstream flow will involve iterations of the Fluvia calculation program to find the flow that satisfies the upstream conditions chosen.

Reminder: Incoming or outgoing flows in SIC follow a sign convention which is that incoming flows are positive and outgoing flows are negative. This is generally valid for external boundary conditions (upstream and downstream), internal (intakes-offtakes) and seepage/infiltrations or contributions distributed along the sections. But, to avoid the classic errors that we often encountered with users, we make exceptions to this convention when there is no ambiguity. For example,

a downstream boundary condition of type Q(Z) can only be outgoing, and we therefore expect positive flows at the data entry interfaces whereas according to the convention we should choose them negative. The calculator will choose them positive, but they will indeed be outgoing flows.

Chapter 64

Cross structures description

To enter a cross structure: double-click on the corresponding singular section.

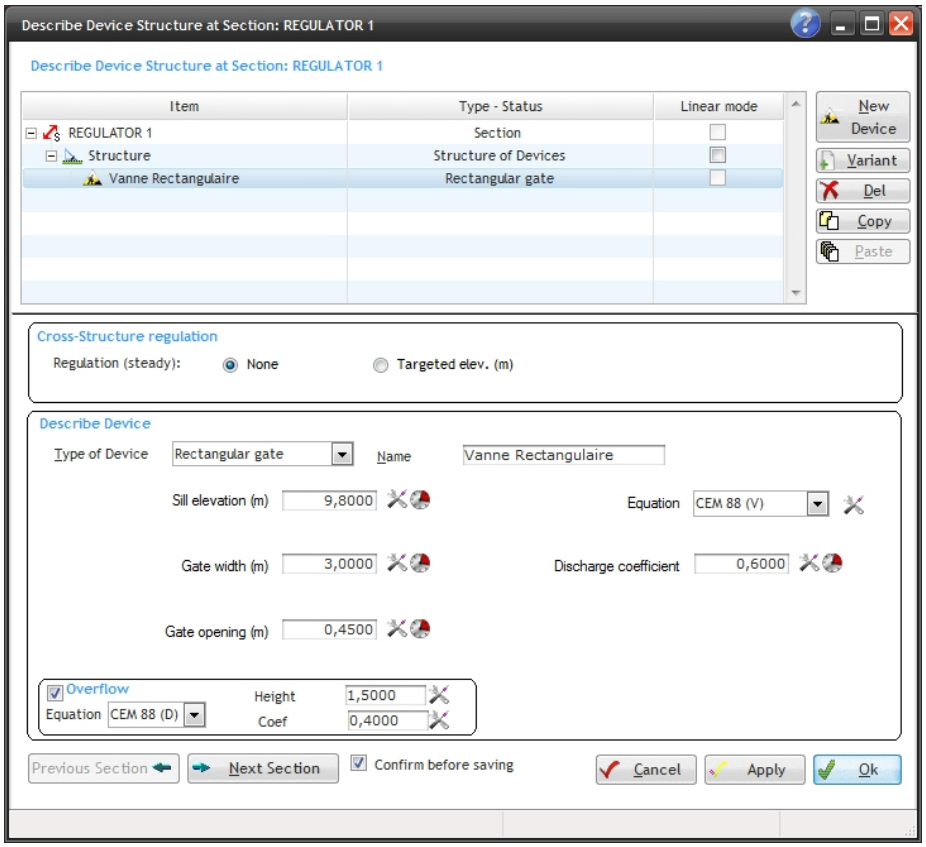


Figure 64.1: descriptionouvragetravers

A cross-structure is composed of one or more devices in parallel.

64.1 Regulation (steady flow only) : target elevation

In steady flow, it is possible to ask FLUVIA to calculate a gate opening or a sill elevation or any other caracteristic of a device to achieve a water elevation upstream the structure across.

In the “Cross-Structure regulation” part, choose “Targeted elevation (m)” and set the elevation that is to be obtained at the upstream of the structure. Then, select the device that is to be regulated and click the calculator  in front of the device feature (valve opens, width, dimension strike ...) that will be regulated to achieve the objective. The selected field shows in green the reference “In Calculation” instead of its value and a box at the bottom right allows you to

enter mini and maxi bounds and an initial value of the calculation.

Chapter 65

Consistency of the hydraulic scenario

For the hydraulic simulation can be started, the hydraulic scenario must meet the rules of consistency.

Consistency tests can be run in several ways: - Automatically when saving the XML file or when validating the windows of description of offtakes or description of cross-structures - Manually by the button “Network Consistency”

The consistency of the hydraulic scenario window lists the fatal errors and warnings found in the script and allows whenever possible to directly edit the element concerned by double clicking on the line of the error.

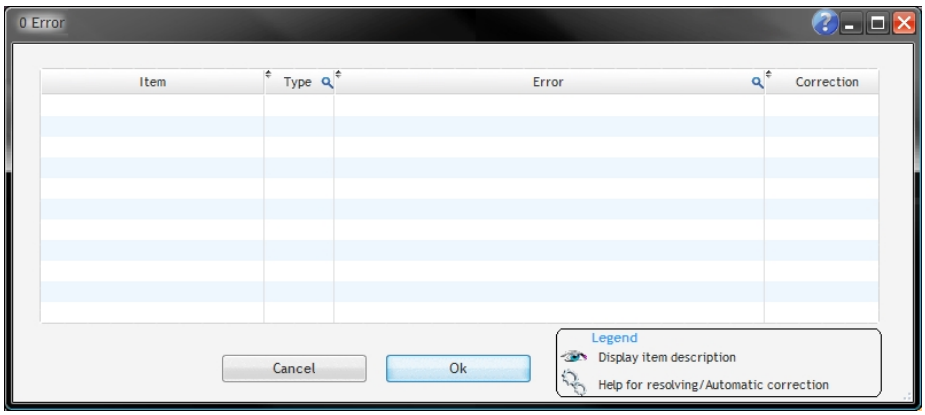


Figure 65.1: erreurs_flu

Among the tests performed, the software checks that initial flows are positive in

all reaches of the network. This condition is necessary to perform a calculation in steady flow, but can be ignored for unsteady flow calculation.

Chapter 66

Checking of initial discharges

A hydraulic calculation in steady flow (Fluvia) is initialized by initial flow in each reach estimated from the boundary conditions (see Sorting mesh reaches to calculate steady - initially Solution). These flows should be positive in all reaches. The software therefore provides a control of the initial flow rates from the network boundary conditions and a display of flow rates upstream and downstream for each reach.

This check is performed automatically during a consistency test.

Displaying the initial flow verification window can be done: - From the network coherence window if there was actually a discharge error on a reach by double clicking on the error. - In the Tools menu > Initial discharges

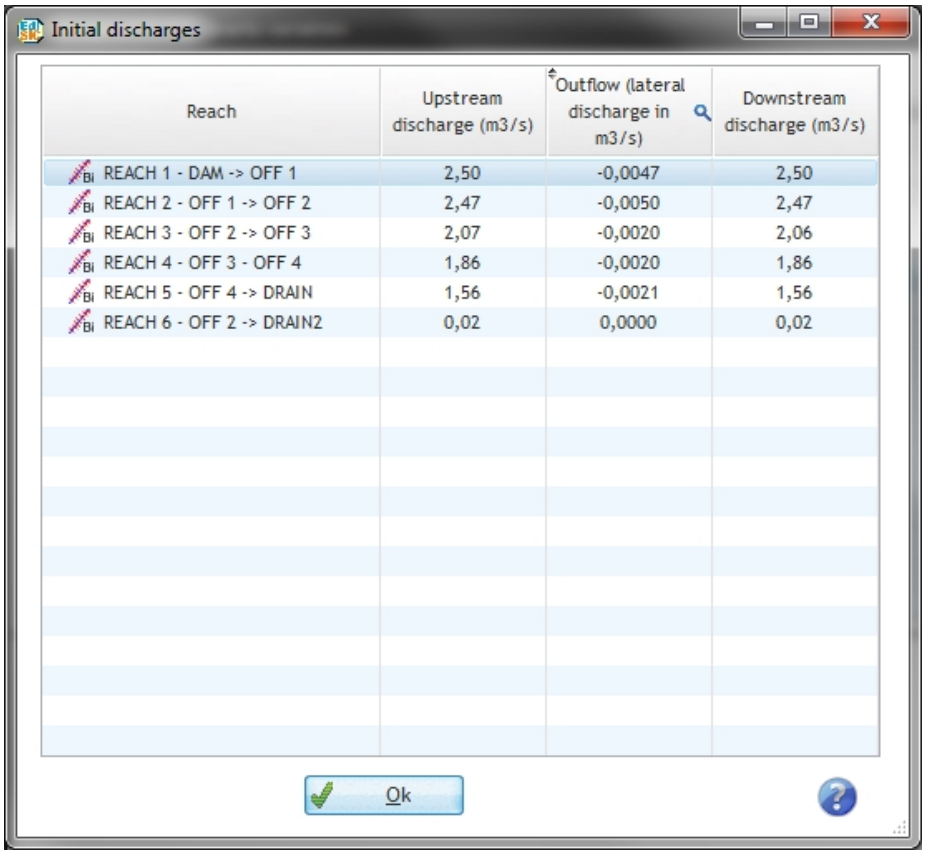


Figure 66.1: debitsinitiaux

Initial discharges are also displayed directly on the network graph to the upstream and downstream nodes and on mouse-over the reaches.

Chapter 67

Time laws

On the boundary conditions (fixed rate or rating), and on the characteristics of the devices (width, threshold elevation, opening ...), it is possible to define fixed values or values that change over time.

Wherever possible, to move from a fixed value to a law function of time, just click on the symbol  at the right of the field to enter.

Then the events are defined in a table by setting the date and the new value at that date.

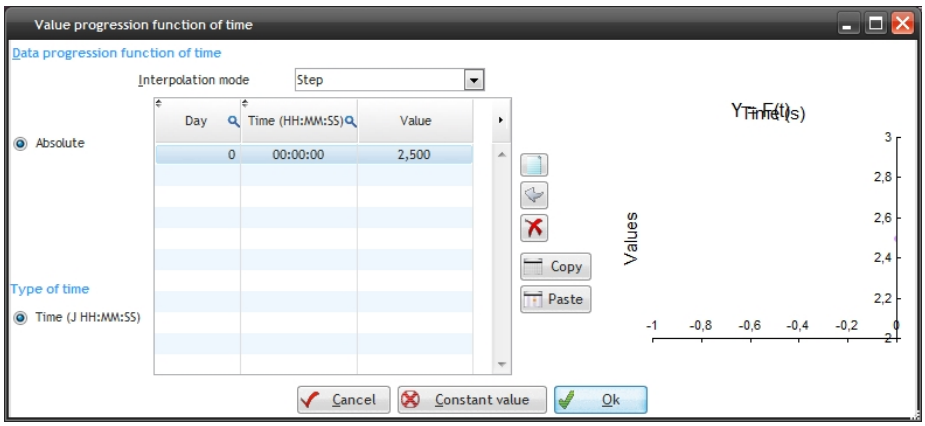


Figure 67.1: donneesft

There are two interpolation methods of data between two events: - the step mode - the ramp mode

The mode level corresponds to a one-time operation at a given time step. The operation is effective one time step after the date of the event.

The ramp mode is a linear evolution between two events. The value calculated at each time step is an interpolation between the previous event and the next event.

It is possible to return to a fixed value not dependent on time by clicking the "Fixed value" button.

Chapter 68

Management of variables (synthetic visualization)

To access the window summarizing all adjustable hydraulic variables for scenarios and variants: - From the Project Explorer, select a scenario or variant and click the button “ Management of variables”. - From the hydraulic mode of editing a script, go to the Tools menu> Management of variables or click on the .

The window provides a table with the list of all the modifiable variables. These variables correspond to all editable items as part of a scenario and a variant (see Scenario and variants: instructions for a description of these variables).

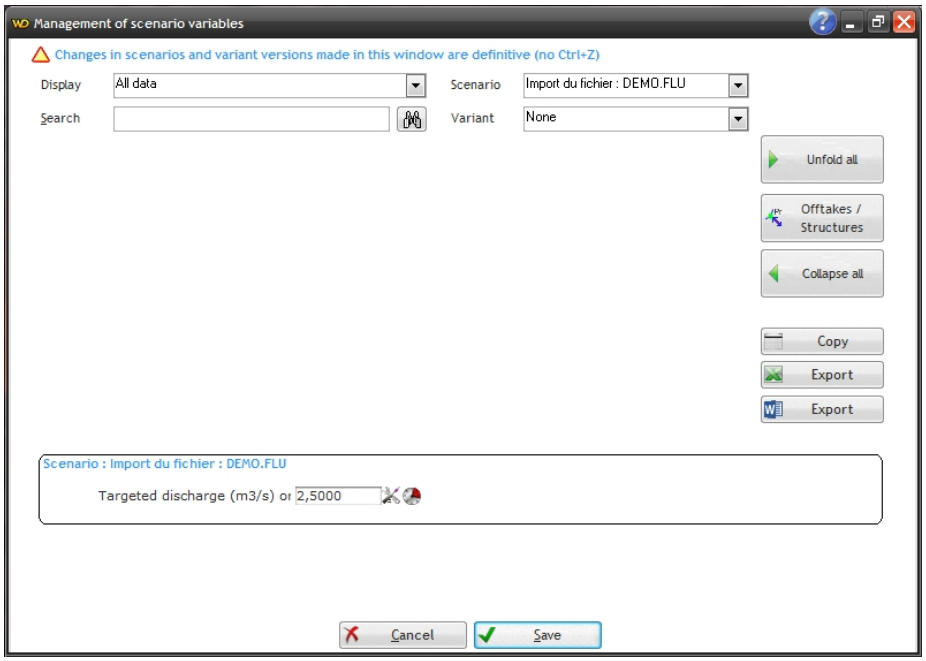


Figure 68.1: gestionvar_1

To edit a variable, select the corresponding line in the table and make the change in the field provided for this purpose at the bottom of the window.

68.1 Entering variant data

To change the data of the variants, one can: - Open the “Management of variables” from the project explorer, having previously selected a variant. - Choose a variant in the “Variant” drop-down list in the window “Management of variables”

The data of the selected variant appear in a second column adjacent to the scenario data.

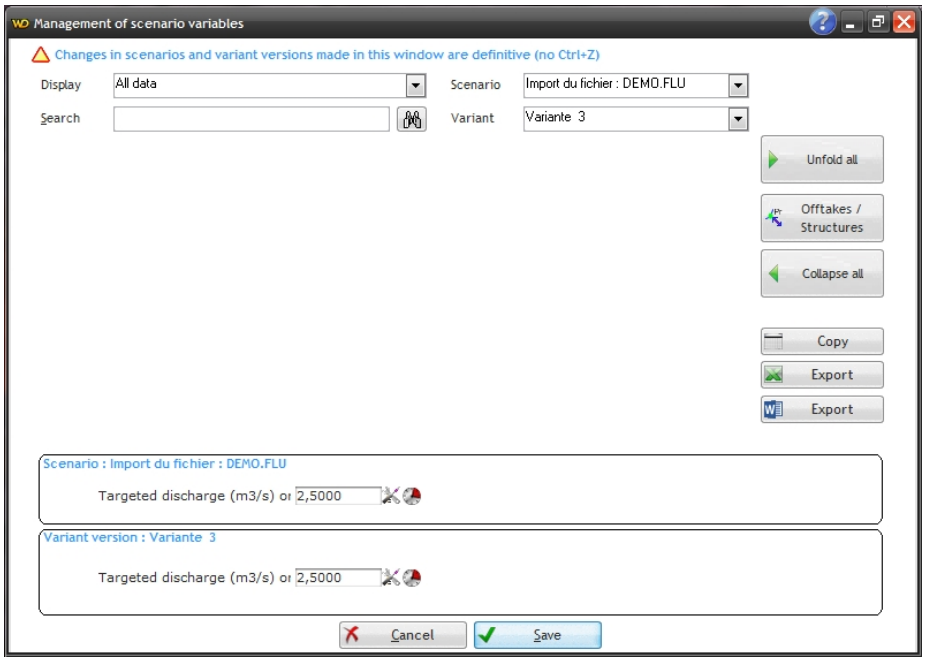


Figure 68.2: gestionvar_2

To enter a variant data, select the relevant data and enter this data using the dedicated field at the bottom of the window. Warning: the first field changes the given scenario, the second that of the variant.

If a variant data has been entered, you can delete it by clicking the button “Delete variant”.

Chapter 69

Scenario and variant: operating instructions

In a scenario, all the data needed to launch a calculation in steady or unsteady flow are defined. Variant (of a scenario) will allow to overwrite some parameters of the scenario while keeping the data as a reference scenario for all non-modified settings.

Changing the settings modified by the variant is only possible in the variable management table.

The following table defines where items of the hydraulic mode can be set: | *Item* | *Scenario* | *Variant* | | *Calculation settings:* | < | < | | –> Steady or Unsteady flow | X | | | –> Time | X | | | –> Results writing | X | | | –> General settings | X | | | *Nodes description:* | < | < | | –> Create, delete offtakes | X | | | –> State of the offtake (Active / Inactive) | X | X | | –> Modify the type of the downstream boundary condition | X | | | –> Modify water elevation or discharge of downstream boundary condition | X | X | | –> Create, delete, modify type of devices at offtakes | X | | | –> Modify characteristics of devices (width, sill elevation, opening…) | X | X | | *Cross-structures description:* | < | < | | –> Create, delete, modify type of devices | X | | | –> Modify characteristics of devices (width, sill elevation, opening…) | X | X | | *Roughness (Manning) and infiltrations* | X | |

As an unsteady flow calculation requires an initial condition, it is advisable to perform the various simulations (steady and one or more unsteady flow) in a scenario and create a variant by unsteady flow simulation you want to perform.

Chapter 70

Running a steady state calculation (FLUVIA)

For a steady state simulation, the minimum required information are: - The roughness (Manning) on the entire network - Inflows (positive) on all upstream nodes of the network - Downstream boundary conditions (Water elevation, Q(Z) or Q alpha) on all downstream nodes of the network

 | The upstream-downstream direction of flow in steady state was defined when entering the network topology. If reaches are in the wrong direction, we must return to the geometry mode, click on the reaches to reverse and choose the right-click of the mouse “Reverse reach.” Then you have to run again the mesh (Talweg) before returning to hydraulic mode. |

If these three items of information are entered, it is possible to realize a first calculation in steady state. Then the user can complexify its system by describing: - Cross-structures located on the singular sections, - Offtakes at intermediate nodes that may be flow boundary conditions (negative or positive flow) or devices.

To run a steady simulation from editing a scenario, just click on the . This can also be done from the project explorer by selecting the line of the script and clicking on the button “ Calculation steady”. Before the calculation, the software offers to save the changes and performs tests of consistency, including the direction of flows in reaches to ensure that the simulator has the necessary data to calculate.

During its execution the steady state solver (FLUVIA) will: - Calculate the water line and record the results requested in calculation parameters> Writing Results, - Screen display warnings and ongoing miscalculations, - Write a log file, named `[Name Of Xml Project File]_FLUVIA_[Scenario number]_[Variant number].log` where you will find details of errors encountered.

Chapter 71

Running an unsteady-state calculation (SIRENE)

To start a simulation in unsteady-state conditions from the edition mode of a scenario, just click on the  button. This can also be done from the Project Explorer by selecting the scenario line and clicking the “ Transient Calculation” button. Before the calculation, the software proposes to save the modifications made and performs consistency tests.

 |The coherence tests are currently designed for steady-state, if the boundary conditions and direction of flow do not correspond to those accepted in steady-state, these warnings should be disregarded. |

During its implementation, the unsteady-state calculation program (SIRENE) will: - Calculate the water line at each time step and record, at the frequency defined in time parameters, the results requested in calculation parameters > Writing Results - Display on the screen the warnings and errors being calculated

Chapter 72

Overview of results

After performing a simulation in steady flow (FLUVIA) or unsteady flow (SIRENE), the result is available from the Project Explorer.

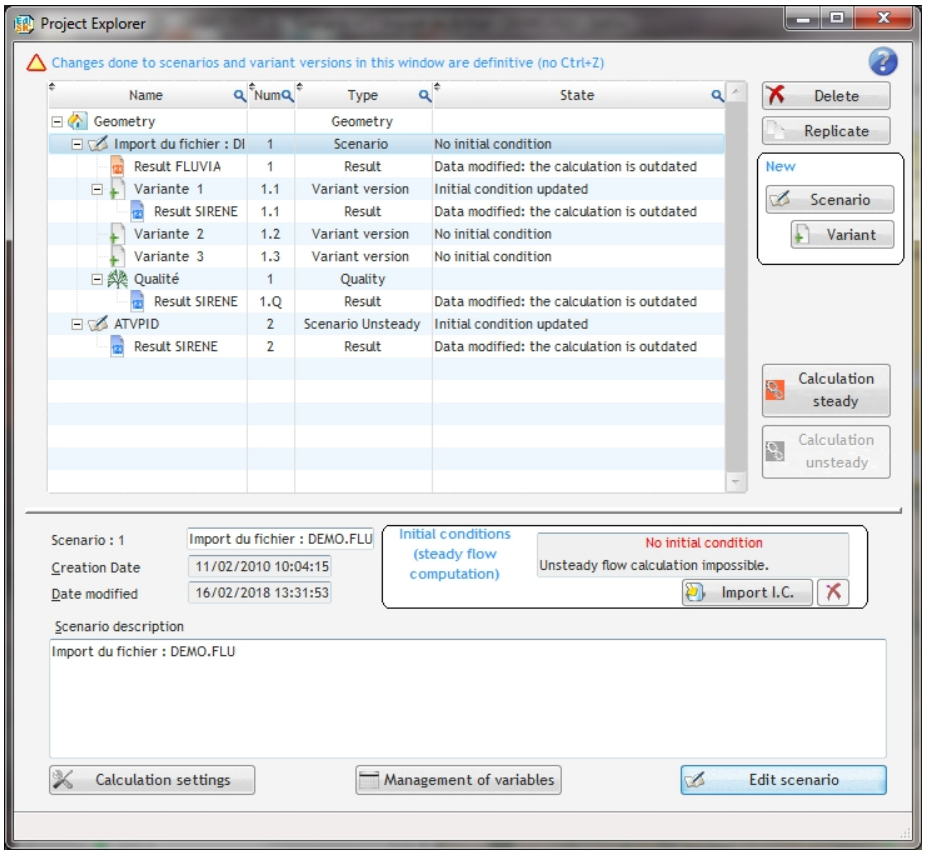


Figure 72.1: explorateurprojet

For a scenario or a scenario variant, a single result is saved. Running a simulation overwrites the previously result saved.

The use of the result mode is achieved by selecting a row of result in the Project Explorer and clicking “Edit result”.

Several visualization modes are available:

-* Longitudinal profil of water elevation and flows -* Results for objects of the same type time step by time step -* Chronicle of results for a selected object -* Calculation of water volume for a channel portion (Calculation of average delay)

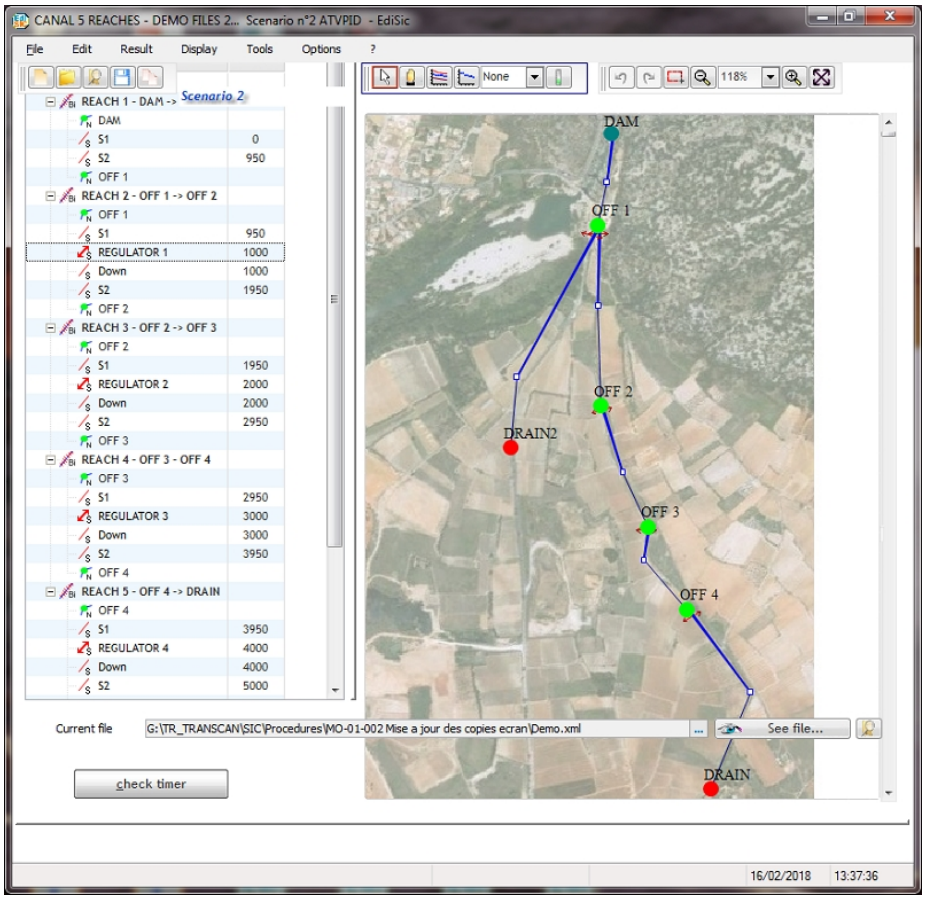
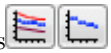


Figure 72.2: premiere_res

Chapter 73

Longitudinal profiles of water elevation and discharge

The display of longitudinal profiles is done using the buttons  or from the Tools > Discharges or Tools > Elevations menus. Before the selection of the tool or during its use, one should select the desired part of the network to display. A graphic is then shown at the bottom of the main window with the desired profil.

In case of diffluence or confluence that is part of the selection, a window opens with the list of the available upstream-downstream paths of the water. The selection of a path displays this path in red color on the network graph and the corresponding longitudinal profil.

In case of an unsteady flow or succession of steady flows simulations, discharges and elevations for different time steps are available through the tools located in the bottom of the profil window.

For example, the discharges profil allows to locate the intake or outake points:

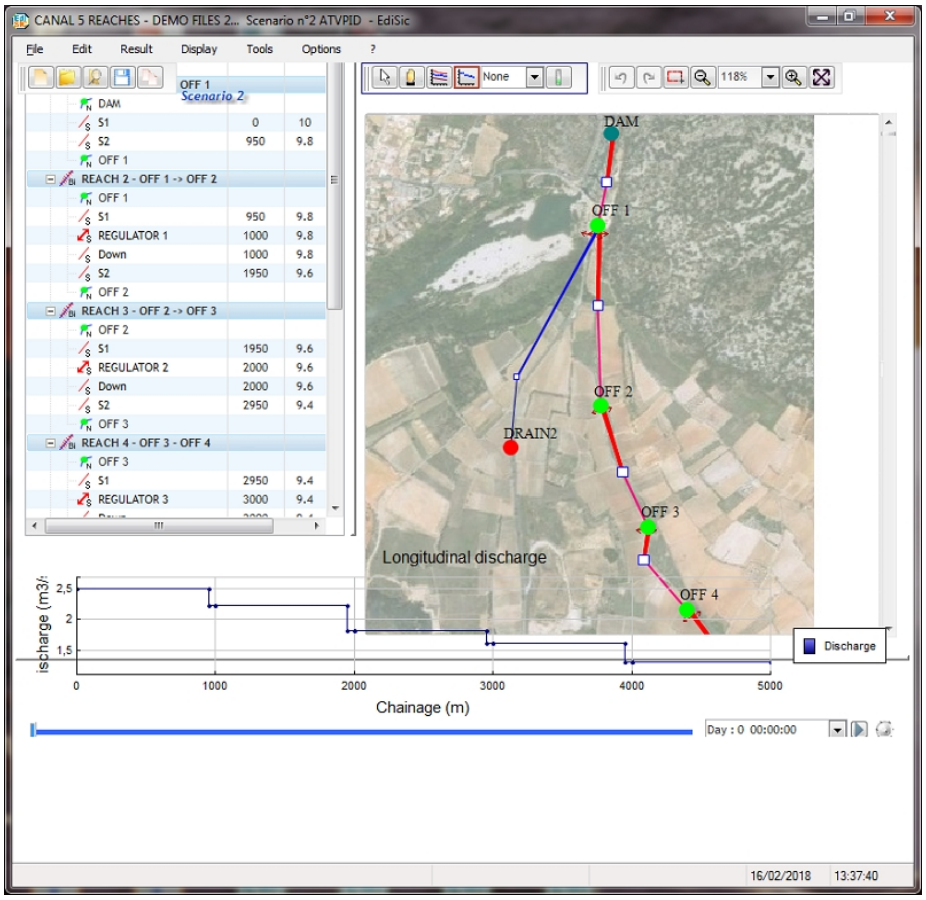


Figure 73.1: premiere_resdebit

Water elevation profils allow to locate zones of singular headlosses and overflows:

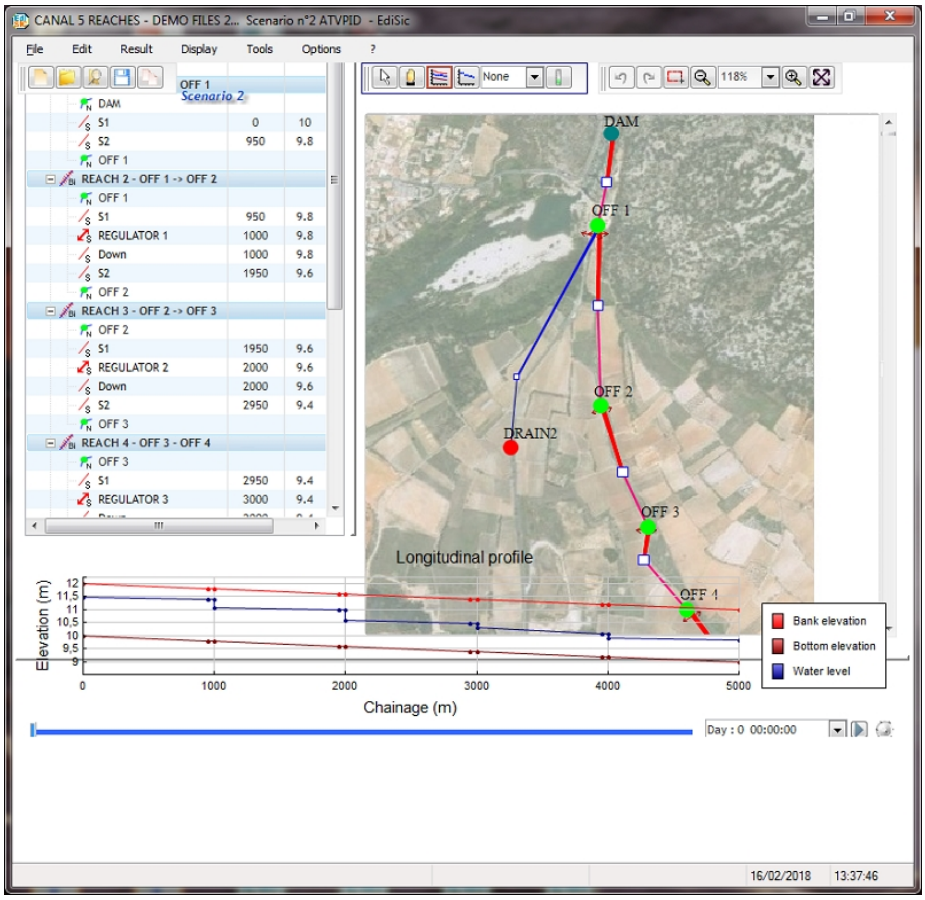


Figure 73.2: premiere_resprofil

Chapter 74

Data available in the results

Recorded data during simulations and available in results mode are those defined in Settings for results writing.

In addition to the variables defined in these settings, all variable subject to regulation (Time laws, Offtake regulation for obtaining an objective discharge, regulation at a cross structure for obtaining a target upstream elevation, Variable regulated by a regulation module) will added to the results and available for display.

Time steps available will be those defined in the simulation time settings.

Chapter 75

The results tables

Simulation results are presented in the form of tables on which multiple actions are available.

75.1 Export data

Export the table displayed to an Excel 2003 file is done with the button .
The  button allows to copy the contents of the table into the clipboard and then, it can be pasted into a spreadsheet or text editor.

75.2 Data sorting

By clicking on any column header, you can sort them in ascending or descending order.

75.3 Display graphics

At the bottom of the window, a combo “Display graph” is available to display graphics for the different variables in the columns of the table.

Example for a flow time serie to a section:

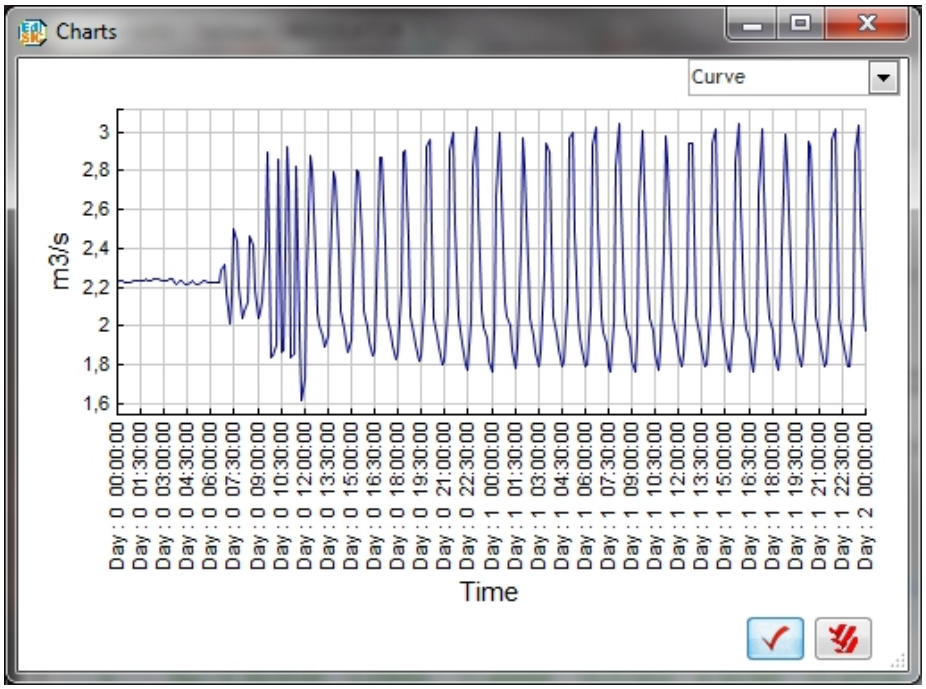


Figure 75.1: graphique1_section1_graphe_3

In the case of display results for objects of the same type for one time step, the minimum and maximum encountered during simulation appear respectively in light and dark green lines on the graph.

Example for discharge at Intakes:

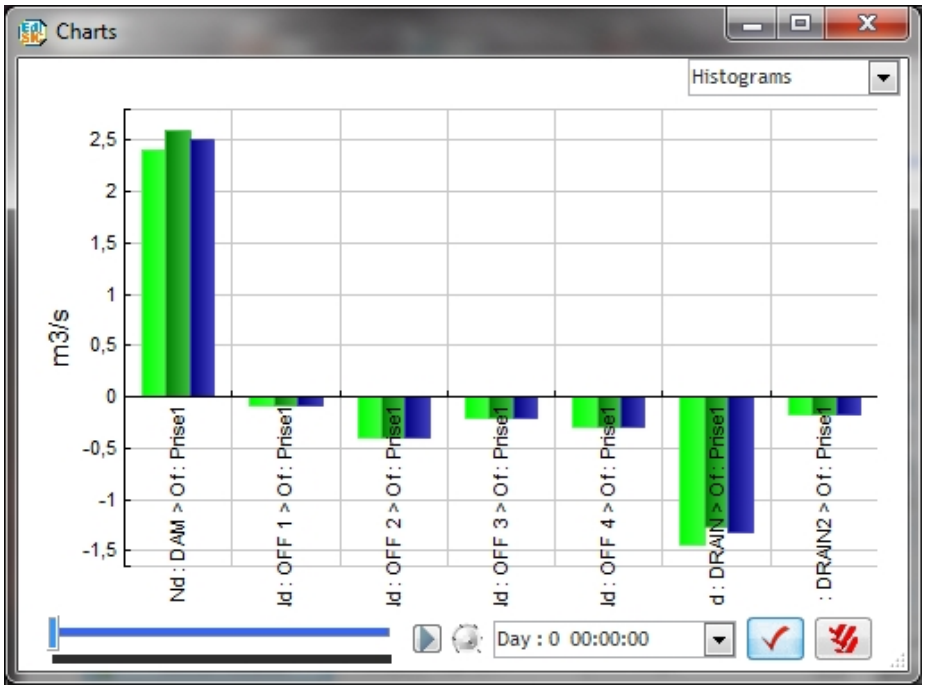


Figure 75.2: graphique1_type_5_graphe_3

75.4 Other possible actions

The windows of the results tables are also provided with a button “Advanced Settings” that lets you change the display format and delete columns or add new columns containing data computed from the existing results.

For tables of Sections, color code is used to represent the overflow states.

Finally, help is available to know the meaning of status and error codes on different Network objects.

Chapter 76

Results for objects of the same type

Results are available for different types of objects in SIC²:

-* The Sections -* The Singular Sections -* The Devices in Cross-Structures -* The Nodes -* The Intakes -* The devices in Intakes

You can view the results of all or part of these objects by making a selection on the network graph.

To view the results for a particular type of object, possibly select a network portion, and click this drop-down list in the toolbar:

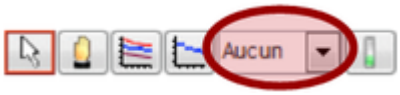


Figure 76.1: doc98

The results are displayed for the selected object type:

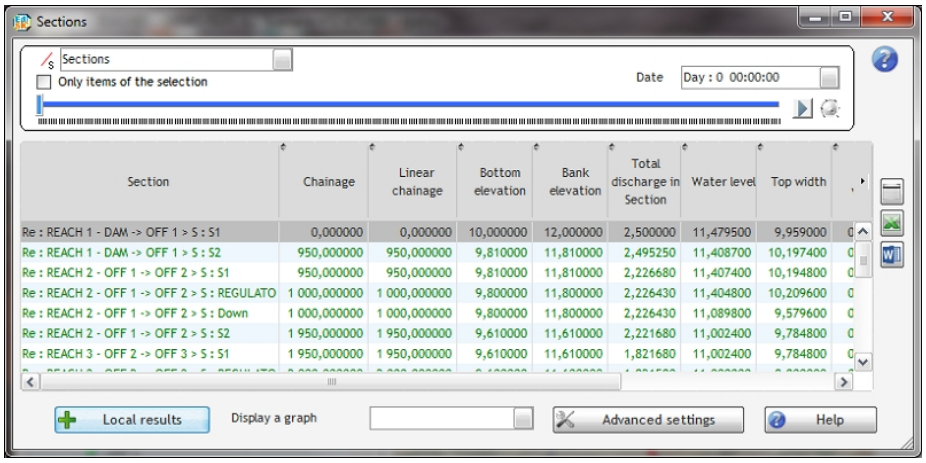


Figure 76.2: resultatselements_1

If the simulation result includes several time steps, the first time step is displayed by default. You can display other time steps using a scroll bar and a drop down list:



Figure 76.3: doc103

This function is also available for graphics displayed from this window. The play button and the potentiometer to its right allow to launch a dynamic animation of the results over time. It is also possible to record the animation of a graphic in the form of a video.

In addition to the common options present on the tables of results, this viewing mode offers different actions:

-* Directly select another type of object to be displayed using this drop-down list at the top of the window. -* The ability to switch between the display of all objects and selection using the checkbox “Only the elements in the selection” at the top of the screen. -* The ability to display the time serie result for an item by double-clicking the item from the list or by selecting it and clicking the “local results” button at the bottom of the window.

Chapter 77

Results time series for a selected object

These time series are available in either double-clicking the tree of network objects or on the objects of the network graph or else by double clicking on a line of results tables for an object type of Network.



Figure 77.1: resftnoeud1 — image absente de la source

For the possibilities offered by the results tables in general, see the dedicated documentation.

The upper part of the window lets you navigate between objects on the same location or nearby. In the example above, we can see the Node results, or the one on the offtakes or devices located in the various structures of the node. One can also easily view the data of adjacent nodes using the arrows surrounding the logos  and  and the data of the upstream section of diffluent reaches and

the data of the downstream section of the confluent reaches by double-clicking on a row of the table of adjacent reaches to the node.

Chapter 78

Advanced settings of results tables

At the bottom of the results window for objects of the same type or the results of time serie window for a selected object, the button “Advanced Settings” allows to change the data format, in particular, one can choose to:

-* Select the number of decimal places used for the display. -* Convert length in different units: m (metric system, ISU=International System of Units), feet (UK, USA). -* -* Convert discharges in different units: m3/s (ISU), Ml/day (Mega liters per day, Australia), cfs (cubic feet per second, USA).

Warning, these units are modified only in these hydraulic results (not in the edition of geometric or hydraulic data which remain in metric units, nor in the storage of the results in the xml project file or binary files if applicable). The longitudinal abscissas remain in meters (m). The results in m2 (areas, surfaces) and m3 (volumes) are kept in metric units (we can change it in a next version if you wish). On the tables there is no reference to the unit that has been changed, in the label at the top of the columns. On the other hand, on the graphs the units are correctly mentioned on the legend of the y-axis (m, feet or pieds, m3/s, Ml/day or cfs, etc.). You must therefore be very careful when using this option, which is stored in the Edisic settings (Edital.ini, in the exe subdirectory, with for example UnitLResults=2, UnitQResults=3), and is therefore retained between different uses of Edisic. A warning and confirmation request is made at the opening of a project to clarify this option. Exports of results by the regulation modules (e.g. PRINT module) or by SicExport remain in metric units.

In this window you can also choose not to display some columns and display additional data derived from existing data.

For sections, we will be able to calculate:

-* Water depth and free-board; -* Discharge and hydraulic surface of the minor bed in the case of a minor-medium bed; -* The speed of wave (c) and V+c and V-c where V is the flow velocity (V=Q/S). To get these variables the calculation must have been done with calculation of wetted area (for c and V) and top width (for c); -* DTVPC: The time (in seconds) needed to get a Courant number of 1 (dt=dx/(V+c)). Please note that DX is calculated only based on the display sections; this should be done for the calculation sections. Therefore, all calculation sections should first be switched to display sections; -* The lateral discharge; -* The Manning of minor and medium beds.

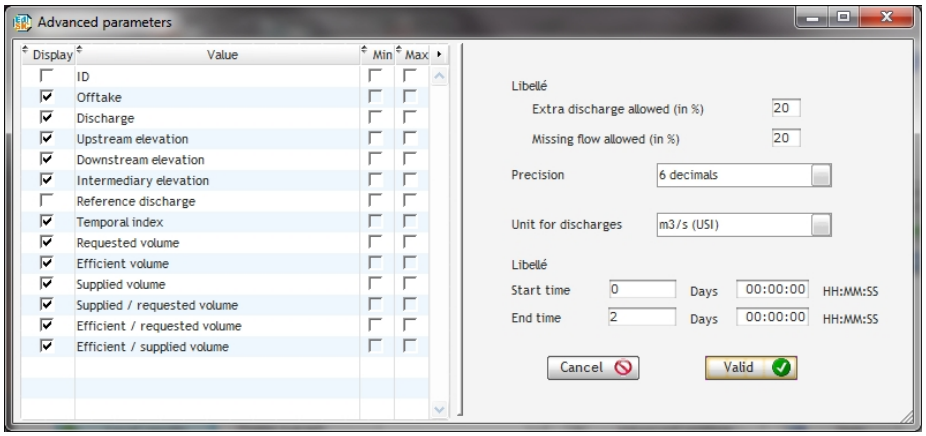


Figure 78.1: parametresavances

In the case of displaying the list of an object type on several time steps, it is possible to display the minimum and maximum for each variable over time and elapsed volume. For all the volumes calculated, the window offers to define the begin time and end time to be used for the calculation.

For the offtakes additional indicators are available:

-* The volume requested calculated from the integration of reference discharge at offtakes; -* The effective volume (the lower and excess tolerance compared to the requested discharge can be parametrized in this same window); -* Different ratios: supplied / requested, efficient / requested, effective / supplied; -* The time index.

The method of calculating volumic indexes and the time index are described in the theoretical documentation.

Chapter 79

Color status used in results tables

79.1 Colors used for nodes

The information about the nodes can appear in three colors according to calculation method for the reaches adjacent to the node: - Blue: All the reaches are in water elevation equality mode. - Green: All the reaches are in head equality mode - Red: Some reaches are in water elevation equality mode some others in head equality mode.

79.2 Colors used for the sections

Information on sections can appear in three colors considering the water level in the section: - Green: the water level is less than the bank level and the overflow level of the minor / medium bed if it exists. - Orange: the water level is between the overflow level of the minor / medium bed and the bank level. - Red: the water level is higher than the bank level.

Chapter 80

Flags meaning

80.1 Meaning of the codes "Error flag" at the offtake

The error flag at the offtake is used to control the results of a target discharge regulation at an offtake. Its meaning is as follow:

Code | *Meaning* |
-4 | Non convergence |
-3 | Downstream elevation superior to upstream elevation (the offtake is closed!) |
-2 | Sill elevation or downstream weir elevation superior to the upstream elevation |
-1 | The required discharge cannot be satisfied |
0 | The required discharge is null (the offtake is closed!) or there are no regulation at the offtake |
1 | The required discharge is satisfied |

80.2 Meaning of the codes "Error Flag" at a section

The error flag (IF1 variable in results and/or XML file) has the following meanings:

Code | *Meaning* |
10 | Targeted water elevation at the regulator < downstream water elevation |
9 | Targeted water elevation at the regulator too high |
8 | Targeted water elevation at the regulator too low |
7 | Non convergence of regulator calculation |

6 | Device oversized relative to the geometry of the river (water depth < critical depth) |
5 | The surface width varies too quickly: unaccuracy calculation. Froude number is >1 while the calculation is located on a branch of the specific head HS(Y) |
4 | Large variation of elevation after calculation with small steps and slope of the water line still steep |
3 | Water line slope steep with small steps calculation |
2 | Passing by the critical depth in the first section |
1 | Subcritical flow |
0 | Supercritical flow |
-1 | Water elevation less than bottom elevation |
-2 | Enormous water level in the section |
-3 | Non convergence of critical depth calculation |
-4 | Negative discharge in the section |
-5 | Non convergence of water level calculation |
-6 | Non convergence of cross-structure calculation |

Meaning of the "Type of flow" codes at a device

The type of flow (ITEC variable in the XML file) has the following meanings:

Code | *Meaning* |
-2 | Free flow, open surface flow |
-1 | Submerged flow, open surface flow |
0 | Nil flow rate |
1 | Free flow, piped flow |
2 | Submerged flow, piped flow |
8 | Free flow, open surface flow with overflow |
9 | Submerged flow, open surface flow with overflow |
11 | Free flow, piped flow with overflow |
12 | Submerged flow, piped flow with overflow |

Meaning of "Gec gates errors" codes at a device

The error codes of the GEC gates (IERR variable in the XML file) has the following meanings:

Code | *Meaning* |
0 | OK |
-1 | J = Z1-Z2> JM for AVIS gates |
-2 | Closed gate - Water level below the level to be regulated - Increase Z1 (AMIL) or Z2 (AVIO / S) |
-3 | Fully opened gate - Water level above the level to be regulated - Decrease Z1 (AMIL) or Z2 (AVIO / S) |
-4 | Gates AMIL, ZMAX is not between 0 and D |
-5 | If R=<0 or HA-ZR>R or D=<0 or H=<0 or HA-H-ZR<-R (all gates except mixed gates) |
-5 | If R=<0, D=<0 or H=>0 (all mixed gates) |
-6 | AMIL with overflow (Z1>ZR+W+EC) |

-7 | Problem with the targeted flow too high for the selected gate |
-8 | AVIO with overflow (normally case not possible) |
-9 | AVIS with overflow (Z1>ZR+W+EC) |
-10 | If J = Z1-Z2 <0 |

Chapter 81

Create a video file from a graphic animation

For results with several time steps (calculation with a succession of permanent regimes or transient computation), the data displayed for a set of objects can be animated as a function of time.

Here is a method that allows to save all the images of the animation in a folder and make a video (.AVI file).

81.1 Save animation images in EdiSIC

<Sic|img=graphique1_type_1_graphe_3>

To start the animation, each graphic has a “play” button.

To save the JPG animation images to the disc, press the [Ctrl] key when the play button is clicked. At this time, EdiSIC proposes to choose a folder where to save the images. Then animation and recording start simultaneously. Animation and recording can be stopped at any time by clicking anywhere on the screen.

81.2 Convert images to a video file with MakeAVI

Once the images have been recorded, a program must be used to convert this series of images into video. The rest of this documentation proposes to use MakeAVI a minimalist program and free software downloadable at the following address:

- http://sourceforge.net/projects/makeavi/files/latest/download?source=files



Figure 81.1: doc68

MakeAVI is very simple to use, it needs in this order: - Click on “Add files”, select all the image files (Ctrl + A to select all the files of a directory) with which one wants to make a video - Choose the “Playback frame rate”, that is, the number of frames per second of the video. (Eg for a simulation of 100 steps of time, we will put a frame rate of 10 fps to get a video of 10 seconds) - Click on “Begin!” To choose the location and name of the created .AVI file, choose the codec [[If this term does not mention anything for you in the list, choose “Intel Indeo” or if it is not present “Radius Codec Cinepak” by leaving the default settings]] used and creating the video file.

Chapter 82

The AMIL method



| These gates can be modelled directly in Hydraulic mode from SIC Version 4.0. It is better to use this new option. Nevertheless the previous option using a regulation module is still available. |

For the AMIL gates, the variables D, d, HA and FILTRE must be given under the format (3(F8.2,1X),F8.2):

- `D d HA FILTRE`

  PS= 220. .10 1.50 0.8

- D is the gate reference from the constructor list : 80, 90, 100, 110, 125, 140, 160, 180, 200, 220, 250, 280, 315, 355, 400, 450, 500, 560, 630, 710, 800,

- d is the decrement (equal to L/50 for AMIL, where L the width of the gate, Cf. Documentation of the constructor), given in meters (m), - HA is the level of the gate rotation axis measured from the sill (in m).

- FILTER is the coefficient of the first order filter used to damp the fluctuations of the controlled water level. It should be in the range [0 .. 1]. The filter is stronger for higher coefficients. This filter is usually compulsory to get a stable gate.

Remark: The radius of the gate R is calculated from D and the constructor table (63, 63, 63, 63, 90, 90, 90, 125, 125, 125, 160, 160, 200, 200, 250, 250, 315, 315, 400, 400, 450),

Remark: D and R are given in the above tables in centimeters. In the software, these variables are converted into (m), when reading the .REG file. These

converted variables (in meters) are the ones written on the .LST file. If the decrement d is not provided (i.e. equal to zero), we take d = 2% of D (d=0.05*D).

An example of regulation file (.REG) for the AMIL gate is given bellow. The calculation routine CAMIL for the AMIL gate is given in the following section.

-

82.1 Controlleur 1

D=AMIL

- DUMin DUMax UMin UMax DT ('PU',1X,4(F5.2,1X),F6.0) PU= 0.00 0.50 0.00 2.00 0 U=AWS(X, 2415.0, 3) Y=TYS(R, +0, +0)
- `D d HA FILTER (3(F8.2,1X),F8.2)`

  PS= 220 .10 1.50 0.8

Chapter 83

Exemple of calculation routine CAMIL (Amil gates)

```
SUBROUTINE CAMIL(U,Y,PARA)
    PARAMETER(NBCM=$NBCM,NBUM=$NBUM,NSING=$NTO,NBIEF=$IB)
    PARAMETER(NPOIN=$NPOIN,NSECT=$NSC)
    CHARACTER CFIC1*3
    DIMENSION Y(*),U(*),PARA(*)
C-----index of the current controller (INURE),
C-----current number of calls of the controller n (INUAP(n))
    COMMON/P1/IFIC1,INURE,INUAP(NBCM),IFIC2
    COMMON/P2/INU,IFIC3(2),FIC1(3)
    COMMON/REGU/IFIC4,CFIC1(NBUM),IFIC5(NBUM,2),NSUG(NBUM)
1           ,IFIC6(NBUM),NKSU(NBUM),FIC2(NBUM,2)
    COMMON/REG2/NUD(NBCM),IFIC7(NBCM),BFIC1(NBCM)
    COMMON/NGEO/IFIC8(NBIEF,2),IP1(NSECT),IFIC9(NSECT,2)
    COMMON/RGEO/IFIC10,FIC3(NSECT),YGEO(NPOIN),FIC4(NPOIN,3)
1           ,FIC5(NSECT,3)
    COMMON/RSIN/FIC6(NSING),ZS0(NSING),FIC7(NSING,3)
C-----Equations de Goussard
C-----Affectations
C-----PARA(1) Modele (D parmis LISTD) de vanne AMIL
    D=PARA(1)
    Dec=PARA(2)
    R=PARA(3)
    HA=PARA(4)
C-----On met un filtre d'ordre 1 : S/E = (1-a)Z-1 / 1-aZ-1 (FILTRE = a)
```

```
    FILTRE=PARA(5)
C-----ZD : hauteur du radier % cote de fond
    ZD=ZSO(NKSU(NUD(INURE)))-YGEO(IP1(NSUG(NUD(INURE))))
C-----Calcul pour cette methode
    alfmax=asin(0.45*D/R)-asin(0.2205*D/R)
    alfb=HA/R
    alfb=min(alfb,1.)
    alfb=max(alfb,-1.)
    alfb=acos(alfb)
    alfc=alfb+alfmax
    UMAX=HA-R*cos(alfc)
C-----Y(1) est en tirant d'eau (mode TYS obligatoire sur le fichier .REG)
    Z=Y(1)-HA-ZD
C-----Initialisation de Zold au premier appel
    IF(INUAP(INURE).EQ.1) PARA(10)=Z
    Z=FILTRE*PARA(10)+(1.0-FILTRE)*Z
C-----On stocke Zold filtré dans PARA(10) pour le filtre
    PARA(10)=Z
C-----Regular setting
    if (Z.LE.-Dec) then
        alfa=0
        c=0
    elseif (z.GE.0.) then
        alfa=alfmax
        c=1
    else
        alfa=alfmax-asin(-Z/Dec*sin(alfmax))
        c=(1 - (0.45*D/R)**2)**0.5 * sin(alfa)
        1  / (0.2295*D/R) + 1.9608 * (1 - cos(alfa))
    endif
C-----Calcul de la commande
    U(1)=c*UMAX
C-----Si plusieurs U (SIMO) on duplique la meme ouverture
C-----(interessant dans le cas de plusieurs vannes identiques en parallele
C      en mode V par exemple)
    IF(INU.GT.1) THEN
        DO 10 I=2,INU
            U(I)=U(1)
        10    CONTINUE
    ENDIF
C-----Calcul du debit
C     En réglage classique Zmax=0, en réglage alternatif Zmax=Dec
C     Kj=1 + 2.2*zmax/D - 1.6*(zmax/D)**2
C     Kq=1 + 2.82*zmax/D - 34.8*(zmax/D)**2 + 164*(zmax/D)**3
C     x=Y(1)-z2
```

```
C     if (x.LT.0) then
C        Q=0.
C     elseif (x.LE.(0.45*(1 - 0.7211*c**0.3)*Kj*D)) then
C        Q=0.7096*c*Kq*D**2.103*x**0.397
C     else
C        Q=0.3113*c*Kq*Kj**0.397*D**2.5
C     endif
END
```

Chapter 84

The AVIS - AVIO method

- ALFMAX d R HA STEP FILTER PS= 60.00 .10 2.00 1.50 0.03 0.8 ** - ALFMAX is the maximum opening angle of the gate measured compared to the closed position,

- d is the decrement (equal to R/20 for AVIS and AVIO, where R is the gate radius, Cf. Documentation of the constructor), - R is the gate radius,

- HA is the level of the gate rotation axis measured from the bottom. - STEP is the height of the step (if any) through the gate between the upstream and the downstream cross sections. It is positive if the bed elevation is lower at the downstream section. The unit used is (m).

- FILTER is the coefficient of the first order filter used to damp the fluctuations of the controlled water level. It should be in the range [0 .. 1]. The filter is stronger for higher coefficients. This filter is usually compulsory to get a stable gate.

<doc107|center>

An example of regulation file (.REG) for the AVIS gate is given bellow. The calculation routine for the AVIS and AVIO gates is given in the following section.

-

84.1 Controlleur 1

D=AVIS

- DUMin DUMax UMin UMax DT ('PU',1X,4(F5.2,1X),F6.0) PU= 0.00 0.50 0.00 2.00 0 U=AWS(X, 2415.0, 3) Y=TYS(R, +0, +1)
- ALFMAX d R HA STEP FILTER (5(F8.2,1X),F8.2) PS= 60.00 .10 2.00 1.50 0.03 0.8

Chapter 85

Exemple of calculation routine CGEC (Avis and Avio gates)

```
SUBROUTINE CGEC(U,Y,PARA)
    PARAMETER(NBCM=$NBCM)
    DIMENSION Y(*),U(*),PARA(*)
C-----index of the current controller (INURE),
C-----current number of calls of the controller n (INUAP(n))
    COMMON/P1/ILNPS,INURE,INUAP(NBCM),IREGS
    COMMON/P2/INU,INY,INZ,DTU,DTY,DTZ
C-----Equations de Goussard
C-----Affectations
C-----PARA(1) en degres, ALFMAX en radiants
    ALFMAX=PARA(1)*0.01745
C-----Dec est negatif dans la formule de Goussard
    Dec=-PARA(2)
    R=PARA(3)
    HA=PARA(4)
C-----CHUTE &gt; 0 si ca descend vers l'aval
    CHUTE=PARA(5)
C-----On met un filtre d'ordre 1 : S/E = (1-a)Z-1 / 1-aZ-1 (FILTRE = a)
    FILTRE=PARA(6)
C-----Calcul pour cette methode
C-----Y(1) est en tirant d'eau (mode TYS obligatoire sur le fichier .REG)
    Z=Y(1)-CHUTE-HA
C-----Initialisation de Zold au premier appel
    IF(INUAP(INURE).EQ.1) PARA(10)=Z
```

```
    Z=FILTRE*PARA(10)+(1.0-FILTRE)*Z
C-----On stocke Zold filtré dans PARA(10) pour le filtre
    PARA(10)=Z
    IF (Z.GE.0.) THEN
        ALFA=0.
    ELSEIF (Z.LE.Dec) THEN
        ALFA=ALFMAX
    ELSE
        ALFA=0.
        IF(Dec.NE.0.) ALFA=SIN(ALFMAX)*Z/Dec
        ALFA=MIN(ALFA,1.)
        ALFA=MAX(ALFA,-1.)
        ALFA=ASIN(ALFA)
    ENDIF
C-----Calcul de U
    ALFB=0.
    IF(R.NE.0.) ALFB=HA/R
    ALFB=MIN(ALFB,1.)
    ALFB=MAX(ALFB,-1.)
    ALFB=ACOS(ALFB)
C-----Calcul de la commande
    U(1)=HA-R*COS(ALFA+ALFB)
C-----Si plusieurs U (SIMO) on duplique la meme ouverture
C-----(interessant dans le cas de plusieurs vannes identiques en parallele
C      en mode V par exemple)
    IF(INU.GT.1) THEN
        DO 10 I=2,INU
            U(I)=U(1)
        10    CONTINUE
    ENDIF
END
```

Chapter 86

The PID method

The PID method corresponding to a conventional PID controller is programmed according to the routine CPID shown below. You can use a copy of this routine to write your own calculation routine CUSER if you want to test a variant of PID.

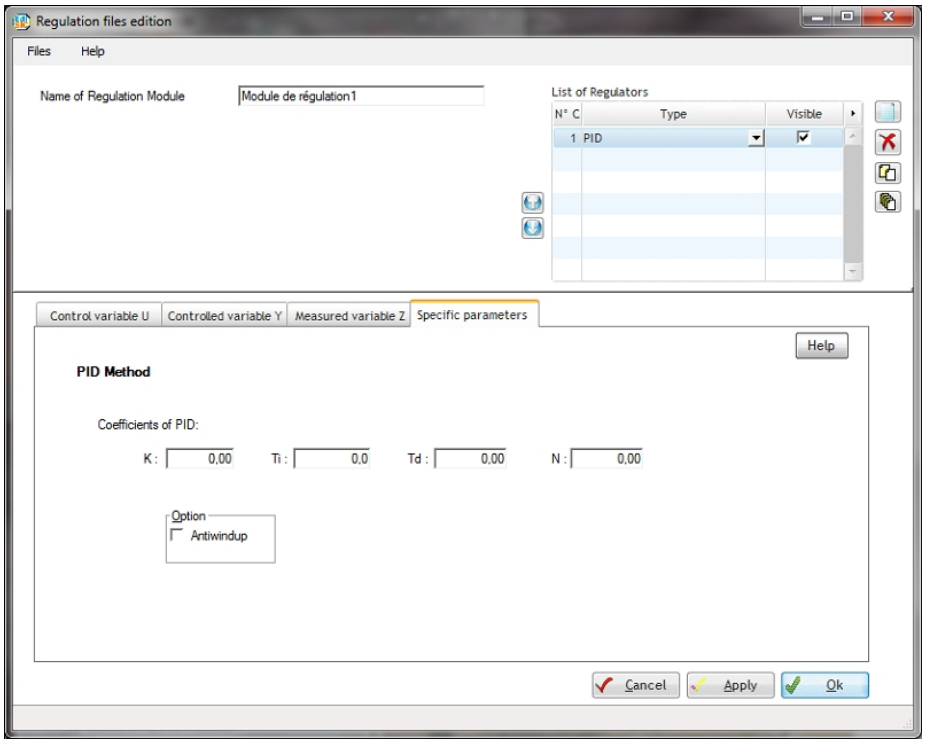


Figure 86.1: editregv5_pid

For the PID method this coefficients are: K, Ti, Td, N and Antiwindup with the format `(4(F8.2,1X),F8.2)`.

```
* Specific Parameters
*          K       Ti        Td         N  AntiWup
PS=    -0.23  1245.20      0.00      5.00      0.00
```

The routine for the reading of these parameters LPID is given below. You can use a copy of this routine to write your own routine LUSER to read parameters for a USER control module (USER1 to USER9).

The goal of the antiwindup is to stop the integration of the error terms between the controlled variable and the corresponding targets, when the actuator is saturated. This is to avoid having too much to catch up when the saturation is finished.

Chapter 87

Example of LPID reading routine

Subroutine to read the PID specific parameters :

```
!===============================================================================
!> Lecture des parametres specifiques pour la methode PID
!===============================================================================
SUBROUTINE LPID(U,PARA)
   USE SIRENE_parametres, ONLY : LONG
   USE F1       , ONLY : IERROR,IWARN,CWARN

   IMPLICIT NONE

   CHARACTER, INTENT(IN) :: U*(*)
   REAL(KIND=LONG), DIMENSION(*), INTENT(OUT) :: PARA
   INTEGER :: I

!-----Debut de la routine
!-----Lecture des parametres specifiques pour cette methode

!-----Debut de la routine
      READ(U,'(4(F8.2,1X),F8.2)',ERR=10) (PARA(I),I=1,5)
      PARA(6)=0
      PARA(7)=0
      PARA(8)=0
      RETURN
   10 CONTINUE
      IERROR=2
```

```
END SUBROUTINE LPID
```

Chapter 88

Example of calculation routine CPID

The PID method corresponding to a classic PID controller is programmed as explained below. It includes an optionnal filter term of the derivative term, an antiwindup and deadband zone:

```
SUBROUTINE CPID(Reg,U,Y,YT,PARA,ISIMO)
   USE SIRENE_parametres, ONLY : LONG
   USE IFICH   , ONLY : JREGS
   USE TEMPS   , ONLY : T
   USE D_REGULATION, ONLY : Regulateur_t
   USE D_GLOBAL, ONLY : sProgActif

   IMPLICIT NONE

   TYPE(Regulateur_t), INTENT(INOUT) :: Reg
   INTEGER, INTENT(IN) :: ISIMO
   REAL(KIND=LONG), INTENT(IN) :: Y,YT
   REAL(KIND=LONG), INTENT(INOUT) :: U,PARA

   INTEGER :: IU,IY
   REAL(KIND=LONG) :: KP,N,TI,TD,AWU,EOLD,SE,DE,E,STI,A1,A2,U2

!-----Si ISIMO=0 c'est un PID monovariable, si ISIMO=1 et s'il y a plusieurs U on duplique U
!-----Si ISIMO=I<0 il est appelé à partir d'ATVPID et c'est l'indice du U et du Y qu'on doit
!-----   utiliser pour décaller le U et Y global pour DYMAX et les variables de l'antiwindup
      INTEGER, PARAMETER :: NMES=$NSIRE,LMES=$LSIRE
!-----On gère l'antiwindup
      LOGICAL SATURE
```

```
      CHARACTER MESP*(LMES)

!-----On utilise DTU et non plus DT (POM 2/6/99)
      COMMON/SIRE/MESP(NMES)

      DIMENSION Y(:),YT(:),U(:),PARA(:)
!-----Affectations
      KP=PARA(1)
      TI=PARA(2)
      TD=PARA(3)
      N=PARA(4)
      AWU=PARA(5)
      EOLD=PARA(6)
      SE=PARA(7)
      DE=PARA(8)
!-----Calcul pour cette methode
!-----Nouvel ecart (e = y* - y)
      E=YT(1)-Y(1)
!-----Recherche de l'indice global pour Y et pour U (POM 14/3/5)
      IF(ISIMO.LT.0) THEN
         IY=ABS(ISIMO)
         IU=ABS(ISIMO)
      ELSE
         IY=1
         IU=1
      ENDIF
!-----On gère une bande morte (POM 21/07/04)
      IF(ABS(E).LT.Reg%tY(IY)%DYMax) THEN
         IF(sProgActif=='SIRENE') THEN
           WRITE(JREGS,MESP(87)) T,'DYMax',Reg%num,IY,Reg%tY(IY)%DYMax,E,0.
         ELSEIF(sProgActif=='FLUVIA') THEN
           WRITE(JREGS,MESP(76)) T,'DYMax',Reg%num,IY,Reg%tY(IY)%DYMax,E,0.
         ENDIF
         E=0.
      ENDIF
!-----Terme integral (approximation de Tustin)
!-----On gère l'antiwindup (POM 16/07/04)
      IF(TI.NE.0.) THEN
         STI=KP*Reg%tU(IU)%VarU%DT/TI*(EOLD+E)/2
      ELSE
         STI=0.
      ENDIF
      IF(AWU.EQ.1.) THEN
         SATURE=Reg%tU(IU)%FiltreU%iFlagBloc>=3 .AND. Reg%tU(IU)%FiltreU%iFlagBloc/=5
         IF(.NOT.SATURE) THEN
```

```
            SE=SE+STI
         ENDIF
      ELSE
         SE=SE+STI
      ENDIF
!-----Terme derive
      IF(TD.NE.0.) THEN
         IF(N.EQ.0.) THEN
!-----------Terme derive sans filtre (backward difference)
!-----------Avec une consigne y* constante
            DE=KP/Reg%tU(IU)%VarU%DT*TD*(E-EOLD)
         ELSE
!-----------Terme derive avec filtre (backward difference)
!-----------Avec une consigne y* constante
            A1=1/(1+N*Reg%tU(IU)%VarU%DT/TD)
            A2=KP*N*A1
            DE=A1*DE+A2*(E-EOLD)
         ENDIF
      ENDIF
!-----Calcul de la commande
      U(1)=KP*E+SE+DE
!-----Ancien ecart
      EOLD=E
!-----Affectations
      PARA(6)=EOLD
      PARA(7)=SE
      PARA(8)=DE
!-----Si plusieurs U (SIMO) on duplique la meme ouverture
!-----(interessant dans le cas de plusieurs vannes identiques en parallele, en mode V par exemple
      IF(ISIMO.GT.0) THEN
         U2=U(1)
         U(:)=U2
      ENDIF

END SUBROUTINE CPID
```

If you want another PID equation, you can always copy and paste this subroutine into a USER regulation module and modify it as you want (do the same with the LPID -> LUSER subroutine to read the parameters).

Chapter 89

Regulation modules

In SIC^2, a regulation module corresponds to a set of regulators (also called controllers) which will operate simultaneously during a simulation. It is possible to define several control modules in a project, thus stored in the xml file and potentially available for all scenarios and variants. For each scenario, it is necessary to define which control module to use before starting a simulation. If a regulation module is selected for a scenario, it will also be active for all its variants. To deactivate it in a variant, it must be disabled in the parent scenario. It is of course possible to select none, but at most one. It can be said that a regulation module can make it possible to define a logic for managing a network, possibly comprising several regulators or controllers.

The regulation modules can be used both in unsteady-state calculation and steady-state calculation. Apart from the fact that the iterations of the regulation modules (they will be called regularly to read Y and Z data and calculate U commands) will be done either on the instants of the transient regime or on the instants of the successions of steady state regimes, the calculation principles are very similar. There is however a difference on the mode of calculation of the variables in relative mode. Indeed this relative mode considers a variable in relation to its initial value at t=0, and is therefore relevant in transient. However on the successions of calculations in steady state regimes there is no notion of initial time and therefore the relative mode will in fact work like the absolute mode.

The regulation modules can be accessed from hydraulic mode of SIC^2 either via the menu Tools> Regulation modules, or by right-clicking on a blank area of the network graph.

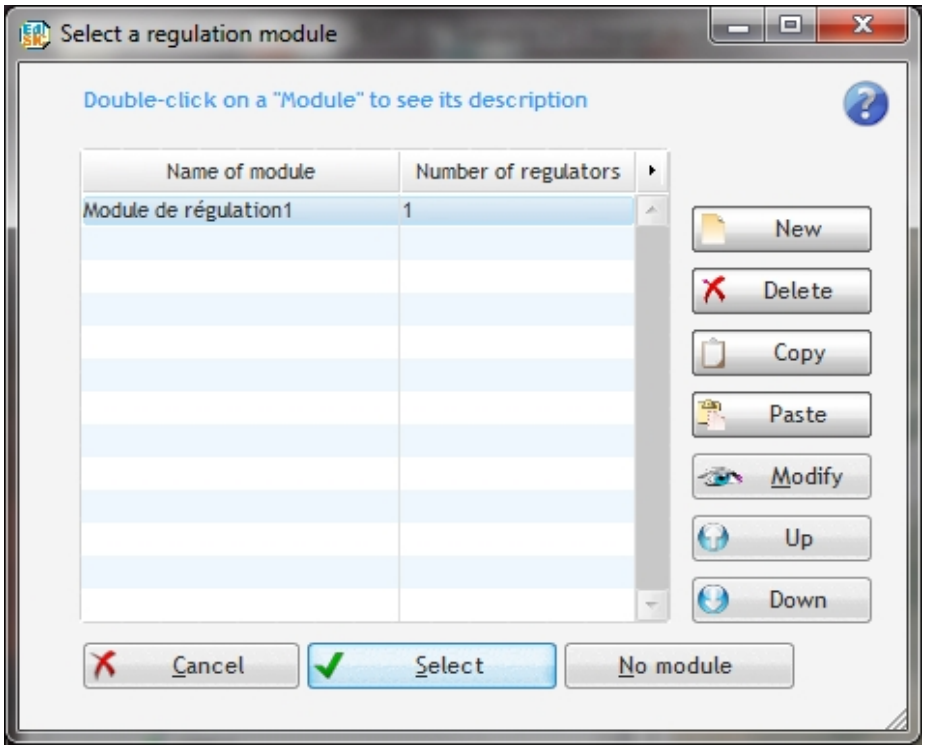


Figure 89.1: choixmodulereg

The window that appears allows you to: - Add, delete and modify regulation modules in the current project. - Select a regulation module to be used for the simulation.

89.1 Choosing a regulation module

If you do not want to use a regulation module, click the “No Module” button. If not, choose a regulation module from the list and click on “Select”.

In hydraulic mode, for scenarios and variants editions, all the controllers of the selected regulation module will appear on the network graph. Unless of course, for example to lighten the display, or to hide some controllers less important than others, some of them are declared as non-visible (click or not on the box Visible to the right of the name of the Controller in the list). The variables U are connected to the box representing the regulator with a red line. Y variables with a dark green line. Variables Z with a light green line.

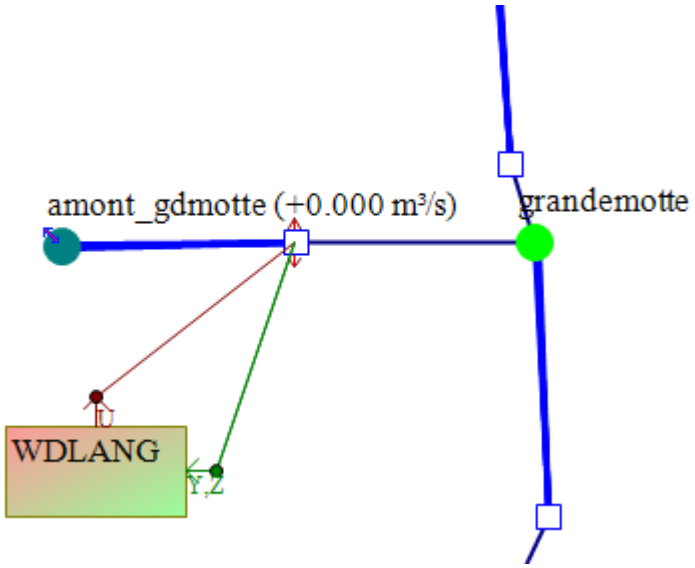


Figure 89.2: doc90

89.2 Editing a regulation module

The principle of the regulation modules in this version is, on the one hand, to define the measuring points (measured variables Z), controlled variables (controlled variables Y and their setpoints as a function of time YT, if necessary) and control action variables (U), and on the other hand to specify the method of regulation acting on these points, whatever their nature and location.

This method can be chosen from an existing library. The regulation methods proposed for now are:

- AMIL (Gec-Alsthom control gates, but old option available in version 4 of SIC, it is now possible to directly select this device in the library of devices, which is much better)
- AVIS (same remark as for AMIL)
- AVIO (same remark as for AMIL)
- ATV (Auto Tuning Variation, to automatically calibrate a PID controller)
- ATVPID (Method that applies the ATV method initially and then starts the PID as soon as ATV has supplied the parameters of the PID, and switches from one to another. The temporisation can be used to wait for hydraulic stabilization before switching to the next U)
- BIVAL (Method controlling a virtual intermediate point obtained as a linear combination of an upstream point and a downstream point, which, if well chosen, corresponds more or less to controlling a volume in this section.) Method developed by the Sogreah in the 80s)

- BOLST (open loop provided as a text file)
- BOMAT (open loop provided as a MatLab .mat file)
- BOSCI (open loop provided as a .dat SciLab file)
- DIGEST (simulator of the SCADA digester inspired by that of the Gignac irrigation canal, cf. SCADA below)
- DSS (discrete controller in the state space whose matrices A, B, C and D are given in a MatLab .mat file, allowing to simulate any LTI = Linear Time Invariant controller)
- FILEX (data exchange U, Y, YT and Z in an external file)
- GPC (Generalized Predictive Controller. Not active in this release)
- LP (calculation of the LP norm of a chosen Z signal)
- MATLAB (DDE link with MatLab)
- PID (Controller Proportional, Integral, Derivative)
- PLOT (drawings, in development. Not active in this release)
- PUMPS (pumps start and stop according to water elevations, see example and specific documentation)
- PRINT (writing of the variables U, Y, YT and Z chosen on the .LST file)
- QSCP (typical scenario of irrigation canal offtake withdrawals)
- SCADA (simulator of the SCADA inspired by that of the Gignac channel, cf DIGEST above)
- SCILAB (DDE link with SciLab)
- STEPS (automatic generation of steps)
- STOP (interruption of the calculation allowing to mimic a manual regulation)
- USER1 to USER9 (9 modules open to the user, to be programmed in FORTRAN in DLLUSERn.FOR (n = 1 to 9) and then to compile with the Digital or Intel compiler, cf MkDllUser.bat under sic/prog. It will replace the initial empty dll under sic/exe).
- WDLANG (possibility to write a script in WdLanguage of Windev)

Examples are provided for each of these methods.

These control methods use the variables U, Y, YT and Z or sometimes just a sub-part of these variables. For example, the LP method does not calculate a variable U, and does not use the variables Y or YT since it simply computes the norm Lp of the measured variables Z. If the minimum number of these variables is not supplied, an error message will be indicated when the calculation is executed.

The variables used by the different methods are shown in the following table. This table is important because it indicates, for example, that if you use the PID control method, you will have to set the controlled variable as Y variable, not as Z variable.

Method | U (control variables) | Y (controlled variables) | YT (target on controlled variables) | Z (measured variables) | Mono or Multivariable |
AMIL, AVIS, AVIO | X | X: must be given by water depth | | | SISO |
ATV | X | X | X | Option if calculating parameters from Z instead of U. Option

also if ATV is to be applied to a weighted combination of Z such as for a Bival controller | SISO but duplicated if nu>1 |
ATVPID | X | X | X | Option if calculating parameters from Z instead of U | MIMO with successive SISOs sent to ATV then PID |
BIVAL | X | X: a variable Y(1) must be defined, even if it is recalculated from Z(1) and Z(2), but thus makes available the trajectories over time YT | X | X | 2ISO but duplicated if nu>1 |
BOLST | X | | | | SISO but we can have several that share the same parametrization file BOLST.TXT |
BOMAT, BOSCIL | X | | | | MO |
DIGEST | X | X | X | X | MIMO |
DSS | X | X | X | X | MIMO |
FILEX | X | X | X | X | MIMO |
GPC | X | X | X | X | Not active |
LP | | | | X | MI |
LQR | X | X | X | X | Not active |
MATLAB | X | X | X | X | MIMO |
PID | X | X | X | | SISO but duplicated if nu>1 |
PUMPS | X | X | X | | SISO but SI2O if nu=2: the flow is reinjected elsewhere: U(2) = - U(1) |
PRINT | X | X | X | X | MIMO |
QSCP | X | | | | SISO but duplicated if nu>1 |
SCADA | X | X | X | X | MIMO |
SCILAB | X | X | X | X | MIMO |
STEPS | X: All U are moved in the same way | Option: written on LST file | | X: the trigger threshold is calculated on the min of all Z | MIMO |
STOP | X | | | X | MIMO |
USER1 to USER9 | X | X | X | X | MIMO |
WDLANG | X | X | X | X | MIMO |

Chapter 90

General parameters for U

- DUMin DUMax UMin UMax DTU IFU1 IFU2

Format: 'PU=',4(F5.2,1X),F6.0,1X,I1,1X,I1

Example: PU= 0.02 0.20 0.01 2.00 600. 0 0

Where:

- DUMin and DUMax are the minimum and maximum variation, respectively of the U command between T and T+DTU (where DTU is the regulation time step).
- UMin is the minimum value (and Umax the maximum) of the U command (expressed in absolute value whatever the mode is: R, I or A defined below).

Remark: Even in the case of a U in discharge mode Q (except in the case IFU2=4 for a pump) the above DUMin, DUMax, UMin and Umax values are applied to the calculated gate opening.

- DTU is the command time step. If DTU=0, the command time step will be taken equal to the simulation time step DT. You can enter different DTU for each U inside a regulation module, but so far we just use the last one for the entire regulation module.
- Flags IFU1 and IFU2 are useful only in mode orders in flow (B=Q below). They make it possible in this case to choose the way in which the flow Q given by the module of regulation will be translated into opening W applied to the device:
- IFU1 = Flag mode 1 for U -* 0: normal, i.e. action at each DTU, -* 1: action at each DT calculation in mode Q
- IFU2 = Flag mode 2 for U -* 0: normal, -* 1: method of the characteristics in mode Q for DZ1 and DZ2, -* 2: method of the characteristics in mode

Q for DZ2 alone, -* 3: method of the characteristics in mode Q for DZ1 alone, -* 4: pump mode i.e. no conversion into gate opening).

In the case of the use of the modes with correction by the method of the characteristics, the correction is carried out with the formula:

- $dz_1 = \frac{dQ}{L(V-C)}$
- $dz_2 = \frac{dQ}{L(V+C)}$

These parameters are valid until changed.

Chapter 91

General parameters for Y

- DYMax DTY(s) (Format 'PY=',F5.2,1X,F6.0) PY= 0.02 600.

where DYMax is the precision required for the controlled variable. This means that if the difference between Y and the corresponding order YT is inferior to DYMax, the difference is considered equal to 0. This parameter is stored into the P4 common accessible for the USER regulation module, and used in the PID regulation module. This parameter is valid until changed.

DTY is the measure time step for the controlled variables Y. If DTY=0, the measure time step for the controlled variables will be taken equal to the simulation time step DT.

Chapter 92

Targeted values for Y

We can assign some YT targets to the controlled variables Y.

Use mouse right click to : - Create a new targeted value for Y - Delete a targeted value - Copy/Paste a value on to another

On the .REG file, these values are stored by indicating “T”, then the number of the controlled variable Y, the day, hour, minutes and the corresponding value of YT with the format (’ T’,I3,6X,3(I2,2X),F8.3):

T1 0 00 00 0.000 T1 0 04 00 0.200 T1 0 10 00 0.100

Warning: The targeted values at each time step are linearly interpolated between the previous described values (ramps).

Chapter 93

General parameters for Z

- DTZ(s) (Format 'PZ=' ,F6.0) PZ=600.

where DTZ is the measure time step. This parameter is valid until changed.

If DTZ=0, the measure time step will be taken equal to the simulation time step DT.

Chapter 94

Specific Parameters

The specific parameters of the control method depends on the method chosen. They are entered in the “Specific parameters” tab of the regulation edition window.

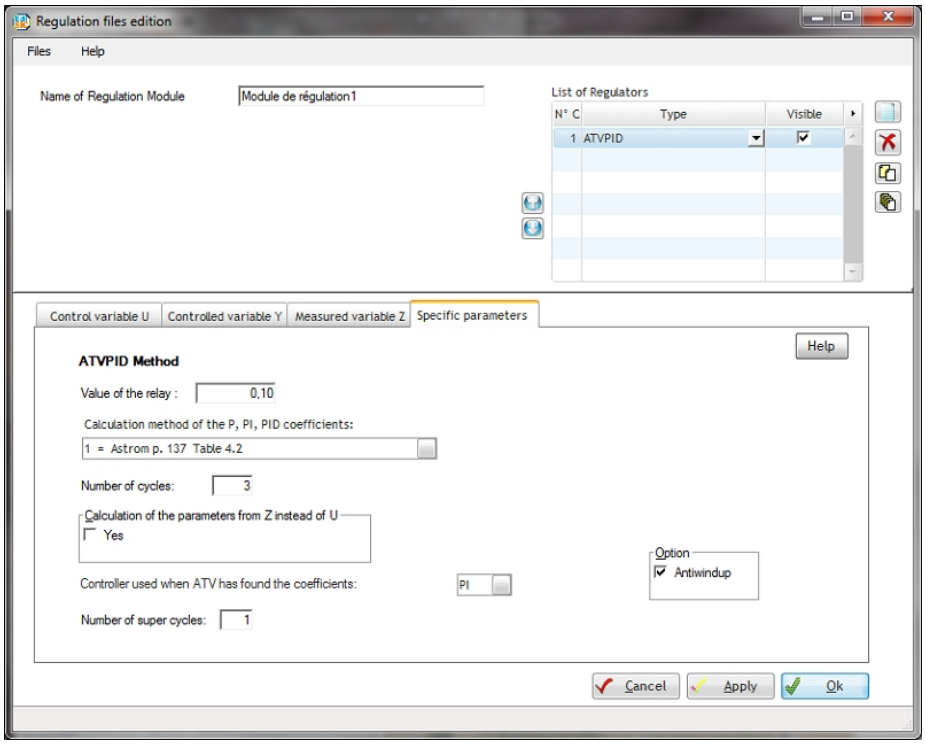


Figure 94.1: editregv5_para

94.1 Definition of specific parameters by text file

It is possible to override these settings in the project using a text file . This is especially useful when you need to redefine parameters of regulation modules on the fly in case of using SIC² in batch mode.

To do this, create a file named `REGPSn.TXT` in the same directory as the XML project file with `n` the number of the controller for which you wish to overwrite specific parameters. This file contains one or more lines starting with "`PS=`" and supplemented by parameters. The format of the parameters of PS lines is defined in the documentation for each controller.

94.2 Example for a PID controller

The control module used for the simulation contains a PID controller. We can redefine the specific parameters of this controller by creating a file named `REGPS1.TXT` in the same folder as the XML project file and this file will contain the following line:

```
PS=    8.01  8036.75     0.00     0.00     0.00
```

This corresponds to a PI controller with coefficients respectively 8.01 and 8036.75 for the proportional and integral parameters.

Chapter 95

Compilation

The compilation and the link edition are done thanks to the batch file MKREG.BAT.

MKREG compiles sirene.exe. It uses the FORTRAN Digital-Compaq-Microsoft compiler (6.1 Version or latter) and the NMAKE utility of this compiler. It seems, from the tests we performed, that we cannot use the Intel Fortran Compiler for the moment since it is not compatible with the MatLab DDE routines.

You must then copy the SIRENE.EXE file in the EXE directory. Please save the initial SIRENE.EXE program under another name or in another directory, before deleting it in case you want to recover it without re-installing SIC again.

Chapter 96

Remarks on regulation modules

If no regulation module is active the Q(t), W(t), Z(t), and L(t) laws are normally read in the (.SIR) file.

Contrary to the 2.1version, the regulation module of the 3.1 version has the priority on the .SIR file. Thus, if the regulation module controls some nodes or singular sections, the Q(t), W(t), Z(t), and L(t) laws of the (.SIR) file will be ignored.

Before beginning a hydraulic calculation, the subroutine LUSERn (n = 1 to 9) is called to read regulation parameters in the (.REG) file. At each calculation time step, before the double sweep, the SUSERns and CUSERns subroutines are called to calculate cross structures openings, manipulations at offtakes to the values and the downstream Z(t) if necessary.

In case of several regulation modules manipulating several times the same control variable U, only the latest one will have an effect in modifying U. Beware of that, for example, if you use a PRINT and PID controller for respecting printing and using a PID module. They should be defined in this order otherwise the PID will have no effect (since it will be the PRINT the latest one, and a PRINT module does not change the value of a U).

Chapter 97

The ATV method

The ATV method (Auto Tuning Variation) is a simple method to automatically calibrate PID controllers without automatic control knowledge, through a simple experiment on the model of the channel, canal or river.

The method applies by default to a variable Y, with its possible setpoint YT. This setpoint trajectory should be used with caution because ATV must be applied in the absence of disturbances other than those generated by ATV, to capture the dynamics to be controlled. But there may be small differences at the start of the transient that we want to remove by moving the variable Y of this gap precisely by using YT. It is also possible to apply the ATV method not on Y(1), but on a linear combination of Z. By default the coefficients of the linear combination are the same (1/number of Z), which gives the arithmetic average. But it is possible to choose other coefficients by writing them, 1 per line, in an ATV.TXT file. This can be useful for stalling the PID coefficients of a Bival regulator for example.

Example of ATV.TXT file:

```
0.4
0.6
```

In this case, the controlled variable will be 0.4*Z(1)+0.6*Z(2)

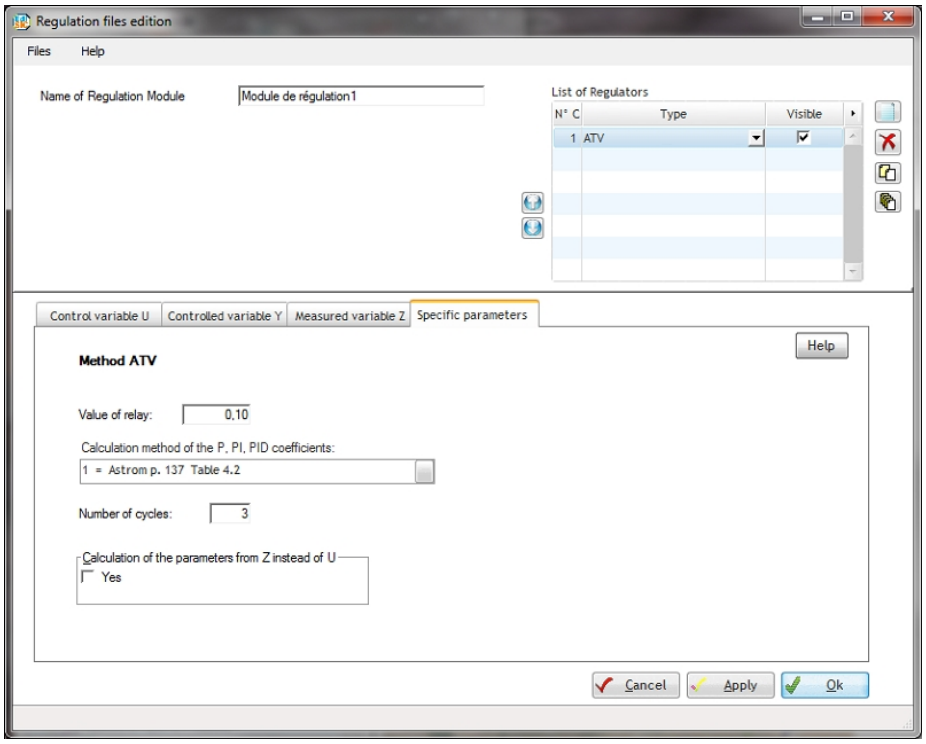


Figure 97.1: editregv5_atv

For the ATV method we read 6 parameters (Format of the line PS : `(5(F8.2,1X),F8.2)`) which are, in the following order:

```
*          R      METH     CYCLE      UouZ        MG        MP
PS=     0.20      1.00      0.00      0.00      0.00      0.00
```

- R is the switch value (in meters, or m^3/s if there is a master-slave controller structure). It is positive for the classical feedback counter-reaction (e.g. downstream control) and negative for inverse configuration (e.g. upstream control).
- METH is the method used to calculate the P, PI and PID coefficients: *-1 = Astrom p. 137 Table 4.2* -2 = Astrom p.141-142 (r=0.5, Phi = 20°) *-3 = Astrom p.141-142 (r=0.41, Phi = 61°)* -4 = Astrom p.141-142 (r=0.29, Phi = 46°) *-5 = Flaus p. 72, Table 2.7* -6 = Flaus p. 72, Table 2.8 slight overshot *-7 = Flaus p. 72, Table 2.8 without overshot* -8 = Cemagref from gain and phase margins
- CYCLE is the number of cycles you want (or manual validation if = 0)
- UouZ allows (si =1) to use the measure Z instead of the control action U to compute Ku (this is usefull in case you have a saturation of the U, or if

you have a limited precision of the actuators and that you are using real data using the SCADA link). In this case you define the corresponding Z as the measure of the gate opening, and this way you will get the real applied gate opening instead of the theoretical one.

- MG is the desired Gain Margin (for option 8)
- MP is the desired Phase Margin (for option 8)

The K, T_i and T_d coefficients of the PID are obtained from K_u and T_u of the ATV method using the following below. T_u is the period of the oscillation cycles. K_u is computed by $K_u = 4\frac{R}{\pi E_{max}}$. If UouZ=0 R is the value of the desired relay and if UouZ=1 R is the value of the relay actually measured on the channel ($= Z(1)$). The values of the coefficients K, T_i and T_d of the PID are then calculated from K_u and T_u of the ATV method as follows:

Astrom p. 137, Table 4.2:

| *K* | *Ti* | *Td* |

|P | 0.5 Ku | | | |PI | 0.4 Ku | 0.8 Tu | | |PID | 0.6 Ku | 0.5 Tu | 0.125 Tu |

Astrom p. 141-142, r=0.5, Phi=20°:

| *K* | *Ti* | *Td* |

|P | 0.5 Ku | | | |PI | 0.47 Ku | 0.4373 Tu | | |PID | 0.47 Ku | 0.4546 Tu | 0.1136 Tu |

Astrom p. 141-142, r=0.41, Phi=61°:

| *K* | *Ti* | *Td* |

|P | 0.5 Ku | | | |PI | 0.1988 Ku | 0.0882 Tu | | |PID | 0.1988 Ku | 1.23 Tu | 0.3077 Tu |

Astrom p. 141-142, r=0.29, Phi=46°:

| *K* | *Ti* | *Td* |

|P | 0.5 Ku | | | |PI | 0.2015 Ku | 0.1537 Tu | | |PID | 0.2015 Ku | 0.7878 Tu | 0.197 Tu |

Flaus p. 72, table 2.7:

| *K* | *Ti* | *Td* |

|P | 0.5 Ku | | | |PI | 0.45 Ku | 0.833 Tu | | |PID | 0.6 Ku | 0.5 Tu | 0.125 Tu |

Flaus p. 72, table 2.8, slight overshot:

| *K* | *Ti* | *Td* |

|P | 0.5 Ku | | | |PI | 0.45 Ku | 0.833 Tu | | |PID | 0.33 Ku | 0.5 Tu | 0.333 Tu |

Flaus p. 72, table 2.8, without overshot:

| K | Ti | Td |

|P | 0.5 Ku | | | |PI | 0.45 Ku | 0.833 Tu | | |PID | 0.2 Ku | 0.5 Tu | 0.333 Tu |

For the method “Cemagref from gain and phase margins” we specify the desired performance in terms of gain and phase margin. When a value is given for a gain margin, the interface automatically offers the maximum phase margin that can be asked. You can decrease but not increase it (or it would lead to aberrant coefficients). Contact us for details or see our scientific publications on this topic.

Bibliographical references:

Cheng-Ching Yu. Autotuning of PID controllers. A relay feeback approach. 2nd Edition. Springer, 2006.

X. Litrico and P.-O. Malaterre. Test of auto-tuned automatic downstream controllers on gignac canal. In USCID conference on SCADA, editor, USCID conference on SCADA, Denver, 2007.

X. Litrico, P.-O. Malaterre, J.-P. Baume, P.-Y. Vion, and J. Ribot-Bruno. Automatic tuning of PI controllers for an irrigation canal pool. Journal of Irrigation and Drainage Engineering, 133:27–37, February 2007.

Chapter 98

The ATVPID method

The ATVPID method consists in running the ATV method, and then apply the PID and then move on to the next device and do the same thing so on all the devices selected for this method ATVPID (if we have therefore several U).

To understand this method it is advisable to read the documentation of the ATV method and on the PID method. A paper details these procedures: https://hal.archives-ouvertes.fr/hal-00450874/document.

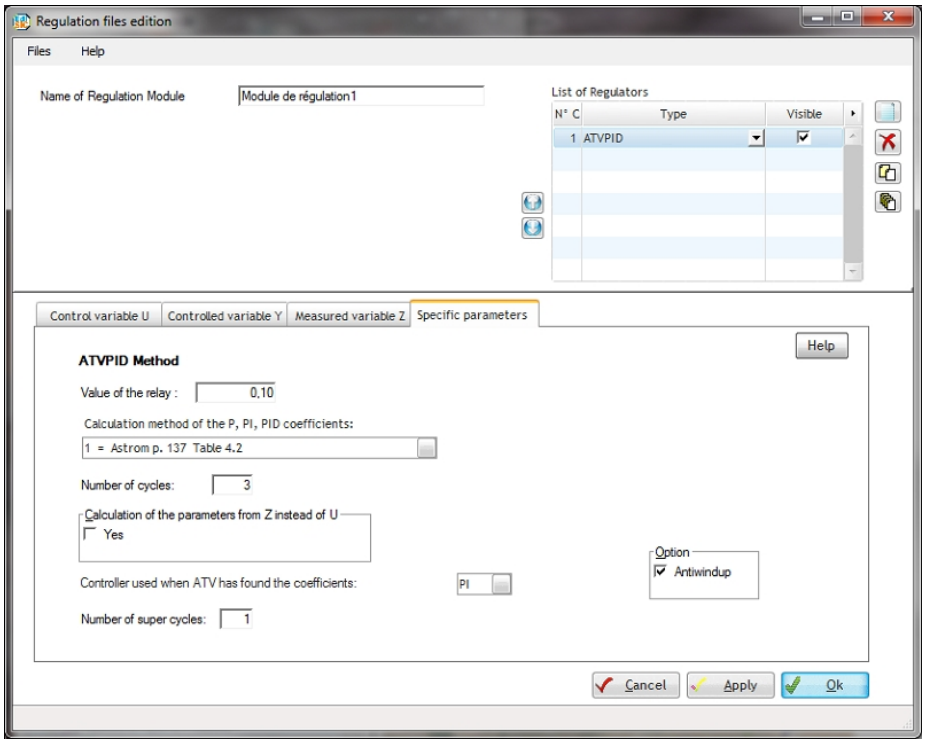


Figure 98.1: editregv5_atvpid

For the ATVPID method we read 8 parameters under the format (7(F8.2,1X),F8.2):

```
*          R      METH     CYCLE      UouZ     CHOIX   AntiWup        MG        MP
PS=     0.20      8.00      3.00      1.00      2.00      1.00     10.00     61.00
```

-*R* is the switch value (by default it is the same for all U if you have several, but can be changed with ATVPID.TXT, cf below). It is positive for the classical feedback counter-reaction (e.g. downstream control) and negative for inverse configuration (e.g. upstream control).

-METH is the method used to calculate the P, PI and PID coefficients ATV: *-1 = Astrom p. 137 Table 4.2* -2 = Astrom p.141-142 (r=0.5, Phi = 20°) *-3 = Astrom p.141-142 (r=0.41, Phi = 61°)* -4 = Astrom p.141-142 (r=0.29, Phi = 46°) *-5 = Flaus p. 72, Table 2.7* -6 = Flaus p. 72, Table 2.8 slight overshot *-7 = Flaus p. 72, Table 2.8 without overshot* -8 = Cemagref from gain and phase margins

- CYCLE is the number of cycles you want (or manual validation if = 0)
- UouZ allows (si =1) to use the measure Z instead of the control action U to compute Ku (this is usefull in case you have a saturation of the U, or if

you have a limited precision of the actuators and that you are using real data using the SCADA link). In this case you define the corresponding Z as the measure of the gate opening, and this way you will get the real applied gate opening instead of the theoretical one.

- CHOIX corresponds to the type of controller we activate when ATV has found the correct coefficients: P (CHOIX=1), PI (CHOIX=2) or PID (CHOIX=3).
- AntiWup is the Antiwindup option as defined for the PID controller (cf. routine CPID).
- MG is the desired Gain Margin (for option 8)
- MP is the desired Phase Margin (for option 8)

As can be understood from a reading of the above parameters to provide, the R relay, by default, is the same for all the control variables U. This is restrictive because there are many cases where it is desired to have different values, for example when tuning automatically in cascade flow controllers and gate opening controllers, or with upstream controllers for certain structures and downstream controllers for the others. Since the release of version 4.29g this is possible by creating a ATVPID.TXT file in the subdirectory of the current data. This file must contain a priori as many lines as U of variables, each line indicating the desired value of the variable relay for the corresponding U. These lines correspond to the U in the same order as shown in the list of the control action variables. If lines are missing or do not correspond to a real value, then the default value from R parameter above will be used.

This file also allows adding timers, ie a time (in seconds) to wait before triggering the next ATV on the following U (all of course in case there are several U).

Example of ATVPID.TXT file:

```
*
*
*
*
0.5
3600.
3600.
3600.
3600.
3600.
```

The second value of the relay (for U(2)) will be the one from the reg file because the line contains a "*" character, which is therefore not a real acceptable value. The relay of the first variable U is -0.11, -0.22 for the third variable, the variable -0.12 fourth and fifth variable +0.33.

The values of the relay for U(1) to U(4) are those from the above parameters $R = 0.2$ since the line contains a "*" character, which is therefore not a real acceptable value. Relay for the fifth variable U(5) will be 0.5.

The temporizations of 1h = 3600s are added, ie that the ATVPID regulation module will wait 1h before launching the next ATV on the variable U that follows.

When running the transient calculation with this module, the results obtained and the corresponding PID coefficients are displayed on the screen in the normal message window (see below), and also on a ATVPIDCOEF.TXT file. The example provided with SIC (ex2) gives, for example:

1.70 | 3598.78 | 0.00 | 0.00 | 0.00 | 2400.00 | 4.59 | 1200.00 | 19.78 |
2.42 | 3598.78 | 0.00 | 0.00 | 0.00 | 2400.00 | 6.54 | 1200.00 | 31.30 |
3.85 | 3598.78 | 0.00 | 0.00 | 0.00 | 2400.00 | 10.41 | 1200.00 | 35.26 |
3.01 | 3598.78 | 0.00 | 0.00 | 0.00 | 2400.00 | 8.13 | 1200.00 | 29.83 |
6.99 | 4498.48 | 0.00 | 0.00 | 0.00 | 3000.00 | 18.91 | 0.00 | 0.00 |

The columns correspond, respectively to: K_p, T_i, T_d, N (filter), Antiwindup (0 if not actif or 1 if actif), T_u, K_u, $\widetilde{T_u}$ $\widetilde{K_u}$ (for the report downstream-upstream, cf bibliographical references bellow for the details of the équations)

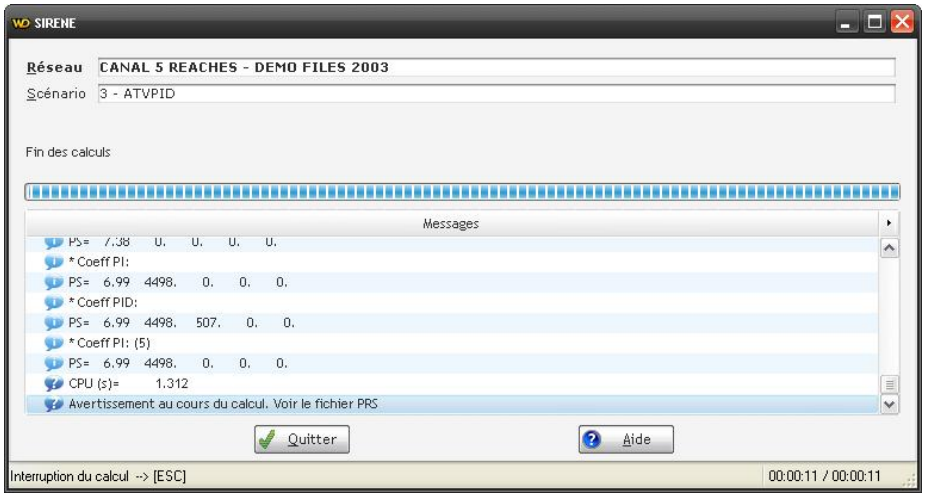


Figure 98.2: doc76

Chapter 99

The BIVAL method

- ALPHA at format (F8.4) PS=0.5
- K Ti Td N at format (3(F8.2,1X),F8.2) PS=0.23 1245.2 0.00 5.00

ALPHA is the weighting coefficient between Z(1) and Z(2):

Y1(1)=ALFA*Z(1)+(1-ALFA)*Z(2)

On this variable Y1(1) we then apply a PID controller with the K, Ti, Td and N provided coefficients.

Chapter 100

The BOLST method

The BOLST method is a method for imposing U command changes from a .txt text file (eg, ASCII or ANSI text format). These changes are indicated according to the time at which they are desired. The time is not necessarily at regular time steps. The values will then be used in step mode (visible at the following time step, as for a law depending on the time entered at the interfaces: it is applied at time T and it is visible in the results at time T+DT, so if we want to see it at time T we have to shift it from DT, with Tlag for example) or in ramp mode (linear interpolation, in this case the values will be visible in the results at the indicated times). These are therefore modifications of the Open Loop type, since they are not calculated from controlled variables Y nor from measured variables Z.

100.1 Create a .txt command file in ASCII or ANSI or other text format

The text file .txt produced must contain the application times of the command in the first columns and the value of the various U commands in the last column. The formats of time and control are relatively free, or in any case can be indicated with an free format (see below) but must be indicated in FORTRAN format in a BOLST.TXT parameterization file. This time is indicated however in days, months, years, hours, minutes, seconds, even if the format can be chosen. Caution, reading in Fortran format is relatively strict, it is necessary to respect the columns indicated in the format, and decimal points preferably. If you are not used to these formats, it is better to read a documentation on the subject. Otherwise the module may have crashes, or unwanted behaviors.

Various BOLST controllers, each monovariable, can be used, and thus acting on a single control U, but the parameterization file BOLST.TXT is common to all these BOLST controllers, with a configuration line for each BOLST controller.

An example is provided under dat/ex8_bolst, with 2 BOLST modules. One of the 2 files acting on a U command in discharge (file QPRISE_02092001.TXT) contains 4 changes of values:

```
02:09:2001:00:00:00:0.000
02:09:2001:05:00:00:-.100
02:09:2001:15:00:00:0.100
02:09:2001:21:00:00:0.300
```

Be careful not to leave empty lines at the bottom of this file, otherwise these lines will be read as if all the numbers are at 0, which will give T=0 and a-priori will create an error message according to which the times are not increasing (or empty line).

The format of the dates and times here is DD:MM:YY:HH:MM:SS, and the variable U in F10.3 format, as indicated on the corresponding line of the BOLST.TXT file with the FORTRAN standard format notations: (2(I2,1X),I4,1X,3(I2,1X),F10.3)

This file must be saved in a named file with a name of its choice, which will be specified in the BOLST.TXT file. In the example provided, there are 2 files: `QPRISE\_02092001.TXT` and `HCRUE\_G\_02092001.TXT` located in the same folder as the project XML file. But even if you specify it in the BOLST.TXT file, these files can also be located elsewhere, in one or more other subdirectories, with the command "DIR=":

```
% This line is optional and ignored if it starts with DIR=DEF
DIR=DEFAUT
```

But, if you specify a subdirectory with the code "DIR=", then the following files read lower in the BOLST.TXT file will be searched in this other subdirectory. It is necessary to put the full name C:/ ..., or it is possible that it also accepts the relative paths relative to the subdirectory of the data ..DAT2 ! It is also possible that under some OS it is necessary to double the \. To test if necessary … You can put as much command "DIR=" as you want in a file BOLST.TXT, and so the following files will be searched and read in this or these subdirectories.

```
% In this case the data files of the BOLST modules (QPRISE\_02092001.TXT and HCRUE\_G\
DIR = C:/DATA/SICBOLST
```

The latter case allows several SIC projects located in different subdirectories to read the same data files for BOLST modules in one or more subdirectories common to these various SIC projects.

Warning: for versions of Fluvia and Sirene programs compiled with the Digital compiler, the same .TXT file can not be opened simultaneously several times by several modules. In this case it is necessary to duplicate it with different names used by the corresponding BOLST modules (or use a version of Fluvia or Sirene compiled in Intel, this is indicated in the selection of the desired version).

100.2 Configuring the BOLST module in SIC

To configure a BOLST module in SIC, you must:

- Select the BOLST module in the editing window of the control modules,
- Choose the (only one, otherwise the others will be identical to the first one and only one read on the corresponding .txt file) command U. The Y, YT and Z variables are not used in this module because it is a module of Type Open Loop,
- Define the specific parameters of the BOLST method.

100.3 Specific parameters of the BOLST module

The specific parameters are the same as those of the BOMAT or BOSCI method: - Scaling factor “a” (multiplicative factor, or called “homothety”) - Additive constant “b” (then U = ax + b) - Ramp or Step Mode - Write or not the data of the file .txt read on the file Nom_Sirene.lst

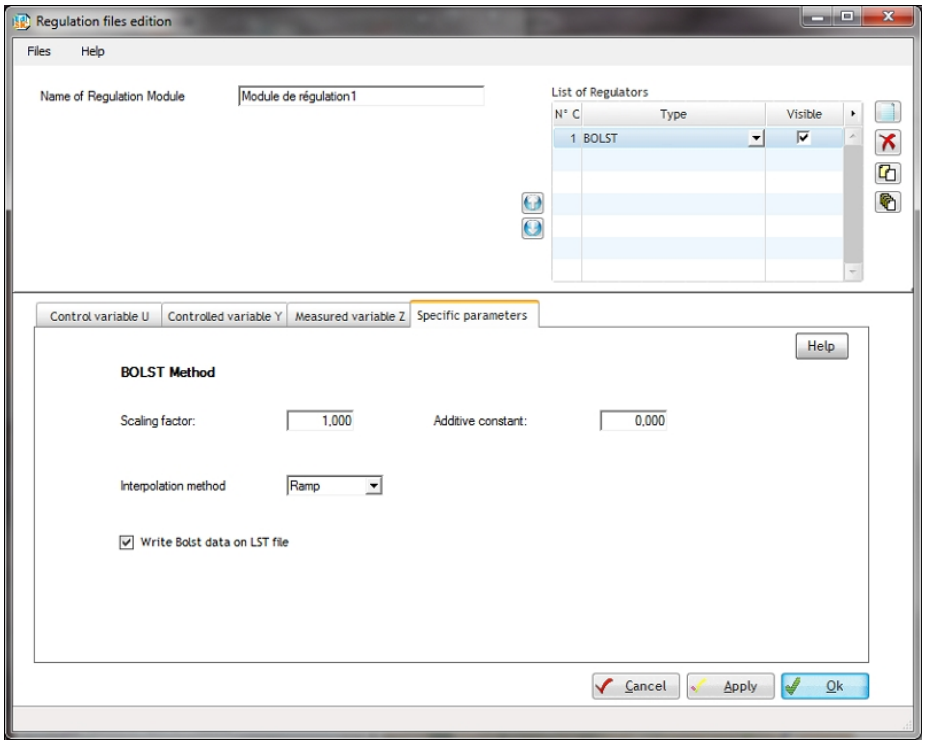


Figure 100.1: editregv5_bolst

100.4 Scaling Parameter

This is a multiplier that will be applied to the values in the .txt file before the command is applied.

A second multiplier can be added in the BOLST.TXT file which will multiply to that one to obtain a global modified factor. These coefficients will be written and indicated on the file Nom_Sirene.lst

100.5 Constant additive

This additive constant will be added to the value read, possibly modified by the multiplicative factor mentioned above. The variable U injected into the model will therefore be of the form U = ax + b where a is the multiplicative factor, b the additive constant and x the value read on the .TXT file (or rather interpolated for the current time in ramp mode) .

100.6 Interpolation method

The two available interpolation methods are the same as those of the laws of time.

100.7 Write to LST file

The data read from the .TXT files can be written to the File_Name.lst file to see if it is read correctly. In this case, the format is as follows:

```
Number of the BOLST    T-Tlag    Tbefore    Tafter    U(1)
BOLST n°    1    -3000.          0.       3600.       2.500
BOLST n°    2   -13800.      -3600.      14400.       0.000
```

The Number of the BOLST is the number as indicated in the BOLST file (end of the line, cf. bellow). Tbefore and Tafter are the 2 times played in the .TXT file on 2 consecutive lines that frame or will frame the current time possibly offset by the delay time Tlag: T-Tlag

100.8 Restriction

One can have several controllers of this type (BOLST) for a simulation. Each BOLST is monovariable. All BOLST methods retrieve the file name and some parameters on a common BOLST.TXT file:

```
% We can put comment, with lines beginning by %, ! or /
DIR=DEFAUT               % This line is optional and ignored if starts by DIR=DEF
TLAG=3600.               % This line is optional. It gives a time shift (in s) to appl
```

```
% From version 5.38c3 the scaling factor format changes from 4.1 to 7.3
% The format is then : FORMAT(A20,1X,A36,1X,F7.3,1X,2(I2,1X),I4,1X,3(I2,1X),I2)
HCRUE\_G\_02092001.TXT      (2(I2,1X),I4,1X,3(I2,1X),F10.3)   1.000 02/09/2001 00:00:00  1
% Time Lag can be different for each BOLST module (as DIR also)
TLAG=14400.               % This line is optional. It gives a time shift (in s) to apply to the
QPRISE\_02092001.TXT       (2(I2,1X),I4,1X,3(I2,1X),F10.3)   1.000 02/09/2001 01:00:00  2
```

TLAG= allows to shift the read values in time. If TLAG > 0 a delay is added. If TLAG < 0 change of value is anticipated. One can have several lines of definition of Tlag which will then apply to the files of the BOLST indicated following this line.

In the BOLST.TXT file, there are also as many rows of BOLST modules used to define the parameters of these modules. The format of this file must be respected. In particular it must be found on each line (illustrated in the example provided):

```
This line is read in the format: (A20,1X,A36,1X,F7.3,1X,2(I2,1X),I4,1X,3(I2,1X),I2)
QPRISE\_02092001.TXT: the file name up to 20 characters
(2(I2,1X),I4,1X,3(I2,1X),F10.3): the read format with up to 36 characters
1.000: an additional scaling factor which will multiply to that of the specific parameters of thi
02/09/2001 01:00:00: a reference time, which will be subtracted from the times read on the lines
2: the number of the BOLST module, since we can have several ones, in format I2
```

The number of the BOLST module is the number in the order of the control modules defined in the SIC interfaces. They can be in the natural order 1 to n, which is the most classic or in any order if for one reason or another it is simpler so. We can also put the number 0, for all, in this case they will be applied in the order of the lines read.

In the BOLST.TXT file you can put comment lines that begin with the characters %, ! or / in the first column.

The version of Fluvia and/or Sirene used is free except the above notice on Digital or Intel to have the possibility to use the same .TXT file several times by different BOLST modules.

The formats of the BOLST.TXT file and other files containing the data must be stricktly respected since Fortran programs Fluvia and Sirene are strict when reading such files.

Soon, the date and hour format will be able to be choosen among a large list of standart formats, such as for Matlab for example:

Table 1: Standard MATLAB Date format definitions

Number	String	Example
0	'dd-mmm-yyyy HH:MM:SS'	01-Mar-2000 15:45:17
1	'dd-mmm-yyyy'	01-Mar-2000
2	'mm/dd/yy'	03/01/00

```
  3             'mmm'                   Mar
  4             'm'                     M
  5             'mm'                    03
  6             'mm/dd'                 03/01
  7             'dd'                    01
  8             'ddd'                   Wed
  9             'd'                     W
 10             'yyyy'                  2000
 11             'yy'                    00
 12             'mmmyy'                 Mar00
 13             'HH:MM:SS'              15:45:17
 14             'HH:MM:SS PM'            3:45:17 PM
 15             'HH:MM'                 15:45
 16             'HH:MM PM'               3:45 PM
 17             'QQ-YY'                 Q1-96
 18             'QQ'                    Q1
 19             'dd/mm'                 01/03
 20             'dd/mm/yy'              01/03/00
 21             'mmm.dd,yyyy HH:MM:SS'  Mar.01,2000 15:45:17
 22             'mmm.dd,yyyy'           Mar.01,2000
 23             'mm/dd/yyyy'            03/01/2000
 24             'dd/mm/yyyy'            01/03/2000
 25             'yy/mm/dd'              00/03/01
 26             'yyyy/mm/dd'            2000/03/01
 27             'QQ-YYYY'               Q1-1996
 28             'mmmyyyy'               Mar2000
 29 (ISO 8601)  'yyyy-mm-dd'            2000-03-01
 30 (ISO 8601)  'yyyymmddTHHMMSS'       20000301T154517
 31             'yyyy-mm-dd HH:MM:SS'   2000-03-01 15:45:17
```

Table 2: Free-form date format symbols

```
Symbol  Interpretation of format symbol
=====================================
yyyy    full year, e.g. 1990, 2000, 2002
yy      partial year, e.g. 90, 00, 02
mmmm    full name of the month, according to the calendar locale, e.g.
        "March", "April" in the UK and USA English locales.
mmm     first three letters of the month, according to the calendar
        locale, e.g. "Mar", "Apr" in the UK and USA English locales.
mm      numeric month of year, padded with leading zeros, e.g. ../03/..
        or ../12/..
m       capitalized first letter of the month, according to the
        calendar locale; for backwards compatibility.
dddd    full name of the weekday, according to the calendar locale, eg.
        "Monday", "Tueday", for the UK and USA calendar locales.
```

```
ddd     first three letters of the weekday, according to the calendar
        locale, eg. "Mon", "Tue", for the UK and USA calendar locales.
dd      numeric day of the month, padded with leading zeros, e.g.
        05/../.. or 20/../..
d       capitalised first letter of the weekday; for backwards
        compatibility
HH      hour of the day, according to the time format. In case the time
        format AM | PM is set, HH does not pad with leading zeros. In
        case AM | PM is not set, display the hour of the day, padded
        with leading zeros. e.g 10:20 PM, which is equivalent to 22:20;
        9:00 AM, which is equivalent to 09:00.
MM      minutes of the hour, padded with leading zeros, e.g. 10:15,
        10:05, 10:05 AM.
SS      second of the minute, padded with leading zeros, e.g. 10:15:30,
        10:05:30, 10:05:30 AM.
FFF     milliseconds field, padded with leading zeros, e.g.
        10:15:30.015.
PM      set the time format as time of morning or time of afternoon. AM
        or PM is appended to the date string, as appropriate.
```

Chapter 101

The BOMAT method

The BOMAT method is a method for imposing U command changes from a .mat file in MatLab format. These changes are indicated according to the time at which they are desired. So these are Open Loop modifications.

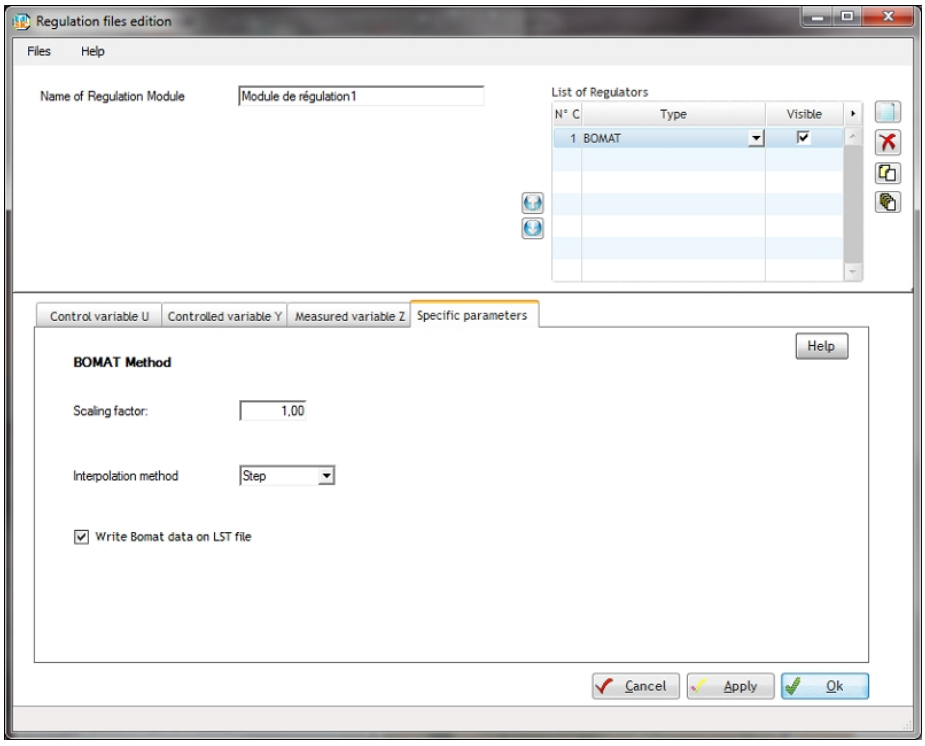


Figure 101.1: editregv5_bomat

- SCALING INTERPOL WRITE at format (F8.2,1X,F8.2,1X,F8.2) PS=1.0 0. 1.

The file name is arbitrary because it can be selected from the list of .mat files. If a Bomat.mat file exists then it will be chosen without requiring interfaces. The name of the matrix must be U if possible. Otherwise there will be a warning on the screen and on the .LST file. But even in this case, the first matrix found will be read and the program will continue. The matrix must include the times in seconds in the first line and the various U at these instants in the following lines. The file must be created by MatLab, for example by a command:

save Bomat.mat U;

We read a scaling parameter, an interpolation parameter, and a flag to write or not the data of the matrix .mat on the file Nom_Sirene.lst, on the .xml file on the tag line PS=

READ(U,‘(F8.2,1X,F8.2,1X,F8.2)’,ERR=10) PARA(1),PARA(2),PARA(3)

If the interpolation parameter PARA(2) = 0 then the U changes are made by step, if PARA(2) = 1 then the U changes are done by ramps (linear interpolation).

Attention you can only have one module of this type (BOMAT).

At the output of reading, we have: PARA(4)=NU PARA(5)=NT

The BOMAT module can handle the U variables in absolute and relative mode, but not in incremental mode (there will be no error message but the U command will be applied at all intermediate instants between two time values, which is not a priori desired in incremental mode).

A DDE link version with Matlab must be selected as an option for this module to be active. The programming of this module uses routines provided by MatLab. If this is a problem we can program a reading of .mat files in the -V4 format which can be done in direct binary reading without using internal MatLab routines. The very close BOSCIL method for matrices in Scilab .dat files does not require a Scilab DDE link version, but just a compiled version with the Intel or Digital compiler, which we provide (see options).

Chapter 102

The BOSCIL method

The BOSCIL method is a method for imposing U command changes from a Scilab binary file. These changes are indicated according to the time at which they are desired. So these are Open Loop modifications.

102.1 Create a command matrix with Scilab

The matrix produced in Scilab must contain the times of application of the command in the first line and the value of the various commands U in the following lines.

Example to create two commands U which take values 5, 10, 5 and 4, 2, 4 to 0, 1 and 2 hours respectively:

```
U=[0, 3600, 7200 ; 5, 10, 5; 4, 2, 4]
```

The name of the variable does not matter. This must be saved in a file named `BOSCILAB.DAT` located in the same folder as the project XML file using the command:

```
save("Directory of the XML project/BOSCILAB.DAT",U)
```

102.2 Configuring the BOSCIL module in SIC

To configure a BOSCIL module in SIC, you must:

- Select the BOSCIL module in the editing window of the regulation modules
- Define a number of control variables U corresponding to that of the matrix created under Scilab
- Define the specific parameters of the BOSCIL method

102.3 Specific parameters of the BOSCIL module

The specific parameters are the same as those of the BOMAT method: - Scaling factor (multiplicative factor, or even called homothety) - Ramp or Step mode - Write or not the data of the matrix read on the file Nom_Sirene.lst

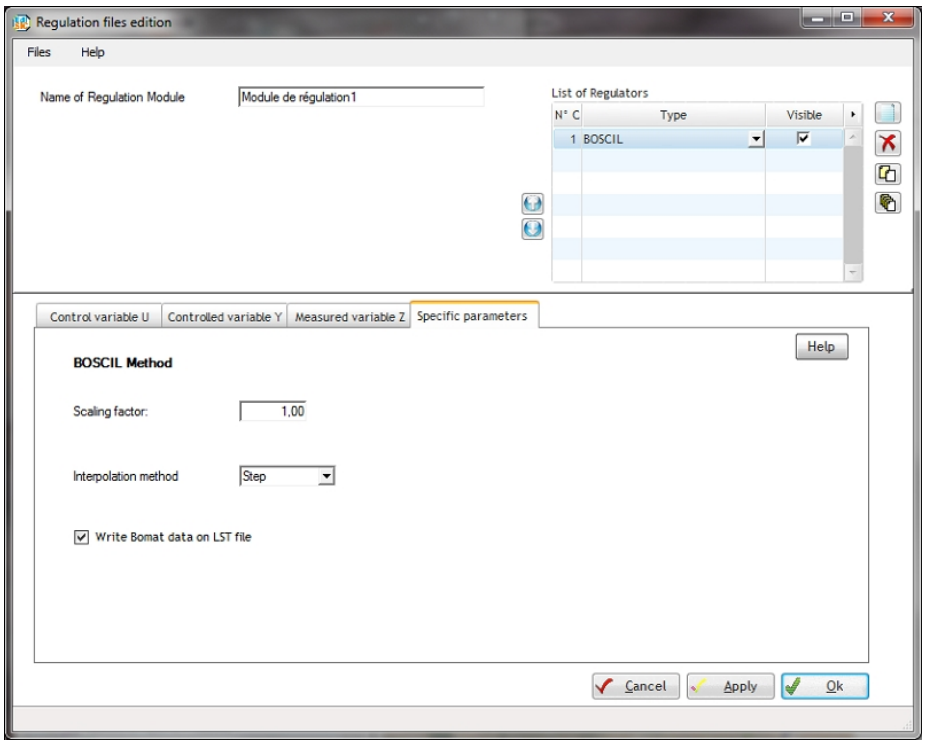


Figure 102.1: editregv5_boscil

102.4 Scaling Parameter

It is a multiplier that will be applied to the values present in the matrix before the command is applied.

102.5 Interpolation method

The two available interpolation methods are the same as those of the time laws.

102.6 Write to LST file

The data read on the BOSCILAB.DAT matrix in Scilab format can be written to the file Name_Sirene.lst, to check if it is read correctly.

102.7 Restriction

You can only have one module of this type (BOSCIL) for a simulation.

The version of Fluvia and/or Sirene used is not required to contain the DDE Scilab link. Before version 5.36c included, however, a version compiled with the Fortran Intel compiler was required (in the Matlab or Scilab DDE link options, the compiler used is indicated for the corresponding version). Starting with version 5.36d this option is available for all versions of Fluvia and Sirene, compiled with Fortran compilers Intel or Digital, with or without DDE link Matlab and/or Scilab.

Chapter 103

The DIGEST method

The DIGEST method is a method that simulates the DSA digester on the Gignac canal. This method can for example be used by a Sirene instance while another instance is working with the SCADA module. The Digester module is a module retrieving data measured on the hydraulic system studied, and writes them in a Data.txt file. It waits in return for a Csg.txt file in which there are instructions to apply to mobile devices. Once this file read it is destroyed, to wait for the next one.

In order to avoid writing problems in the Data.txt file, knowing that a program may be waiting for it to compute a command and then destroy it, an intermediate Data0.txt file is first generated, then once completed and closed is renamed to Data.txt.

This method is managed by the editor of Edisic's control modules.

The first parameter to provide is the time (in s) between each generation of data.txt files. If we put a time lower than the time used in the instance of Sirene with the SCADA module, we will have an acceleration compared to the real time. This avoids spending too much time in front of the computer while having the same results (at least as long as the algorithms do not use the real time variable).

The second parameter is the full name of the REGULATOR subdirectory that will be used with its subdirectory structure described in additional documentation. Before version 5.38c7 this name was necessarily C:/REGULATEUR. Since version 5.38c7 it is possible to change this name.

Chapter 104

The DSS method

The DSS method is a method for testing any LTI (linear time invariant) controller given by the matrices of its representation in the state space. These matrices (A, B, C and D) are provided in a .Mat file in MatLab format. This method therefore makes it possible to test controllers of the Hinfini, H2, l1, etc. type.

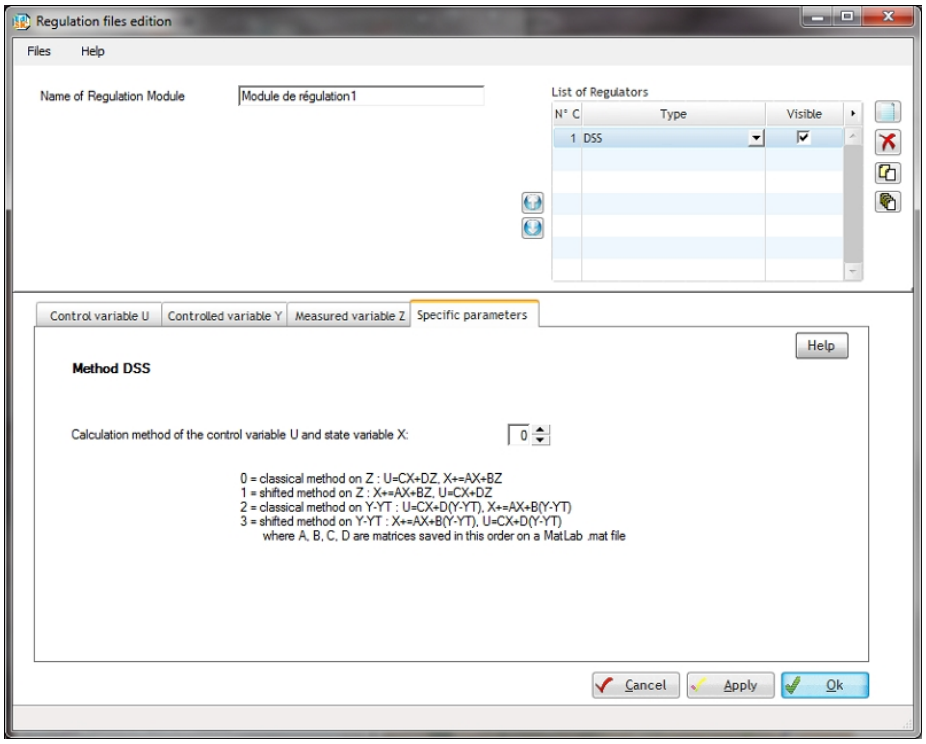


Figure 104.1: editregv5_dss

For the DSS method, one reads a parameter in the format (F8.2) : * N PS= 1.

-* N=0 for the classical method on Z (U=CX+DZ, X+=AX+BZ) -* N=1 for the shifted method on Z (X+=AX+BZ, U=CX+DZ) -* N=2 for the classical method on Y-YT (U=CX+D(Y-YT), X+=AX+B(Y-YT)) -* N=3 for the shifted method on Y-YT (X+=AX+B(Y-YT), U=CX+D(Y-YT))

Notes: The file name is arbitrary because it is selected in the .mat files list The names of the matrices must be A, B, C, and D and must be in this order in the .mat file

The file has therefore to be created by a command:

save Gains.mat A; save Gains.mat B -APPEND; save Gains.mat C -APPEND; save Gains.mat D -APPEND;

At the output of reading, we have:

PARA(1)=NX (Number of states) PARA(2)=NU (Number of control variables) PARA(3)=NZ (Number of measured variables)

Chapter 105

The FILEX method (FILe EXchange)

The FILEX method is a method similar to that of digesteur. It allows to send to a third-party application the data collected on the model (the commands U, the controlled variables Y, the setpoints YT and the measures Z) and to retrieve commands U to be applied to the system and calculated by the third application.

At each regulation time step, the module writes the SIC_U.txt, SIC_Y.txt, SIC_YT.txt, SIC_Z.txt files comprising, respectively, the commands, the controlled variables, the setpoints and the measures. The write format is `F12.5` with one line per variable.

After writing these 4 files, the module writes a file SIC_GO.txt indicating to the third-party application that it can read the 4 data files. The third-party application must then delete the SIC_GO.txt file, read the data files, overwrite the SIC_U.txt file with the new commands and write a SIC_OK.txt file to tell SIC that it can read the SIC_U.txt file.

During this time, the FILEX module checks the presence of the SIC_OK.txt file, and if it is present, it erases the SIC_OK.txt file and reads the SIC_U.txt file. The commands read in the file are then used as is.

This module contains two specific parameters: - The waiting time between each loop verifying the presence of SIC_OK.txt (by default 0.1 seconds) - The maximum waiting time to check the presence of SIC_OK.txt. If the file is not present after this time, the simulation is stopped (by default 120 seconds)

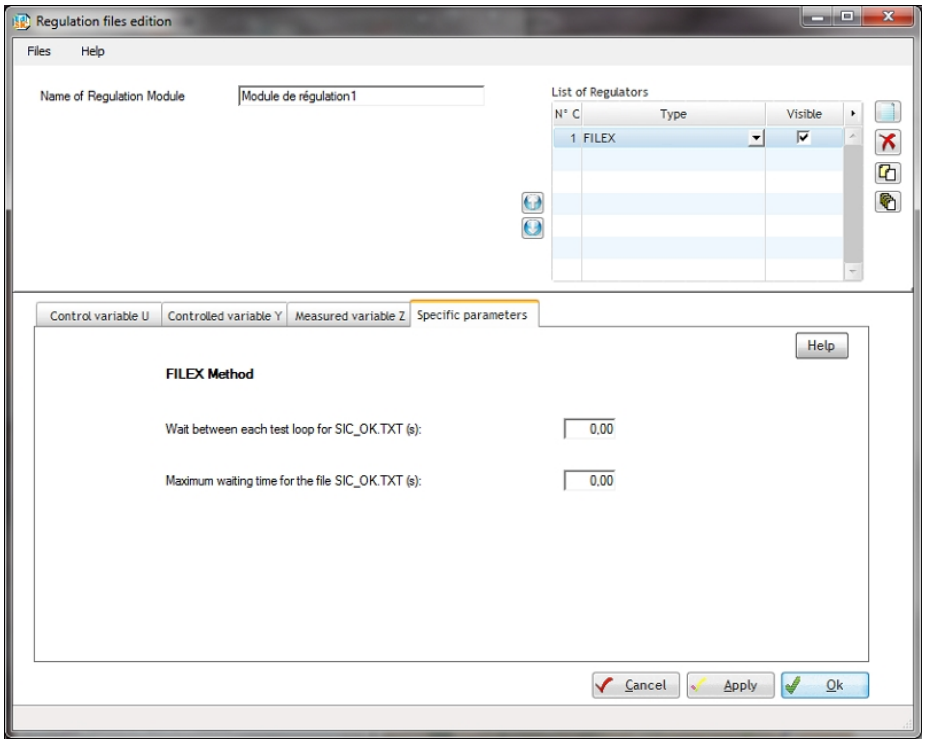


Figure 105.1: editregv5_filex

Example of a SIC_Y.txt file produced by the FILEX module for 4 variables controlled and read by the third-party application:

```
98.45955
98.60769
99.01830
99.61246
```

Example of a SIC_U.txt file produced by the third-party application for 4 control variables and read by the FILEX module:

```
0.248462
0.284745
0.195793
0.177652
```

Chapter 106

The LP method

This method does not calculate U control action variables to be applied to structures, but calculates the Lp-norm of a chosen Z measured variable.

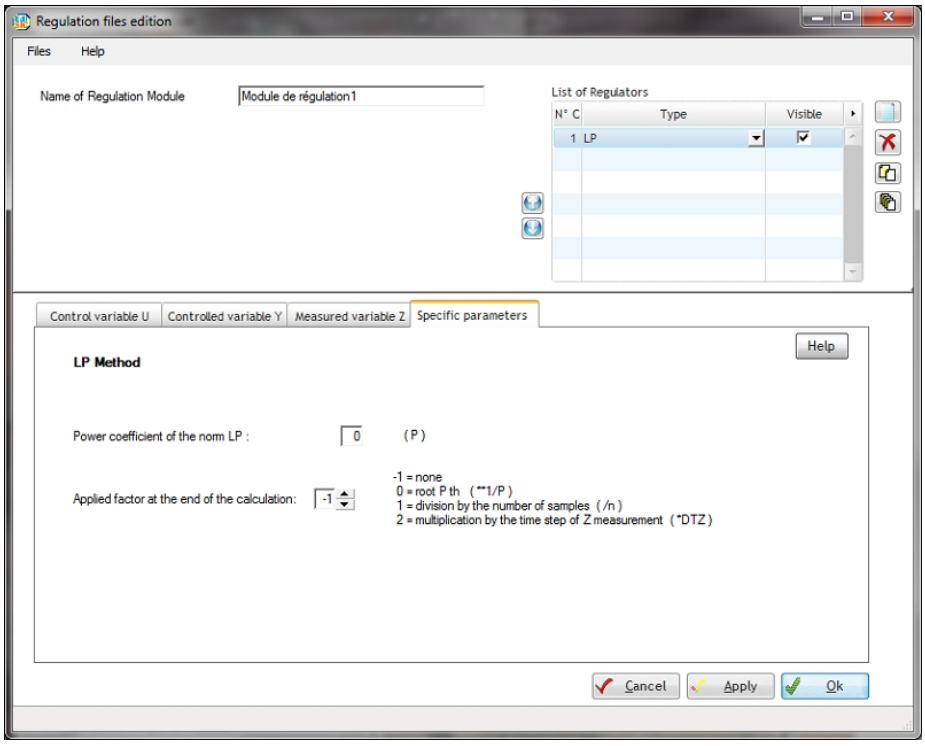


Figure 106.1: editregv5_lp

For the LP method we read 2 parameters in the format (F8.2,1X,F8.2): * P N PS= 2. 1.

P: power coeff of the LP norm N: flag for different options (0: **1/P, 1: /n, 2:*DTZ, otherwise no modification)

```
C-----Calculation of the power p, for the norm LP:
C-----|Z|lp  = (Sum((Z)**p))**(1/p) (for p>0 integer)
C-----|Z|inf = Max(abs(Z)) (for p=0)

IF(IP.NE.0) THEN
     PARA(3)=PARA(3)+ABS(Z(I))**IP
ELSE
     PARA(3)=MAX(PARA(3),ABS(Z(I)))
ENDIF
```

The flag N is used at the end of the calculation:

```
IF(INT(PARA(2)).EQ.0) THEN
C------Classical norm LP
     IF(IP.NE.0) PARA(3)=PARA(3)**(1./IP)
ELSEIF(INT(PARA(2)).EQ.1) THEN
C------We divide by the number of samples
     IF(PARA(4).NE.0) PARA(3)=PARA(3)/PARA(4)
ELSEIF(INT(PARA(2)).EQ.2) THEN
C------Standard used for PID optimization for example
     PARA(3)=PARA(3)*DTZ
ENDIF
```

Chapter 107

The MATLAB method

The MATLAB method (of the SIRENE or FLUVIA calculation program of SIC) starts the Matlab program at the initial time t=dt opening a DDE link and permits you to execute a .m Matlab file at each regulation time step dtu for this module (dtu can be a multiple of the numerical time step dt).

The Matlab .m script file will be used as detailed hereafter. The name of this file can be specified in the specific parameters of the method MATLAB (pidsic.m in the example provided in dat/exemple/matlab). It may indicate that name with either .m extension (ex .: pidsic.m) or without this extension (ex .: pidsic). In the case where the extension is not present it will be automatically added. If this file is found in the current sub-directory where the xml file of the current project is, then it will be the one used without offering the user to choose another one, which facilitates the user's task. If this file does not exist in the current subdirectory, then a file can be selected from .m existing files in the data subdirectory. If no name is specified, then a default name is generated and sought: module2.m, module3.m, etc., or a generic module"n".m, where "n" is the controller number, starting usually at 2 (for the index 1 is used for the laws function of time, which are managed also internally as a regulator if necessary).

Be careful not to use a name for this .m file already used by an intrinsic Matlab function (example pid.m). With some versions of Matlab this does not pose any problem (ex 7.1) but with some other this will generate an error (ex Matlab 2010b).

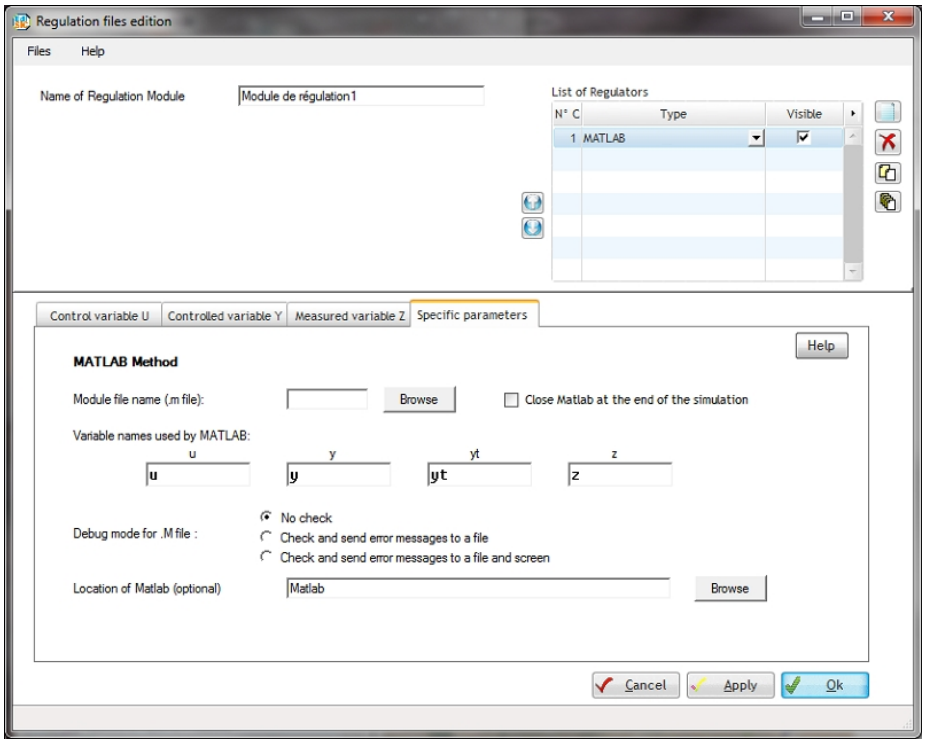


Figure 107.1: editregv5_matlab

In the interface when calculating steady flow scenarios (FLUVIA Program) or transient scenarios (SIREN Program), the names of these .m files from the project (xml file), automatically generated or selected by the user are shown.

At this initial time step (t=dt) the MATLAB method (of the SIRENE or FLUVIA calculation program of SIC) sends to the Matlab environment (workspace) : the calculation time step dt, the regulation time step dtu, the final time tfin and an “interf” flag allowing to manage the interfaces in the .m file according to the interface mode as defined in the option menu.

For this and every later time step t dt (multiple of dtu, see the discussion of dtu and dt), MATLAB method sends to the Matlab environment (workspace) controlled variables y, corresponding targets yt, measured variables z, current control variable positions u and current time t. The names of these variables can be redefined. These variables are vectors since they can be multiple. In Scilab the vectors are by defaut columns (size(u) = [nu 1]), contrary to Matlab where they are lines (size(u) = [1 nu]). These vectors y, yt and z are those of the current instant t. The vector u is that calculated at the current time t and which will be applied and effective from the moment of following calculation t + dt. If you want to have the history of these vectors it is for you to store them

in variables in the Matlab memory space (the variables are stored and saved in memory between 2 calls), or possibly on a file.

The SIRENE (or FLUVIA) program then executes the selected Matlab .m file (at each time step t=n.dtu, n=1 to nmax such as t=nmax.dtu=tfin).

The SIRENE (or FLUVIA) program then retrieves the control action variables u from the Matlab workspace (as calculated by the .m file) and applies them into SIRENE (or FLUVIA).

This MATLAB regulation method permits you to write and test quickly a regulation method while taking advantage of all Matlab development environment. In addition, it does not require any compilation or link edition of the SIRENE (or FLUVIA) program. On the other hand, the regulation module execution is slower because of data exchanges between SIRENE (or FLUVIA) and Matlab, and because a .m file is interpreted by Matlab and not compiled directly in executable code. You can check this execution time since we provide the same example with the intrinsic PID regulation module, the Scilab and the MatLab DDE link and the wdlang module.

Example of .m file (see example provided under dat/ex4_matlab):

```
% file corresponding PIDSIC.M to a controller PID
% initialisation
% Matlab and Scilab manage vectors differently
% In Matlab, by default, a vector is a line (size (u) = 1 nu)
% In Scilab , a default vector is a column (size (u) = nu 1)
% To avoid unnecessary duplication of indexing, we adapt to that convention
% Matlab and Scilab scripts are then slightly different

if t == dt
   eold=0;
   se=0;
   de=0;
   [n1 nu]=size(u);
   vu=zeros(n1,nu)+1;
   [n1 ny]=size(y);
   vy=zeros(n1,ny)+1;
end;

% useful coefficient calculation
------
Kp=[14.78 1.44 4.22 2.51 1.53];
Ti=[1440 1680 1200 1440 1440];
Td=zeros(n1,nu);
N =zeros(n1,nu);
for i=1:nu,
   if Ti(i)==0;
```

```
      Ki(i)=0;
   else
      Ki(i)=Kp(i)/Ti(i)*dt/2;
   end
   if Td(i)==0;
      Kd1(i)=0;
      Kd2(i)=0;
   else
      if N==0
         Kd1(i)=0;
         Kd2(i)=Kp(i)*Td(i)/dt;
      else
         Kd1(i)=1/(1+N(i)/Td(i)*dt);
         Kd2(i)=Kp(i)*N(i)*Kd1(i);
      end;
   end;
end;

% calculation of the vectorial pid
------
e=yt-y;
se=se+Ki.*(eold+e);
de=Kd1.*de+Kd2.*(e-eold);
u=Kp.*e+se+de;

% old deviation saving
------
eold=e;

% drawing of y and u
------
if interf == 1
   if t == dt
      subplot(211)
      xlabel('time (h)')
      ylabel('y, yt')
      axis([0 25 -.5 .5])
      title('Outputs, Targets');
      subplot(212)
      xlabel('time (h)')
      ylabel('u')
      axis([0 25 -1.5 1.5])
      title('Commands');
   end;
   subplot(211),plot(t*vy/3600,y,'g+',t*vy/3600,yt,'b+');
```

```
   subplot(212),plot(t*vu/3600,u,'r+');
end
```

The parameters for the MATLAB method are the name of the default .m file and the name of the variables in the Matlab workspace. These indications are optional because default names are provided.

In debug mode = 0 no syntax checking of the .m file is performed, in debug mode = 1 checks are performed and any messages are written to a matlab.dbg file (be careful to delete it at the end of the simulation because it is opened in "append" mode and the messages are concatenated at the end of the file if it already exists, it can therefore become very large), in debug mode = 2 the messages are, moreover, displayed on the screen. It is advisable to use mode 2 during debugging of the .m file and then to switch to mode 0 to do the routine simulations (time saving).

The location of the matlab.exe program can be specified, which is useful if different versions of Matlab are installed on the computer. But the desired version of Matlab must also be defined in the corresponding option in the "Options/Options" menu that manages the SIRENE and FLUVIA exe with the correct version of Matlab. The PATH system must also contain the reference to the Matlab subdirectory (of the correct version to be used), and that there is no reference before to another subdirectory also containing libraries like lapack.dll. Otherwise there will be a conflict (for example with another version of MatLab, or a version of Scilab, see below). In this case we can get an error message saying that Fluvia or Sirene does not find libmx.dll ("The program can not start because libmx.dll is missing from your computer"). The solutions can then be either to modify this path, or to temporarily rename the installation subdirectories of these other programs (you usually have administrator rights to do this).

You can also specify if you want to close or not Matlab at the end of the simulation (tick box selected to close Matlab).

The Matlab DDE link was compiled with Matlab version 6.5 (release 13) for versions 4.07 to 4.21, with version 7.1 (release 14) for version 4.22, and also with versions 7.4 (Release 2007a), 7.5 Release 2007b), 2008 and 2009. If you have another version, please contact us. With the libraries of your version we can normally compile a specific version adapted to your version of Matlab (currently only in 32 bits). Some versions of Matlab provide backwards compatibility with earlier versions of the Matlab DDE link, but Matlab's policy on this point is unclear. Recent versions like the 2010b version of Matlab (Release 7.11.0.584 to be precise) is compatible with our versions of Fluvia and Sirene with the DDE MatLab 7.1 link. So even if your version of Matlab is not explicitly proposed in our list, we recommend that you test the different versions we offer with your version of Matlab before concluding that this is not available.

In the event of a problem when launching the DDE link (problem to launch

Matlab, message that dlls are not found, or that procedure entry points are not found in dlls, or function not found) is that you probably have a conflict between various MatLab-related dll and Scilab installed on your computer. The problem is that these programs use a number of dlls, some of which have the same name, such as lapack.dll in particular. If you choose the option "No DDE link" (in the "Hydraulics" mode, go to the Options/Options menu, the "Select" button in the "Version of the DDE Matlab or Scilab" link), then the active FLUVIA and SIRENE programs are those that do not have DDE links with MatLab or Scilab. There will be no more conflict but you will not be able to use this DDE link. You will get a message in the display window during the calculation. You will have to choose the version of FLUVIA and SIRENE that corresponds to the version of MatLab or Scilab you use (installed on your computer) by choosing the "MatLab 6", "MatLab 7.1", "MatLab 7.4", "MatLab 7.5"," MatLab 2008 "," MatLab 2009 "," Scilab 4 "or" Scilab 5 ". In this case, the corresponding FLUVIA and SIRENE exe, as well as the dlluser *.dll are copied to the SIC/EXE subdirectory and become the active versions.

If you have multiple versions of Matlab installed on your computer, you will need to define the one you want to use. To do this, follow the instructions on the Matlab forum (Register MATLAB as Automation Server):

https://fr.mathworks.com/help/matlab/matlab_external/register-matlab-as-automation-server.html?requestedDomain=nl.mathworks.com

Otherwise you might have a problem running Fluvia or Sirene when they attempt to open Matlab with the message "MATLAB can not be launched".

Chapter 108

The PLOT method

Under development ...

Chapter 109

The POMPES method

This control method mimics the operation of a pumping station comprising a number of identical pumps which start or stop depending on a low or high water level and a time delay.

```
SUBROUTINE CPOMPES(U,Y,YT,PARA)

   USE D_GLOBAL, ONLY : REPSI
   USE SIRENE_parametres, ONLY : LONG
   USE TEMPS, ONLY : T,TDEB,TFIN
   USE IFICH, ONLY : JREGS
   USE P1, ONLY : INURE
   USE P2, ONLY : INU,INY,DTU

   IMPLICIT NONE

   REAL(KIND=LONG), INTENT(IN) :: Y,YT
   REAL(KIND=LONG), INTENT(INOUT) :: PARA, U

   INTEGER :: NBMIN,NBINI,NBMAX,NBNOW
   REAL(KIND=LONG) :: ZH,ZB,Z1,Z2,Q,TLASTON,TLASTOF
   REAL(KIND=LONG) :: TEMPOonoff,TEMPOoffon,TEMPOonon,TEMPOoffoff
   REAL(KIND=LONG) :: RHO,P,H,PMAX,ETOT
   LOGICAL :: Bon,Boff

   DIMENSION U(*),Y(*),YT(*),PARA(*)
!-----Affectations
      ZH=PARA(1)
      ZB=PARA(2)
      NBMIN=INT(PARA(3))
      NBINI=INT(PARA(4))
```

```
      NBMAX=INT(PARA(5))
      Q=PARA(6)
      TEMPOonoff=PARA(7)
      TEMPOoffon=PARA(8)
      TEMPOonon=PARA(9)
      TEMPOoffoff=PARA(10)
      TLASTON=PARA(11)
      TLASTOF=PARA(12)
      PMAX=PARA(13)
      ETOT=PARA(14)
      NBNOW=PARA(15)

!-----Protection
      IF (Q.EQ.0.) RETURN

!-----Calcul des booléens pour savoir si on peut faire un on et/ou un off
!-----en tenant compte des tempos. Par défaut on ne peut pas (=FALSE)
      Bon=.FALSE.
      Boff=.FALSE.
      IF(TLASTON<=TDEB.AND.TLASTOF<=TDEB) THEN
      !--On n'a rien fait encore donc on peut tout faire (on ou off)
         Bon=.TRUE.
         Boff=.TRUE.
      ELSEIF(TLASTON<=TDEB) THEN
      !--On a déjà fait un off mais pas encore un on
         IF((T-TLASTOF).GE.TEMPOoffon) Bon=.TRUE.
         IF((T-TLASTOF).GE.TEMPOoffoff) Boff=.TRUE.
      ELSEIF(TLASTOF<=TDEB) THEN
      !--On a déjà fait un on mais pas encore un off
         IF((T-TLASTON).GE.TEMPOonon) Bon=.TRUE.
         IF((T-TLASTON).GE.TEMPOonoff) Boff=.TRUE.
      ELSE
      !--On a déjà fait un on et un off
         IF(TLASTON<TLASTOF) THEN
         !--Le dernier réalisé est un off
            IF((T-TLASTOF).GE.TEMPOoffon) Bon=.TRUE.
            IF((T-TLASTOF).GE.TEMPOoffoff) Boff=.TRUE.
         ELSE
         !--Le dernier réalisé est un on (ou les 2 en mm temps mais pas possible)
            IF((T-TLASTON).GE.TEMPOonon) Bon=.TRUE.
            IF((T-TLASTON).GE.TEMPOonoff) Boff=.TRUE.
         ENDIF
      ENDIF

      IF (Y(1).GT.ZH) THEN
```

```
         IF (Q.GT.0) THEN
!--------Arret d'une pompe (alimentation) si possible si Q>0 (N>NBMIN et TEMPO)
            IF(NBNOW.GT.NBMIN.AND.Boff) THEN
               NBNOW=NBNOW-1
               TLASTOF=T
               WRITE(JREGS,830) 'POMPE ',INURE,' - Arret d''une pompe (alimentation) à T=',T,' :
               ,' . U=',U(1),' -> ',NBNOW*Q
            ENDIF
         ELSE
!--------Démarrage d'une pompe (extraction) si possible si Q<0 (N<NBMAX et TEMPO)
            IF(NBNOW.LT.NBMAX.AND.Bon) THEN
               NBNOW=NBNOW+1
               TLASTON=T
               WRITE(JREGS,830) 'POMPE ',INURE,' - Démarrage d''une pompe (extraction) à T=',T,'
               ,' . U=',U(1),' -> ',NBNOW*Q
            ENDIF
         ENDIF
      ELSEIF (Y(1).LT.ZB) THEN
         IF (Q.GT.0) THEN
!--------Démarrage d'une pompe (alimentation) si possible si Q>0 (N<NBMAX et TEMPO)
            IF(NBNOW.LT.NBMAX.AND.Bon) THEN
               NBNOW=NBNOW+1
               TLASTON=T
               WRITE(JREGS,830) 'POMPE ',INURE,' - Démarrage d''une pompe (alimentation) à T=',T,
               ,' . U=',U(1),' -> ',NBNOW*Q
            ENDIF
         ELSE
!--------Arret d'une pompe (extraction) si possible si Q<0 (N>NBMIN et TEMPO)
            IF(NBNOW.GT.NBMIN.AND.Boff) THEN
               NBNOW=NBNOW-1
               TLASTOF=T
               WRITE(JREGS,830) 'POMPE ',INURE,' - Arret d''une pompe (extraction) à T=',T,' : ',
               ,' . U=',U(1),' -> ',NBNOW*Q
            ENDIF
         ENDIF
      ENDIF
!-----Affectation du débit dans la commande U
      U(1)=NBNOW*Q
!-----Vérification des cotes de l'eau
      IF (Q.LT.0) THEN
!-----Vérification de la cote de l'eau à l'aspiration (Q<0)
!-----Récupération de la cote d'aspiration Z1 pour le calcul du gain de charge
       IF(Y(1).LT.YT(1)) THEN
          WRITE(JREGS,800) 'POMPE ',INURE,' - Cote de l''eau trop basse à l''aspiration au temps
          Z1=YT(1)
```

```
       ELSE
          Z1=Y(1)
       ENDIF
!-----Récupération de la cote de refoulement Z2 pour le calcul du gain de charge
!-----Si on a un Y2, en option on prend YT(2), sinon on prend YT(1)+10m
       IF(INY.LT.2) THEN
          Z2=YT(1)+10.0
       ELSE
!-----Si on a un U2 on regarde alors la valeur de Y(2) % YT(2) sinon on prend YT(2)
          IF(INU.GT.1) THEN
             IF(Y(2).LT.YT(2)) THEN
!---------------Le refoulement est dénoyé
                Z2=YT(2)
             ELSE
!---------------Le refoulement est noyé
                Z2=Y(2)
             ENDIF
          ELSE
             Z2=YT(2)
          ENDIF
       ENDIF
      ELSE
!-----Vérification de la cote de l'eau au refoulement (Q>0)
!-----Récupération de la cote de refoulement Z2 pour le calcul du gain de charge
       IF(Y(1).GT.YT(1)) THEN
          WRITE(JREGS,800) 'POMPE ',INURE,' - Cote de l''eau plus haute au refoulement
          Z2=Y(1)
       ELSE
          Z2=YT(1)
       ENDIF
!-----Récupération de la cote d'aspiration Z1 pour le calcul du gain de charge
!-----Si on a un Y2, en option on prend YT(2), sinon on prend YT(1)-10m
       IF(INY.LT.2) THEN
          Z1=YT(1)-10.0
       ELSE
!-----Si on a un U2 on regarde alors la valeur de Y(2) % YT(2) sinon on prend YT(2)
          IF(INU.GT.1) THEN
             IF(Y(2).LT.YT(2)) THEN
!---------------L'aspiration est dénoyé (pb mais on doit qd mm faire le calcul)
                Z1=YT(2)
             ELSE
!---------------L'aspiration est noyé (ok cas normal)
                Z1=Y(2)
             ENDIF
          ELSE
```

```
            Z1=YT(2)
          ENDIF
        ENDIF
      ENDIF
!-----Calcul de la hauteur de refoulement - hauteur d'aspiration
!-----On néglige les V^2/2g à l'amont et à l'aval ainsi que les pertes de charges dans le disposi
      H=Z2-Z1
!-----Calcul de la puissance hydraulique nécessaire au refoulement
      RHO=1
      P=RHO*9.81*ABS(U(1))*H
!-----Stockage de la puissance max dans PMAX
      IF(P.GT.PMAX) THEN
        PMAX=P
        WRITE(JREGS,810) 'POMPE ',INURE,' - Puissance maximum au temps T=',T,' : ',P,' Watts'
      ENDIF
!-----Calcul de l'énergie totale consommée dans ETOT
      ETOT=ETOT+P*DTU
      IF(T.GE.TFIN) THEN
        WRITE(JREGS,820) 'POMPE ',INURE,' - Energie totale consommée jusqu''au au temps T=',T,' :
      ENDIF
!-----Possibilité de faire communiquer la pompe avec un autre emplacement (si INU>1)
      IF(INU.GT.1) THEN
        U(2)=-U(1)
      ENDIF
!-----Sauvegarde des variables
      PARA(11)=TLASTON
      PARA(12)=TLASTOF
      PARA(13)=PMAX
      PARA(14)=ETOT
      PARA(15)=NBNOW
  800 FORMAT(A6,I2,A55,F12.2,A3,F14.3,A3,F14.3)
  810 FORMAT(A6,I2,A55,F12.2,A3,F12.1,A6)
  820 FORMAT(A6,I2,A55,F12.2,A3,E20.1,A7)
  830 FORMAT(A6,I2,A55,F12.2,A3,F14.3,A3,F14.3,A5,F10.3,A4,F10.3)

END SUBROUTINE CPOMPES
```

If you want a different pumping station control module you can always copy this routine into a USER module and modify it as you wish (in this case, do this for the LPOMPES -> LUSER routine for parameter reading).

```
SUBROUTINE LPOMPES(CHAINE,Reg)
   USE D_REGULATION, ONLY : Regulateur_t
   USE SIRENE_parametres, ONLY : LONG
   USE F1       , ONLY : IERROR,IWARN,CWARN
   USE TEMPS    , ONLY : TDEB
```

```
   IMPLICIT NONE

   CHARACTER, INTENT(IN) :: CHAINE*(*)
   TYPE(Regulateur_t), INTENT(INOUT) :: Reg

   REAL(KIND=LONG) :: TEMPOonoff,TEMPOoffon,TEMPOonon,TEMPOoffoff
   INTEGER :: I

!-----Debut de la routine
!-----Lecture des parametres specifiques pour cette methode

!-----Debut de la routine
!-----En V 5.31e on passe à 10 paramètres
      READ(CHAINE,'(9(F8.3,1X),F8.3)',ERR=10) (Reg%tPara(I),I=1,10)
      TEMPOonoff=Reg%tPara(7)
      TEMPOoffon=Reg%tPara(8)
      TEMPOonon=Reg%tPara(9)
      TEMPOoffoff=Reg%tPara(10)
      Reg%tPara(11)=TDEB-MAX(TEMPOonoff,TEMPOonon)   ! TLASTON
      Reg%tPara(12)=TDEB-MAX(TEMPOoffon,TEMPOoffoff) ! TLASTOF
      Reg%tPara(13)=0        ! Puissance
      Reg%tPara(14)=0        ! Energie
      Reg%tPara(15)=Reg%tPara(4)  ! Nb de pompes actives en temps réel (NBNOW=NBINI)

      RETURN
   10 CONTINUE
      IERROR=2

END SUBROUTINE LPOMPES
```

Chapter 110

The PRINT method

This method does not calculate U-commands to apply to some devices, but prints the selected U, Y, YT, and Z variables into a LST text file.

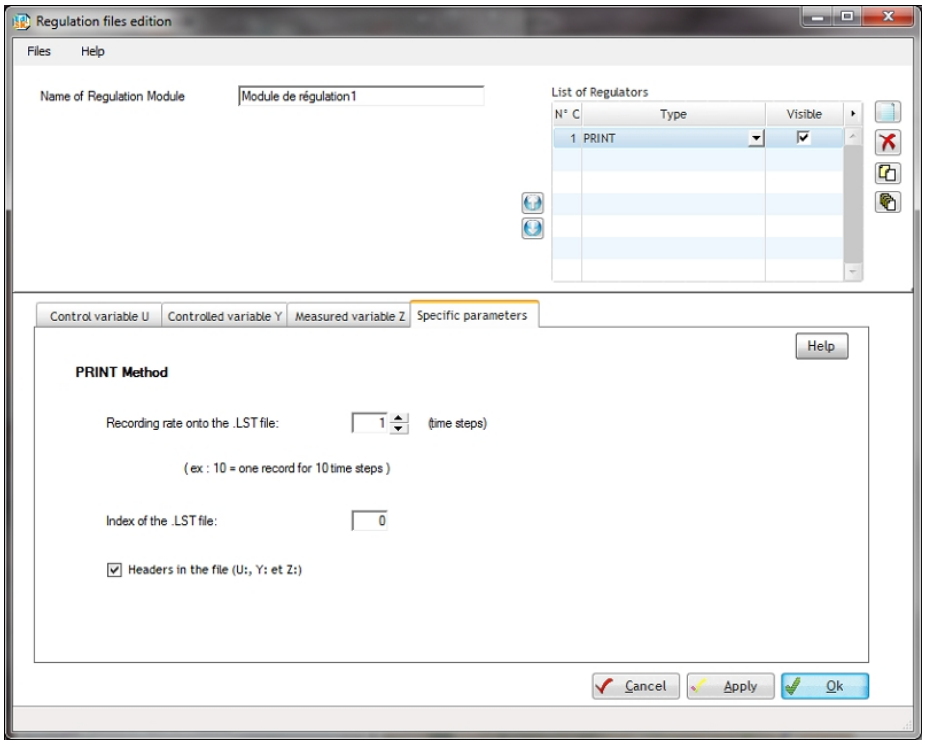


Figure 110.1: editregv5_print

The frequency of writing the selected data into the LST file can be chosen. For example if you choose 10, in this case one writes one time step out of 10.

We can choose to have the data written into a different file than the standard Project_Program_Scenario_Variant_LST files. In this case the name of the file will be REGULPRINTn.LST, where n is an index number in the file name. This is useful in the case we have several PRINT regulators, in order to have separate files. This is also useful when different regulators write different information into the LST file, to have a more readable file.

It is possible finally to choose whether or not to have the headers in the first column U :, Y: YT: or Z:. If you want to import these files into an external software (eg Matlab), it is probably better not to have these headers in order to have only numerical data.

In the LST file, the first column indicates the time T-DT, the second column the time T, and the subsequent columns indicate the requested values. The regulators of the active control module are calculated before the hydraulic calculation. Thus, in general, the values written at the instant T, available in memory, are in fact derived from the hydraulic calculation carried out previously, and therefore corresponding to the time T-DT. In some cases, the results of other controllers of the active control module can be requested in the Z variables. Therefore, if this PRINT regulator is located (in the order of the regulators) before the regulator targeted by the Z of the PRINT, the corresponding value will be that of the instant T-DT, and if not T (because the U will have been calculated just before the PRINT call). For this reason, the 2 times T-DT and T are indicated so that you can use the one you are interested in.

The number of variables that can be written to a PRINT file is limited to 100. If you want more, use multiple PRINT modules.

Chapter 111

The QSCP method

The QSCP method is used to generate disturbances at the offtakes of type encountered on irrigation channels operating on demand (model developed using data provided by the Société du Canal de Provence).

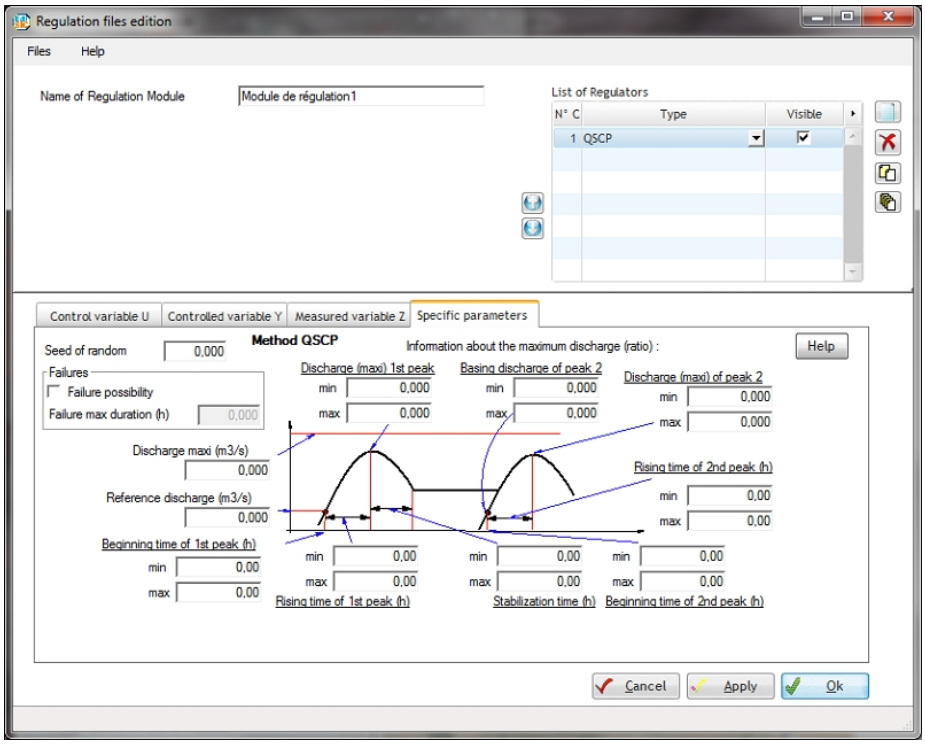


Figure 111.1: editregv5_qscp

```
* The parameters indicated define the min and max values
*  Seed    Qbase     Qmax      TMAXP              IFLAGP ('PS',1X,(4(F10.0),I10))
PS=10.       -0.350    -0.700    10.                 1
*  T01MIN    T01MAX
PS=5.5       7.
*  TM1MIN    TM1MAX
PS=2.5       4.75
*  TSMIN     TSMAX
PS=1.5       4.5
*  QP1MIN    QP1MAX
PS=0.457     0.857
*  Q02MIN    Q02MAX
PS=0.267     0.508
*  QP2MIN    QP2MAX
PS=0.576     0.929
*  T02MIN    T02MAX
PS=14.25     16.75
*  TM1MIN    TM1MAX
PS=3.5       5.75
```

Chapter 112

The SCADA method

The “Digesteur” is the name given by DSA to the interface of communication between the fronts (1 and 2 in redundancy) and the Cemagref (Irstea, Inrae) tools described below allowing real-time automatic control experiments on the Gignac canal.

The SCADA module marked “Scada_Sic” below is the name given by Cemagref (Irstea) to the regulation module which manages the communication between the files of the Digester given below (csg.txt, data.txt as well as the parameterization files) and the Sirene SIC transient calculation program including standard regulation modules.

112.1 Global computing architecture

112.2 Variables handled by the SIC Scada regulation module

The variables written to the data.txt file and read on the csg.txt file (by the DSA digester) are defined in the SQL database using the “ItemRegulateur” field. If this field is not specified, then this variable is not managed by the digester, and therefore by the Cemagref control tools. It is very easy to add a new variable in this management. It is enough to fill in this field “ItemRegutaleur”. Names follow a certain logic (eg, Data.Avencq.DebitAmont for measured data, Csg.Avencq.PosVanneReg for control variables) that it is good to follow, even if it is not mandatory. The variables thus available are given below as an example for the Avencq site.

|| List of U-control variables available in Avencq || | *Tag* | *Libelle* | *ItemRegulateur* | *Unite* | *TableHisto* | |3076 | Consigne débit aval CEMAGREF vanne

régulation Avencq | Csg.Avencq.DebitAvalVanneReg | l/s | H_Avencq_Consignes | | 3081 | Consigne niveau amont CEMAGREF vanne décharge Avencq | Csg.Avencq.NivAmontVanneDec | cm | H_Avencq_Consignes | | 3079 | Consigne position CEMAGREF vanne décharge Avencq | Csg.Avencq.PosVanneDec | cm | H_Avencq_Consignes | | 3074 | Consigne position CEMAGREF vanne régulation Avencq | Csg.Avencq.PosVanneReg | cm | H_Avencq_Consignes | | 3127 | Seuil bas niveau canal amont Avencq | Csg.Avencq.SeuilBasAmont | cm | H_Avencq_Consignes | | 3126 | Seuil haut niveau canal amont Avencq | Csg.Avencq.SeuilHautAmont | cm | H_Avencq_Consignes | | 3128 | Seuil haut niveau canal aval Avencq | Csg.Avencq.SeuilHautAval | cm | H_Avencq_Consignes |

|| List of Z-measured variables available in Avencq || | Tag | Libelle | ItemRegulateur | Unite | TableHisto | | 3015 | Débit amont Avencq | Data.Avencq.DebitAmont | l/s | H_Avencq_DebAmont | | 3020 | Débit aval Avencq | Data.Avencq.DebitAval | l/s | H_Avencq_DebAval | | 3019 | Débit aval (vitesse, hauteur) Avencq | Data.Avencq.DebitAvalVH | l/s | H_Avencq_DebAval1 | | 3018 | Débit déversoir Avencq | Data.Avencq.DebitDeversoir | l/s | H_Avencq_DefbDeversoir | | 3017 | Débit vanne décharge Avencq | Data.Avencq.DebitVanneDec | l/s | H_Avencq_DebVDecharge | | 3016 | Débit vanne régulation Avencq | Data.Avencq.DebitVanneReg | l/s | H_Avencq_DebVRegule | | 3053 | Défaut calcul débit v décharge Avencq | Data.Avencq.DefCalculDebitVanneDec | | H_Avencq_Defauts | | 3051 | Défaut calcul débit amont Avencq | Data.Avencq.DefDebitAmont | | H_Avencq_Defauts | | 3056 | Défaut calcul débit aval (vanne-déversoir) Avencq | Data.Avencq.DefDebitAval | | H_Avencq_Defauts | | 3055 | Défaut calcul débit aval (vitesse-hauteur)Avencq | Data.Avencq.DefDebitAvalVH | | H_Avencq_Defauts | | 3054 | Défaut calcul débit déversoir Avencq | Data.Avencq.DefDebitDeversoir | | H_Avencq_Defauts | | 3033 | Défaut capteur position v décharge Avencq | Data.Avencq.DefDebitVanneDec | | H_Avencq_Defauts | | 3052 | Défaut calcul débit v régulation Avencq | Data.Avencq.DefDebitVanneReg | | H_Avencq_Defauts | | 3030 | Défaut capteur niveau canal amont Avencq | Data.Avencq.DefNivAmont | | H_Avencq_Defauts | | 3031 | Défaut capteur niveau canal aval Avencq | Data.Avencq.DefNivAval | | H_Avencq_Defauts | | 3131 | Défaut niveau trop bas canal amont Avencq | Data.Avencq.DefNivBasAmont | | H_Avencq_Defauts | | 3133 | Défaut niveau trop bas canal amont Avencq | Data.Avencq.DefNivBasAval | | H_Avencq_Defauts | | 3130 | Défaut niveau trop haut canal amont Avencq | Data.Avencq.DefNivHautAmont | | H_Avencq_Defauts | | 3132 | Défaut niveau trop haut canal aval Avencq | Data.Avencq.DefNivHautAval | | H_Avencq_Defauts | | 3032 | Défaut capteur position v régulation Avencq | Data.Avencq.DefPosVanneReg | | H_Avencq_Defauts | | 3089 | Défaut surverse Avencq | Data.Avencq.DefSurverse | | H_Avencq_Defauts | | 3034 | Défaut capteur vitesse aval Avencq | Data.Avencq.DefVitesseEauAval | | H_Avencq_Defauts | | 3008 | Niveau canal amont Avencq | Data.Avencq.NivAmont | cm | H_Avencq_NivAmont | | 3009 | Niveau canal aval Avencq

| Data.Avencq.NivAval | cm | H_Avencq_NivAval | | 3011 | Position vanne décharge Avencq | Data.Avencq.PosVanneDec | cm | H_Avencq_PositionV2 | | 3010 | Position vanne régulation Avencq | Data.Avencq.PosVanneReg | cm | H_Avencq_PositionV1 | | 3012 | Vitesse de l'eau aval Avencq | Data.Avencq.VitesseEauAval | m/s | H_Avencq_VitesseAval |

112.3 Files managed by the digester and Scada_SIC

112.4 Architecture of the subdirectories:

The Digester handles its various input and output files in a set of subdirectories with the following structure:

-* REGULATEUR -** Commande -** Csg -** CsgRepli -** Data -** ParamDigesteur -** SaveCsg -** SaveData

112.5 Sub-directory Commande

This sub-directory contains 2 files:

112.6 *CdeF.txt:

This file contains a single line with:

STOP or START

To indicate whether the Digester is in action or not. It is the Scada_Sic module that writes this variable "START" (at the start of the calculation) or "STOP" (at the end of the simulation) in this file. It is possible at any time to write in this file what one wants with any text editor, such as Notepad for example. This will stop the test in an emergency. This may also be forced at the interface of the Scada_Sic module.

112.7 *Config.txt:

This file contains information about time periods and the backup option of the data.txt and csg.txt files. The PeriodeData variable is very important since it is the one that will be the cadence of the measurements and creations of the data.txt files. The Scada_Sic module reads this variable and verifies that it is identical to that of the control module. A warning message will be displayed if there is a difference.

```
'time out (in s) on the sending of instructions from the setpoint file. Afterwards, tal
TimeoutCsg=900

'Period (in s) of generation of the acquisition file
PeriodeData=300

'Switch to find out if data files are backed up (1 for Yes, 0 for No)
ActivationTraceData=1

'Switch to find out if csg files are backed up (1 for Yes, 0 for No)
ActivationTraceCsg=1
```

112.8 Sub-directory Csg

It is in the Csg sub-directory that the files csg.txt containing the instructions to be applied in real time to the control structures (instructions sent to the programmable controllers on sites) will be read periodically. As soon as a file csg.txt is read by the digester, it is destroyed. The units are those specified in the database. These are often cm for openings and l/s for flow. Be sure to check these units. These csg.txt files will be created periodically by the Scada_Sic module. The index is incremented with each new file. Attention this variable is important because verified by the digester and must correspond to the index of the file data.txt read. The timestamp is the creation time of the csg.txt file. This timestamp is not managed by the Digester, it is just written for information.

112.9 *Csg.txt:

```
[Header]
Horodate=25/10/2004 14:43: 7
Index=    1
[Data]
CSG.AVENCQ.POSVANNEDEC                                    ;25/10/2004 14:43: 7;
CSG.AVENCQ.POSVANNEREG                                    ;25/10/2004 14:43: 7;
[End]
```

112.10 Sub-directory CsgRepli

The file CsgRepli.txt is a file used a priori in the event of a problem, when a file csg.txt has not been supplied during a time specified in the timeoutCsg variable in the file param.txt described below. It has the same format as the csg.txt files. Neither the index nor the timestamp are used in this case.

112.11 *CsgRepli.txt:

```
[Header]
Horodate=01/01/2000 00:00:00
Index=1
[Data]
CSG.AVENCQ.POSVANNEDEC                                    ;25/10/2004 14:43: 7;      +0.520E+01
CSG.AVENCQ.POSVANNEREG                                    ;25/10/2004 14:43: 7;      +0.298E+02
[End]
```

112.12 Sub-directory Data

This sub-directory will contain the data.txt files generated periodically by the digester at a period specified in the config.txt file described above (variable PeriodeData). These data.txt files will be read by the Scada_Sic module and destroyed at the end of the reading, to leave room for a new file. This file contains all the variables having an ItemRegulator in the SQL database. It is the file location.txt that will indicate the variables that interest us for the Scada_Sic module and which will link the field variables to the internal variables of the Scada_Sic module (U, Y and Z variables of the control module).

112.13 *Data.txt:

```
[Header]
Horodate=26/10/2004 17:03:02
Index=01
[Data]
DATA.AURELLE.DEFPIEZZO                    ;20/03/2004 10:31:16;0.0000000000E000
DATA.AURELLE.DEFSURVERSE                  ;20/03/2004 10:31:16;0.0000000000E000
DATA.AURELLE.NIVPIEZZO                    ;26/03/2004 15:51:07;7.0000000000E001
DATA.AVENCQ.DEBITAMONT                    ;26/03/2004 15:51:27;5.3100000000E002
DATA.AVENCQ.DEBITAVAL                     ;26/03/2004 15:51:27;5.3100000000E002
DATA.AVENCQ.DEBITAVALVH                   ;26/03/2004 15:51:27;7.0100000000E002
DATA.AVENCQ.DEBITDEVERSOIR                ;26/03/2004 15:51:27;4.7700000000E002
DATA.AVENCQ.DEBITVANNEDEC                 ;26/03/2004 10:55:54;0.0000000000E000
DATA.AVENCQ.DEBITVANNEREG                 ;26/03/2004 15:51:27;5.4000000000E001
DATA.AVENCQ.DEFCALCULDEBITVANNEDEC        ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFDEBITAMONT                 ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFDEBITAVAL                  ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFDEBITAVALVH                ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFDEBITDEVERSOIR             ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFDEBITVANNEDEC              ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFDEBITVANNEREG              ;20/03/2004 10:31:26;0.0000000000E000
```

```
DATA.AVENCQ.DEFNIVAMONT                ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFNIVAVAL                 ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFNIVBASAMONT             ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFNIVBASAVAL              ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFNIVHAUTAMONT            ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFNIVHAUTAVAL             ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFPOSVANNEREG             ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFSURVERSE                ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.DEFVITESSEEAUAVAL          ;20/03/2004 10:31:26;0.0000000000E000
DATA.AVENCQ.NIVAMONT                   ;26/03/2004 15:50:57;8.5200000000E001
DATA.AVENCQ.NIVAVAL                    ;26/03/2004 15:51:27;5.5200000000E001
DATA.AVENCQ.POSVANNEDEC                ;26/03/2004 10:37:08;0.0000000000E000
DATA.AVENCQ.POSVANNEREG                ;26/03/2004 10:06:57;0.0000000000E000
DATA.AVENCQ.VITESSEEAUAVAL             ;26/03/2004 15:51:27;8.7000000000E-001
DATA.BELBEZET.DEBAMONT                 ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEBAVAL                  ;26/03/2004 07:07:45;1.0370000000E003
DATA.BELBEZET.DEBVANNEDECHARGE         ;26/03/2004 07:00:45;4.2000000000E001
DATA.BELBEZET.DEBVANNEDROITE           ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEBVANNEGAUCHE           ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEBVANNEMILIEU           ;26/03/2004 07:07:45;1.0370000000E003
DATA.BELBEZET.DEFDEBAMONT              ;26/03/2004 07:00:39;1.0000000000E000
DATA.BELBEZET.DEFDEBDECHARGE           ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFDEBDROITE             ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFDEBGAUCHE             ;26/03/2004 07:00:39;1.0000000000E000
DATA.BELBEZET.DEFDEBMILIEU             ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFNIVAMONT              ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFNIVAVAL               ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFNIVBASAMONT           ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFNIVBASAVAL            ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFNIVBASRETENUE         ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFNIVHAUTAMONT          ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFNIVHAUTAVAL           ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFNIVHAUTRETENUE        ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFNIVHERAULT            ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFPOSVANNEDECHARGE      ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFPOSVANNEDROITE        ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.DEFPOSVANNEGAUCHE        ;26/03/2004 07:00:39;1.0000000000E000
DATA.BELBEZET.DEFPOSVANNEMILIEU        ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.NIVAMONT                 ;26/03/2004 07:00:45;1.1060000000E002
DATA.BELBEZET.NIVAVAL                  ;26/03/2004 07:07:45;8.7800000000E001
DATA.BELBEZET.NIVHERAULT               ;26/03/2004 07:00:45;1.2550000000E002
DATA.BELBEZET.NIVHERAULTNGF            ;26/03/2004 07:00:39;7.2050000000E001
DATA.BELBEZET.POSVANNEDECHARGE         ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.POSVANNEDROITE           ;26/03/2004 07:00:39;0.0000000000E000
DATA.BELBEZET.POSVANNEGAUCHE           ;26/03/2004 07:07:45;3.4400000000E001
```

```
DATA.BELBEZET.POSVANNEMILIEU                ;26/03/2004 07:00:39;2.4200000000E001
DATA.CEYRAS.DEBITCANALAMONT                 ;25/03/2004 09:52:27;4.0777196875E005
DATA.CEYRAS.DEBITDEVERSOIR                  ;25/03/2004 09:52:27;0.0000000000E000
DATA.CEYRAS.DEBITREJET                      ;25/03/2004 09:52:27;0.0000000000E000
DATA.CEYRAS.DEBITVANNERESERVOIR             ;25/03/2004 09:52:27;0.0000000000E000
DATA.CEYRAS.NIVBASCANALAMONT                ;25/03/2004 09:51:53;0.0000000000E000
DATA.CEYRAS.NIVCANALAMONT                   ;25/03/2004 09:52:27;5.4615091173E001
DATA.CEYRAS.NIVHAUTCANALAMONT               ;25/03/2004 09:51:53;0.0000000000E000
DATA.CEYRAS.PCTFERMETUREVANNEDEV            ;25/03/2004 09:51:53;7.1900511177E001
DATA.GIGNAC.DEFPIEZZO                       ;20/03/2004 10:31:11;0.0000000000E000
DATA.GIGNAC.DEFSURVERSE                     ;20/03/2004 10:31:11;0.0000000000E000
DATA.GIGNAC.NIVEAUPIEZZO                    ;26/03/2004 15:50:28;7.2200000000E001
DATA.LAGAREL.DEBITAMONTTH                   ;26/03/2004 15:51:28;3.7000000000E001
DATA.LAGAREL.DEBITAVALTH                    ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEBITAVALVH                    ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEBITDEVERSOIR                 ;26/03/2004 15:51:28;3.7000000000E001
DATA.LAGAREL.DEBITVANNEDEC                  ;25/03/2004 15:15:37;0.0000000000E000
DATA.LAGAREL.DEBITVANNEREG                  ;20/03/2004 10:31:26;-3.2768000000E004
DATA.LAGAREL.DEFDEBITAMONTTH                ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFDEBITAVALTH                 ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFDEBITAVALVH                 ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFDEBITDEVERSOIR              ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFDEBITVANNEDEC               ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFDEBITVANNEREG               ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFNIVCANALAMONT               ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFNIVCANALAVAL                ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFNIVEAUBASAMONT              ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFNIVEAUBASDEVERSOIR          ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFNIVEAUHAUTAMONT             ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFNIVEAUHAUTDEVERSOIR         ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFPOSITIONVANNEDEC            ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFPOSITIONVANNEREG            ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEFSURVERSE                    ;25/03/2004 18:29:52;0.0000000000E000
DATA.LAGAREL.DEFVITESSEEAUAVAL              ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.DEVERSOIR                      ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.NIVCANALAMONT                  ;26/03/2004 15:50:58;8.6000000000E001
DATA.LAGAREL.POSITIONVANNEDEC               ;24/03/2004 09:56:29;0.0000000000E000
DATA.LAGAREL.POSITIONVANNEREG               ;20/03/2004 10:31:26;0.0000000000E000
DATA.LAGAREL.VITESSEEAUAVAL                 ;20/03/2004 10:31:26;0.0000000000E000
DATA.MOBILE.DEFPIEZZO                       ;20/03/2004 10:35:53;0.0000000000E000
DATA.MOBILE.NIVPIEZZO                       ;26/03/2004 17:02:53;9.2000000000E001
DATA.PARTITEUR.DEBDEVERSOIRAMONT            ;26/03/2004 11:13:17;0.0000000000E000
DATA.PARTITEUR.DEBITAMONTTH                 ;26/03/2004 17:03:00;1.6090000000E003
DATA.PARTITEUR.DEBITDECHARGEDROITE          ;25/03/2004 18:00:19;0.0000000000E000
DATA.PARTITEUR.DEBITDECHARGEGAUCHE          ;20/03/2004 10:31:25;0.0000000000E000
```

```
DATA.PARTITEUR.DEBITDROITE                ;26/03/2004 17:03:00;9.7300000000E002
DATA.PARTITEUR.DEBITGAUCHE                ;26/03/2004 17:03:00;6.3600000000E002
DATA.PARTITEUR.DEBITREGULDROITE           ;26/03/2004 17:03:00;8.2100000000E002
DATA.PARTITEUR.DEBITREGULGAUCHE           ;26/03/2004 17:03:00;7.4600000000E002
DATA.PARTITEUR.DEFDEBITAMONT              ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFDEBITDECHARGEDROITE     ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFDEBITDECHARGEGAUCHE     ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFDEBITDROITE             ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFDEBITGAUCHE             ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFDEBITREGULDROITE        ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFDEBITREGULGAUCHE        ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFNIVBCANALCOMMUN         ;20/03/2004 10:31:25;0.0000000000E000
DATA.PARTITEUR.DEFNIVBCANALDROITE         ;20/03/2004 10:31:25;0.0000000000E000
DATA.PARTITEUR.DEFNIVBCANALGAUCHE         ;20/03/2004 10:31:25;0.0000000000E000
DATA.PARTITEUR.DEFNIVCANALCOMMUN          ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFNIVCANALDROITE          ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFNIVCANALGAUCHE          ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFNIVHCANALCOMMUN         ;20/03/2004 10:31:25;0.0000000000E000
DATA.PARTITEUR.DEFNIVHCANALDROITE         ;20/03/2004 10:31:25;0.0000000000E000
DATA.PARTITEUR.DEFNIVHCANALGAUCHE         ;20/03/2004 10:31:25;0.0000000000E000
DATA.PARTITEUR.DEFPOSITIONVANNEDECDROITE  ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFPOSITIONVANNEDECGAUCHE  ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFPOSITIONVANNEREGDROITE  ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFPOSITIONVANNEREGGAUCHE  ;25/03/2004 11:50:19;0.0000000000E000
DATA.PARTITEUR.DEFVITESSEEAUDROITE        ;20/03/2004 10:31:25;0.0000000000E000
DATA.PARTITEUR.DEFVITESSEEAUGAUCHE        ;20/03/2004 10:31:25;0.0000000000E000
DATA.PARTITEUR.NIVCANALCOMMUN             ;26/03/2004 17:03:00;6.8900000000E001
DATA.PARTITEUR.NIVCANALDROITE             ;26/03/2004 17:02:23;1.0510000000E002
DATA.PARTITEUR.NIVCANALGAUCHE             ;26/03/2004 17:03:00;8.2100000000E001
DATA.PARTITEUR.POSITIONVANNEDECDROITE     ;25/03/2004 18:00:19;0.0000000000E000
DATA.PARTITEUR.POSITIONVANNEDECGAUCHE     ;25/03/2004 11:37:30;0.0000000000E000
DATA.PARTITEUR.POSITIONVANNEREGDROITE     ;26/03/2004 16:35:18;3.5700000000E001
DATA.PARTITEUR.POSITIONVANNEREGGAUCHE     ;26/03/2004 16:27:31;1.4300000000E001
DATA.PARTITEUR.VITESSEEAUDROITE           ;26/03/2004 17:03:00;6.9800000000E-001
DATA.PARTITEUR.VITESSEEAUGAUCHE           ;26/03/2004 17:03:00;5.3100000000E-001
DATA.PONTLUSSAC.DEBITAMONT                ;25/03/2004 07:03:41;5.6315850000E005
DATA.PONTLUSSAC.DEBITAVAL                 ;25/03/2004 07:03:41;2.0273704960E009
DATA.PONTLUSSAC.DEBITDEVERSOIR            ;25/03/2004 07:03:41;8.0417953491E001
DATA.PONTLUSSAC.DEBITREJET                ;25/03/2004 07:03:41;8.0417953491E001
DATA.PONTLUSSAC.DEBITVANNECONDUITE        ;25/03/2004 07:03:41;0.0000000000E000
DATA.PONTLUSSAC.DEBITVANNEPONT            ;25/03/2004 07:03:41;0.0000000000E000
DATA.PONTLUSSAC.NIVBASCANALAMONT          ;25/03/2004 07:03:41;0.0000000000E000
DATA.PONTLUSSAC.NIVCANALAMONT             ;25/03/2004 07:03:41;6.1597619593E001
DATA.PONTLUSSAC.NIVHAUTCANALAMONT         ;25/03/2004 07:03:41;1.0000000000E000
DATA.PONTLUSSAC.PCTFERMETUREVANNECONDUITE ;25/03/2004 07:03:41;0.0000000000E000
```

```
DATA.PONTLUSSAC.PCTFERMETUREVANNEPONT     ;25/03/2004 07:03:41;2.1484702831E000
DATA.ROUVIERE.DEBITAMONT                  ;20/03/2004 10:31:27;1.3600000000E002
DATA.ROUVIERE.DEBITAMONTVH                ;26/03/2004 15:51:29;2.5400000000E002
DATA.ROUVIERE.DEBITAVAL                   ;26/03/2004 15:51:29;1.1400000000E002
DATA.ROUVIERE.DEBITDEVERSOIR              ;26/03/2004 15:51:29;1.1400000000E002
DATA.ROUVIERE.DEBITVANNEREG               ;23/03/2004 07:29:40;0.0000000000E000
DATA.ROUVIERE.DEFBASNIVCANAL              ;20/03/2004 10:31:27;0.0000000000E000
DATA.ROUVIERE.DEFDEBITAMONTVH             ;20/03/2004 10:31:27;0.0000000000E000
DATA.ROUVIERE.DEFDEBITAVAL                ;20/03/2004 10:31:27;0.0000000000E000
DATA.ROUVIERE.DEFDEBITDEVERSOIR           ;20/03/2004 10:31:27;0.0000000000E000
DATA.ROUVIERE.DEFDEBITVANNEREG            ;20/03/2004 10:31:27;0.0000000000E000
DATA.ROUVIERE.DEFHAUTNIVCANAL             ;20/03/2004 10:31:27;0.0000000000E000
DATA.ROUVIERE.DEFNIVCANAL                 ;20/03/2004 10:31:27;0.0000000000E000
DATA.ROUVIERE.DEFPOSVANNEREG              ;20/03/2004 10:31:27;0.0000000000E000
DATA.ROUVIERE.DEFSUSRVERSE                ;20/03/2004 10:31:27;0.0000000000E000
DATA.ROUVIERE.DEFVITESSEEAUAMONT          ;20/03/2004 10:31:27;0.0000000000E000
DATA.ROUVIERE.NIVCANAL                    ;26/03/2004 15:51:29;7.3800000000E001
DATA.ROUVIERE.POSVANNEREG                 ;23/03/2004 07:29:40;0.0000000000E000
DATA.ROUVIERE.VITESSEEAUAMONT             ;26/03/2004 15:51:29;1.4600000000E-001
DATA.STJEAN.DEFSUSRVERSE                  ;20/03/2004 10:31:10;0.0000000000E000
DATA.STJEAN.NIVEAUPIEZZO                  ;26/03/2004 15:51:38;5.6700000000E001
DATA.STJEAN.POSVANNE                      ;26/03/2004 15:51:38;1.5600000000E001
[End]
```

112.14 Sub-directory ParamDigesteur

The ParamDigester sub-directory contains the Param.txt file containing a set of useful parameters for the digester. The Scada_Sic module does not read this file. The TimeoutCsg and PeriodeData variables defined below (attention: in ms and not in s) will be overwritten by those defined in the config.txt file described above.

112.15 *Param.txt:

```
'N° PORT SOCKET
Sck_Port=***

'Adresse IP Connexion serveur CIA
Sck_AdrIP=***

DBPARAMTYPE=MYSQL
DBPARAMPATH=reference_gignac
DBPARAMSERVER=***
```

```
DBPARAMUSER=***
'DBPARAMPWD=***

'time out (en ms) prise en compte du fichier de consignes de repli
TimeoutCsg=900000

'Période (en ms) de génération du fichier d'acquisition
PeriodeData=60000

'Chemin du fichier d'acquisition (Chemin FTP)
PathData=/REGULATEUR/Data

'Chemin de sauvegarde des fichiers d'acquisition (Chemin FTP)
TraceData=/REGULATEUR/SaveData

'Chemin du fichier de consigne (Chemin FTP)
PathCsg=/REGULATEUR/Csg

'Chemin de sauvegarde des fichiers de consigne (Chemin FTP)
TraceCsg=/REGULATEUR/SaveCsg

'Chemin du fichier de command (start et stop) (Chemin FTP)
PathCdeFrontal=/REGULATEUR/Commande/CdeF.txt

'Chemin du fichier de consignes de repli (chemin local)
PathCsgRepli=C:/Outils_DSA/digesteur/CsgRepli

'Paramètres du serveur FTP
Ftp_AdrIP=***
Ftp_Login=***
Ftp_Pwd=***
```

112.16 SaveCsg

This sub-directory contains backups of all csg.txt files created with a filename including the timestamp for creating the file. These backup files are created if the ActivationTraceCsg option has been enabled (= 1) in the config.txt file:

112.17 SaveData

This sub-directory contains backups of all data.txt files created with a file name including the timestamp for creating the file. These backup files are created if the ActivationTraceData option has been enabled (= 1) in the config.txt file:

112.18 Example of controller development

112.19 Development under SIC

Develop the regulation module under SIC with the usual procedures (modules ATV, DSS, PID, MATLAB, USER1 to 9, etc.). For the moment, the Scada_Sic module can control only one SIC regulation module. For complex control (requiring several SIC regulation modules, such as 2 simultaneous PID modules, for example), it may be necessary to use the MATLAB module or a USER module. This limitation will be removed in a future release.

112.20 Craation of the Lieu.txt file

The Lieu.txt file links the field data as manipulated in the csg.txt and data.txt files and the Sic data (U, Y, and Z variables of the Scada_Sic module). This file also allows you to define a scaling factor and an offset (SicValue = a * FieldValue + b). This file is very important because it allows to translate the data comprehensible by the Digester and the module Scada_Sic, in both directions. The order of the variables is also important since the correspondence of the U, Y and Z will be done in this order:

```
[Header]
[Z]
Data.Avencq.NivAmont                                    ;0.01      -
.852
Data.Lagarel.NivCanalAmont                              ;0.01      -
.86
[Y]
Data.Lagarel.NivCanalAmont                              ;0.01      -
.86
[U]
Csg.Avencq.PosVanneDec                                  ;0.01      -
.052
Csg.Avencq.PosVanneReg                                  ;0.01      -
.298
[End]
```

In the example given above: - the variable Z(1) will correspond to 0.01 * Data.Avencq.NivAmont - 0.852, - Z(2) will be 0.01 * Data.Lagarel.NivCanalAmont - 0.86, - Y(1) will be 0.01 * Data.Lagarel. NivCanalAmont - 0.86, - Csg.Avencq.PosVanneDec will correspond to (U(1) + 0.052) / 0.01, - Csg.Avencq.PosVanneReg will correspond to (U(2) + 0.298) / 0.01.

The time stamps of the Y and Z variables are stored in A19 format in the COMMON/HOROYZ/HORODY(100),HORODZ(100)

If the offset “b” is to be initialized automatically to the initial value, it is sufficient

to put a non-numeric code. For example:

```
Data.Lagarel.NivCanalAmont                                    ;0.01      -
xx6
Csg.Avencq.PosVanneDec                                        ;0.01      -
.*
```

As soon as the reading of this offset returns a read error, the offset is initialized to the initial value read in the data.txt file. On the screen scada interface will appear the value -9999.0 to mean that this option was used. A file Lieu2.txt is created with the values of offsets updated automatically. This option is useful when retrieving and manipulating variables in relative mode, as is often the case in linear controllers, without having to enter these values manually, which is cumbersome and error-prone.

Note that sometimes the labels (ItemRegulator) of certain variables of the Csg.*** type are not the same as those of the corresponding variables in Data.***. For example, Csg.Partiteur.DebVanneRegDroite and Data.Partiteur.DebitRegulDroite. It would have been more judicious to respect this coherence. It is possible to improve this by modifying the reference SQL database that contains these ItemRegulator labels. In any case, it is essential that the file Lieu.txt contain in its headings [U] the correct label of the variable Csg. Because it is this variable that will wait and decode the Digester (Csg.Partiteur.DebVanneRegDroite in our example above). There will be a warning that this U-control variable has not been found in the DATA.TXT file, so that the number of U variables of the .REG file is not the same as the number of U correctly found in the DATA.TXT file. This is just a warning not posing any particular problem. The only possible disadvantage is that the corresponding variable U which is normally initialized to its values from the field will not be correctly initialized and will therefore be at 0. A future evolution of the SCADA module will be to make it possible to enter 2 labels, one in Csg. and the one in Data., in the case where these two labels are different:

```
[U]
Csg.Partiteur.DebVanneRegDroite                               ;0.001     -
.652
Data.Partiteur.DebitRegulDroite                               ;0.001     -
.652
[End]
```

The wording in Csg. will be used for the generation of the CSG.TXT files and the label in Data. will be used to initialize the corresponding U value.

112.21 Activation of the module Scada_Sic

Once the control module has been developed in SIC, simply activate the SCADA module with the parameters (PS lines) the same parameters as those of the origi-

nal module, preceded by a line of parameters in the format (A6,1X,I1,1X,I1,1X,I4) indicating the name of the module (eg MATLAB in the example below), the synchronization flag (0 below), the writing option on the csg.txt files (1 below) and a number of synchronization preheating iterations during which the Scada module does not erase the data.txt to speed up the calculation iterations to synchronize as quickly as possible field data, if necessary:

112.22 Editing the control data for the SCADA module

If the synchronization flag is 1, then the U commands will not only be supplied to the digester via the csg.txt files, but will also be applied to the calculation in SIC. This can make it possible to check the evolution in the simulator and in the real channel. On the other hand, it can be dangerous if sudden maneuvers create computational problems in SIC (torrential, dry run, etc.). Indeed, any SIC crash in this kind of simulation will stop the generation of csg.txt files. The instructions given in the fallback file will then be applied after the TimeoutCsg waiting period specified in the config.txt file.

Once the simulation is launched, an interface allows to follow the evolution of the exchanges of files and variables between Sic (via Sirena Scada_Sic) and the Channel (via the Digester):

112.23 Interface of the Scada module of Sic

Chapter 113

The SCILAB method

The SCILAB method starts the Scilab program at the initial time t=dt opening a DDE link and permits you to execute a .sce Scilab file at each regulation time step dtu for this module (dtu can be a multiple of the numerical time step dt).

The Scilab .sce script file will be used as detailed hereafter. The name of this file can be specified in the specific parameters of the method SCILAB (pid.sce in the example provided in dat/exemple/scilab). It may indicate that name with either .sce extension (ex .: pid.sce) or without this extension (ex .: pid). In the case where the extension is not present it will be automatically added. If this file is found in the current sub-directory where the xml file of the current project is, then it will be the one used without offering the user to choose another one, which facilitates the user's task. If this file does not exist in the current subdirectory, then a file can be selected from .sce existing files in the data subdirectory. If no name is specified, then a default name is generated and sought: module2.sce, module3.sce, etc., or a generic module"n".sce, where "n" is the controller number, starting usually at 2 (for the index 1 is used for the laws function of time, which are managed also internally as a regulator if necessary).

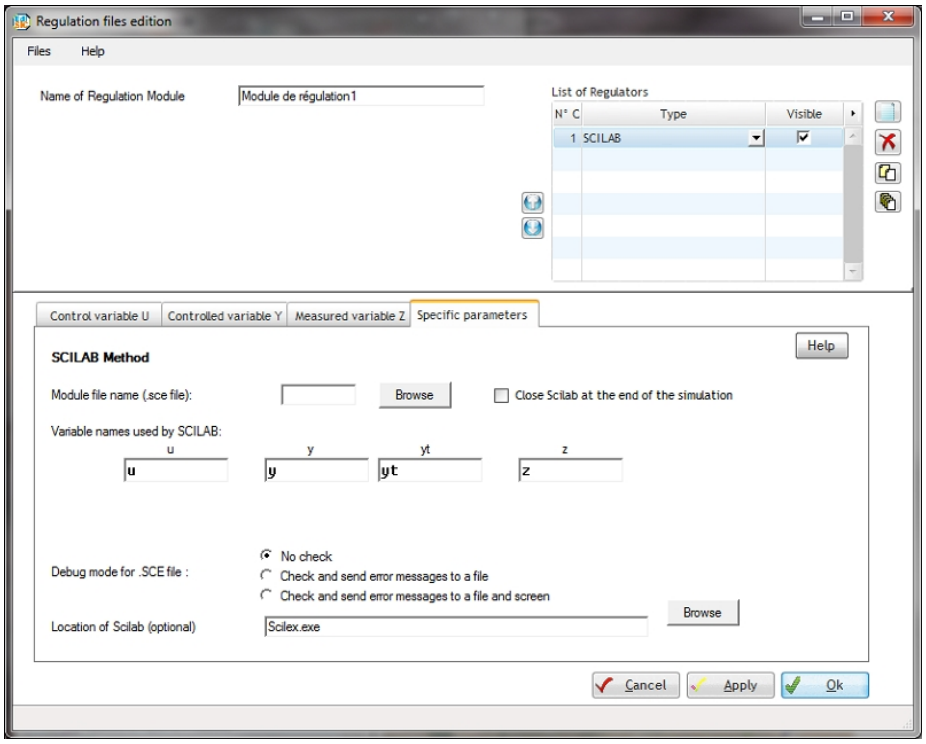


Figure 113.1: editregv5_scilab

In the interface when calculating steady flow scenarios (FLUVIA Program) or transient scenarios (SIREN Program), the names of these .sce files from the project (xml file), automatically generated or selected by the user are shown.

At this initial time step (t=dt) the SCILAB method (of the SIRENE or FLUVIA calculation program of SIC) sends to the Scilab environment (workspace) : the calculation time step dt, the regulation time step dtu, the final time tfin and an "interf" flag allowing to manage the interfaces in the .sce file according to the interface mode as defined in the option menu.

For this and every later time step t dt (multiple of dtu, see the discussion of dtu and dt), SCILAB method sends to the Scilab environment (workspace) controlled variables y, corresponding targets yt, measured variables z, current control variable positions u and current time t. The names of these variables can be redefined. These variables are vectors since they can be multiple. In Scilab the vectors are by defaut columns (size(u) = [nu 1]), contrary to Matlab where they are lines (size(u) = [1 nu]). These vectors y, yt and z are those of the current instant t. The vector u is that calculated at the current time t and which will be applied and effective from the moment of following calculation t + dt. If you want to have the history of these vectors it is for you to store them

in variables in the Scilab memory space (the variables are stored and saved in memory between 2 calls), or possibly on a file.

The SIRENE (or FLUVIA) program then executes the selected Scilab .sce file (at each time step t=n.dtu, n=1 to nmax such as t=nmax.dtu=tfin).

The SIRENE (or FLUVIA) program then retrieves the control action variables u from the Scilab workspace (as calculated by the .sce file) and applies them into SIRENE (or FLUVIA).

This SCILAB regulation method permits you to write and test quickly a regulation method while taking advantage of all Scilab development environment. In addition, it does not require any compilation or link edition of the SIRENE (or FLUVIA) program. Scilab is a free software developed under a free license and can be downloaded from the Internet. On the other hand, the regulation module execution is slower because of data exchanges between SIRENE (or FLUVIA) and Scilab, and because a .sce file is interpreted by Scilab and not compiled directly in executable code. You can check this execution time since we provide the same example with the intrinsic PID regulation module, the Scilab and the MatLab DDE link and the wdlang module.

Example of .sce file (see example provided under dat/ex_scilab):

```
% Pid.sce file for a PID controller
// Initialize ----------
// Matlab and Scilab manage vectors differently
// In Matlab, by default, a vector is a line (size (u) = 1 nu)
// In Scilab , a default vector is a column (size (u) = nu 1)
// To avoid unnecessary duplication of indexing, we adapt to that convention
// Matlab and Scilab scripts are then slightly different

if t == dt
   [nu n1]=size(u);
   [ny n1]=size(y);
   eold=zeros(ny,1);
   se=eold;
   de=eold;
end;

// calcul des coefficients utiles ----------
Kp=[14.78;1.44;4.22;2.51;1.53];
Ti=[1440;1680;1200;1440;1440];
Td=zeros(nu,1);
N =zeros(nu,1);
for i=1:nu,
    if Ti(i)==0;
       Ki(i)=0;
    else
```

```
        Ki(i)=Kp(i)/Ti(i)*dt/2;
    end
    if Td(i)==0;
        Kd1(i)=0;
        Kd2(i)=0;
    else
        if N==0
            Kd1(i)=0;
            Kd2(i)=Kp(i)*Td(i)/dt;
        else
            Kd1(i)=1/(1+N(i)/Td(i)*dt);
            Kd2(i)=Kp(i)*N(i)*Kd1(i);
        end;
    end;
end;

// calcul du pid vectoriel ----------
e=yt-y;
se=se+Ki.*(eold+e);
de=Kd1.*de+Kd2.*(e-eold);
u=Kp.*e+se+de;

// sauvegarde ancien ecart ----------
eold=e;

// dessin de y et u ----------
if interf==1
//   plot2;
end;
```

For the SCILAB method, the specific parameters to give are the default name of the .sce file and the name of the variables in Scilab workspace.

A debug option allows to decide if you want messages in case of problems (with 0 you have no message, with 1 you have messages on a scilab.dbg file in the current directory, with 2 you have the same + messages displayed on the screen). The scilab.dbg file is opened in append mode and you have therefore to delete it before a new simulation if you want to forget about previous messages. We recommend using the mode 2 during the development of .sce file and switch mode 0 for routine simulations (time gain) . [RMQ : this management is functional in the DDE link Matlab, but not yet in the DDE link Scilab , Scilab module is therefore more difficult to debug from SIC, it is good to test it first directly into Scilab using a small script garnishing vectors y, yt and z, as well as variables t, dt, tfin, etc. running the script .sce testing and verifying the vector u calculated in both dimension and values].

You can specify the location of the Scilab.exe file, but in fact the Scilab.exe program that will be run will correspond to the version of the Scilab version indicated in the Windows path and to the version selected from the Options/Options menu. It also requires that the system PATH contains the reference to the good Scilab version subdirectory (the correct version you want to use), and there is no before reference to another subdirectory also contains dynamic libraries such as lapack.dll. Otherwise there will be a conflict (eg with another version of Scilab, or a version of MatLab, see below). The solutions can then be either change the path or temporarily rename the installation subdirectories of these other programs (it generally requires administrator rights to do this). It also requires the environment variable TMPDIR is set (Windows Start Menu, System, Advanced System Properties, Environment Variables, TMPDIR = C:/ tmp for example).

You can also specify if you want to close or not Scilab at the end of the simulation.

Different versions are available: from the SIC version 5.29 it is compatible with the Scilab version 4.0, 5.0.2, 5.2.2 and 5.3.3. In February 2013 the Scilab 5.4.0 version became available, but the DDE link routines have changed (the library libscilab.lib is no longer available but replaced by call_scilab.lib). Ditto for 5.5.1 and later. These new versions have not yet been tested, but we will soon, or under urgent request ! If you are using a different version, please contact us, with your libraries we can certainly compile a specific version (only in 32 bits mode for the moment).

In case you have an error or warning message (for example about the lapack.dll library), you may have a conflict between some MatLab and/or Scilab versions installed on your computer. The one found first into the Windows path definition, and the one selected into the Options/Options must be the same. You can also temporary change the name of the installation directory of some other version(s) to prevent this conflict. The reason is that different versions of MatLab and Scilab use different versions of some libraries (including the lapack one).

Chapter 114

The STEPS method

The STEPS method is used to generate step changes on the U command(s).

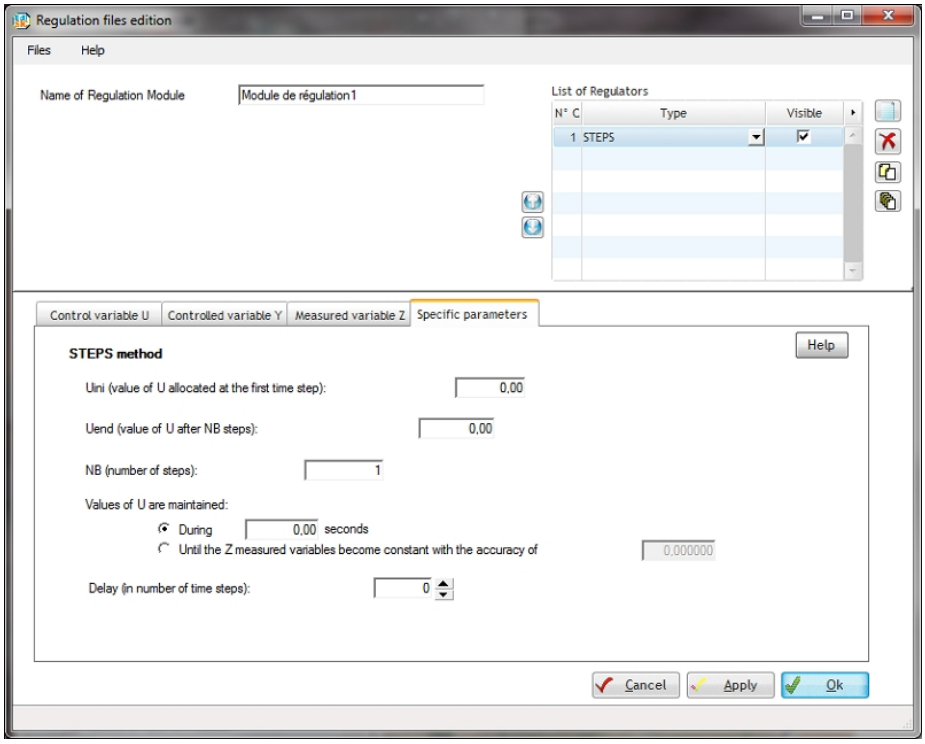


Figure 114.1: editregv5_steps

The following parameters are read on the .xml file on the PS= tags

READ(U,900,ERR=10) UMIN,UMAX,NB,MODE,XSWITCH,IDELAI 900 FORMAT(2(F8.2,1X),I2,1X,I1,1X,F8.2,1X,I2)

where: STEPS is a module that will modify the values of U between UMIN and UMAX, UMIN is allocated to U from the first time step, via NB steps of the same value (NB $>$ 0), and up to UMAX, (note: you can have UMIN $<$ or $>$ UMAX) MODE defines how to go from one value to another: MODE = 0: the values of U are held during XSWITCH seconds, MODE = 1: the values of U are maintained until the measured variables Z are constant to the accuracy XSWITCH. In this case, the selected Z variables must be entered in I (incremental) mode. IDELAI allows to delay the step (for example to check that it is in steady state, or to synchronize with other modules like USER1, LQG for example). It is given in number of time steps (integer I2 $>=$ 0).

Remark: If Y variables are described on the .xml file for this module, their values will be written to the .lst file just before the step is generated. This makes it possible to check the status of certain hydraulic variables just before generations of steps.

Chapter 115

The STOP method

The STOP method interrupts the SIRENE program calculation, indicates the first 5 measured variables z1 to z5 and the first 5 control action variables u1 to u5 to the user. This allows the user to input new values for the control action variables u and to choose the next intervention time (in number of time steps).

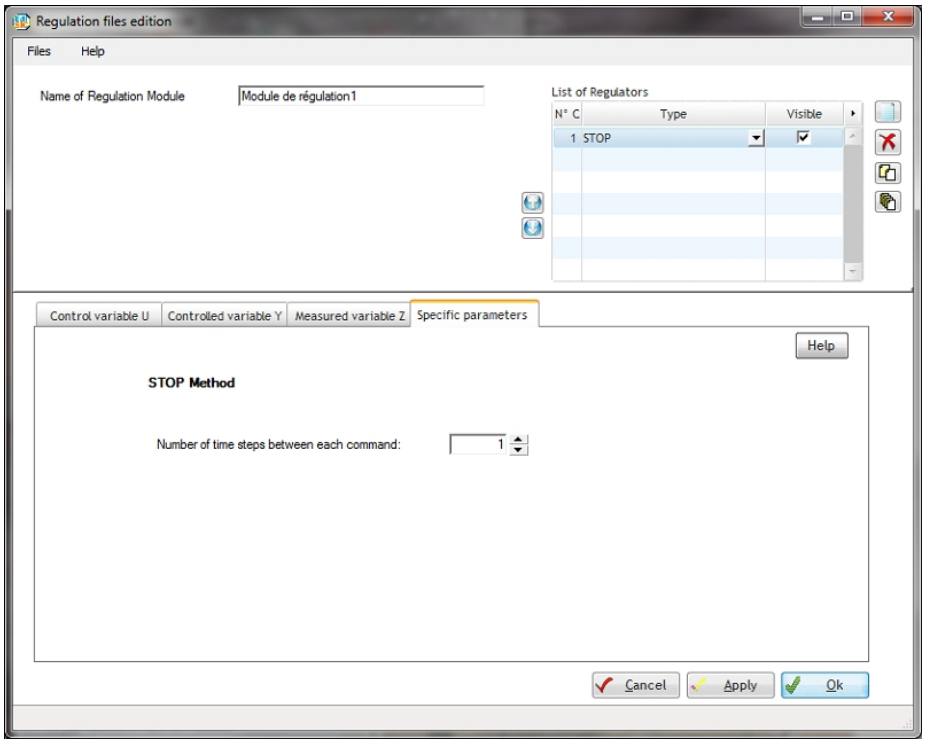


Figure 115.1: editregv5_stop

For the “STOP” method it is the number of time steps between each command with the format (F8.2).

PS = 1.00

Chapter 116

The USER1 to USER9 methods

The method USER1 permits you to write directly in FORTRAN a new regulation module. We can program it in the same way as for the CPID routine (see above). This method requires a compilation of this module by the Microsoft Digital 6.1 or above compiler (32 bit compiler) and a new link edition by the linker associated. On the other hand, the program execution is very fast. To write this module you can take the inspiration from the PID module presented above. We can imagine that we finalise a method thanks to the MATLAB method and when it is finalised and tested, we integrate it in the SIC software by the USER method to increase the execution speed.

We can program up to 9 regulation modules by using USER1 to USER9 methods. For each one, there are 3 routines that may need to be informed in the SIRVID.FOR file:

```
SUBROUTINE SUSER1
END

SUBROUTINE LUSER1(CHAINE,PARA)
CHARACTER CHAINE*(*)
DIMENSION PARA(*)
END

SUBROUTINE CUSER1(U,Y,YT,Z,PARA)
DIMENSION U(*), Y(*), YT(*), Z(*), PARA(*)
END
```

The SUSER1 routine is a supervision routine called at each time step. It gives you the possibility to change parameters of a regulation module, of method type,

etc.

The LUSER1 (CHAINE,PARA) routine is only called at the beginning of the program, at the initial time. It is called every time that a "PS=" line is met for this method USER1 in the .REG file. It permits you to read the specific parameters of the USER1 method, in the chain of character (CHAIN) placed after a "PS=" code . If several "PS=" lines must be read then it is necessary to manage a line meter in this LUSER1 routine, permitting you to read different variables in different lines. The PARA vector is sized at 50 (it was 10, 20 or 22 in older versions of SIC), this means that 50 maximum parameters can be read and can be stored by this reading routine. If you want to read and to store more than 50 parameters you must choose and use another common. Common names already used (and therefore to be avoided) are specified at the beginning of the SIRVID.FOR file.

The routine CUSER1 is a regulation routine called for each time step, after the SUSER1 supervisory routine. It permits you to calculate control action variables U from the controlled variables Y, corresponding orders YT, measured variables Z and from parameters placed in the table PARA. The number, type and site of these variables are described in the regulation file .REG.

Chapter 117

The WDLANG method

The WDlang method is used to write a control module in WDlanguage as defined in the Windev development environment. Help on this language is available on the site http://www.pcsoft.fr.

This functionality, carried out in WinDev, is intended to offer an alternative to the current DDE link with MATLAB or SCILAB or to the Fortran interface used with the FLUVIA and / or SIRENE programs for programming regulation modules. This new functionality does not require any third-party software, unlike the DDE MatLab link which requires the MatLab software, the DDE SciLab link which requires the SciLab software (free software under the GNU license) or the Fortran interface which requires a Fortran compiler (Digital or Intel).

This window offers an operation similar to that of the DDE MATLAB or SCILAB links. It applies laws to input data (Y, YT, Z) to generate output data (U) calculation results and possibly using parameters (PARA). Like MATLAB, this window offers the user the possibility of writing their own calculation laws. The variables Y, YT, Z and U are optional, you don't have to define and use them all. One can for example just use Z variables to visualize them numerically or graphically, or one can just use U variables to modify them according to time for example, or read them on an external file (Open Loop type command). All of them can of course be used for example to calculate U commands from measured variables Z, from controlled variables Y, possibly assigned with YT setpoints (Closed Loop type command).

Here is an overview of the window, composed in the upper part of the data tables, results, and in the lower part of the algorithm to be executed.

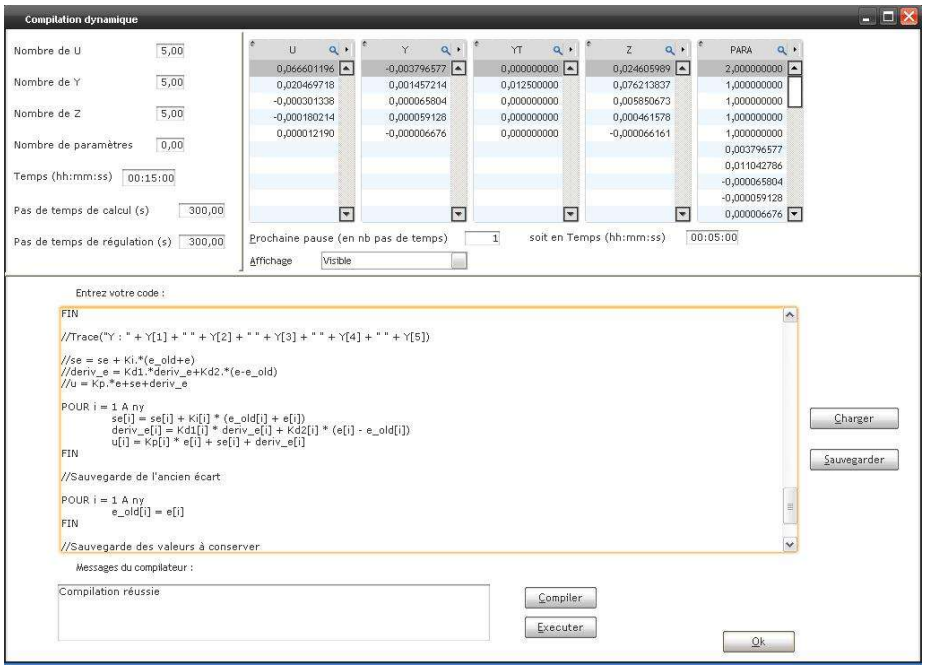


Figure 117.1: doc109

The window receives the data in the form of four tables of reals called “Y, YT, Z and PARA” that the Fluvia and Sirene programs for hydraulic calculation in permanent and transient mode respectively (written in Fortran) provide information at each regulation time step.

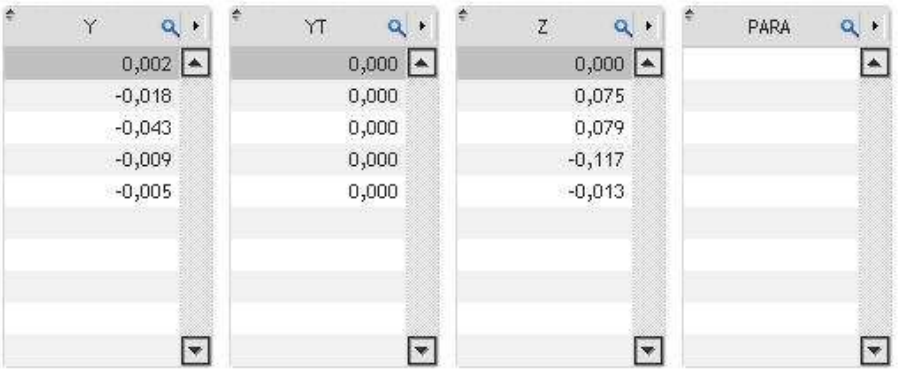


Figure 117.2: doc110

The window will then execute a code (calculation law), entered by the user or even loaded from a text file. If the “wdroutine.txt” file exists in the data subdirectory, it will be loaded on the first call. Otherwise it will be possible to select another file with a .txt extension, or to write it in the input window (and save it).

The purpose of this law is, for example, to inform the result table “U” by following the algorithm of the calculation law (written in WLanguage).

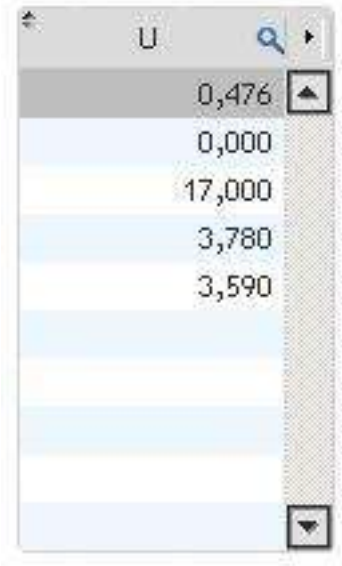


Figure 117.3: doc111

The Fluvia or Sirene program informs the input tables (Y, YT, Z PARA) at each regulation time step, starts the calculation of the algorithm, recovers the calculated output variables (U) as well as the other modifiable variables on the screen.

You can switch the window to invisible mode to speed up the calculations (the display of the window and the calculations are not performed until the next time you get started). One can also ask to take back control only in several steps of computation time (or at a future moment).

The user can then write code in WLanguage in the control provided for this purpose in the lower part of the window, save it as a text file, or load an existing file.

Compile button: Compiles the code written in the text area. This phase is necessary if the code has been modified, or if another file has been loaded. Messages are indicated if there are compilation errors, or if on the contrary everything went well.

Execute button: Executes the last successfully compiled code, without incrementing the time, and without returning to the hydraulic calculation of Fluvia or Sirene.

Continue button: Executes the last successfully compiled code and gives control to the Fluvia or Sirene program, to repeat the hydraulic simulation.

A certain number of variables are accessible and modifiable from the code:

The PARA variables correspond to the parameter table. It is displayed on the right of the window. This array is sized to 50 real variables in current versions of SIC^2, but may be increased as needed for future versions (it was 10, 20 or 22 variables in previous SIC versions). These are persistent variables between 2 calls to the code. The first 5 PARA variables are used by the WDLANG regulation module. They can be modified by the code but this will have consequences on the functioning of the module:

PARA [1] for visible window with refreshing of the values at each DTU (if PARA [1] = 2) or invisible (if PARA [1] = 1) except screen input (read on the .xml data file, in the data module specific, modifiable on screen) PARA [2] for counter for screen entry (max value) (idem) PARA [3] for counter for screen entry (current value) PARA [4] for counter for execution of the calculation (max value) (read on .reg, not modifiable on the screen) PARA [5] for counter for execution of the calculation (current value)

Other variables are accessible: T: current time in seconds DT: numerical calculation time step in seconds DTU: no control module call time in seconds TFIN: simulation end time NGR: regulator number INU: number of U variables INY: number of Y variables INZ: number of Z variables INPARA: number of parameters (currently 50 by default)

T and DT are controlled by the Fortran Fluvia and / or Sirene program. Modifying them will have no consequence, except in the lines of code following this modification, and for the current execution. They will be reset to their initial values on the next call. DTU can be modified either in the code, or thanks to the field “Next pause” in Number of time steps or in Time. TFIN can be changed in the code, but only by shortening it. Otherwise there will be problems writing the results to the file, at least in binary format. The other parameters above cannot be changed, or in any case it will have no effect.

You can also have access to 2 Chart controls (see this type of control in the Windev WdLangue language documentation on their website www.windev.com) to draw whatever you want, and for example the U variables (s) , Y, YT and / or Z. But also other things like intermediate calculations of the programmed algorithm for example. These Graphs fields are named GRFU and GRFZ. They are visible on the 2nd tab of the window. The programming code to use them is for example:

```
i is an integer
for i=1 to INU
   grNuageAjouteDonnéeXY(GRFU,i,t,u[i])
end
for i=1 to INZ
   grNuageAjouteDonnéeXY(GRFZ,i,t,z[i])
end
grDessine(GRFU)
grDessine(GRFZ)
```

These variables, as well as the u, y, yt, z, para, etc. can be used in upper or lower case. Before compilation, there is a formatting and modification of the source of the programmed code, so that it uses the names of the variables understandable by Windev, i.e. by using the name of the global variables (by adding the COL_RegWDLang. extension) or referring to the name of the Windev window (regWDLang.). The source thus modified is visible on the 3rd tab. This information is only given for experts who want to understand how it works. This formatting also changes the lowercase -> uppercase variables, and its visualization on the 3rd tab also indicates the line number, which facilitates debugging, since this line is indicated in the dynamic compiler message returns.

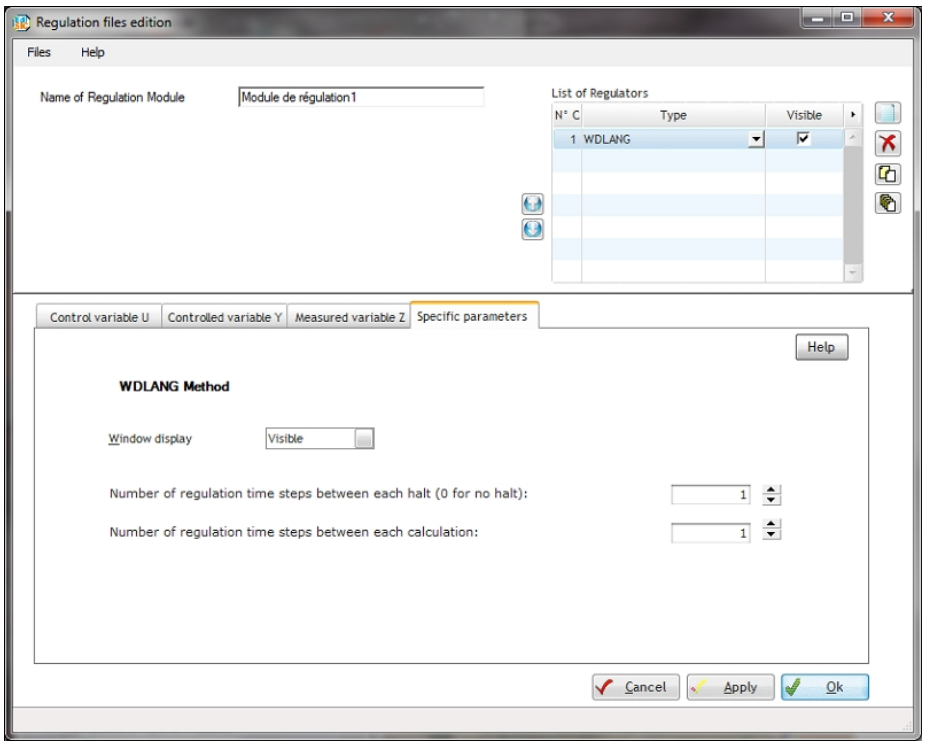


Figure 117.4: editregv5_wdlang

Format of the line PS : `(F8.2,1X,F8.2,1X,F8.2)`

Chapter 118

Presentation

The ALGEQUEAU module adds quality simulation capabilities to FLUVIA and SIRENE. It was first developed to medel algal development in irrigation networks.

An hydraulic scenario may not contain more than one ALGEQUEAU module, but hydraulic data used for quality simulation can come from any variant of this scenario.

The first step when building a quality simulation is to define the quality classes, which may be : - a temperature (°C) ; - a nutrient concentration (mg/L) ; - a concentration of drift algae (kg/m3) ; - a linear density of fixed algae (kg/m) - a sediment concentration in water (g/m3) ; - a linear density of sediment in the bed (m3/m).

Their evolution is ruled by transport and exchanges, following exchange laws depending on hydraulic and quality parameters.

For now, the module can : - transport temperature, nutrient, drift algae and sediments along the stream, following the transport equation ; - calculate at a section the exchanges between classes ; - calculate at a node the downstream repartition of transported classes.

Chapter 119

Adding an Quality module to an hydraulic scenario

One Quality module can be created for each hydraulic scenatio.

When editing an hydraulic scenario, one can enter the Quality module by chosing «Quality module» in the Tools menu.

The next step is to define the quality classes.

The module can then be accessed through the project explorer if at least one quality class is defined. The module can also be deleted through the project explorer.

Chapter 120

Definition of the quality classes

The definition of quality classes is accessible through the  button or Tools >  Quality classes definition.

One can add a new quality class with the “add a quality class” button.

Each quality class is given a name and a type. Depending on the type, different functionalities are available.

120.1 Drift quality classes

Drift quality classes such as nutrients, temperature, and drif algae or sediment must have boundary conditions on offtakes. Their initial condition is then obtained by a steady-state simulation.

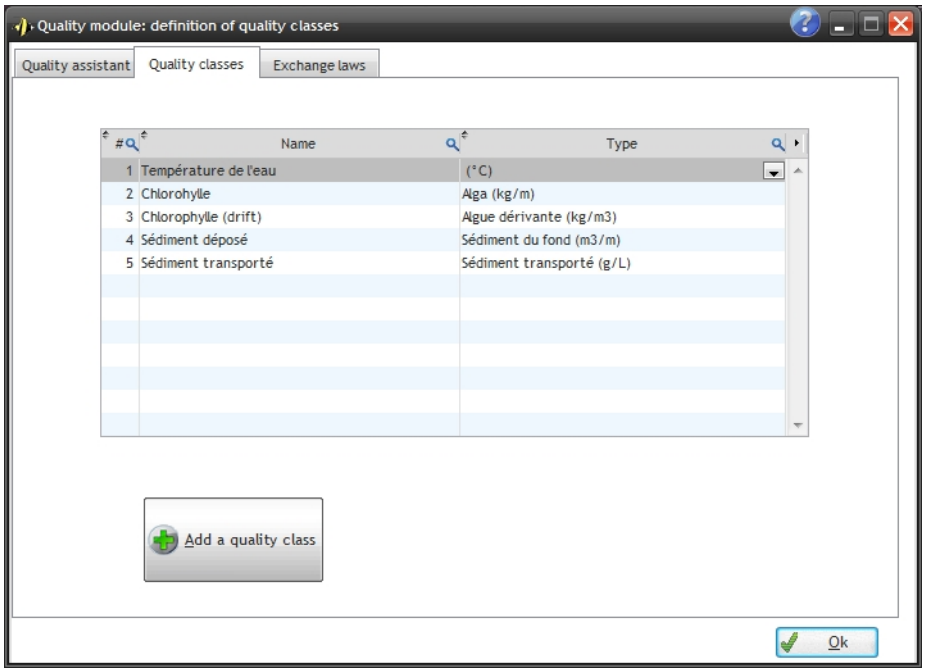
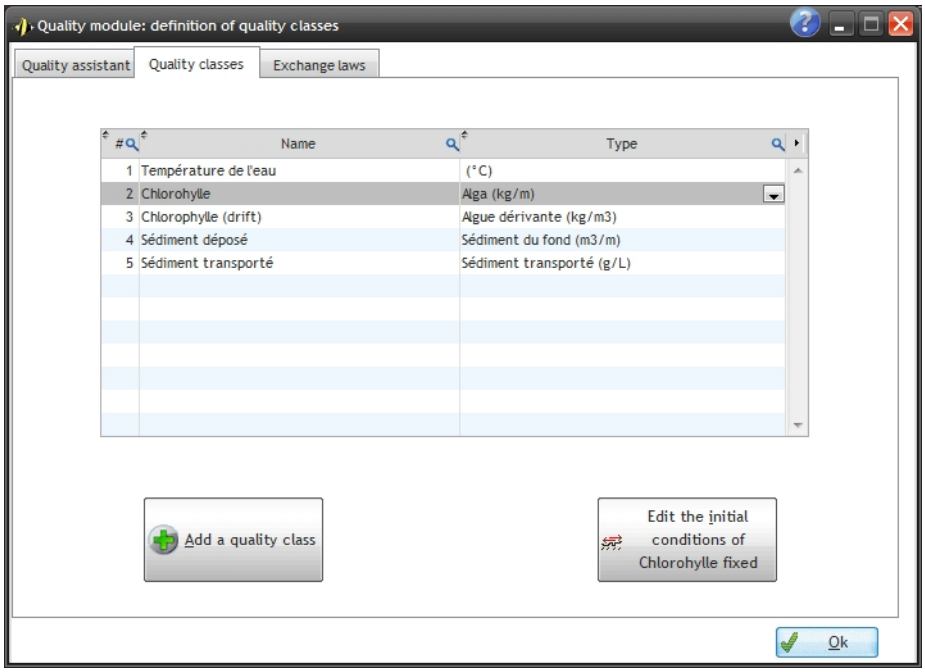


Figure 120.1: qualite_defclasse_temperature

120.2 Fix quality classes

Fix quality classes, which are not transported by the stream and only vary through user-defined exchange laws can be defined as fix sediment or fix algae. The channel geometry is modified according to the quantity of deposited or eroded sediment. For these classes, the user is invited to define the initial value.


Quality module: definition of quality classes
Quality assistant
Quality classes
Exchange laws
Name Type
1 Température de l'eau (°C)
2 Chlorohylle Alga (kg/m)
3 Chlorophylle (drift) Algue dérivante (kg/m3)
4 Sédiment déposé Sédiment du fond (m3/m)
5 Sédiment transporté Sédiment transporté (g/L)
Add a quality class
Edit the initial conditions of Chlorohylle fixed
Ok

Chapter 121

Initial conditions for fix classes

Initial conditions definition for fix classes is accessible through the quality classes definition window, with the “Edit the initial conditions of …” button.

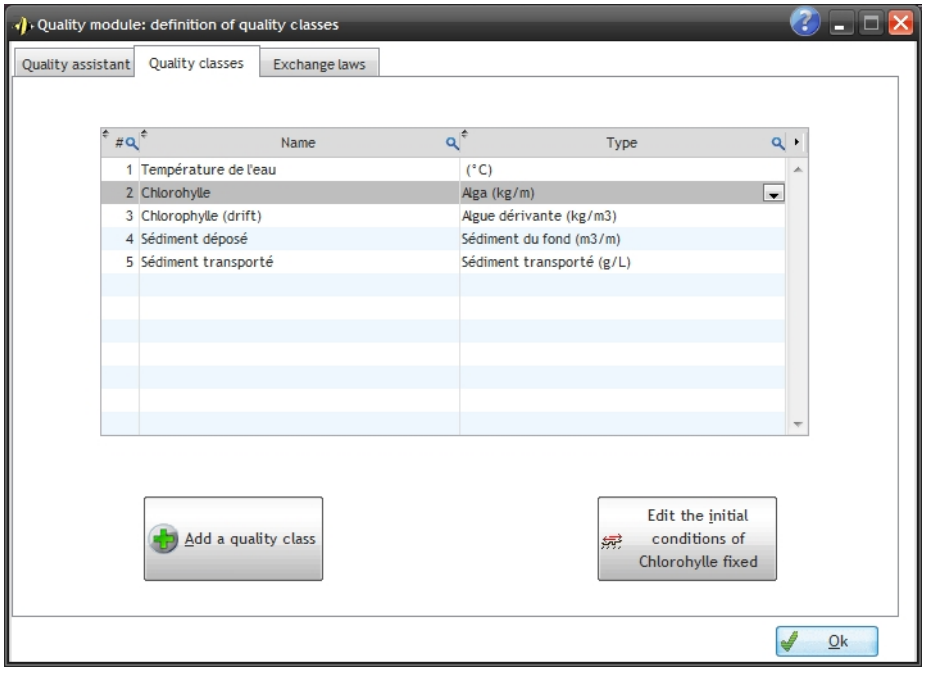


Figure 121.1: qualite_defclasse_algues

Initial conditions are defined the same way as Manning-Strickler coefficient or infiltration data in hydraulic mode : one value is given for a reach part, with the upstream and downstream limits abscissas for this reach part.

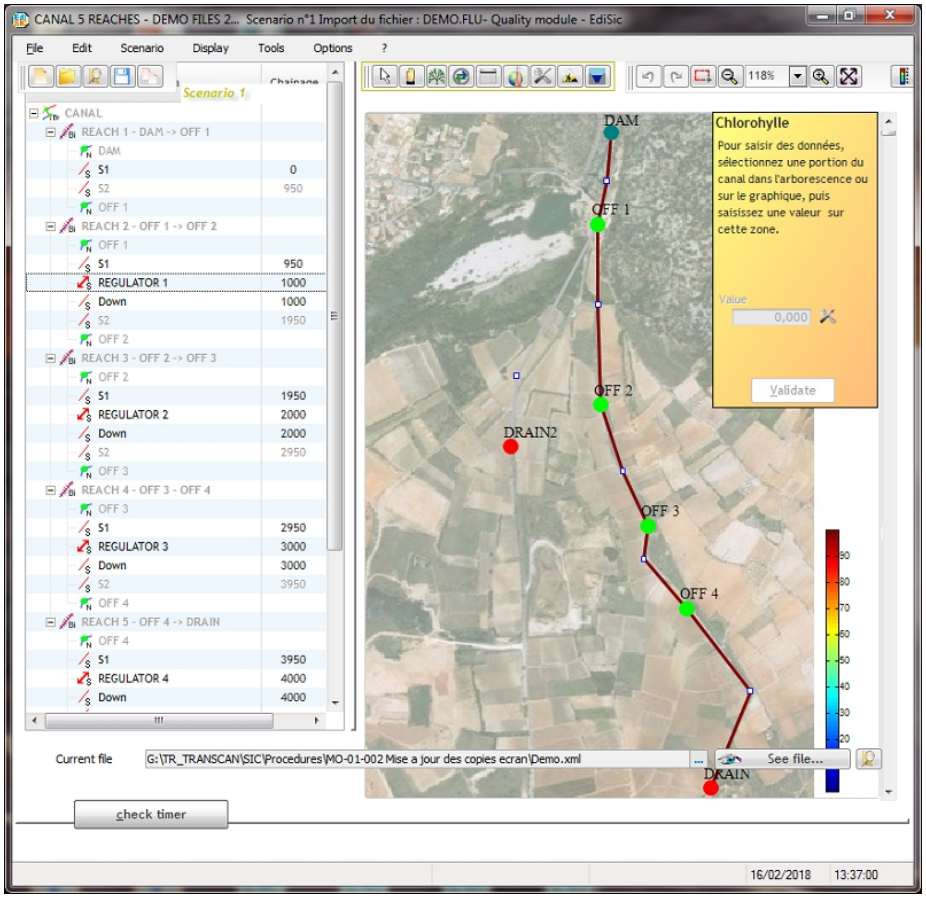


Figure 121.2: premiere_algequeau_ci

Chapter 122

Exchange laws

A set of exchange laws should be defined to model the interactions between classes.

For each law, the user should click the “add a law” button, and then write a description, and select the type.

For instance, for the temperature evolution law, only one acting class should be defined, the temperature.

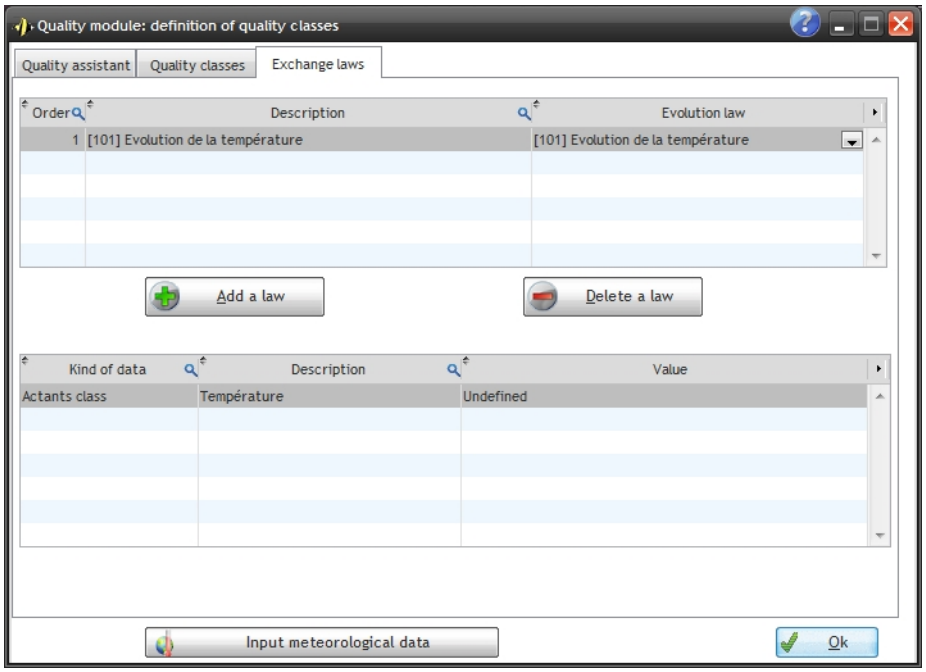


Figure 122.1: qualite_defclasse_loi_101

Chapter 123

Boundary conditions

Drift quality classes simulated in ALGEQUEAU are transported from upstream to downstream, and the input concentrations are therefore needed. Furthermore, the distribution at a node can be heterogeneous and requires the definition of a distribution coefficient. All these parameters can be set in through the boundary conditions window, accessible through the  button or the Tools > Boundary conditions menu.

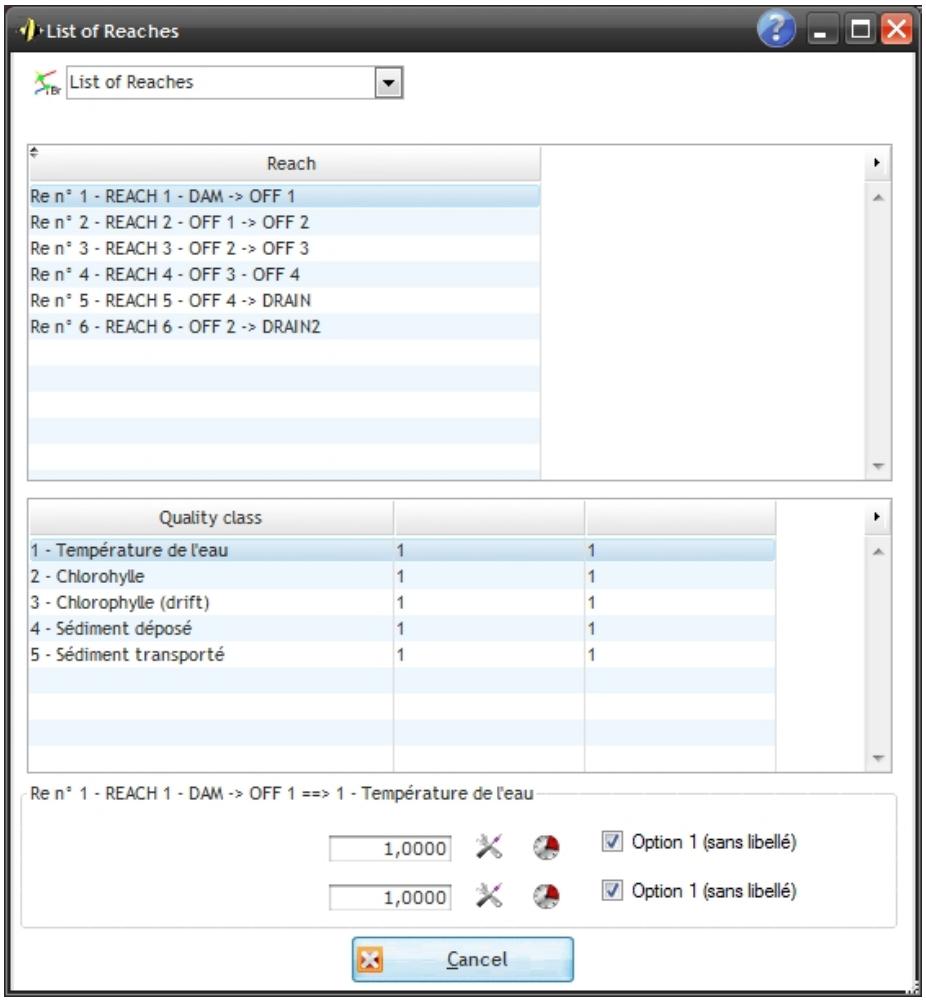


Figure 123.1: qualite_condlim

The first drop-down menu allows to choose on which type of network element we want to intervene : - Reaches : k on upstream and downstream (in case of flow-inversion) ; - Cross-structures : k ; - Node : k for infiltration in the pond ; - offtake : input (= boundary condition) for inflow and k for outflow.

The second menu allows the user to choose the element, and the third is for the quality class.

The user can then set the required parameters at the bottom of the window.

Distribution coefficients k may be adjustable in order to ensure mass conservation.

Chapter 124

Meteorological parameters

Meteorological parameters are entered by clicking on the  button or through Tools > meteorological parameters :

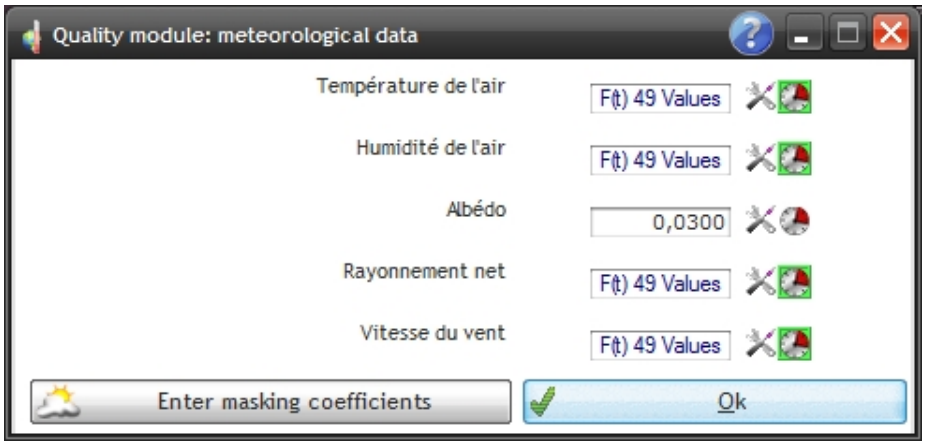


Figure 124.1: qualite_meteo

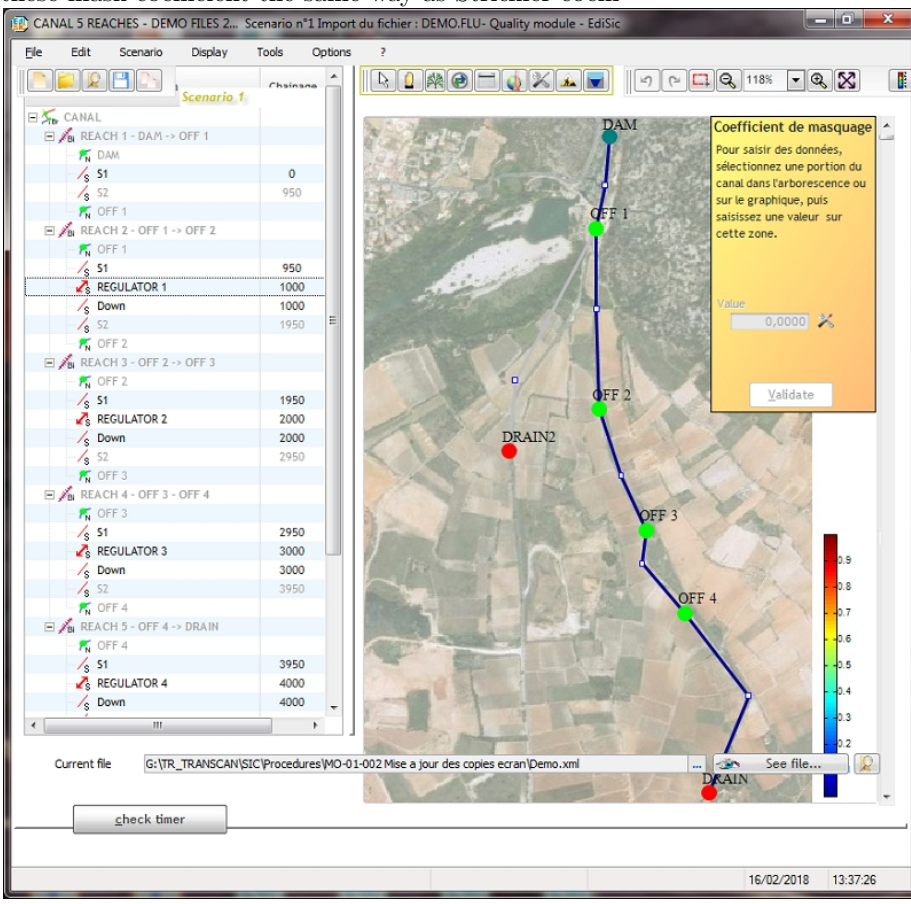
In this windiw, the user can type in the following parameters : - Air Temperature - Air moisture (between 0 et 1) - Albedo (between 0 et 1) - Radiation (W.m-2) - Wind velocity (m.s-1)

All these parameter can vary with time.

124.1 Mask coefficients

The presence of tunnels or shade is raken into account through a mask coefficient between 0 and 1, accessible by clickink on the “Mask coefficient” button at the bottom of the Meteorological parameters window.

The user can then define these mask coefficient the same way as Strickler coefficients or infiltration data.



The mask coefficient is a constant, and shade varying with the angle of the sun during the day is not yet possible.

Chapter 125

Calculation settings

Calculation settings are available through the  button or the Tools > Calculation settings menu.

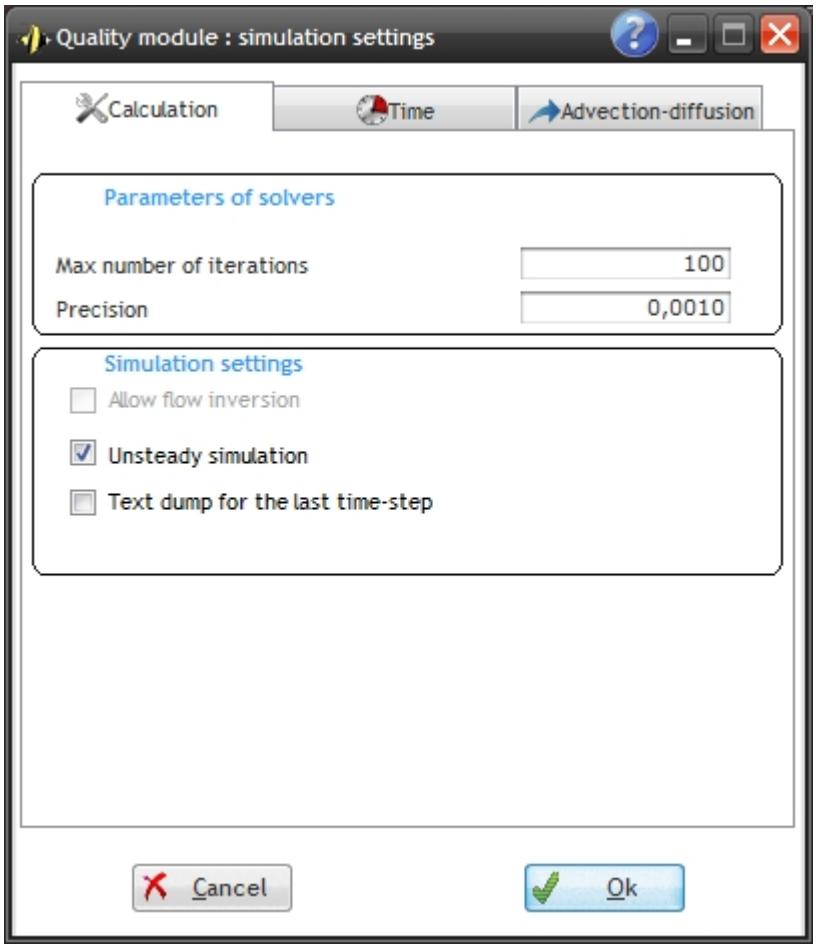


Figure 125.1: qualite_param

125.1 Iteration parameters

Calculation at a section or a node use an iteration method (Newton or conjugate gradient). The user can define a maximum number of iteration and a precision.

If the required precision is not reached in the maximum number of iterations, the calculation will stop with an error warning : “Convergence problem in the section.”

125.2 Simulation settings

125.3 Transient simulation

The first time-step of any transient calculation is calculated with the steady-state algorithm, in order to have an initial condition. If this option is unchecked, all time-steps are calculated in steady-state.

125.4 Text-dump of last time-step

When this option is activated, all data for the last time-step are written in a text file with fixed columns.

The file is created in the same folder as the XML project file, and is named [XML-File's name]_FLUVIA.ALG or [XMLFile's name]_SIRENE.ALG depending on the hydraulic program used.

Le fichier sera crée dans le même dossier que le fichier XML et sera nommé NomDuFichierXml_FLUVIA.ALG ou NomDuFichierXml_SIRENE.ALG en fonction du calcul hydraulique utilisé.

125.5 Time

These parameters have the same definition as their hydraulic counterparts.

The beginning and end time of a simulation and the time-step are not necessarily the same for quality and hydraulic simulations :

- is the time steps are different, hydraulic data used for quality calculation are interpolated ;
- the actual simulated period will be the intersection of hydraulic and quality time period (cf. beyond).

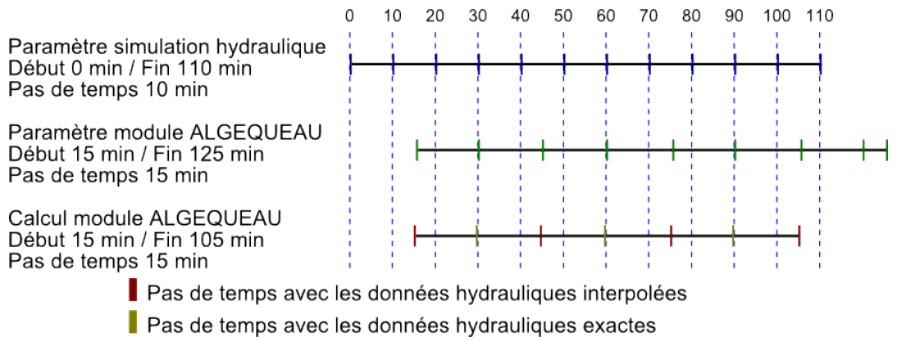


Figure 125.2: doc34

Chapter 126

How to make a network

Launch EdiSIC with the program's icon  located on the Windows® desktop or in the "Start" menu.

If a project is already open, create a new project (File Menu> New or  button) and save the file in XML format.

It is possible to use a base map to facilitate the capture and understanding of the network (see Using a base map).

The first step in creating the network is to describe its topology (ie, draw nodes and reaches). This step is described on page Creating a network (nodes and reaches).

Then, describe the geometry of nodes and reaches: - The geometry of the nodes only means to describe the pond located in the node (see Description of node and pond). - The geometry of the reaches is to describe sections in each reach (see Creating sections and description of cross sections).

At each record of the file, if there are inconsistencies in the network, the software prompts to view and correct them (see Managing inconsistencies in the network).

Once we have obtained a coherent network, we can start the mesh network (see Execution of the mesh (Talweg)).

Chapter 127

How to perform a steady state simulation (FLUVIA)

To perform a simulation in steady state, we must first have built a Network and successfully run Talweg on it.

Then you have to edit an hydraulic scenario for entering the data required to perform a simulation in steady state. To do so, open the Project Explorer, select the scenario created by default or create a new scenario, and click the “Edit scenario” button.

Type of information that can be entered are summarized on the presentation page of the three modes (geometry, hydraulic and result).

The minimum information to be entered and execution of the calculation software (FLUVIA) are described in the “Running a steady state calculation (FLUVIA).”

If the calculation was successful, it is then possible to view the results.

Chapter 128

How to perform a transient simulation (SIRENE)

In order to perform a transient calculation, it is first necessary at least to have been able to carry out a steady-state calculation or another calculation in transient mode which will serve as an initial condition for the calculation. In addition, the calculation used as an initial condition must have been carried out with the parameter “Write in all calculation sections” checked. This parameter is checked by default for the permanent calculation parameters and is unchecked by default for transient calculation parameters.

128.1 Creating a new scenario or variant

The calculation of this new simulation will be carried out either in a new scenario or in a variant of the scenario serving as an initial condition. The first operation to be performed in the project explorer is therefore to choose: - Duplicate the initial condition scenario - Create a new variant in the initial condition scenario



| Since transient computation requires an initial condition, it is important that the boundary conditions and the parameters of the start of the transient simulation are compatible with the initial waterline originating from a previous calculation. To obtain this guarantee, it is advisable to create a variant in the scenario where the calculation is used as the initial condition for the transient calculation. |

128.2 Import of initial condition

In the Project Explorer, go to the line of the scenario or variant where the transient calculation is to be done and click on the " Import C.I." button (C.I. = Initial Condition). Choose from the list of results present the calculation results which will serve as initial condition.

Once this is done, the " Transient Calculation" button becomes active.

128.3 Edit Transitional Computing Events

The user will then enter the changes that will take place on the system over time (Variation of supplied or retrieved flow rates, operation of devices …). This is done by the seizure of laws as a function of time.



| If the types of boundary conditions do not change between the permanent scenario and the transient calculation, it is advisable to enter the time-dependent laws directly in the permanent scenario. The permanent calculation will be carried out on the first time step only and the transient calculation in a variant (which does not contain any modification with respect to the scenario) will calculate from the second time step on the data of the scenario. |

128.4 Calculation Parameters

The user will have to define the time parameters of the simulation as presented on the page Time Settings. It may also possibly modify the parameters of the transient computation and the writing parameters of the results.

128.5 Start simulation

Follow the instructions on page Performing a Transient Calculation (SIRENE)

Chapter 129

Simulating a check valve with a WDLANG module

SIC does not include any device or regulation module of the "check valve" type, a device that only allows the discharge to flow in one direction. To model this type of device, you must program a regulation module yourself. This is what this tutorial proposes to do using the WDLANG module.

129.1 Creation of the module and connection to the devices

In hydraulic mode, go to the tools menu > Regulation module

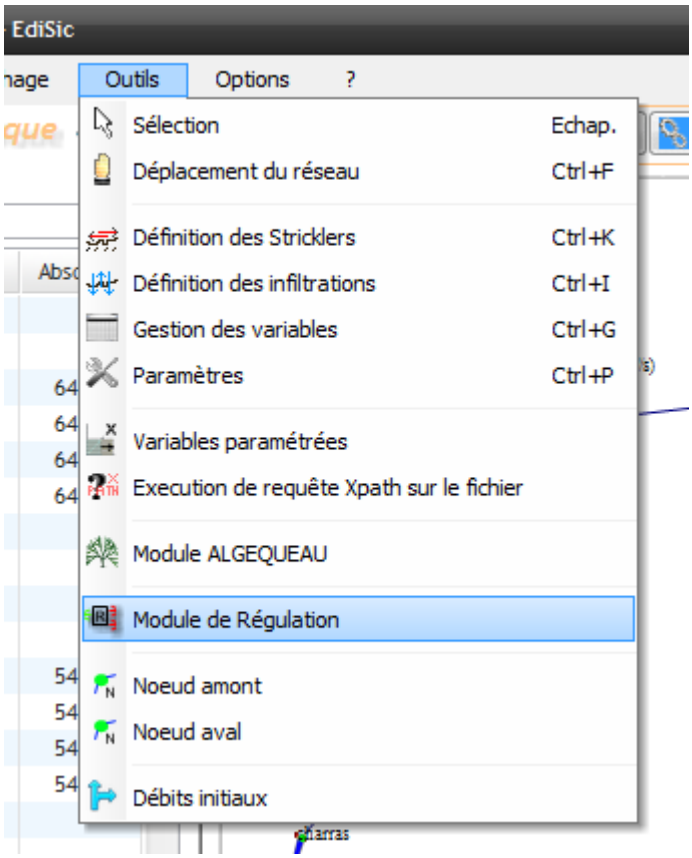


Figure 129.1: doc81

In the window that opens, click on « New », give a name to the module, then click on « Modify »:

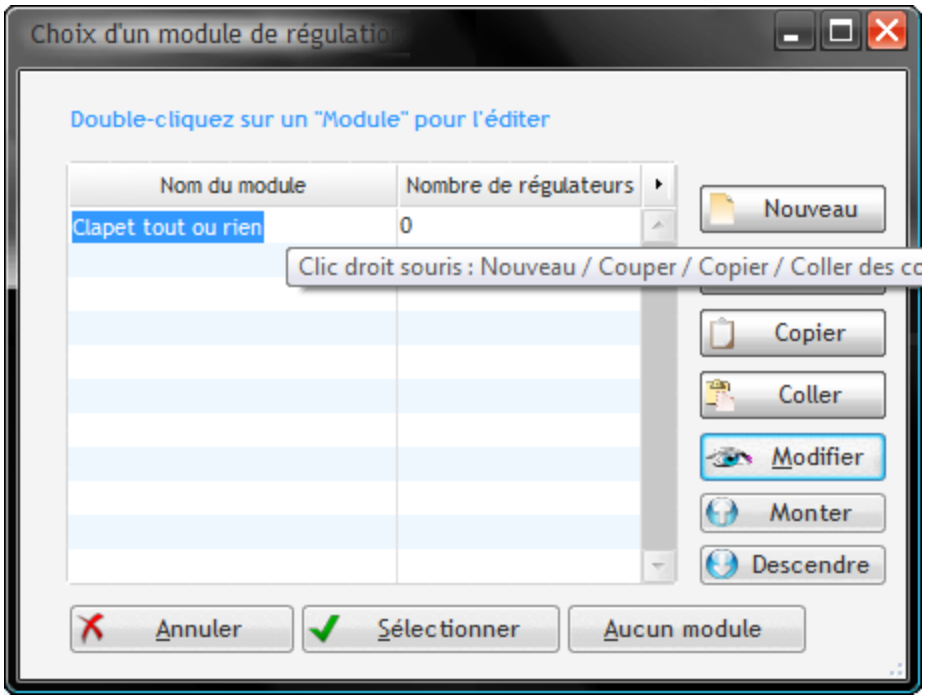


Figure 129.2: doc82

In the regulation modules editing window, add a WDLANG regulator by choosing in the drop-down list at the top right:

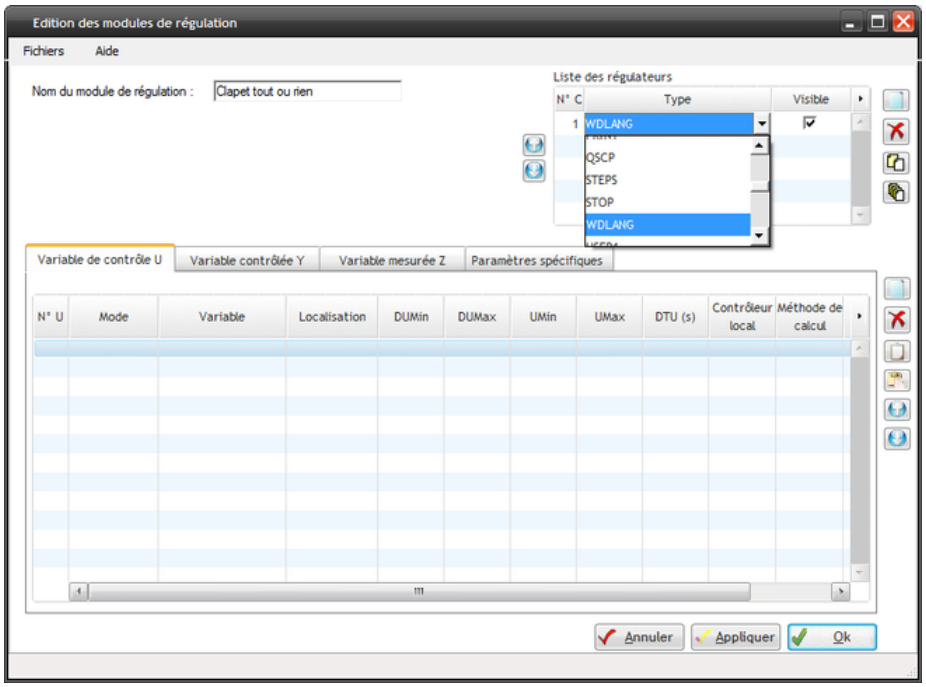


Figure 129.3: doc83

Add a control variable U by clicking on the first row of the table and choose the opening of a gate where the check valve must be placed:

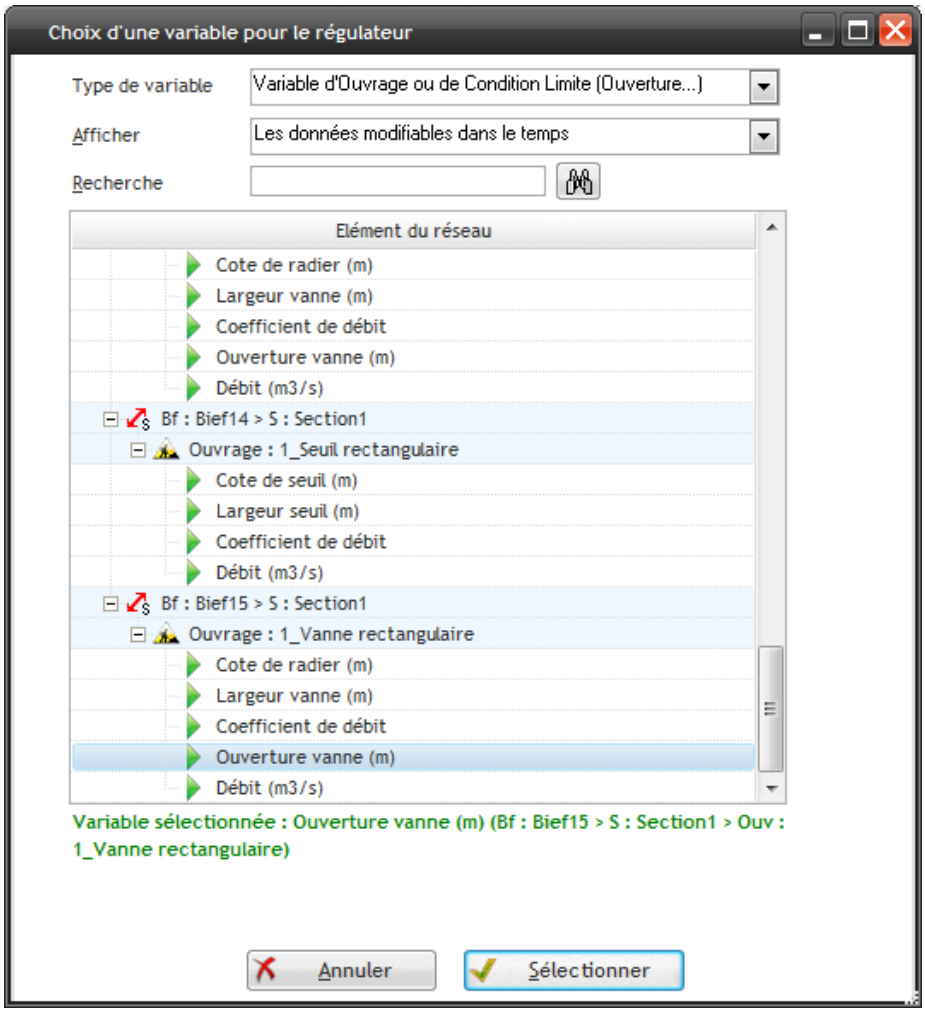


Figure 129.4: doc84

Define the parameters of the control variable: for a check valve, you must use an absolute mode command (i.e. the value sent by U will be equal to the value of the opening) and you can optionally define Umin and Umax so that they correspond to the minimum and maximum opening of the gate in case the regulation algorithm would return values outside the range. Then, we must define the hydraulic measurements that will allow us to command the check valve. I suggest putting the upstream water elevation in the controlled variable (Y) and the downstream water elevation in the measured variable (Z).

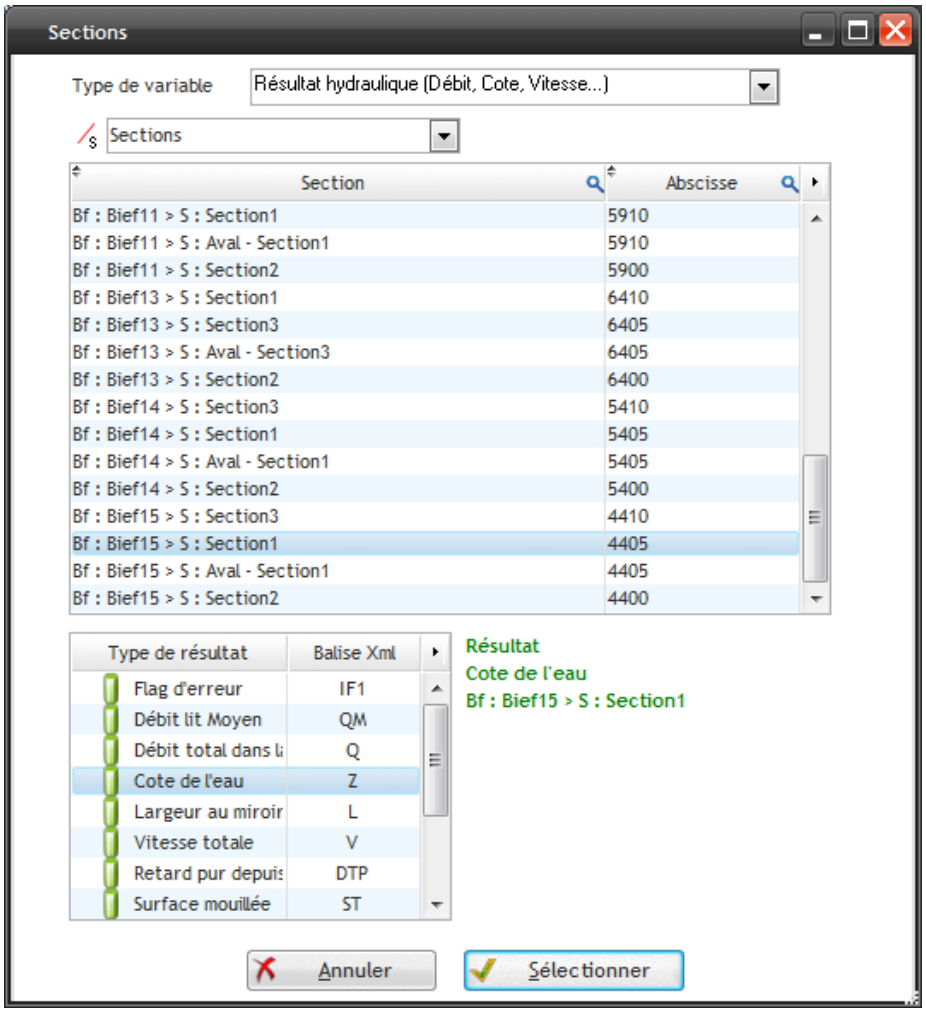


Figure 129.5: doc85

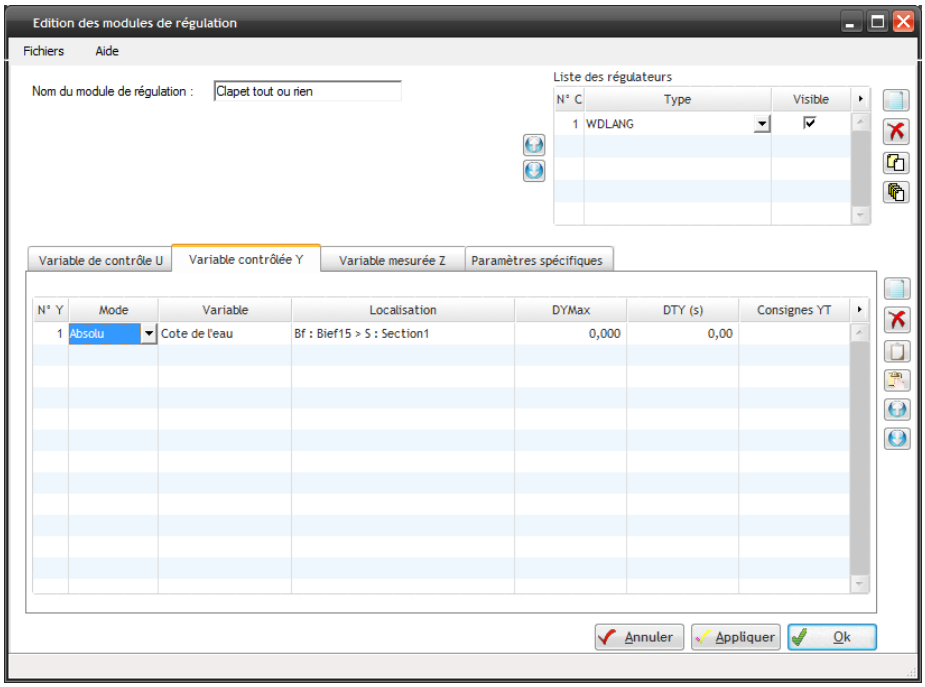


Figure 129.6: doc86

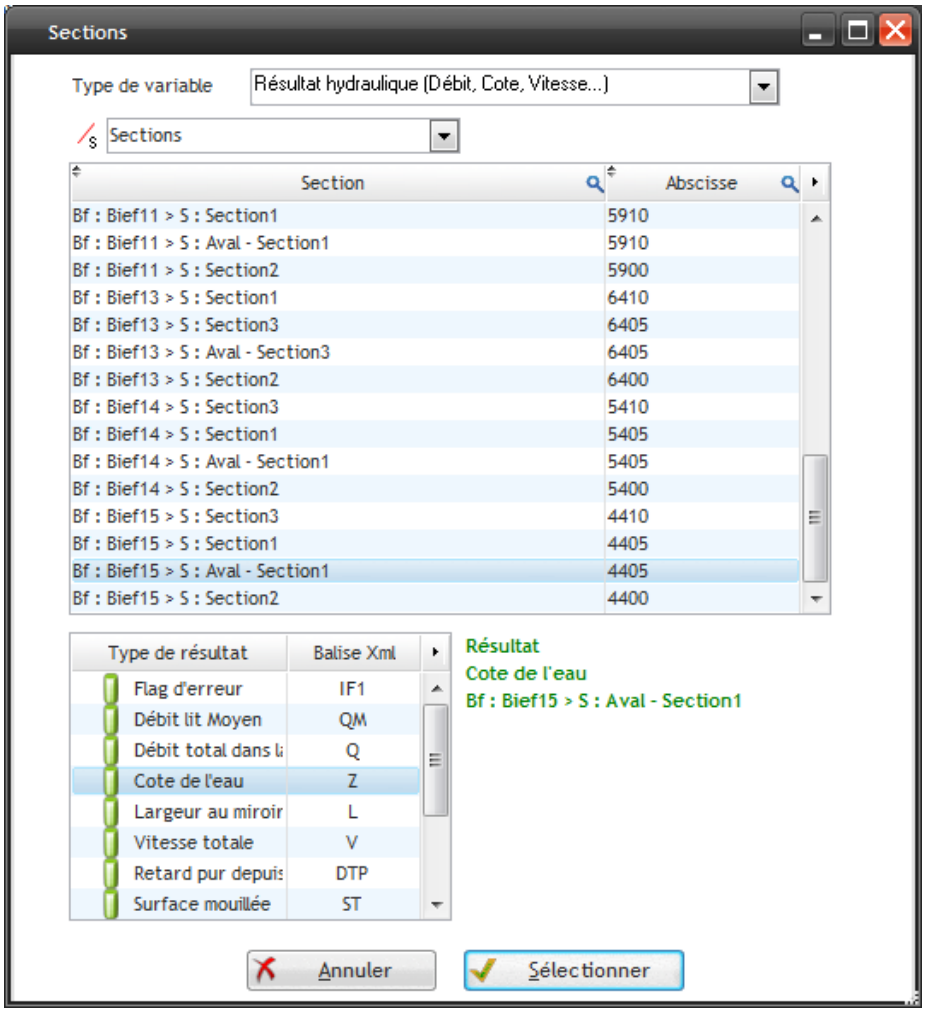


Figure 129.7: doc87

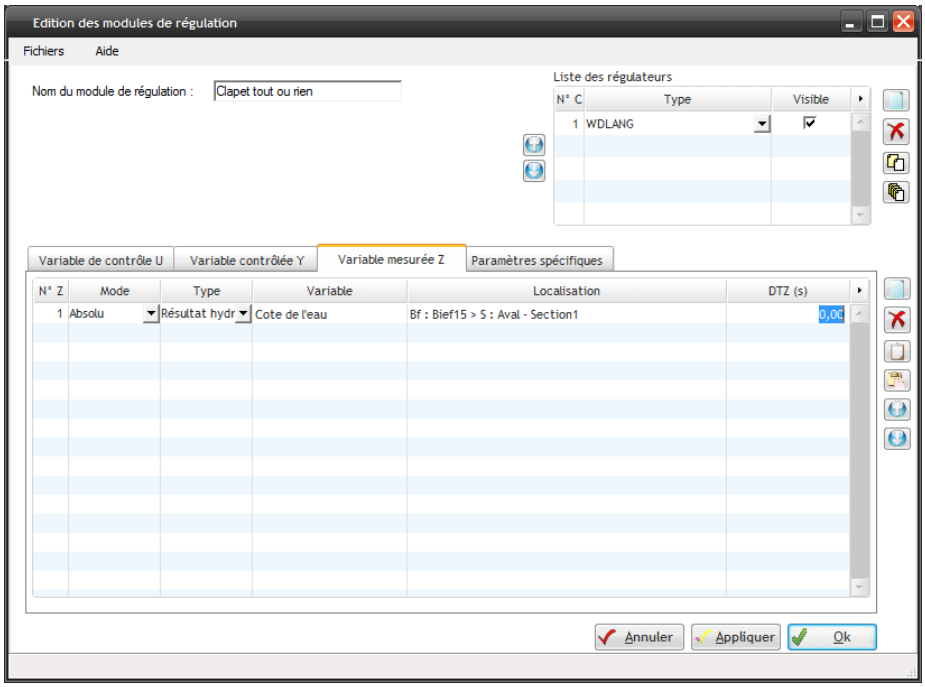


Figure 129.8: doc88

I set Y and Z to absolute in order to be able to compare the water elevations Z of the two sections.

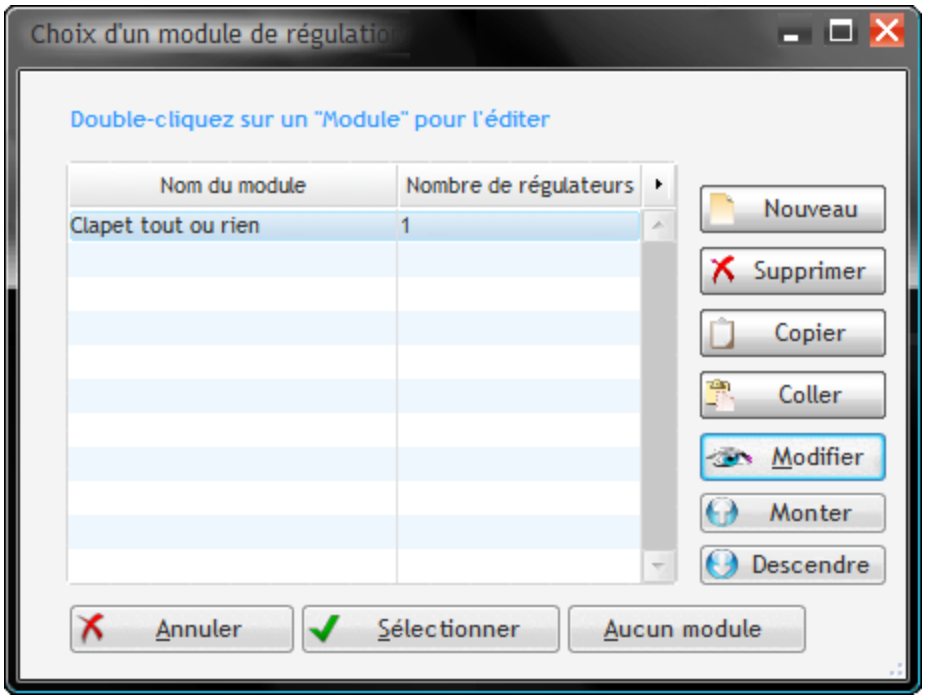


Figure 129.9: doc89

Click on « Select ».

On the network chart, a « WDLANG » box must be connected to the gate:

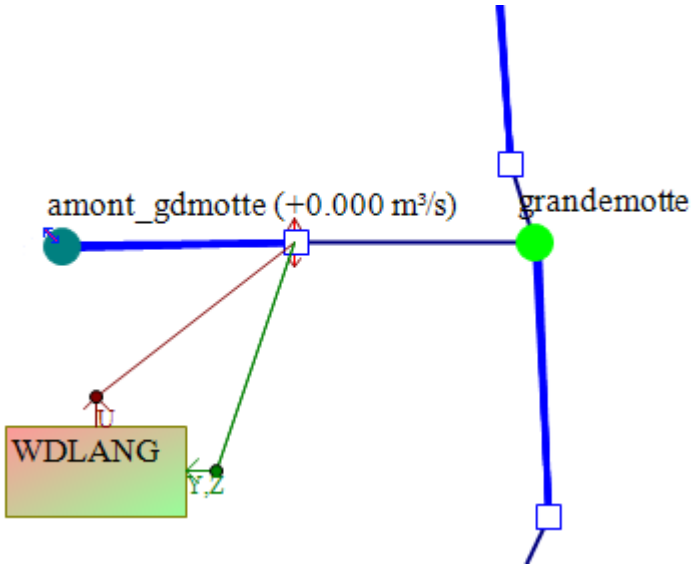


Figure 129.10: doc90

129.2 Writing the regulation script

Then, we can launch SIRENE and we will be able to write the regulation script during the simulation.

At the launch of SIRENE, a window opens with a first message:

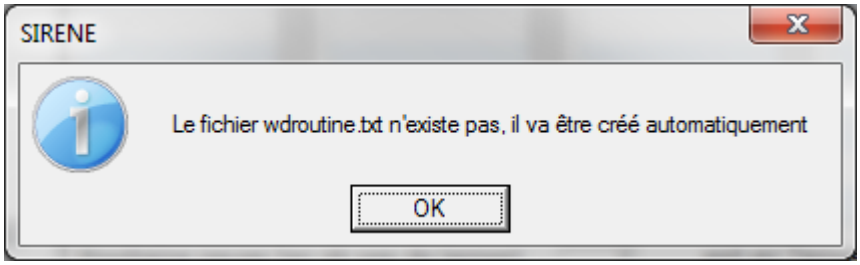


Figure 129.11: doc91

We can write our code in the WDLANG module window:

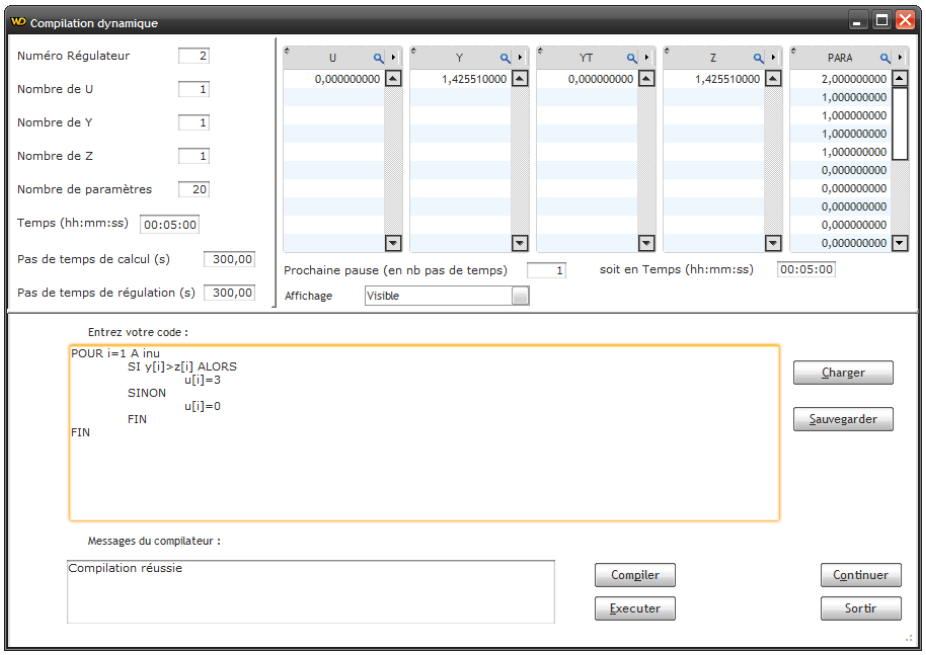


Figure 129.12: doc92

In the U, Y and Z tables, we see in real time respectively, the opening, the upstream water elevation and the downstream water elevation of the gate.

The portion of code in WDLangage above loops over all the U and, for each U, will open the gate to 3 meters if the upstream water elevation is greater than the downstream water elevation and close the gate otherwise. In this way, we can add other check valves by adding lines in the U, Y and Z of the regulation modules editing window.

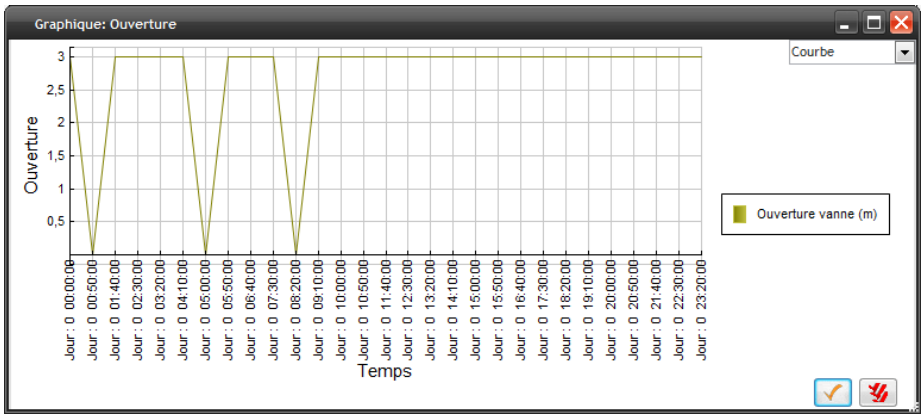
The script must be saved and compiled with the buttons ad hoc. Then, if we click on Run, the script will update the value of U.

If we click on Continue, it will run the simulation until the time defined by the "Next pause (in number of time steps)" field.

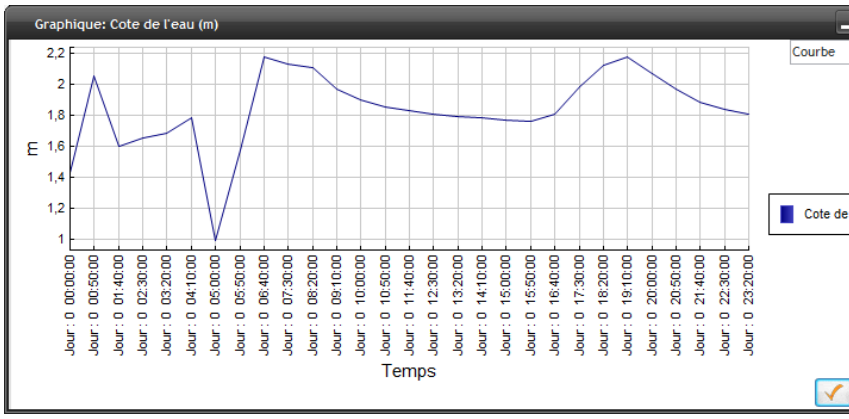
129.3 Simulation parameters and results

Here is an example where we can see the check valve working:

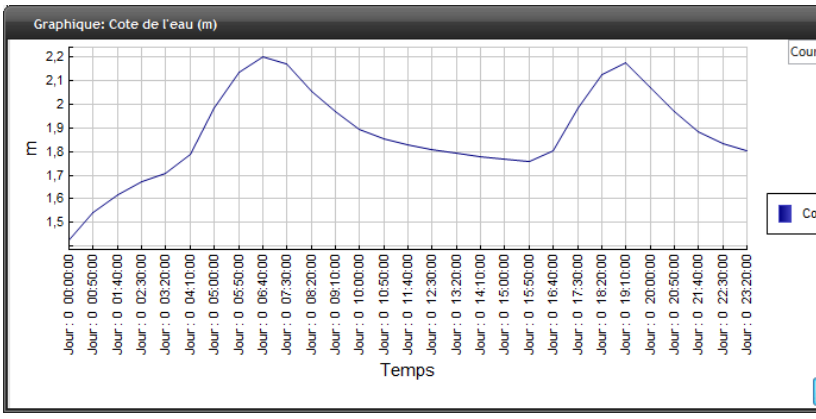
Opening of the gate:



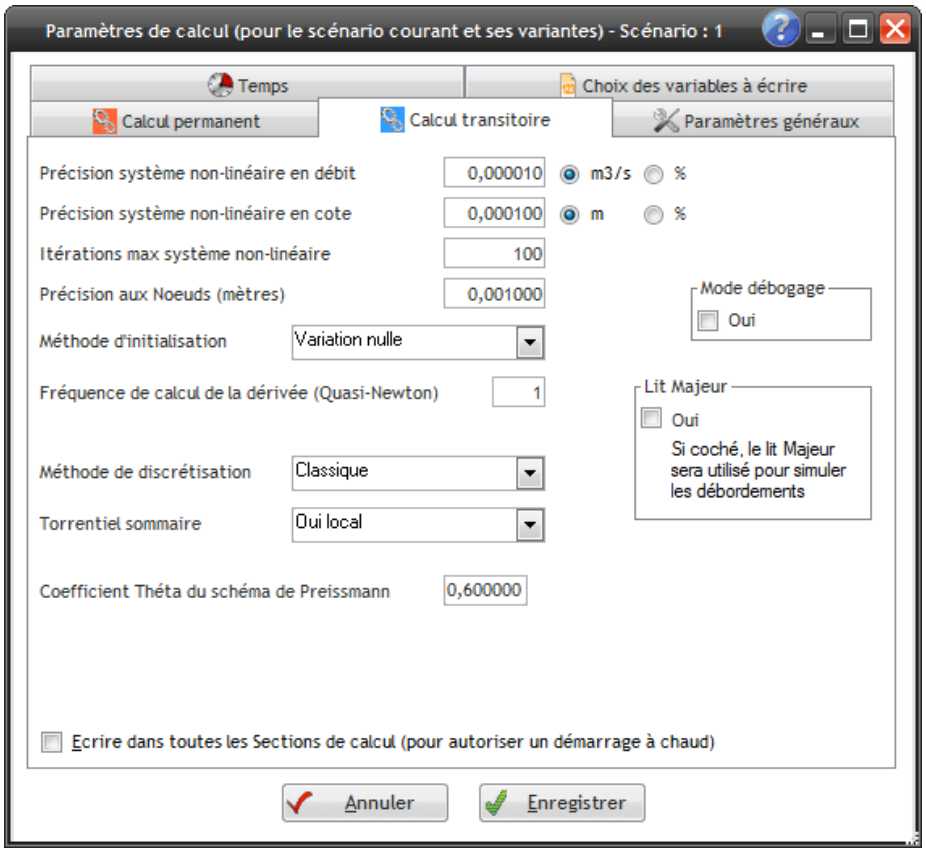
Upstream water elevation of the check valve:



Downstream water elevation of the check valve:



We note that the sudden change of opening of the gate creates numerical instabilities. The calculation must be done in nonlinear mode by putting 1 in “Frequency for computing the derivative (Quasi-Newton)” of the unsteady flow calculation pa-



rameters:

Another point to note is that the opening command of the check valve is carried out with one time step of delay on the measurements. That is to say that the check valve closes if at the previous time step the discharge started to flow in the reverse direction, and does not reopen until the upstream water elevation is greater than the downstream water elevation at the previous time step. It is therefore necessary, if the opening/closing frequency of the check valve is high, to possibly decrease the calculation time step.

Chapter 130

Introduction

The various functions of SIC can be used from the editing and visualization program "EdiSIC". This is sufficient for classical use of the software. Users wishing to perform more complex operations such as optimization will need to automate the current operations (execution of the various TALWEG, FLUVIA, SIRENE, results processing …) programs without going through the user interface.

This section describes the different software features available from a command line also called "Batch mode". This opening will allow scripts (.BAT, Scilab, Matlab, R, Python …) written by the user to automate these operations.

Chapter 131

Import of .TAL and .FLU files in batch mode

This feature allows you to launch EdiSIC, open a *.TAL* file, save it in Xml, launch the mesh with the Talweg program, and then if everything went right, import one or more *.FLU* files. These files are from those with the format of version SIC 4.**.

131.1 Syntax

```
EDISIC.EXE "working_directory" IMPORT "File.TAL" "File1.FLU" ["File2.FLU" ...]
```

131.2 Parameter Description

- "EDISIC.EXE": The name of the executable program with the possibility to give its complete name if the third party program running the batch mode is not located in this directory, or if the executable program is not defined in the path
- "Work_directory": Working directory (in which the output XML file and the .log file will be generated)
- IMPORT: EdiSIC launch mode - opening and saving one or more files
- "File.TAL": Name of a .TAL file (full path or relative to the working directory)
- "File1.FLU", "File2.FLU" ...: Name of a .FLU file (full path or relative to the working directory)

The option works for as many .FLU files as you want. It is even possible to put several batch of .TAL followed by several .FLU.

131.3 Command line limitation: using EDISIC.MAC

Since the command line size is limited, it may be useful to set the command line in the EDISIC.MAC file and to run EdiSIC in “macro” mode.

131.4 Command Line Example

```
EDISIC.EXE "C:/Files/SicV5/dat/ex1" IMPORT Demo.tal Demo.flu
```

or

```
C:/Sic538f5/exe/Edisic.exe "C:/Fichiers/Sic V5/dat/ex1" Import Demo.tal Demo.flu
```

The following actions will be performed: - Opening of Demo.tal - Registration in xml format - Mesh of the xml file by Talweg - Import file Demo.flu

The Demo.flu file will be used to create a scenario. In order to better adapt to executions in batch mode, several things will be defined and activated for this scenario: - A variant will be created, in case we want to use it for example for an unsteady simulations (in batch mode we can import an initial condition in a scenario or a variant and then run a steady or unsteady calculation) - The management of the .par file will be positioned in the mode where this .par file will not be rewritten by Edisic. It will thus be possible to create it by a third-party program and to take it into account during the calculations (otherwise by default Edisir rewrites it each time the .xml file is saved). - The mode of steady state calculation for several time instants and not only for the first moment is also activated, to allow these multiple calculations in batch mode. We can actually think that this feature will be useful in batch mode. It will be possible anyway in the calculation parameters of the program Fluvia to specify the end times (TFIN=) and the time step (DT=) if we do not want to perform every possible moment.

In the case of using an edisic.mac file, it will be necessary to put:

```
"C:/Files/SicV5/dat/ex1" IMPORT Demo.tal Demo.flu
```

this file being positioned in the sub-directory of the project, where the files to be imported will also be present. And Edisic must also have this sub-directory as the current directory (open a file in this directory before and then close Edisic).

and just run in command line:

```
EDISIC.EXE
```

or

```
C:/Sic538f5/exe/Edisic.exe
```

Chapter 132

Import sections in text format

This feature allows you to launch EdiSIC, create a new blank document or modify an existing project, import files containing section data into text format and save it as TAL (for SIC Version 4.xx) or XML (for SIC version 5 .xx). For more information on the text import formats of the sections, see article dedicated.

Modifying an existing project allows you to keep all the data of hydraulic scenarios, quality modules and regulation modules. The reaches are then imported in the order of their numbering.

132.1 Syntax

```
EDISIC.EXE "Work_directory" Import_Mode "File to create" "File1.txt" ["File2.txt" ...]
```

132.2 Parameter Description

- "Work_directory": Working directory (in which the output XML or TAL file and a .log file will be generated). This working directory can be different from the current subdirectory, cf below;
- Import_Mode: this parameter can take 4 different values (and possible alternatives with _new or _update, see below) depending on the chosen import mode (see below import modes);
- "File to create": Name of the XML or TAL file to be created at the end of the import without the extension. A file of the same name with the .log extension will be created and will contain error messages for the operation, if any;

- "File1.txt" [, "File2.txt" ...]: Name of one or more file(s) .txt to import (full path or single name). Each .txt file represents a reach. The file order defines a branch from upstream to downstream. If the names of these files are not given with a full path, they will be searched by default in the current subdirectory, which is the last one used by Edisic (and stored in the sic.ini file under exe). If they are not found, the current subdirectory is changed into the working subdirectory defined above, which allows to have a library of txt files in a current subdirectory, and to create various XML or TAL files in subdirectories and with selected names as needed.

132.3 Import modes

There are 4 modes (and possible alternatives with _new or _update, see below) of choice to define the type of file created (Xml or Tal) and the format of the text file (normal or transposed): | *Import mode* | *Text file format* | *Generated File Format* | | ImportXml | Normal | XML (SIC version 5.xx) | | ImportTal | Normal | TAL (SIC version 4.xx) | | ImportXmlTrans | Transposition | XML (SIC version 5.xx) | | ImportTalTrans | Transposition | TAL (SIC version 4.xx) |

If the project file already exists, the interface asks if the user wants to overwrite the project, modify the existing project or cancel the operation. It is possible to override this issue by adding either _NEW or _UPDATE to the import mode (Example: use IMPORTXML_NEW to overwrite an existing XML project file). Lowercase or uppercase can be used for these options.

132.4 Command line limitation: using EDISIC.MAC

Since the command line size is limited, it may be useful to set the command line in the EDISIC.MAC file and to run EdiSIC in "macro" mode.

132.5 Command Line Example

```
EDISIC.EXE "C:/Files/SicV5/dat/test" ImportXml "Demo" "bief1.txt" "bief2.txt"
```

The following actions will be performed: - Creation of the file C:/Files/SicV5/dat/test/Demo.xml; - Creation of the reach1 upstream; - Import the cross-section of the reach 1 and their cross section from the file C:/Files/SicV5/dat/test/bief1.txt; (Or the bief1.txt file located in the current subdirectory if it exists) - Creation of the reach 2 downstream of the reach 1; - Import of cross sections from C:/Files/SicV5/dat/test/bief2.txt file; (Or the bief2.txt file located in the current subdirectory if it exists) - Save the file C:/Files/SicV5/dat/test/Demo.xml. If this file already exists, a question is asked to allow overwriting.

Chapter 133

EdiSIC macro mode : an alternative to the command line

The EdiSIC features such as importing sections in text format may require the use of a command line longer than the 256 characters allowed by the Windows® operating system.

So there's a mode of writing longer command lines in a file EDISIC.MAC. The location of EDISIC.MAC is defined in this order:

-# in the folder defined as first argument of the command line if one argument is present when EdiSic is launched; -# in the current folder when Edisic is launched; -# in the folder of the last project opened with EdiSic.

Example for loading the EDISIC.MAC file located in a folder defined in the command line:

```
C:/MyDocs/MyInputs>C:/SIC/exe/edisic.exe C:/MyDocs/MyFolderWithEdisicMac
```

133.1 Syntax of EDISIC.MAC file

You just have to write the arguments used in the command line usually used for asking EdiSIC to do a specific task. See examples in the section "Control SIC from other programs (Batch mode)".

For example, for a file edisic.mac:

```
D:/SIC/Po ImportXml_new Po exportdata1.txt  exportdata2.txt  exportdata3.txt  exportdata4.txt  ex
```

Then just run Edisic.exe on the command line, or directly in the usual way (double click on the Windows shortcut), and the instructions written in the Edisic.mac file will be executed. Here in this example importing 10 section data files into a Po.xml file that will be saved in the D:/SIC/Po data subdirectory. Even if the file already exists a new file will be created (because import option ImportXml_new). To stop the use of this automatic mode in macro mode just delete (or rename) the edisic.mac file.

Chapter 134

Importing an initial condition for a transient flow simulation

This command simply allows importing the result of a steady or transient simulation of a scenario or variant into the initial conditions of a scenario or variant, so that a transient flow simulation (SIRENE) can be launched.

134.1 Command line syntax:

```
EDISIC.EXE "Working_directory" CI "File.XML" [ResSce] [ResVar]
[ResTps] [CiSce] [CiVar]
```

134.2 Description of the command line parameters

- `"Working_directory"` : directory where the model XML file is located
- `CI` : command to EdiSic to launch the import of initial conditions
- `"File.XML"` : File where the model to be processed is located. `-[ResSce]` : number of the scenario containing the simulation result to import
- `[ResVar]` : number of the variant containing the simulation result to import
- `[ResTps]` : time (in seconds) of the simulation result to import
- `[CiSce]` : Scenario to which the initial condition is imported
- `[CiVar]` : Variant to which the initial condition is imported

134.3 Usage example on the file C:/titi/toto.xml

To import the result at time 0 of scenario 1 into variant 1 of scenario 1:
`EDISIC.EXE "C:/titi" CI "toto.xml" 1 0 0 1 1`

To import the result at 6 hours of variant 1 of scenario 1 into variant 2 of the same scenario: `EDISIC.EXE "C:/titi" CI "toto.xml" 1 1 21600 1 2`

Chapter 135

Running TALWEG, FLUVIA and SIRENE from the command line

The SIC executables can be launched from the command line with various options that allow automating the launch of a simulation.

The launch syntax for TALWEG, FLUVIA and SIRENE is as follows:

```
PROGRAM.EXE [XML file name] [Interface mode] [Fluvia and Sirene options]
```

All the command line arguments are optional; if the XML file name is not specified, it will be requested from the user through the standard Windows® dialog box.

135.1 Meaning of the arguments:

135.2 [XML file name]

Name of the XML file to open. If the path is not specified, the current path is used by default.

135.3 [Interface mode]

This is the type of display and interaction with the user during the execution of the program. The different modes are the following:

- 0: No interface on screen (reads all options from the command line)

- 1: No interface (executes the instructions of the macro file PROGRAM.MAC)
- 2: Normal interfaces
- 3: Normal interfaces + writing of the inputs to PROGRAM.MAC (macro recording)
- 4: Normal interfaces except calculation monitoring (obsolete in version 5)
- 5: Normal interfaces except file opening (gets file name from PROJECT.INI)
- 6: No interface at all (i.e. same as 1 without the small window)

In the absence of a corresponding second argument in the command line, mode 2 is used.

135.4 [Fluvia and Sirene options]

The various available options are written in the form `OPTION=VALUE` separated by spaces. The different possible options are: - `INTERF=n` : interface mode n (see above). Default `INTERF=2`. - `SCE=n` : launch the hydraulic simulation on hydraulic scenario n. This parameter is mandatory. - `VAR=n` : launch the simulation on variant n. By default, the simulation is launched on the scenario data (`VAR=0`). - `REG=n` : launch the simulation with regulation module n. By default, the regulation module selected in EdiSic is used. - `TFIN=x` : set the end time of the simulation to x seconds. - `DT=x` : set the time step of the simulation to x seconds. - `MODRES=n` : set the format for writing the results: -** `MODRES=0` (default) : keeps the writing format defined in the scenario parameters -** `MODRES=1` : forces the writing of the results in binary format -** `MODRES=2` : forces the writing of the results in the project XML file

135.5 Usage example

To launch SIRENE with the following options: - File `C:/DOC/DEMO.XML` - Scenario 2 - Variant 1 - Regulation module 1 - No interface on screen at all

The command line will be the following:

```
SIRENE.EXE C:/DOC/DEMO.XML INTERF=0 SCE=2 VAR=1 REG=1
```

or also

```
SIRENE.EXE C:/DOC/DEMO.XML 0 SCE=2 VAR=1 REG=1
```

Chapter 136

SicExport: text-format results export utility

SicExport is an executable to be used from the command line, provided in the EXE directory of the SIC installation. It can be launched directly from the DOS or Windows command line, but also from scripts written in other languages (MatLab, Scilab, R, etc), for example to chain calculations with their pre- and post-processing.

It allows quickly creating a tab-separated text file with the results of a water surface profile at a given time, or the results of a precise location in the network for all times.

It can handle results of steady or transient flow calculations.

This utility allows exporting the results whether they are written in XML format directly in the project file, or also in binary format in an external RES file. This format is chosen before the launch of the steady (Fluvia program) or transient (Sirene program) calculation, in the general parameters, and this information is stored in the XML file. SicExport will read this mode and look for the information in the appropriate file (XML or RES). If the RES file corresponding to the scenario and variant concerned does not exist, an error message will be displayed.

136.1 Usage syntax

The usage syntax of SicExport is the following:

```
SicExport.exe FichierXml /sce=n /out=FichierTexte [Options]
```

There must be no space between the slash «/» and the option, and between the option, the equal sign and the value of the option, otherwise the options will be

ignored.

136.2 Mandatory parameters

- `FichierXml` : Name of the project file to use. Names with spaces must be quoted.
- `/sce=n` : n° of the scenario where the result to extract is located.
- `/out=FichierTexte` : Name of the text file where the extracted results will be stored. If the path is not specified, that of `FichierXml` is used. Names with spaces must be quoted.

136.3 Optional parameters

- `/var=n` : n° of the variant where the result to extract is located. `/var=0` corresponds to the scenario result without variant.
- `/t=n` : Water surface profile at time n. If this parameter is present the generated file contains the results of all sections of the network at time n in reach number format.
- `/bf=n` : reach number
- `/sn=n` : n° of minor bed section
- `/x=n` : Chainage. Can only be used if the option `/bf=n` is present and the option `/sn=n` absent.
- `/nd=n` : n° of node
- `/pr=n` : n° of offtake (valid only if the option `/nd=n` is present)
- `/st=n` : n° of structure. Used only to extract the data of a device at an offtake (valid only if the options `/nd=n` and `/pr=n` are present)
- `/ouv=n` : n° of device
- `/quiet=0;1;2` : Options for the file headers (see below).

If none of these options is present, the water surface profile at time 0 is exported (option `/t=0`).

136.4 *Options on the file headers: the /quiet parameter

- Option absent or `/quiet=0` : Complete headers with comments on 3 lines and the variable names on 1 line after a carriage return
- `/quiet=1` : Table header (variable names) only
- `/quiet=2` : No header (no comments, no variable names)

136.5 Example

Example of output file with the command `SicExport "C:/SIC/DAT/asce1/asce1.xml" /sce=1 /out=test.txt /t=0`

```
% Projet : C:/DocDD/SIC/Validation/asce1/asce1.xml
% Scenario 1
% Simulation FLUVIA Version 5.12 du 17/03/2011 15:20:10

Bief    Section Abscisse    Q   Z   L   V   ST
1   1   0   8.000000E-01    100.703850  3.111552    5.528833E-
01    1.446960
1   2   95  8.000000E-01    100.700073  3.670238    3.849038E-
01    2.078442
2   1   95  7.000000E-01    100.700073  3.670238    3.367907E-
01    2.078442
2   2   100 7.000000E-01    100.699997  3.699981    3.309729E-
01    2.114977
2   3   100 7.000000E-01    99.163811   2.091429    1.244785    5.623461E-
01
2   4   1295    7.000000E-01    97.300064   3.670175    3.368033E-
01    2.078365
...
```

Chapter 137

Appendix

Chapter 138

Supercritical discharge modelling

However, the problem won't be solved in unsteady flow, since some sections will be in critical flow, which generate numeric instabilities and stop the calculations because of passages in supercritical flow. A better solution for guaranteeing a calculation in steady flow and also in unsteady flow is to define a weir at every important fall, passing by critical depth Yc (curve F2). In this case, the upstream depth of the fall will be calculated by this weir equation, and won't be Yc. Hydraulically, we do not calculate the depth immediately upstream from the fall (Yc), but slightly upstream from this fall, in a zone where the water trickle contraction is negligible and where the hypothesis of hydrostatic distribution of pressures holds true (Cf. usual demonstrations of weir equations).

The problem that remains is to define the parameters of this weir. Normally, it would be necessary to give the real geometry of the section. However this is not possible (except in the simple case of a rectangular section), since only the horizontal weirs can be modelled in SIC. However, if the section is almost a rectangular section, a horizontal weir were frame level is equal to the bottom level, is a good approximation. Otherwise, the use of several weirs in parallel as indicated in the previous chapter that deals with the triangular weirs permits you to model weirs of a more complex shape. In any case, it will also be necessary to provide a discharge coefficient for the weir, that can arbitrarily be equal to 0.4 or calibrated by measures on the ground.

The same problem can also occur at abrupt geometry changes (widening), with local passage to Yc. The solution to use is the same. when these corrections have been done, the calculated flow profile is more realistic.

In spite of these corrections that permit you to better model existing falls at some point in the system there still may remain many points where the discharge

is supercritical (Cf. the .BAK file).

Often these passages in supercritical take place far from the offtakes. This is relatively normal because in general a gravity offtake is placed in a fluvial discharge zone, in order to control its feed level. We can thus imagine simplifying the supercritical zone model by using abrupt falls equipped with weirs. when these modifications have been done, we get an entirely fluvial flow profile (Cf. .BAK file). The passage problem at the critical depth Yc or the passage problem in supercritical discharge can be solved by doing a modification in the geometry of the system. It will be necessary during the calibration of the model, not to pay attention to zones where these modifications have taken place, but to be vigilant elsewhere, and to check that these geometry modifications don't change the results in fluvial discharge zones too much.

Chapter 139

Dry bed modelling

Another problem observed on some networks is the dry reach problem. In steady flow calculation there is no particular problem. On the other hand, it is impossible to do a simulation in unsteady flow on such a network. Two solutions are conceivable. The first one consists of adding a fictional discharge to a dry zone head and to subtract this discharge at the end of that zone, in order not to disturb the flow somewhere else. A second solution is to extract a subsystem from the complete system that only contains flowing reaches. This second solution is not satisfactory anymore if the dry level(s) begin(s) to flow during the simulation. This "dry bed" problem is hard to solve correctly. Cemagref is working on this problem and a next SIC version should integrate this possibility.

Chapter 140

Triangular weir modelling

140.1 Weirs Laws

```
function [Q] = vnotch(T,h,alfa,Cd,e)
% T   : 1 for true V-Notch
% T   : 2 pour V-Notch equivalent
% h   : upstream level
% Q   : discharge
% Initialisation ----------------------------------------
R2G = sqrt(2*9.81);
if (T==1)
% triangular weir
  Q=1.32*tan(alfa/2)*h^2.47;
else
% Equivalent horizontal weir
if (h>0)
  L=e*tan(alfa/2);
  Q=Cd*R2G*L*h^1.5;
end
if (h>e)
  Q=Q+Cd*R2G*2*L*(h-e)^1.5;
end
if (h>2*e)
  Q=Q+Cd*R2G*2*L*(h-2*e)^1.5;
end
if (h>3*e)
  Q=Q+Cd*R2G*2*L*(h-3*e)^1.5;
end
if (h>4*e)
```

```
  Q=Q+Cd*R2G*2*L*(h-4*e)^1.5;
end
end
```

140.2 Comparison of the results obtained by using the two laws on an example

```
% Parameters --------------------------------------------
pas=0.005;
val=100;
alfa=pi/2;
e=0.1;
Cd=0.38;
% Calculation -------------------------------------------------
for i=1:val
   [Q(i,1)]=vnotch(1,i*pas,alfa,Cd,e);
   [Q(i,2)]=vnotch(2,i*pas,alfa,Cd,e);
end
% discharge deviation
plot(Q);
% relative deviation in %
plot((Q(:,2)-Q(:,1))/Q(:,1));
```

Chapter 141

Equation and devices calibration (gates and weirs)

We provide below the scripts of 2 Matlab .m functions (Qouvrage.m and calage_f.m) and a script file .m (calage_r.m) usefull for calibrating a discharge coefficient from mesurement datas using the equations programmed in SIC.

141.1 Devices modelisation (gates and weirs) (Script to copy in a file Qouvrage.m)

```
function [Q,C,w] = Qouvrage(T,h1,h2,w,hv,L,Cd)
% T=1 : Seuil Orifice avec surverse éventuelle (SIC)
%   2 : Seuil Vanne avec surverse éventuelle (SIC)
%   3 : Classique Seuil
%   4 : Classique Orifice
%   5 : Classique Noyee
%   6 : Cunge avec differentes conditions d'ecoulement
% h1  : cote amont (par rapport au radier)
% h2  : cote aval (par rapport au radier)
% w   : ouverture (par rapport au radier)
% hv  : hauteur de la vanne (pour le calcul de la surverse éventuelle)
% L   : largeur
% Cd  : coefficient de debit
% Q   : debit
% C   : code couleur correspondant aux conditions d'ecoulement
%       C=0; debit nul
%       C=1; surface libre denoye
%       C=2; surface libre noye
```

```
%       C=3; charge denoye
%       C=4; charge noye partiel
%       C=5; charge noye total
%       C=C+10; si surverse
% Initialisation ----------------------------------------------
R2G = sqrt(2*9.81);
R32 = 3*sqrt(3)/2;
% Tests -------------------------------------------------------
Q=0;
C=0;
if (w==0 & hv==0)
  Q=0;
  return
end
if (h2>h1)
  Q=0;
  return
end
if (h1==0)
  Q=0;
  return
end
% Seuil - Orifice ===========================================
if (T==1 & w>0)
% Conditions d'ecoulement -------------------------------------
  if (h1<=w)
     surfacelibre='oui';
  else
     surfacelibre='non';
  end
  alfa=2/3;
  if (h2<=alfa*h1)
     denoye='oui';
  else
     denoye='non';
     if (h2<=2/3*h1+w/3)
        partiel='oui';
     else
        partiel='non';
     end
  end
% Seuil - Denoye ----------------------------------------------
  if (surfacelibre=='oui') & (denoye=='oui')
     Q=Cd*L*R2G*(h1^1.5);
     C=1;
```

```
  end
% Seuil - Noye ------------------------------------------------
  if (surfacelibre=='oui') & (denoye=='non')
     Cs=R32*Cd;
     Q=Cs*L*R2G*((h1-h2)^0.5)*h2;
    C=2;
  end
% Orifice - Denoye --------------------------------------------
  if (surfacelibre=='non') & (denoye=='oui')
     Q=Cd*L*R2G*(h1^1.5-(h1-w)^1.5);
     C=3;
  end
% Orifice - Noye ----------------------------------------------
  if (surfacelibre=='non') & (denoye=='non')
%  Ennoyement partiel ----------------------------------
     if (partiel=='oui')
        Q=Cd*L*R2G*(R32*h2*(h1-h2)^0.5-(h1-w)^1.5);
        C=4;
%  Ennoyement total ------------------------------------
     else
        Cs=R32*Cd;
        Q=Cs*L*R2G*((h1-h2)^0.5)*w;
       C=5;
     end
  end
end
% Seuil - Vanne ===============================================
if (T==2 & w>0)
% Calcul de parametres ----------------------------------------
  mu0=2/3*Cd;
% Conditions d'ecoulement -------------------------------------
  if (h1<=w)
     surfacelibre='oui';
     muf=mu0-0.08;
     alfa=0.75;
  else
     surfacelibre='non';
     mu =mu0-0.08/(h1/w);
     mu1=mu0-0.08/(h1/w-1);
     alfa=1-0.14*h2/w;
     if (alfa<0.4)
        alfa=0.4;
     end
     if (alfa>0.75)
        alfa=0.75;
```

```
      end
   end
   if (h2<=alfa*h1)
      denoye='oui';
   else
      denoye='non';
      x=sqrt(1-h2/h1);
      beta=-2*alfa+2.6;
      if (x>0.2)
         KF=1-(1-x/sqrt(1-alfa))^beta;
      else
         KF=5*x*(1-(1-0.2/sqrt(1-alfa))^beta);
      end
      alfa1=1-0.14*(h2-w)/w;
      if (alfa1<0.4)
         alfa1=0.4;
      end
      if (alfa1>0.75)
         alfa1=0.75;
      end
      if (h2<=alfa1*h1+(1-alfa1)*w)
         partiel='oui';
      else
         partiel='non';
      end
   end
% Seuil - Denoye ------------------------------------------------
   if (surfacelibre=='oui') & (denoye=='oui')
      Q=muf*L*R2G*(h1^1.5);
      C=1;
   end
% Seuil - Noye --------------------------------------------------
   if (surfacelibre=='oui') & (denoye=='non')
      Q=KF*muf*L*R2G*(h1^1.5);
      C=2;
   end
% Vanne - Denoye ------------------------------------------------
   if (surfacelibre=='non') & (denoye=='oui')
      Q=L*R2G*(mu*h1^1.5-mu1*(h1-w)^1.5);
      C=3;
   end
% Vanne - Noye --------------------------------------------------
   if (surfacelibre=='non') & (denoye=='non')
      x1=sqrt(1-(h2-w)/(h1-w));
      beta1=-2*alfa1+2.6;
```

```
     if (x1>0.2)
        KF1=1-(1-x1/sqrt(1-alfa1))^beta1;
     else
        KF1=5*x1*(1-(1-0.2/sqrt(1-alfa1))^beta1);
     end
%  Ennoyement partiel --------------------------------
     if (partiel=='oui')
        Q=L*R2G*(KF*mu*(h1^1.5)-mu1*(h1-w)^1.5);
        C=4;
%  Ennoyement total ----------------------------------
     else
        Q=L*R2G*(KF*mu*(h1^1.5)-KF1*mu1*(h1-w)^1.5);
        C=5;
     end
  end
end
% Surverse dans cas 1 et 2 ====================================
if hv==0
  hv=inf;
end;
if (T==1) | (T==2)
  if (h1>w+hv)
     surverse='oui';
     alfa=2/3;
     if (h2-w-hv<=alfa*(h1-w-hv))
        denoyesurverse='oui';
     else
        denoyesurverse='non';
     end
  else
     surverse='non';
  end
  % Surverse - Denoye ------------------------------------------
  if (surverse=='oui')
     if (denoyesurverse=='oui')
        Q=Q+0.4*L*R2G*(h1-w-hv)^1.5;
        C=C+10;
     else
        Q=Q+1.04*L*R2G*((h1-h2)^0.5)*(h2-w-hv);
        C=C+10;
     end
  end
end
% Classique - Seuil ============================================
if (T==3)
```

```
  Q=Cd*L*R2G*h1^1.5;
  C=1;
end
% Classique - Orifice ========================================
if (T==4)
  if (h1>w)
     Q=Cd*w*L*R2G*(h1-w/2)^0.5;
     C=3;
  else
     Q=0;
     C=0;
  end
end
% Classique - Noyee ==========================================
if (T==5)
  if (h1>h2)
     Q=Cd*w*L*R2G*(h1-h2)^0.5;
     C=5;
 else
     Q=0;
     C=0;
  end
end
% Classique Cunge ============================================
if (T==6)
% Conditions d'ecoulement ------------------------------------
  if (h2<=2/3*h1)
     denoye='oui';
     if (w<=2/3*h1)
        surfacelibre='non';
        Q=Cd*L*R2G*w*((h1-w)^0.5);
        C=3;
     else
        surfacelibre='oui';
        Q=Cd*L*R2G/R32*h1^1.5;
        C=1;
     end
  else
     denoye='non';
     if (w<=h2)
        surfacelibre='non';
        Q=Cd*L*R2G*w*((h1-h2)^0.5);
        C=5;
     else
        surfacelibre='oui';
```

```
        Q=Cd*L*R2G*h2*((h1-h2)^0.5);
        C=2;
    end
  end
end
```

141.2 Intermediate function used for calibration

```
function Q = f(Cd)
% fonction utilisee pour le calage dans calage\_r.m
% T=1 : Seuil Orifice (SIC)
%   2 : Seuil Vanne (SIC)
%   3 : Classique Seuil
%   4 : Classique Orifice
%   5 : Classique Noyee
% h1  : cote amont (par rapport au radier)
% h2  : cote aval (par rapport au radier)
% w   : ouverture
% hv  : hauteur de la vanne (pour le calcul de la surverse éventuelle)
% L   : largeur
% Cd  : coefficient de debit
% Q   : debit calcule par Qouvrage
% Qm  : debit mesure pour le calage
% C   : code de couleur correspondant aux conditions d'ecoulement
% Initialisation
T = 2;
hr = 41.95
h1 = 43.28 - hr;
h2 = 42.77 - hr;
hv = 0;
w = 0.66;
L = 1.39;
Qm = 1.909;
% Calcul
[Q,C,w] = Qouvrage(T,h1,h2,w,hv,L,Cd);
Q = Q - Qm;
```

141.3 Calibration of discharge coefficient (script to copy in a file calage_r.m)

```
% Fichier .M utilisé pour caler Cd
% fzero est une fonction MatLab qui trouve le zéro
% d'une fonction proche d'une valeur donnée
Cd = fzero('calage_f',0.5)
```

Once these codes have been copied in the .m file as above, you have to run the .m script (calage_r.m) for calculating the discharge coefficient Cd from measures h1, h2, Qm and the description of the device (h1, hv, w, L) using the equations of the evices programmed in SIC (several choices are proposed).

Chapter 142

Modeling of an in-line pump

Since version 4.12, an in-line pump can be modeled using a control module. Indeed, a control module is created in this case (with any method adapted to the method that it is desired to use for the variations of the pump discharge, for example the STOP, BOMAT, WDLANG, SCILAB or MATLAB method). At the site of the in-line pump a singular section is placed with a gate-type structure. For this same gate, the opening is declared as control U of the control module, in flow mode (slave controler). See doc on Regulation modules and General settings for U for further details. This solution is probably the best but requires the use of a control module.

Chapter 143

Abacus of the Gec-Alsthom gates

We recalculated the abacus from the equation we use. We observe they are similar to those given by the Gec-Alsthom company:

- For AMIL gates with Zmax=0 (Maximum water level set on axis elevation) :

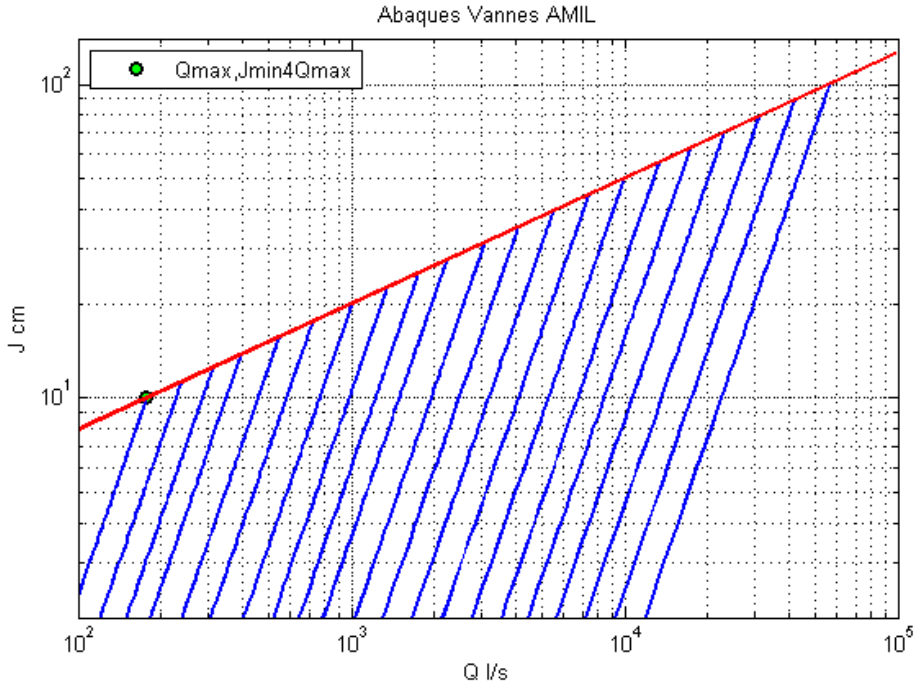
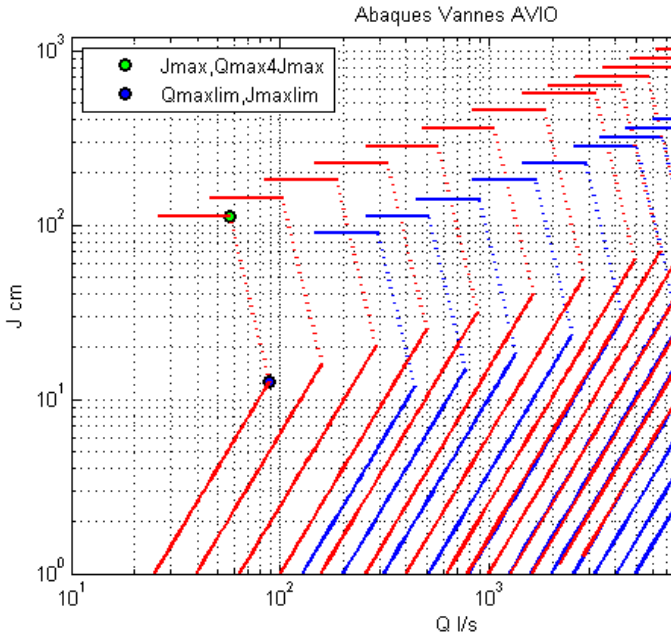
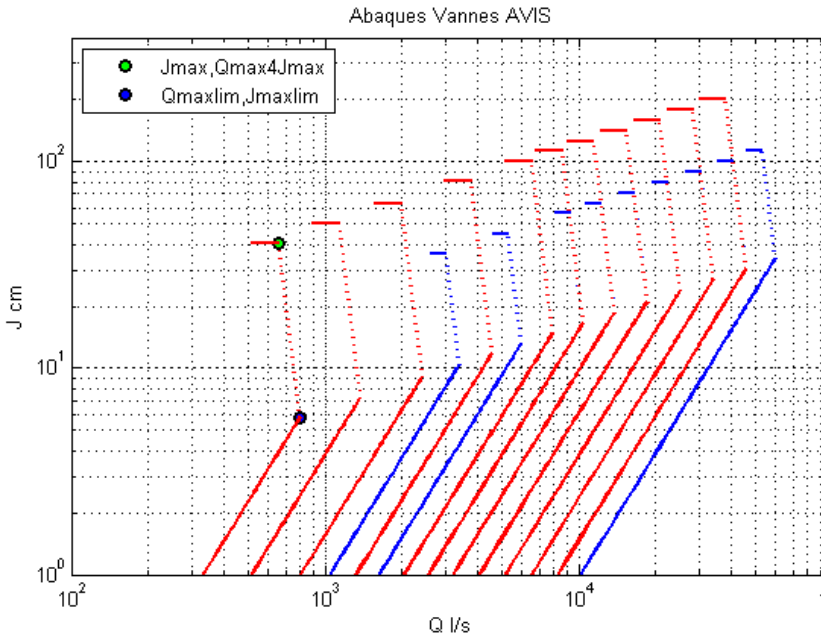


Figure 143.1: doc69



- For AVIO gates (HC in red, BC in blue) :



- For AVIS gates (HC in red, BC in blue) :

From these equations we also obtain the following points:

- For AMIL gates with Zmax=0 :

Model	*Jmin for Qmax*	*Qmax*
AMIL 80/63	0.1004	0.1782
AMIL 90/63	0.1130	0.2392
AMIL 100/63	0.1255	0.3113
AMIL 110/63	0.1381	0.3951
AMIL 125/90	0.1569	0.5438
AMIL 140/90	0.1757	0.7219
AMIL 160/90	0.2008	1.0080
AMIL 180/125	0.2259	1.3532
AMIL 200/125	0.2510	1.7610
AMIL 220/125	0.2761	2.2348
AMIL 250/160	0.3137	3.0763
AMIL 280/160	0.3514	4.0839
AMIL 315/200	0.3953	5.4822
AMIL 355/200	0.4455	7.3917
AMIL 400/250	0.5020	9.9615
AMIL 450/250	0.5647	13.3723
AMIL 500/315	0.6275	17.4020
AMIL 560/315	0.7028	23.1018
AMIL 630/400	0.7906	31.0118
AMIL 710/400	0.8911	41.8139
AMIL 800/450	1.0040	56.3508

- For AMIL gates with Zmax = d = 0.02D :

Model | *Jmin for Qmax* | *Qmax* |
AMIL 80/63 | 0.1044 | 0.1889 |
AMIL 90/63 | 0.1175 | 0.2536 |
AMIL 100/63 | 0.1305 | 0.3300 |
AMIL 110/63 | 0.1436 | 0.4188 |
AMIL 125/90 | 0.1631 | 0.5765 |
AMIL 140/90 | 0.1827 | 0.7653 |
AMIL 160/90 | 0.2088 | 1.0687 |
AMIL 180/125 | 0.2349 | 1.4346 |
AMIL 200/125 | 0.2610 | 1.8669 |
AMIL 220/125 | 0.2871 | 2.3692 |
AMIL 250/160 | 0.3263 | 3.2613 |
AMIL 280/160 | 0.3654 | 4.3295 |
AMIL 315/200 | 0.4111 | 5.8119 |
AMIL 355/200 | 0.4633 | 7.8363 |
AMIL 400/250 | 0.5220 | 10.5606 |
AMIL 450/250 | 0.5873 | 14.1766 |
AMIL 500/315 | 0.6525 | 18.4487 |
AMIL 560/315 | 0.7308 | 24.4912 |
AMIL 630/400 | 0.8222 | 32.8769 |
AMIL 710/400 | 0.9265 | 44.3288 |
AMIL 800/450 | 1.0440 | 59.7399 |

- For AVIO and AVIS gates :

Model | *Jmax* | *Qmax for Jmax* | *Jmax lim* | *Qmax lim* |
AVIO HC 28/6 | 1.1250 | 0.0574 | 0.1264 | 0.0889 |
AVIO HC 36/10 | 1.4175 | 0.1023 | 0.1593 | 0.1584 |
AVIO HC 45/16 | 1.8000 | 0.1859 | 0.2023 | 0.2879 |
AVIO BC 45/32 | 0.9000 | 0.2935 | 0.1184 | 0.4404 |
AVIO HC 56/25 | 2.2500 | 0.3248 | 0.2529 | 0.5029 |
AVIO BC 56/50 | 1.1250 | 0.5128 | 0.1480 | 0.7693 |
AVIO HC 71/40 | 2.8125 | 0.5674 | 0.3161 | 0.8785 |
AVIO BC 71/80 | 1.4063 | 0.8958 | 0.1850 | 1.3440 |
AVIO HC 90/63 | 3.6000 | 1.0518 | 0.4046 | 1.6284 |
AVIO BC 90/125 | 1.8000 | 1.6605 | 0.2368 | 2.4913 |
AVIO HC 110/100 | 4.5000 | 1.8374 | 0.5058 | 2.8448 |
AVIO BC 110/200 | 2.2500 | 2.9007 | 0.2959 | 4.3521 |
AVIO HC 140/160 | 5.6250 | 3.2098 | 0.6322 | 4.9696 |
AVIO BC 140/315 | 2.8125 | 5.0673 | 0.3699 | 7.6028 |
AVIO HC 160/200 | 6.3000 | 4.2611 | 0.7081 | 6.5973 |
AVIO BC 160/400 | 3.1500 | 6.7270 | 0.4143 | 10.0929 |
AVIO HC 180/250 | 7.0875 | 5.7201 | 0.7966 | 8.8563 |
AVIO BC 180/500 | 3.5437 | 9.0304 | 0.4661 | 13.5487 |
AVIO HC 200/315 | 7.9875 | 7.7125 | 0.8978 | 11.9411 |

AVIO BC 200/630 | 3.9937 | 12.1759 | 0.5253 | 18.2681 |
AVIO HC 220/400 | 9.0000 | 10.3939 | 1.0116 | 16.0925 |
AVIO BC 220/800 | 4.5000 | 16.4089 | 0.5919 | 24.6191 |
AVIO HC 250/500 | 10.1250 | 13.9527 | 1.1380 | 21.6026 |
AVIO BC 250/1000 | 5.0625 | 22.0272 | 0.6659 | 33.0486 |
AVIO HC 280/630 | 11.2500 | 18.1573 | 1.2645 | 28.1125 |
AVIO BC 280/1250 | 5.6250 | 28.6651 | 0.7399 | 43.0078 |
AVIS HC 56/106 | 0.4050 | 0.6561 | 0.0576 | 0.7974 |
AVIS HC 71/132 | 0.5040 | 1.1385 | 0.0723 | 1.3826 |
AVIS HC 90/170 | 0.6300 | 1.9978 | 0.0910 | 2.4242 |
AVIS BC 90/190 | 0.3600 | 2.9977 | 0.1045 | 3.3925 |
AVIS HC 110/212 | 0.8100 | 3.7635 | 0.1180 | 4.5631 |
AVIS BC 110/236 | 0.4500 | 5.2602 | 0.1316 | 5.9486 |
AVIS HC 140/265 | 1.0080 | 6.5303 | 0.1479 | 7.9118 |
AVIS BC 140/300 | 0.5625 | 9.2303 | 0.1657 | 10.4304 |
AVIS HC 160/300 | 1.1250 | 8.6122 | 0.1657 | 10.4304 |
AVIS BC 160/335 | 0.6300 | 12.2814 | 0.1863 | 13.8729 |
AVIS HC 180/335 | 1.2600 | 11.4589 | 0.1863 | 13.8729 |
AVIS BC 180/375 | 0.7087 | 16.5254 | 0.2104 | 18.6596 |
AVIS HC 200/375 | 1.4175 | 15.4187 | 0.2104 | 18.6596 |
AVIS BC 200/425 | 0.7987 | 22.3349 | 0.2380 | 25.2093 |
AVIS HC 220/425 | 1.5975 | 20.8392 | 0.2380 | 25.2093 |
AVIS BC 220/475 | 0.9000 | 30.1717 | 0.2693 | 34.0411 |
AVIS HC 250/475 | 1.8000 | 28.1512 | 0.2693 | 34.0411 |
AVIS BC 250/530 | 1.0125 | 40.5980 | 0.3041 | 45.7866 |
AVIS HC 280/530 | 2.0250 | 37.8793 | 0.3041 | 45.7866 |
AVIS BC 280/600 | 1.1250 | 52.9436 | 0.3391 | 59.6890 |

Chapter 144

TAL file format (SIC version 4.33)

The .TAL file is a text file describing all the bathymetric information needed by EDITAL and TALWEG for the old SIC 4.33 versions or earlier. These files can be imported and/or exported from SIC version 5 and later (project editor EDISIC). This is the reason why this format, now obsolete, is nevertheless described in this documentation.

Its structure is hierarchical as regards the order and nature of the information, and includes the following parts, mandatorily in this order:

-* Network description -* Description of the network nodes -* Description of the network reaches & Description of the reach sections -* Description of the branches.

Each part includes a certain number of elements, whose storage structure is organized in lines. The first character of the line identifies the type of line, hence the element to which the line refers. In this version of the software, the following characters are "reserved":

! identifies a network $identifies a node # identifies a reach N identifies a section % identifies a branch * identifies a comment line + identifies a continuation line.

Attention ! In the descriptions below, each comma is translated into a space character in the file, and each line is terminated by a carriage return (the \r\n character in c).

144.1 Network description (1 line)

'!', title, version, consistency, iflagx, xlm, ylm format: A1, A40, F5.3, I1, I1, F10.4, F10.4 where: version is the version of EDITAL (or EDISIC) used to

generate this file, consistency is 1 if the network is consistent and 0 otherwise, iflagx : if it is 1 the longitudinal chainages must be multiplied by 1000 xlm : width scaling factor for the whole network ylm : height scaling factor for the whole network

Nota Bene: TALWEG could test "consistency" to analyze or not the network! For the moment it systematically re-checks the consistency.

144.2 Description of a node

'$', name, x, y, consistency format: A1, A6, I5, I5, I1 where: name is the name of the node x and y are the coordinates of the node in pixels, consistency is 1 if the node is consistent and 0 otherwise (always 1 for this version).

144.3 Description of a reach (2 lines + the sections of the reach)

144.4 1st line

'#', branch name, upstream node name, downstream node name, step, sin, aff, name, xlm, ylm format: A1, A8, A6, A6, I5, F5.3, I1, A40, F10.4, F10.4 where: step: is the calculation step sin: is the sinuosity of the reach = ratio of the medium bed length of the reach / minor bed length of the reach (between 0 and 1). aff: is a parameter that is 0 by default and 1 if one wants all the computational sections of this reach to be stored in the result file in transient flow (.RES). This allows a greater precision in the presentation of the results in transient flow. This is new in versions from 3.3 on. name: the name of the reach xlm: width scaling factor for the whole reach ylm: height scaling factor for the whole reach

Nota Bene: in the old SIC 2.1 version a - sign may appear in the 2nd column if the reach has been inverted. This peculiarity has disappeared in the files of the versions later than 3.0. Now the inverted reach is stored the same way as a normal reach, with its real upstream and downstream nodes.

144.5 2nd line

'+', upstream node number, downstream node number, x, y, consistency format: A1, I5, I5, I5, I5, I1 where: node number is the order of appearance of the node in the list of nodes which precedes, x and y are the coordinates of the "midpoint" of the reach in pixels, consistency is 1 if the reach is consistent and 0 otherwise.

144.6 Description of a section (variable number of lines)

144.7 1st line

'N', name, chainage, type, singular, display, consistency, xlm, ylm format: A1, A14, F8.1, A1, A1, A1, I1, F10.4, F10.4 where: chainage is the chainage of the section, type is: C if the section is Circular D if the section is Culvert R if the section is Rectangular P if the section is Power T if the section is Triangular or Trapezoidal L if the section is Width-elevation A if the section is Abscissa-elevation

singular: is S if the section is a singular section display: is D if the section is a display section consistency is 1 if the section is consistent and 0 otherwise xlm: width scaling factor for the section ylm: height scaling factor for the section

The number of lines depends on the type of section:

for the parametric sections, the number of lines is 2,

for the width-elevation and abscissa-elevation sections, the number of lines is: 4+2*(number of points-1)/8 (unless there is no point, in which case there are only 2 lines).

144.8 2nd line and following

The second line (and the possible following lines) depend on the type of section.

Circular section '+', bottom elevation, bank elevation, radius, overflow elevation of the minor - medium beds format: A1, F7.2, F7.2, F7.2, 7X, F7.2

Rectangular section '+', bottom elevation, bank elevation, width, overflow elevation of the minor - medium beds format: A1, F7.2, F7.2, F7.2, 7X, F7.2

Culvert or Trapezoidal (or Triangular) section '+', bottom elevation, bank elevation, width, slope, overflow elevation of the minor - medium beds format: A1, F7.2, F7.2, F7.2, F7.2, F7.2

Power section '+', bottom elevation, bank elevation, width, coefficient, overflow elevation of the minor - medium beds format: A1, F7.2, F7.2, F7.2, F7.2, F7.2

Abscissa-elevation and Width-elevation section 2nd line '+', reference elevation, overflow elevation of the minor - medium beds

format: A1, F7.2, F7.2

following lines (in pairs) (as long as there are points) '+', (chainage or width [I] I = 1 to 8) '+', (elevation [I] I = 1 to 8) format: 1A, (F7.2), x 8

144.9 Description of a branch

144.10 1st line

'%', name, color, consistency format: A1, A8, I10, I1 where: color is the RGB color of the branch consistency is 1 if the branch is consistent and 0 otherwise.

The number of lines depends on the number of reaches contained in the branch. It is: 2 + (number of reaches - 1)/12 if the branch contains at least one reach, and 1 otherwise.

Each following line is composed of: '+' (number of the reach I = 1 to 12) format: (A1, (I5), x 12) where: number of the reach is the order of appearance of the reach in the EDITAL file.

Chapter 145

FLU file format (SIC version 4.33)

This ASCII file is created by EDIFLU or with any ASCII file editor. It contains the hydraulic data for the steady flow calculation program FLUVIA. It will be checked with the EDIFLU program and a file (.DON) for FLUVIA will be created.

This file is for the old SIC 4.33 versions or earlier. These files can be imported from SIC version 5 and later. This is the reason why this format, now obsolete, is nevertheless described. They can be imported in interface mode, to create a new scenario, or also in batch mode (one or several FLU files), in complement of a TAL geometry file.

No special order is followed to write this data file. The various possible line types are described below, each line starting with a particular emblem.

145.1 The comment line

Contains a "*" character in the first column and a comment after. Study title line '!', TITM, VERNUM in format (A1,1X,A40,1X,F5.3,I1) '!' in column 1, TITM title of the study, VERNUM version number of EDIFLU used to create this file IFLAGQ large discharges mode (0 if no, 1 if yes, i.e. the discharges will be multiplied by 1000)

145.2 Line for the calculation parameters

These data were stored in the .LDA file in the versions earlier than 3.5. This file is not used anymore, this information now being directly in the .FLU file. 'X',

XLAMBQ, XLAMBZ, ERQ, ERZ, NMI, NMAX, IDHA, IWRIT, IPAS in format (A1, 4F10.5, 2I5, 3(1X, I1)) XLAMBQ coefficient_Lambda of error_norm_Q_inflow (oftakes) XLAMBZ coefficient_Lambda of error_norm_Z1 (diffluences) ERQ maximum error on the continuity at the nodes with offtakes ERZ half maximum error at the diffluences = (Z_upstream - Z_average) NMI maximum number of iterations NMAX maximum number of calibration iterations IDHA index for automatic head loss at the expansions (0: no, 1: yes) IWRIT to have more messages in the .LST file created by Fluvia (Debug mode) IPAS to have the automatic small time step enabled when the water slopes are steep.

145.3 Weir line

'D', NB, NS, NUM, XL, ZD, COF, IFOL, IFOC in format (A1, I3, I4, I3, 2F10.4, 10X, F10.4, 30X, 2I2) D in column 1 NB = number of the reach NS = number of the section in the reach NUM = number of the device XL = width of the sill ZD = elevation of the sill COF = discharge coefficient IFOL = equation (0: Cemagref weir/orifice, 1: Cunge, 2: Non-submerged, 3: Cemagref weir/undershot gate, 4: Matlab, 5: Fortran). Modes 4 and 5 are not yet handled IFOC = 0 (not used for the moment)

145.4 Gate line

'V', NB, NS, NUM, XL, ZD, W, COF, Z1, HS, IFOL, IFOC in format (A1, I3, I4, I3, 6F10.4, 10X, 2I2) V in column 1 NB, NS, NUM, XL, ZD same meaning as above W = Opening if IFOC=0 and maximum opening if IFOC=1 (varies the opening) or maximum sill elevation increment if IFOC=2 (varies the sill elevation) COF = discharge coefficient Z1 = target elevation if IFOC=1 or 2 HS = height of the gate (useful in case of overflow) IFOL = equation (0: Cemagref weir/undershot gate, 1: Cunge, 2: Submerged, 3: Cemagref weir/orifice, 4: Matlab, 5: Fortran). Modes 4 and 5 are not yet handled IFOC = 0 for the gates and 1 for the gates adjustable in upstream regulation

145.5 Gec-Alsthom gate line

'G', NB, NS, NUM, XL, ZD, W, COF, Z1, HS, S1S2, IDNS, IFOL, IFOC, NOT in format (A1, I3, I4, I3, 6F10.4, F7.1, I3, 3I2) G in column 1 NB, NS, NUM = same meaning as above XL = Radius of the gate deck (R) ZD = Sill elevation W = Opening (used if IFOL = 2) or otherwise maximum vertical opening COF = Maximum upstream water level (ZMAX for AMIL) relative to the elevation of the axis (between 0 and d) Z1 = Elevation of the axis HS = Decrement (d) S1S2 = Ratio S1/S2 free surface / slot surface for AVIS, AVIO and mixed gate bays IDNS = Shift of the section for Z2 (useful for AVIS, AVIO and mixed gates) IFOL = Type of law (0 Goussard, 1 Cemagref non-locked, 2 Cemagref locked) IFOC = type of gate. AMIL, AVIS, AVIO, high head, low head (0 Amil, 1 Avio

High Head, 2 Avio BC, 3 Avis HC, 4 Avis BC, 5 Mixte 45, 6 Mixte 55, 7 Mixto 45) NOT = Number of the type (various AVIO, AVIS and AMIL models)

145.6 Offtake line

'P', NOEUD, TYPEP, IFPD, TYPEC, QP, XL, ZD, WMAX, W, COF, Z2, XMV, Q0V, Z0V, IFPL in format (A1, A6, A1, I1, A1, F10.3, F9.3, F7.2, F8.3, F8.3, F4.2, F7.2, F4.2, F6.2, F7.2, I2) P in column 1 NOEUD = name of the node on 6 characters in columns 2 to 7 TYPEP = type of offtake on 1 character in column 8 Q = imposed discharge V = gate D = weir A = APM P = pipe L = rating curve IFPD = downstream condition on 1 character in column 9 0 = nothing 1 = constant elevation 2 = weir 3 = rating curve TYPEC = calculation mode on 1 character in column 10 F = off N = on Q = discharge calculation W = opening calculation Z = sill elevation calculation L = sill width calculation I = imposed discharge QP = discharge at the offtake on 10 characters in columns 11 to 20 XL = width or diameter on 9 characters in columns 21 to 29 ZD = sill elevation on 7 characters in columns 30 to 36 WMAX = maximum opening on 8 characters in columns 37 to 44 W = opening on 8 characters in columns 45 to 52 C0F = discharge coefficient on 4 characters in columns 53 to 56 Z2 = downstream elevation or downstream sill elevation on 7 characters in columns 57 to 63 XMV = downstream sill coefficient or power on 4 characters in columns 64 to 67 Q0V = discharge or width on 6 characters in columns 68 to 73 Z0V = elevation corresponding to Q0V on 7 characters in columns 74 to 80. IFPL = equation (0: Cemagref or 1: Cunge)

145.7 Line for the Strickler coefficients

'K', NB, NI, NS, SK in format (A1, I3, 2I4, F10.4) K in column 1 NB = number of the reach on 3 characters in columns 2 to 4 NI = number of the first computational section on 4 characters in columns 5 to 8 NS = number of the last computational section on 4 characters in columns 9 to 12 SK = value of the Strickler coefficient of the section NI to NS on 10 characters in columns 13 to 22

145.8 Line for the lateral flows

'L', NB, NI, NS, QL in format (A1, I3, 2I4, F10.4) L in column 1 NB = number of the reach on 3 characters in columns 2 to 4 NI = number of the first computational section on 4 characters in columns 5 to 8 NS = number of the last computational section on 4 characters in columns 9 to 12 QL = value of the lateral flow of the section NI to NS on 10 characters in columns 13 to 22

145.9 Line for the downstream conditions

'A', NOEUD, QAV, ZAV in format (A1, A6, 3X, 2F10.3) A in column 1 NOEUD = name of the downstream model node on 6 characters in columns 2 to 7 QAV = discharge on 10 characters in columns 11 to 20 ZAV = elevation on 10 characters in columns 21 to 30

145.10 Line for the diffluences

'B', NB, X in format (I3, F10.3) B in column 1 NB = number of the reach X = corresponding discharge (in m3/s, % or area of the downstream reaches. Only the relative values are important)

145.11 Line for calibration elevations

'C', NB, NS, COT in format (2I3, F8.2) C in column 1 NB = number of the reach NS = number of the section COT = corresponding calibration elevation

145.12 Line for the calculation mode at the nodes

By default, Fluvia does the calculation by writing the equality of the elevations at the nodes. The following option proposes to use instead the equality of the total heads at the nodes, with possibly introduction of a head loss at the entry and exit of the reach. This head loss is of the form DH=K.V*V/(2g). 'H', I, PCAM(I), PCAV(I) in format (A1,I3,6X,2F10.5) H in column 1 I = number of the reach PCAM(I) = head loss coefficient at the upstream of the reach PCAV(I) = head loss coefficient at the downstream of the reach

Chapter 146

SIR file format (SIC version 4.33)

This ASCII file is created by EDISIR or with any ASCII file editor. There are some sections that are handled by SIRENE but not entirely by EDISIR. EDISIR can read and write them but not enter them at the interfaces. These sections must therefore be entered with an ASCII file editor. It contains the hydraulic data for the transient flow calculation program SIRENE. This file is directly the one intended for SIRENE, there is no intermediate binary equivalent unlike the steady flow calculation (DON binary equivalent of the ASCII FLU file).

This file is for the old SIC 4.33 versions or earlier. These files can also be imported from SIC version 5 and later (in interface mode, but not in batch mode). Besides, if you have old SIC version 4 projects you may want to understand this format to enter the corresponding data in a SIC version 5 project. This is the reason why this format, now obsolete, is nevertheless described. Preliminary versions of version 5 used at one time files in this format, to support the transition between version 4 and version 5, but that is not the case anymore.

No special order is followed to write this data file. The various possible line types are described below, each line starting with a particular emblem. However, to import data to an object, this object must already be defined (e.g.: a gate must be defined before entering a time-dependent opening law in it).

This file contains all the hydraulic information needed for the transient flow calculation of the SIRENE program. The geometric data are in separate files, the same ones as for the steady flow. When the unit is not specified the SI units are used.

For the version 4 files all the time laws are understood relatively, that is to say that if there is a gap between the initial value in the initial water surface profile and the value at the initial time of this law, then the values of the law will all

be shifted by this gap to have a continuity between the initial regime and the transient one that we want to simulate.

For the version 5 files this way of handling the laws in the calculation has changed and this shifting of the laws is no longer done, the laws then being considered as absolute. This is true for all the laws, except the time-dependent lateral flow laws which are always shifted by the initial gap, if any. For version 5 it is the data editors that handle the possible automatic re-alignment of the laws with respect to the initial values.

Study_title

TITM, VERFIC, IFLAGX, IFLAGQ FORMAT ('!', A40, 1X, F5.3, 1X, I1, 1X, I1)

TITM title of the study, VERFIC version of the file (3.0, 3.1, 4.0, 4.32, 4.33, etc.), IFLAGX large chainages mode (0 if no, 1 if yes, i.e. the longitudinal chainages will be multiplied by 1000), IFLAGQ large discharges mode (0 if no, 1 if yes, i.e. the discharges will be multiplied by 1000)

Reaches and Nodes

IB, N1, N2, TITB FORMAT ('N', 3I3, A40)

IB reach number, N1 upstream node number of the reach IB, N2 downstream node number of the reach IB, TITB title of the reach

From version 5 on the format is modified as follows: FORMAT ('N', 3I3, A40, 2I5, 2F10.5) with the same data as above and 4 new data: IFBPC(I,1), IFBPC(I,2), PCAM(I), PCAV(I) IFBPC(I,1)=1 if equality of the total heads at the upstream, =0 if equality of the elevations, IFBPC(I,2)=1 if equality of the total heads at the downstream, =0 if equality of the elevations, PCAM(I) head loss at the entry of the reach, PCAV(I) head loss at the exit of the reach

Calculation parameters 1

IFLTI, IREP, TREP, IDHA, NEWT, NQUAS, IDISCR FORMAT ('B', 1X, I1, 1X, I1, 1X, F10.0, 1X, I1,1X,I1,1X,I3,1X,I1)

IFLTI flag_inertia_term (1: with classical inertia term, 2: local supercritical, 0 global supercritical), IREP flag_continuation_files_res/rcs (0: no), TREP restart_time (hot start) in mn, IDHA index for automatic head loss at the expansions (0: no, 1: yes), this flag is passed on from the one indicated in the .flu file, NEWT = 0 if fixed point, 1 if Newton, 2 if Fixed point + Newton, NQUAS = 0 if pure Newton, N if Quasi-Newton: derivative calculation frequency, IDISCR discretization index (0: classical, 1: homogeneous with the steady one)

From version 5 on the format is modified as follows:

IFLTI, IREP, TREP, IDHA, INISV, NLSV, IDISCR

INISV=0 classical initialization with the value of the previous step, INISV=1 initialization with the variation of the previous step, NLSV=0 linear calculation,

NLSV>0 nonlinear calculation, gives the Jacobian update every NLSV iterations

Calculation parameters 2

IPS, IFR, IRCS, DT, TDEB, TFIN, XX, WW, XLAMB, ITER1, ERZ0, ITER2, EPSI FORMAT ('C', I3, 1X, 2(I1, 1X), F10.2, 1X, 2(F10.0, 1X), 2(F4.2, 1X), F8.6, 1X, 2(I4, 1X, F8.6))

IPS result_storage_step (every 1 or n calculation steps, to limit the size of the .RES files), IFR regulation_flag (0: without use of a .reg regulation file, 1: with), IRCS full .RCS file flag (0: without creation of this file, 1: with creation of this file which is necessary for a later hot restart, i.e. a transient calculation starting from another transient calculation), DT time_step (in minutes), TDEB start_time of the calculation (in minutes), TFIN end_time of the calculation (in minutes), XX upper_limit of effective_discharge_at_the_offtakes, WW lower_limit of effective_discharge_at_the_offtakes, XLAMB contraction lambda coefficient (0<lambda<=1), ITER1 number of iterations (singular sections, ponds), ERZ0 numerical value of the precision of the nonlinear system, ITER2 number of iterations (oftakes), EPSI maximum error on the local iterations for the offtakes

Initial water surface profile (start_time)

IB, IS, IQL, Q, Z, QL, SK, SKM FORMAT ('I', 3I3, 2F10.5, F10.7, 2F10.5) in version 4.32 FORMAT ('I', I3, I4, I3, 2F10.5, F10.7, 2F10.5) in version 4.33

IB reach, IS relative_section_number_of_reach, IQL lateral_flow_indicator (0: without lateral flow, 1: with lateral flow), this flag comes from what has been entered in the .flu file, it is what will allow identifying the sections on which to possibly distribute the time-variable lateral flows (cf section below), Q discharge (m3/s), Z elevation (m), QL lateral_flow (m3/s/m, i.e. with a 10-6 factor with respect to what has been entered in the .flu file), SK strickler_medium_bed, SKM Strickler_minor_bed

Weirs or gates

IB, IS, KS, (CSG(NSG, I), I=1, 5), IFOL, ISF FORMAT ('D' ou 'V', 3I3, 5F13.6, 2I2) in version 4.32 FORMAT ('D' ou 'V', I3, I4, I3, 5F13.6, 2I2) in version 4.33

IB number of the reach, IS relative_section_number, KS relative_singularity_number, (CSG(NSG, I), I=1, 5) characteristics of the singularity NSG (1 sill_width, 2 sill_elevation, 3 opening, 4 discharge_coef, 5 overflow_height), IFOL = equation (0: Cemagref 88 sill-gate for V or sill-orifice for D, 1: Cunge 80, 2: Classical submerged for V or non-submerged for D, 3: Cemagref 88 sill-orifice for V or sill-gate for D, 4: MatLab, 5: Fortran), ISF = calculation mode (0: nonlinear, 1: linear)

Gec-Alsthom gates

IB, IS, KS, (CSG(NSG, I), I=1, 8), IDNS, IFOL, IFOC, IFLO, ISF FORMAT ('G', 3I3, 7F13.6, F7.1, I3, 2I2, I3, I2) in version 4.32 FORMAT ('G', I3, I4, I3, 7F13.6, F7.1, I3, 2I2, I3, I2) in version 4.33

IB number of the reach, IS relative_section_number, KS relative_singularity_number, (CSG(NSG, I), I=1, 8) characteristics of the singularity NSG (1 radius, 2 sill_elevation, 3 maximum opening, 4 Zamont_max, 5 decrement, 6 axis_elevation framed with respect to ZMIN, 7 actual opening, 8 ratio S1/S2 free surface / slot surface for AVIS, AVIO and mixed gate bays for Versions >= 4.17), IDNS=shift of the section for Z2 (useful for AVIS, AVIO and mixed gates) in number of computational sections (Versions >= 4.17), IFOL = equation (0: Goussard, 1: Goussard for position and Cemagref for discharge, 2: W given by the user and Cemagref for discharge), IFOC = type (0: Amil, 1: Avio High Head, 2: Avio BC, 3: Avis HC, 4: Avis BC), IFLO = model, ISF = calculation mode (0: nonlinear, 1: linear).

Names of the nodes, sum of discharges and elevations at the nodes

TC(NOMAX), ITN(NOMAX), QAE(NOMAX), ZNO(NOMAX), I FORMAT ('S', A6, I3, 2F10.5, 7X, I3)

TC node_name (alphanumeric), ITN node_type = 1 if model_upstream, 2 if intermediate_node, -2 if model_downstream, QAE sum_of_discharges = (Sum_Q_leaving - Sum_Q_arriving) at the non model_downstream node or leakage discharge at the model_downstream node, ZNO sum_of_elevations, I number of the node

Oftakes defined by their geometric characteristics

I, KP(I), KVP(I), QPO(I), QPR(I), X1P(I), X2P(I), X3P(I), X4P(I), X5P(I), X6P(I), X7P(I), X8P(I), X9P(I), IFPL(I) FORMAT ('P', 3I3, 2F7.3, F6.2, 3F10.5, 3F6.2, F7.3, F6.2, I2)

I node (non model_upstream and non model_downstream) (KP) offtake_case = 1 Q(t) gate calculate opening W 2 Q(t) weir calculate Z_sill 3 Q(t) weir calculate sill_width 4 gate calculate Q 5 weir Z(t) calculate Q 6 apm calculate Q 7 pipe calculate Q 8 Q(z) calculate Q 9 weir L (t) calculate Q (KVP) offtake_downstream_case = 1 offtake_downstream_elevation constant 2 offtake_downstream_weir 3 offtake_downstream_Q(z)

the Q(z) laws at the offtakes are introduced in analytical form: Q=Q_base*((Z-Z_sill)/(Z_base-Z_sill))**exponent

(QPO) offtake_target_discharge (QPR) offtake_actual_discharge (X1P) discharge_coefficient (KP=1, 2, 3, 4, 5, 6, 7, 9) or exponent (KP=8) (X2P) Z_sill (X3P) sill_width (KP=1, 2, 3, 4, 5, 6, 9) or diameter (KP=1, 4, 7) or Q_base (KP=8) (X4P) opening (KP=1, 4, 6) or Z_sill_min (KP=2) or Z_base (KP=8) (X5P) maximum_opening (KP=1, 4 and X5P>0) or indicates circular gate (KP=1, 4 and X5P=0) or Z_sill_max (KP=2) or maximum_sill_width (KP=3) or switch (KP=6, 7, 8) (X6P) offtake_downstream_discharge_coefficient (KVP=2) or offtake_downstream_exponent (KVP=3) (X7P) offtake_downstream_elevation (KVP=1) or offtake_downstream_sill_elevation (KVP=2, 3) (X8P) offtake_downstream_sill_width (KVP=2) or offtake_down-

stream_Q_base (KVP=3) (X9P) offtake_downstream_Z_base (KVP=3) IFPL = equation (0: Cemagref or 1: Cunge)

Q(z) laws at the model downstream node

I, Z, Q FORMAT ('A', I3, 6X, 2F10.3)

I downstream_node, Z downstream_elevation (m), Q downstream_discharge (m3/s)

Display parameters (display)

CODEM, BRANM, NAMEM, NBM, NSM, XLONG, ZREF FORMAT ('Y', A1, A8, A6, 2I3, F8.0, F8.2) in version 4.32 FORMAT ('Y', A1, A8, A6, I3, I4, F8.0, F8.2) in version 4.33

CODEM code, BRANM branch_name, NOEUDM node_name, NBM reach, NSM relative_section_number, XLONG chainage, ZREF reference_elevation

Variations of the gate openings

IB, IS, KS, IO, T, X, K, L, ITYPOUV FORMAT ('U', 4I3, 2F10.3,3I3)

IB reach, IS relative_section_number, KS relative_singularity_number, IO device type: 1 (Weir), 2 (Rectangular gate) and 3 (Gec-Alsthom gate) from version 4.07 included (before version 4.07 only the mobile device gates were handled) T time (mn), X opening, K number of the law, L number of the point of the law, ITYPOUV (0: step, 1: ramp)

Hydrograph or law at the offtake at the non downstream node of the model

CH1, J, T, X, I, L, IOD, NOE, ITYPOFF FORMAT ('H', A1, 2X, I3, 3X, 2F10.3, I3, I5, 3I3)

CH1 offtake_law_character: 'Q' offtake_discharge, 'W' offtake_opening, 'S' offtake_switch, 'L' offtake_sill_width, 'Z' offtake_Z_sill, J non_downstream_node, T time (mn), X law_value, I number of the law, L number of the point of the law, IOD(I) type of offtake, NOE(I,3) calculation mode at the offtake, ITYPOFF (0: step, 1: ramp)

Target discharges

J, T, Q, I, L FORMAT ('O', I3, 6X, 2F10.3, 2I3)

J node, T time (mn), Q target_discharge, I number of the law, L number of the point of the law

Pond (S(z) law at the node)

I, Z, S, X FORMAT ('K', I3, 6X, 3F10.3)

I node, Z elevation, S surface area (ha), X evaporation_rate (cm/day)

Lateral hydrograph (handled by Sirene but not by Edisir)

This section allows making the infiltration - evaporation term evolve (if the term is negative) or the inflow term (if the term is positive) as a function of time. In steady flow, this term was a constant, possibly different for each computational section. The value entered for the steady flow was expressed in l/s/km. In transient flow, this term can be defined as a function of time. The value entered is a total discharge (in m3/s) for the reach considered. This discharge will be distributed over the zones within the reach for which there is a lateral flow, proportionally to their lengths. A priori, by default, these zones correspond to the ones thus defined in steady flow. But it is possible to modify them by putting 0s or 1s in the corresponding column of the "I" lines of the .SIR file. The discharge is not modulated proportionally to the initial lateral flows. It will be distributed homogeneously over all the zones having a lateral flow. We therefore assume that in the reach there are zones with and zones without lateral flow, and that, in the zones where there is a lateral flow, the rate of lateral inflow or of lateral loss is identical over all these zones (in discharge/length unit). If this is not the case this reach must be split into several reaches.

If the total initial lateral flow indicated in this lateral hydrograph for this reach is not equal to the total initial lateral flow coming from the steady flow calculation, the hydrograph will be translated to ensure the continuity at the start between the steady flow and the transient regime. This is true for the version 4 of SIC but also for the version 5.

However, it is not possible, for the moment, to enter an infiltration - evaporation term as a function of the water depth. This can possibly be modeled approximately by a succession of offtakes in discharge calculation mode. This limitation seems to us not very constraining in the measure where an estimation of the infiltration - evaporation term is very delicate with a very important uncertainty factor and it is illusory to consider that a q=f(z) law will give a more realistic model.

This "time-dependent lateral hydrograph" section will soon be handled entirely by Edisir (entry at the graphic interfaces) in a later version. Edisir already allows reading and writing it to file, but not filling it through the interfaces (since version 3.8). Sirene handles it perfectly at the calculation level.

IB, K, T, Q FORMAT ('L', 2I3, 3X, 2F10.3)

IB reach number, K step (0) or ramp (1) flag T time (mn), Q lateral_flow (m3/s)

Limnigraph at the model downstream node (handled by Sirene but not by Edisir)

This section allows replacing the Q(z) law entered for the steady flow calculation by a z(t) law allowing for example to represent the downstream influence of a tide.

To be handled by Edisir in a later version. Edisir already allows reading and writing it to file, but not filling it through the interfaces.

The z values are handled relatively with respect to the first value. For example,

if you want to increase the z elevation by 1m at the instant 2h=120mn, it is enough to enter:

Z 2 0 0.000 0.000 Z 2 0 120.000 1.000

Attention: if you enter a z(t) law you must remember to remove the Q(z) law because otherwise it will be kept because it is prioritary with respect to the z(t). A Q(z) law is also prioritary with respect to a pond that would close the channel. If this pond is entered from the interfaces, then the Q(z) will be correctly removed from the .SIR file. However, if this pond is entered by hand directly in the .SIR file then you must remember to remove the Q(z) so that it is no longer used.

The calculation with a z(t) is much faster than with a Q(z) because there are no nonlinear iterations for the calculation at the downstream node. However, this z(t) downstream condition is sometimes considered as "hard" hydraulically and can generate oscillations on the discharges in this zone, for example in the case where an upstream hydrograph arrives at a downstream node with a constant z(t). This z(t) downstream condition is indeed not very physical because the "surface pin" does not exist. The tide is physical but is not a simple imposition of an elevation, but a process that also links the discharges and the elevations via complex physical processes. In the case of a downstream sinusoidal z(t) mimicking a tide there is, a priori, no spurious oscillation.

Even if this is possible at the .SIR file level, it is not recommended to have both a pond and a z(t) for the same downstream node. In this case we would impose the discharge as well as the elevation, which causes numerical problems. In a later version this mode will be explicitly forbidden and in this case the z(t) will be effective and the pond inoperative.

I, K, T, Z

FORMAT ('Z', 2I3, 3X, 2F10.3)

I number of the downstream_node, K step (0) or ramp (1) flag [option since version 3.7] T time (mn), Z downstream_elevation

Chapter 147

Part III

Theoretical concepts

Chapter 148

Description of the hydraulic network

The hydraulic network is divided into homogeneous sections, the reaches being located between an upstream node and a downstream node. Relations between reaches occur only at the nodes.

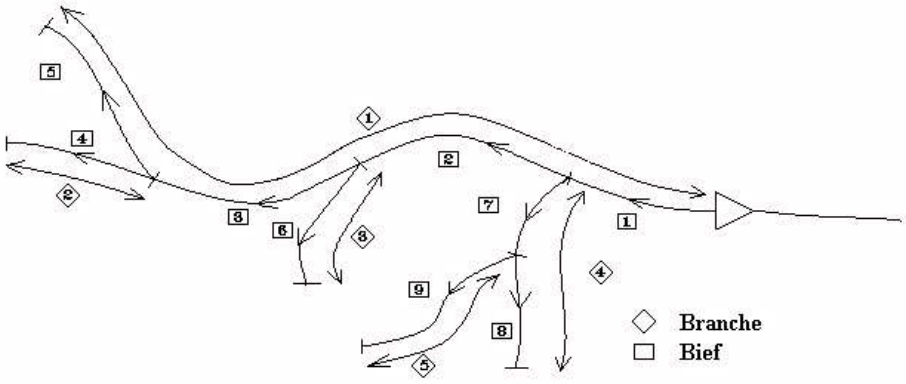


Figure 148.1: doc39

Chapter 149

Choice of reaches

The choice of reaches by the model user is subject to some constraints which are detailed below: - constraints due to network topology. - constraints associated with points of inflows or outflows (they can occur only at model nodes).

The user may however divide any part of the canal into several reaches in order to take into account some particularity, even if such a division is not imposed by the constraints described above.

For instance, one can create a different reach for a lined canal zone (low roughness), and an unlined canal zone (high roughness). One can also create reaches for administrative reasons, etc.

The division into reaches for the user's convenience does not influence the results of the hydraulic calculation. Generally, when one divides a reach artificially, the lowermost cross-section of the upstream reach is the same as the uppermost cross-section of the reach immediately downstream.

If different regulating/control devices exist across the canal, they can be integrated within a reach and do not need any special division. This approach facilitates the modeling of the hydraulic transition from free flow conditions to submerged conditions at such devices.

Chapter 150

Choice of branches

The choice of reaches is principally linked to the hydraulic constraints. In order to let the user visualize portions of the canal that he wants to treat together, he may group a number of reaches into a branch. A branch is therefore a group of reaches serially linked to one another.

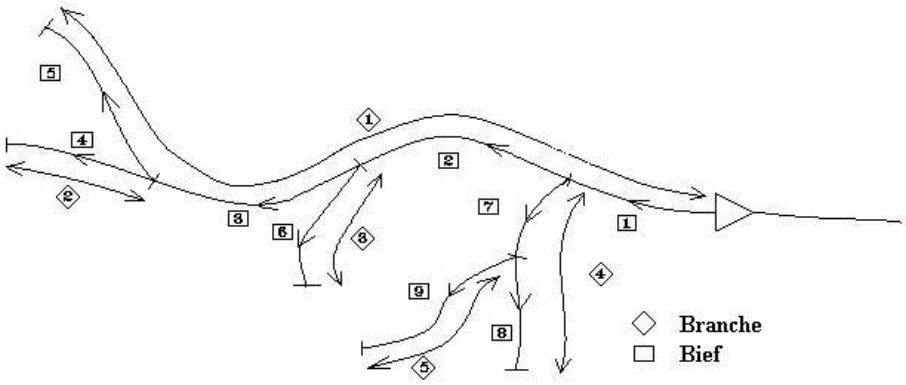


Figure 150.1: doc39

Illustration of how a canal network is subdivided into reaches and branches.

If the canal is not looped nor branched, the division into reaches and branches is thus simplified to the scheme shown in Figure 2 (in this example the system has been divided into two branches).

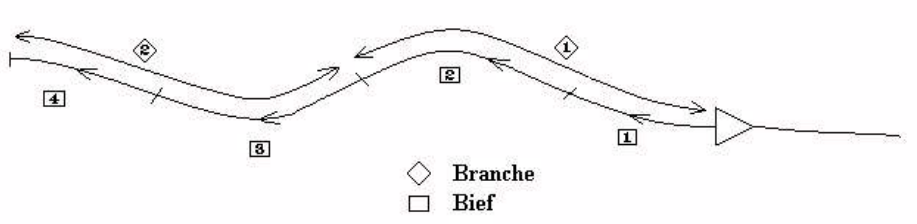


Figure 150.2: doc40

Chapter 151

Classification of reaches

Reaches are identified by their nodes. The position of a reach in the network is entirely defined by the names of its upstream and downstream nodes. The direction of flow is defined at the same time. The network topology is simply described as an oriented graph.

The reaches constitute the arcs of that graph, delineated by the nodes, upstream and downstream. They are automatically numbered by the program according to the order in which they are input in the data file.

Subcritical flow being controlled by the downstream conditions, the calculation of a water surface profile proceeds upwards, commencing at the downstream end.

Therefore, a relationship between water surface elevation and discharge is needed as a downstream boundary condition to start the calculation.

A hydraulic network can be represented by a connected oriented graph, without closed circuits, along which one can move with the help of the following algorithm:

One must look for a node where no arc arrives, and from where only one arc leaves.

The latter arc is added to a list, and is then removed from the graph.

The process is recursively applied on the resulting sub-graph, as long as nodes (as defined above) exist.

Then, either the problem is over, or arcs not added to the list still remain.

In the latter case, one searches in the remaining sub-graph a node where no arcs arrive, and from where more than one arc leaves.

Those are selected, then they are removed from the graph, and the process is recursively applied to the resulting sub-graph, as long as such nodes exist. Then, either the problem is over, or non-listed arcs still remain.

In the latter case, the algorithm is applied again, from the beginning, on the remaining sub-graph.

Reading backwards, the list of arcs thus obtained, one can find a satisfactory partial relationship (Remark: only the upstream nodes of the reaches are processed by this algorithm). From the list, one can also establish the identity of reaches of a given loop. These reaches are gathered starting from the first distributory point (the two first successive reaches with the same upstream node). If the model has several downstream boundary conditions in terms of elevations, then all the reaches following the first distributory point of the list belong to the loop. On the opposite, if the loop is being closed, it stops with the last junction (the last reach with the same downstream node of another reach belonging to the list and located after the beginning of the loop). We will illustrate the use of the algorithm on a simple example, as shown in figure 3.

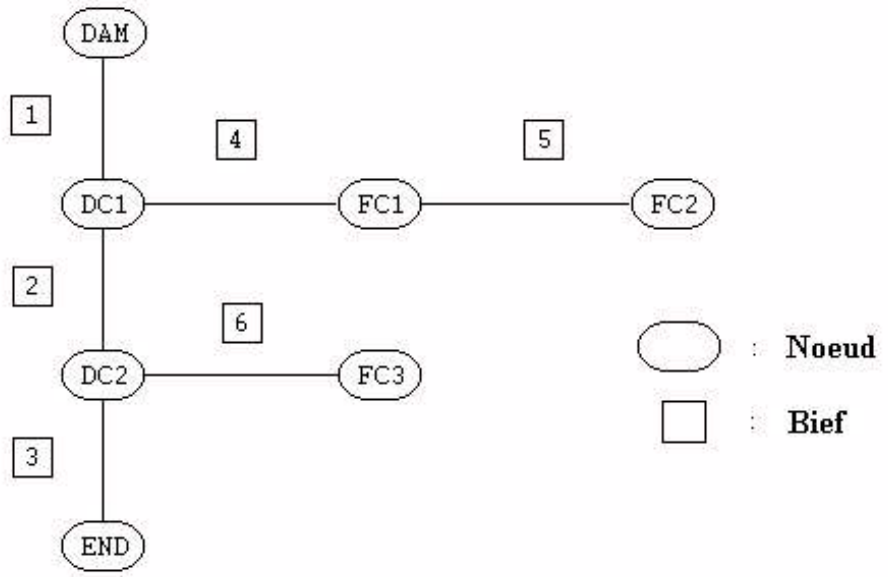


Figure 151.1: doc41

Convention for the reaches of the nodes : - REACH > 0 if it starts from the node - REACH < 0 if it arrives at the node

We have: - DAM [1] - DC1 [2] [4] [-1] - DC2 [3] [6] [-2] - FC1 [5] [-4] - FC2 [-5] - FC3 [-6]

Apply the first criterion:

The first node where no arc arrives and from where only one arc starts is the upstream node.

Add reach *[1]* to the list, and remove it from the graph. Then one obtains:

NODES

- DAM [1]
- DC1 [2] [4] [-1]
- DC2 [3] [6] [-2]
- FC1 [5] [-4]
- FC2 [-5]
- FC3 [-6]
- Liste : [1]

We cannot find any node being the upstream node of one single reach. We look for one node never being a downstream node but being the upstream node of several reaches. Node DC1 is fitting these conditions. Then the reaches number 2 and 4 are added to the list: - DC1 [2] [4] [-1] - DC2 [3] [6] [-2] - FC1 [5] [-4] - Liste : [1], [2], [4]

The same algorithm is applied once again. We get the node named DC2 and the reaches number 3 and 6 are added to the list: - DC2 [3] [6] [-2] - FC1 [5] [-4] - Liste : [1], [2], [4], [3], [6]

We cannot find any other node with two downstream reaches. We then use the first algorithm. We get the node named FC1 and the reach number 5 is added to the list. - Liste : [1], [2], [4], [3], [6], [5]

The loop starts from the first distributory point (reaches 2 and 4, node DC1). All the following reaches of the list belong to the loop, since this loop is not closed (several downstream boundary conditions exist).

Chapter 152

Identification of the upstream and downstream nodes of the model

The program determines the model upstream node(s) where an upstream boundary condition in terms of discharge is required. It also determines the downstream node(s) where a downstream boundary condition (water level or a rating curve) has to be specified.

The upstream nodes of the model can never be simultaneously the downstream nodes of a reach. The downstream nodes can never also be the upstream nodes of a reach. A model which is neither branched nor looped, finally has only one upstream node, corresponding to the main sluice at the dam, and only one downstream node, corresponding to the tail end of the canal.

Chapter 153

Description of data cross sections

The geometry of a reach is determined by the cross sections characteristics of the shape and volume of the channel. Dimensions are identified with respect to a single mark to enable the calculation of the local slopes. Cross sections are identified following the curvilinear abscissa of the channel (x-axis in length).

In general, it is chosen as the origin supply tank dam and orientation as the direction of flow. If the model is meshed or branched we need several long abscissa for the entire model.

The cross sections can be defined in three different ways: abscissa-elevation, width-elevation and parametric. The type of description may vary from one data section to another within a reach.

Chapter 154

Abscissa-elevation

Cross sections provided by the surveyor are usually as indicated in figure 4:

Figure 4

Each point is input in terms of its cross-wise abscissa and its elevation. The section may be introduced either from the left bank or the right bank.

Imagine that one begins from the left bank. The first point need not be entered because the canal bank stops at point 2; similarly for point 9. If one enters points 1 and 9, they will be disregarded by the program. In fact the left and right points of the section have to be disregarded if their elevations are less than or equal to the elevation of the canal bank itself.

You must therefore take these point deletions into account if you enter a section that you want to close to the ceiling. So rather than closing "abruptly" horizontally, or even less well when going down, which would give 2 points at the same height, or an extreme point lower, it will be necessary to maintain a slight increasing slope when returning towards the initial point.

Chapter 155

Width-elevation

Another way to describe a cross section is to enter, for each value of elevation, the section width as drawn in figure 5. That is, one should input couples (Zi, Li), starting from point 1 (decreasing width-elevation), or from point 4 (increasing width-elevation). This description is generally adopted when one does not have precise information on the section or if the section is symmetrical.

Figure 5

If one enters the cross-section in width-elevation couples, any asymmetry in the section is not taken into account. The wetted perimeter is computed assuming a symmetrical section. The only way to take into account an asymmetrical section is to enter it in terms of abscissa-elevation couples. A closed section can also be represented this way, but in this case, width L1 of the figure would be equal to zero.

Chapter 156

Parametric form

Sections of special geometrical shape can be input in parametric form.

Circle: defined by its radius R, the bottom elevation ZF, and the bank elevation ZB. If ZB is not entered, ZB = ZF + 2R.

Figure 6

Culvert: defined by the bed width L, the side slope m (which may be negative) and by the bed elevation ZF and bank elevation ZB.

Figure 7

Power Relationship: defined by XL0, a, the bank elevation and the bed elevation.

L = XL0 * (Y/Y0)a

with:

Y0 = ZB - ZF Y = Z - ZF water depth

The parabolic profile corresponds to a = 0.5

Figure 8

Rectangle: defined by its width, bed and bank elevations.

Figure 9

Trapezium or triangle: Defined by its bed width L (nil for the triangle) and the canal bed and bank elevations. Side slope may be negative in the trapezium case. In case of negative slope, if the section gets closed at a level below ZB, the program makes the necessary adjustment to the bank elevation.

Figure 10

Chapter 157

Siphons

It is possible to model siphons in the SIC software through the so-called "Preissmann slot" [[Cunge J.A., Wegner M., 1964. "Intégration numérique des équations d'écoulement de Barré de Saint Venant par un schéma implicite de différences finies. Application au cas d'une galerie tantôt en charge, tantôt à surface libre". La Houille Blanche, n°1-1964, pp 33-39]] [[Vasconcelos J.G., Wright S.J., 2004. "Numerical modeling of the transition between free surface and pressurized flow in storm sewers". Innovative Modeling of Urban Water Systems, Monograph 12, Chap 10, W. James, Ed., pp 189-214]] [[Ukon T., Shigeta N., Watanabe M., Shiraishi H., Uotani M., 2008. "Correction methods for dropping of simulated water level utilising Preissmann and MOUSE slot models". 11th International Conference on Urban Drainage, Edinburgh, Scotland, UK, pp 1-9]]. This method is applicable both for steady state calculations (Fluvia) or transient calculations (Sirene). The program for calculating the geometry (Talweg) automatically detects the sections which are closed at their upper end. This is possible for sections of width-elevation, abscissa-elevation, culvert (always closed), circular (if bank level high enough zbank >= zbed + 2*r) and trapezoidal (with negative slope and adhoc data) type. Anyway, internally, all these sections are converted into width-elevation format, and then the algorithms are the same for all these sections. The technique known as the Preissmann slot is to create an artificial slot at the top of the section from the ceiling to transform a closed pipe into an open free surface section. But in this artificial slot friction is not taken into account. To be more specific, the wetted perimeter P, section S, velocity V, hydraulic radius Rh=S/P used in the calculation of the pressure drop $J = \frac{Q^2}{(KSR_h^{2/3})^2}$ is limited to the actual section of the siphon. But elevation Z is calculated in the full section including the artificial slot.

In the calculation of the head H, the velocity V is taken in the actual section, and Z in the slot. The width-elevation tables are calculated by Talweg once for all and written in the xml file. When there is a Preissmann slot we can verify

that P does not change in the slot. One can also see that S does not change either in the slot in the tables.

The advantage of this is that all calculations are the same (except for the trick on P, S, V, Rh and H) with or without slot in permanent or transitory flow conditions. The disadvantages are that the volume is stored in this slot, but it must be limited if the slot is thin.

We can show that the equations of Saint Venant pose problem with a width L=0, which one would have without the Preissmann slot from where precisely the idea of this trick.

On the other hand beware: one can have the impression that this is not the case, because when one looks at the wet section for example in the results, one does not find exactly what one would expect. For example, if we have a 9 m wide by 4 m high culvert we would expect to have a surface of 36 m2. In fact we will have 36.0045 m2 (36 + 9/2*0.001). This is due to the fact that the ceiling of the section is slightly modified during the treatment by Talweg, putting a section starting the Preissmann slot to 1mm more than the initial ceiling. Instead of having a flat horizontal ceiling we have a light roof hanging 1mm high, so with a small additional volume. This is to improve the numerical stability of the schema and to have a better defined $\frac{dL}{dZ}$ derivative. To better visualize this one has increased the writing format of S, P and L on the log file of F8.2 to F9.3 (cf Talweg.ans, messages 53, 54, 55 and 56). The wet perimeter P is calculated without this small modification because it only intervenes in the friction term.

The slot width must be chosen small (L=0.01 m in SIC by default, but it can be changed for each section in the section description or using the automatic modification tool) in order not to store too large volumes of water. In sewage systems, where there are manholes, some software uses a trick to reduce the section of those manholes to reduce the volume artificially stored in the slot to get a better volume conservation of the total volume of water into the network.

One can also choose this slot width L (using the automatic modification tool) having a wave speed $c = \sqrt{\frac{gS}{L}}$ close to that of sound waves in water which is about 1500 ms-1.

This modification using the wave speed can be done also using the automatic modification tool. This can lead for example to a slot width in the order of 10-4 m to 10-6 m. With explicit numerical schemes this was a problem because of the very small time step required for the calculation in order to have a Courant number Cr< 1 with $C_r = \frac{(V+c)DT}{DX} < 1$, where V is the velocity of the water, c the celerity of the waves, DT the computation time step and DX the space step between 2 calculation sections. But the implicit Preissmann scheme is not subject to the same constraints. It is unconditionally stable regardless of the Courant number. But a large numerical Cr (or also a very small one, which is not our case) introduces numerical bias (diffusion and phase shift, cf thesis Cunge). By default in SIC we choose a slot width of 0.01 m. For example in

a culvert of 1 m2, for a flow of 1 m3s-1, we get a celerity c of the order of 30 ms-1 with a slot of 0.01 m. For c of the order of 1500 ms-1 we should take a slot width of approximately 0.000005 m. The advantage of a small slot is that the stored volume in the slot is limited. The drawback is that it could be less stable numerically. This is a compromise to find in case of difficulties.

Bibliographical References:

Chapter 158

How to order the reaches in the unknown vector to get an upper triangular matrix H

The sorting algorithm will be described, using the previous example, focusing on the loop reaches (figure 3).

In the NODE column array is placed the list of the upstream nodes of the loop reaches, and in the 2 dimension REACH array is placed the list of the loop reaches connected to these nodes. When a reach joins a node, a negative sign is added to it.

NODEREACH Line 1DC124Ü List of the reachesLine 2DC236-2 connected to the nodeLine 3FC15-4 DC1

In the REACH array, the positive numbers that appear only once are downstream reaches of a loop.

Downstream reaches of a loop: 3, 5, 6

The algorithm can be divided into two steps.

First step: The lines of the REACH array are analyzed. If one (or several) downstream reach of a loop can be found in one line, it is (or they are) stored in the RANGE list. Then the absolute value of all the negative reaches of the line are added to this list.

All the lines matching the criteria are gathered at the top of the REACH array. This way the second step will be applied on the remaining lines.

Summary after the first step:

NODEREACH Line 1DC236-2Line 2FC15-4Line 3DC124

The problem can either be solved (case here), or some reaches can remain. In this latest case, the second step is processed.

Second step: The remaining lines of the REACH array are analyzed. If one positive reach, already belonging to the RANGE list, can be found in a line, then all the absolute values of the negative reaches of the line are added to this list. This line is removed and gathered to the lines processed by the first step. The second step is processed again for each remaining line until all the reaches have been taken.

The order of the RANGE list is inverse of the order of the reaches in the unknown vector. RANGE : 36254 Reaches54321 Corresponding columns in the the loop contains 5 reachesblocks (lines in the unknowns' vector)

Chapter 159

Filling the H, B and A blocks

The H block is filled at the same time as the RANGE list. Some -1 values are placed in this H block, at locations (i, j).

During the first step:

- The column j is given through the downstream reach of the loop
- The line i is given through the negative reach belonging to the same line of the REACH array.

Ex:

=> H(3,5) = -1

=> H(3,4) = -1

=> H(1,2) = -1

During the second step (if any):

- The column j is given through the positive reach
- The line i is given through the negative reach belonging to the same line of the REACH array.

The B block is filled with the equality relations between the water elevations at the distributory nodes. Each line of the REACH array is analyzed and we keep the free pairs of positive reaches.

REACH : Line 136-2Line 25-4 Pairs [3 , 6] and [2 , 4]Line 324

The B block is filled from the bottom. One relation is written for each i line of the B block. The reaches indicates the j columns.

At the last line (i = 5, the loop contains 5 reaches), is written the equality relation between the water elevations at the upstream node of reaches 3 and 6.

=>

At the last but one line (i = 4), is written the equality relation between the water elevations at the upstream node of reaches 2 and 4.

=>

The A block is filled with the discharge continuity equations at the nodes that are not downstream node of a loop.

One i line of the A block corresponds to each line of the REACH array. The j column is given thanks to the reach.

The line i=1 of the A block corresponds to the first line of the REACH array.

Positive reach n°3 => RANGE(j)=3 => j=5 => A(1,5)=1 Positive reach n°6 => RANGE(j)=6 => j=4 => A(1,4)=1 Negative reach n°2 => RANGE(j)=2 => j=3 => A(1,3)=-1

The line i=2 of the A block corresponds to the second line of the REACH array.

Positive reach n°5 => RANGE(j)=5 => j=2 => A(2,2)=1 Negative reach n°4 => RANGE(j)=4 => j=1 => A(2,1)=-1

The line i=3 of the A block corresponds to the third line of the REACH array.

Positive reach n°2 => RANGE(j)=2 => j=3 => A(3,3)=1 Positive reach n°4 => RANGE(j)=4 => j=1 => A(3,1)=1

We have just examined how the model takes into account the network topology. We must now examine how to characterize the geometry of the reaches, basic element of all hydraulic calculations.

Chapter 160

Singular sections

Do not forget to specify the “singular sections” where one or more weirs or gates will be later defined. This definition will be done in the steady flow data editor. You must also enter a section downstream of the weir (or gate) at the same abscissa. It will be then possible to see the discharge and level results at this downstream section of the weir (or of the gate).

<doc108|center>

Chapter 161

Computational sections

Data sections may be unequally distributed along the canal. In fact, the model user should select sections which best represent the canal dimensions, the changes of slopes and so on. Depending on the regularity of the canal, the spacing of the data sections may then be small or large.

For the hydraulic calculation, the spacing between computational sections should be such that a reliable estimation of the water surface profile is possible. This spacing is chosen by the model user depending on his knowledge of the canal hydraulic behavior (when the data sections are too far away, the model interpolates supplementary computational sections in order to allow a better simulation of the water surface profile).

All the entered data sections are retained as computational sections. Irrespective of the manner in which a data section was defined, the program transforms it into width-elevation data for storage and interpolation. Only groups of 4 characteristic values: elevation, width, wetted perimeter (if a section had been entered in terms of abscissa-elevation, the wetted perimeter would take into account the section asymmetry) and area are finally retained.

Chapter 162

Introduction

A main canal network is an open-channel water distribution system, which conveys water from a source (reservoir or river diversion) to various offtakes that deliver water to user groups via secondary and/or tertiary canals.

The hydraulic modeling of such a network needs to take into consideration the real canal topography: the main canal network topology in addition to its geometric description. Unit 1 is responsible for managing all the topographic components used by the model.

Chapter 163

Transformation of a data section into width-elevation format

At any cross section the program looks for all the high and low points. It determines right and left banks and eliminates points outside the bed. Using the high and low points it divides the cross section into channels and for each channel it effects the transformation into width-elevation (see figure 12) format. Then for each elevation it adds the widths of all the channels.

Figure 12

Chapter 164

Interpolation of computational sections

If the distance between two data sections is more than one computational (space) step, the program interpolates computational sections between these two data sections, in accordance with the step decided by the user.

In reality, the space step is adjusted in order to give a whole number of equal computational intervals between the two sections considered. The interpolation is performed at constant water depth.

In the figure below, the interpolated widths in each data section that enable to obtain all the widths of the computational section are shown in broken lines.

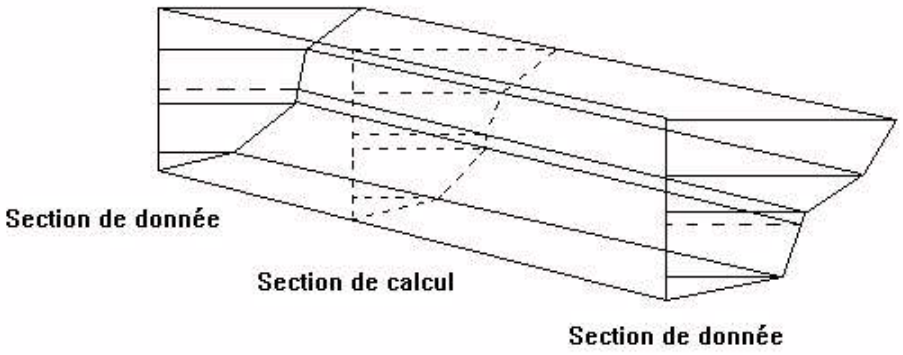


Figure 164.1: doc105

A computational section could thus have up to double the number of points of

the data sections. One must therefore adopt some criteria to eliminate some points and to avoid storing too many points.

Any point of the computational section which does not modify the section area by more than 1 % is eliminated. Similarly, all points which do not modify the wetted perimeter by more than 5 % are excluded. Therefore, whatever the water depth may be, we can expect a precision of 1 % regarding the estimation of the area and 5 % regarding the wetted perimeter.

The computational sections are completed vertically by a fictitious point located 500 m above the canal bank elevation in order to allow calculation even if overtopping occurs.

The computational sections are numbered within each reach, so that if the computational space step is modified in any given reach, the numbering of the sections within other reaches will remain unchanged.

Chapter 165

Interpolation of singular sections

In the case of a singular section it is necessary to have two computational sections at the same abscissa. If the model user entered two data sections at the same abscissa, both these sections are retained as computational sections. If only one section was entered, the downstream computational section is interpolated using the singular data section and the data section immediately downstream with a 1 m step. The interpolated section is then placed at the same abscissa as the singular section (see figure 14).

Figure 14

Section k is interpolated between sections i and j with a 1m step and is then placed at the section i abscissa. Therefore, one has to take care to enter two sections (i and j) at the same abscissa, especially if the bed elevation upstream and downstream of the device is different or if the section dimensions are different.

Figure 15

In figure 15, one will enter the two cross sections (i and j), as shown, but j will be given the same abscissa as i, so that there will not be any interpolation.

Chapter 166

Sort algorithm for the reaches of the loop before the steady flow computation - Initial solution

An initial discharge value is chosen in each reach of the loop, compatible with the discharge continuity at the nodes, using for example a discharge division proportional to the mean cross area of the reach (reach volume at the overtopping bank elevation divided by its length).

Then, the backwater curve is computed in each reach, in sequence and following the above described order (Cf. Classification of reaches). The water elevation at a distributory node (boundary condition for the upstream reach) is taken as the average computed elevation of the downstream reaches starting from this node.

For a given portion of the reach n°i, defined by two sections, the Zij elevation (in the upstream cross section n°j) is to be computed, knowing the Zij+1 elevation (in the downstream cross section n°j+1) and the Qij discharge flowing through the cross section n°j.

This is done by solving (numerically) the backwater equation that can be written as:

$f(Z_j^i, Z_{j+1}^i, Q_j^i) = 0$ *[1]*

Chapter 167

Correcting equations

The corrected backwater curve will be solution of the equation:

$f(Z_j^i + \Delta Z_j^i, Z_{j+1}^i + \Delta Z_{j+1}^i, Q_j^i + \Delta Q^i) = 0$ *[2]*

After a first order Taylor development of *[2]*, we get:

[2] - *[1]* => $a_j^i.\Delta Z_j^i + b_j^i.\Delta Z_{j+1}^i + c_j^i.\Delta Q^i = 0$ *[3]*

with:

$a_j^i = \frac{\partial f}{\partial Z_j^i} \sim \sim \sim b_j^i = \frac{\partial f}{\partial Z_{j+1}^i} \sim \sim \sim c_j^i = \frac{\partial f}{\partial Q^i}$

We know that aij cannot be nil, since *[1]* is solved through the Newton method using the ratio $\frac{f}{a_j^i}$.

Let:

$d_j^i = \frac{b_j^i}{a_j^i}$ et $e_j^i = \frac{c_j^i}{a_j^i}$

For a reach i with n cross sections, the (n-1) equations *[3]* give:

$\Delta Z_1^i = d_1^i.\Delta Z_2^i + e_1^i.\Delta Q^i$

...

$\Delta Z_j^i = d_1^i.\Delta Z_{j+1}^i + e_j^i.\Delta Q^i$

...

$\Delta Z_{n-1}^i = d_{n-1}^i.\Delta Z_n^i + e_{n-1}^i.\Delta Q^i$

After condensation of these equations, we can get one relation representative of the total reach:

$\Delta Z_1^i = D_n^i.\Delta Z_n^i + E^i.\Delta Q^i$ *[4]*

with: - i = index of the reach - 1 = index of the upstream section of the reach - n = index of the downstream section of the reach - $D^i = \prod_{j=1}^{n} d_j^i$ - $E^i = D^i \prod_{k=1}^{n} \frac{e_k^i}{\prod_{j=k}^{n} d_j^i}$ (with: $d_n^i = 1$ and $e_n^i = 0$)

The other available relations are:

167.1 For all the nodes of the loop that are not downstream nodes of the loop:

167.2 *Discharge continuity

$\sum_{i=1}^{k} \varepsilon^i \Delta Q^i = 0$ *[5]*

with: - k = number of reaches of the loop connected to the node - $\varepsilon^i = 1$ upstream reach - $\varepsilon^i = -1$ downstream reach

167.3 *Water elevations equality

$Z_u^i + \Delta Z_u^i = Z_v^l + \Delta Z_v^l$ *[6]*

giving k - 1 equations, with: - u, v = 1 if the reach leaves the node and n if the reach joins the node - i, l = reaches connected to the node

167.4 For all the downstream nodes of the loop:

$\Delta Z_n^i = 0$ *[7]*

if the downstream boundary condition is a given elevation, or

$\Delta Z_n^i = g(\Delta Q^i)$

if the downstream boundary condition is a rating curve (in this case only one reach can join the corresponding downstream node).

Chapter 168

Resolution of the linear system of the correcting equations

Three unknown variables are attached to each reach n°i:

- in terms of water elevation: ΔZ_1^i at the upstream node
- in terms of water elevation: ΔZ_n^i at the downstream node
- in terms of discharge: ΔQ^i anywhere in the reach

Let N be the number of reaches in the loop. It is possible to solve the linear system composed of the 3N equations. But an entire backwater curve computation will be necessary, after correction of the discharges, in order to get the water elevation in each cross section of the reaches.

Therefore, the directly interesting unknowns are theΔQ^i variables.

If the equations are written in a judicious order, some (N.N) blocs appear:

$$\begin{pmatrix} A & B & 0 \\ D & I & F \\ G & H & I \end{pmatrix} \cdot \begin{pmatrix} \Delta Q \\ \Delta Z_1 \\ \Delta Z_n \end{pmatrix} = \begin{pmatrix} C1 \\ C2 \\ C3 \end{pmatrix}$$

Matrix . Vector = Constant

That can be written: $M.V = K$ *[8]*

D, F are diagonal matrices I is the unity matrix H is an upper triangular matrix with a nil principal diagonal

The discharge continuity equations at the nodes which are not downstream nodes of a loop, are placed in the first block line of the linear system (filling the A

block). We then complete the block line with the equality relations between the water elevations at the distributory nodes (filling the B and C1 blocks).

The condensation equations are placed in the second block line.

The downstream boundary conditions at the downstream reaches of the loop, and the equality relations between the upstream and the downstream water elevations of the reaches at the nodes that are not downstream nodes of a loop, are placed in the third block line.

By multiplying *[8]* with a proper transition matrix P:

$(P.M).V = P.K$

we get a N equations system in theΔQ variable:

$A_0.\Delta Q = C_0$ *[9]*

with: $\begin{cases} A_0 = A - B.D + \chi_0.(H.D - G) \\ C_0 = C_1 - \chi_0.C_3/c \\ \chi_0 = B.F.\chi_1^{-1}/c \\ \chi_1 = H.F - I/c \\ I = \text{Unity matrix}/c \end{cases}$

This method can only be applied if theχ_1 matrix can be inverted. Fortunately this matrix can always be transformed into a triangular superior matrix with the diagonal only composed of 1. Thereforeχ_1 can easily be inverted through the Gauss - Jordan method. The system *[9]* is solved through the Gauss method with a partial Gauss elimination.

Chapter 169

Cross structure equations

When cross structures exist on the canal (singular section) the water surface profile equation cannot be used locally to calculate the water surface elevation upstream of the structure.

The hydraulic laws of the different devices present in the section must be applied.

The modeling of these devices is a delicate problem to solve when developing open channel mathematical models. The equations used to represent the hydraulic devices are numerous and do not cover all the possible operating conditions.

In particular, it is rather difficult to maintain the continuity between the different formulations, for example, at the instant of transition between free-flow conditions and submerged conditions, or between open-channel conditions and pipe-flow conditions.

A distinction has been made between devices with a high sill elevation (called hereafter Weir / Orifice) and devices with a low sill elevation (called hereafter Weir / Undershot gates).

Chapter 170

CEM88(D) : Weir / Orifice (high sill elevation)

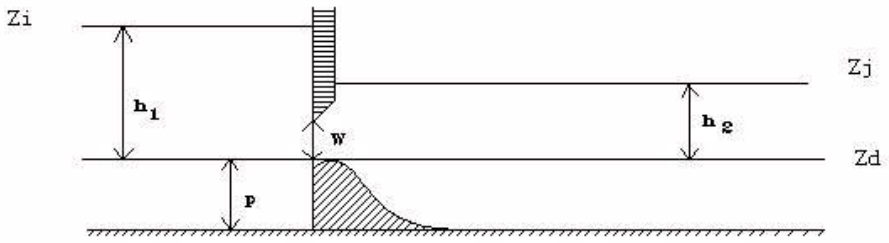


Figure 18. Device schematic view

170.1 Weir - Free flow ($h_1 < W$ and $h_2 \leq \frac{2}{3}h_1$)

$Q = \mu_F L\sqrt{2g}h_1^{3/2}$ *[12]*

Classical equation for the free flow weir ($\mu_F \simeq 0.4$).

170.2 Weir - Submerged ($h_1 < W$ and $h_2 \geq \frac{2}{3}h_1$)

$Q = \mu_S L\sqrt{2g}(h_1 - h_2)^{1/2}h_2$ *[13]*

Classical formulation for the submerged weir.

The free-flow/submerged transition takes place for $h_2 = \frac{2}{3}h_1$

Thus, $\mu_S = \frac{3\sqrt{3}}{2}\mu_F$ for$\mu_F = 0.4 \Rightarrow \mu_S = 1.04$

The equivalent free-flow coefficient can be calculated:

$\mu_F = \frac{Q}{L\sqrt{2g}h_1^{3/2}}$

It indicates the degree of submergence of the weir by comparing it to the introduced free-flow coefficientμ_F. In effect, the reference coefficient of the device considered is that corresponding to the free-flow weir (μ_F close to 0.4).

170.3 Orifice - Free flow ($h_1 \geq W$ and $h_2 \leq \frac{2}{3}h_1$)

An equation of the following type is applied:

$Q = \mu L\sqrt{2g}\left(h_1^{3/2} - (h_1 - W)^{3/2}\right)$ *[14]*

This formulation is applicable to large width rectangular orifices.

The continuity towards the open-channel flow is assured when:

$\frac{h1}{W} = 1$, one then has: $\mu = \mu_F$.

170.4 Orifice - Submerged

Two formulations exist, according to whether the flow is partially submerged or completely submerged.

170.5 Partially submerged flow ($h_1 \geq W$ and$\frac{2}{3}h_1 < h_2 < \frac{2}{3}h_1 + \frac{W}{3}$)

$Q = \mu_F L\sqrt{2g}\left[\frac{3\sqrt{3}}{2}\left((h_1 - h_2)^{1/2}\, h_2\right) - (h_1 - W)^{3/2}\right]$ *[15]*

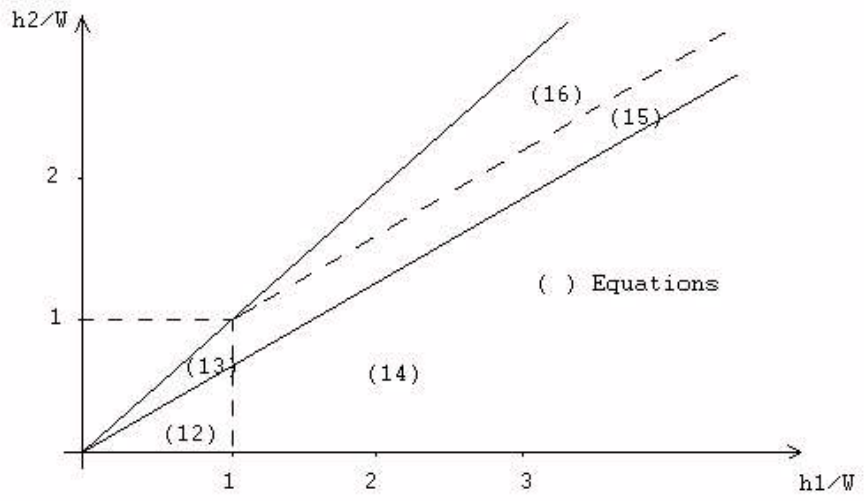
170.6 Totally submerged flow ($h_1 \geq W$ and$\frac{2}{3}h_1 + \frac{W}{3} < h_2$)

$Q = \mu` L\sqrt{2g}(h_1 - h_2)^{1/2}\,[h_2 - (h_2 - W)] \Rightarrow Q = \mu` L\sqrt{2g}(h_1 - h_2)^{1/2}W$ *[16]*

This is the classic equation of the submerged orifice with$\mu` = \mu_S$.

The operation of the weir/orifice device is represented by the above equations and in figure 19. Whatever the conditions of pipe flow, one calculates an equivalent free-flow coefficient, corresponding to the free-flow orifice:

$C_F = \frac{Q}{L\sqrt{2g}W(h_1 - 0.5W)^{1/2}}$



(12): Weir - Free flow (15): Orifice - Partially submerged (13): Weir - Submerged (16): Orifice - Totally submerged (14): Orifice - Free flow *Figure 19. Weir - Orifice*

Equations are also available in a Matlab script file (function Qouvrage) here.

Chapter 171

CEM88(V) : Weir / Undershot gate (small sill elevation)

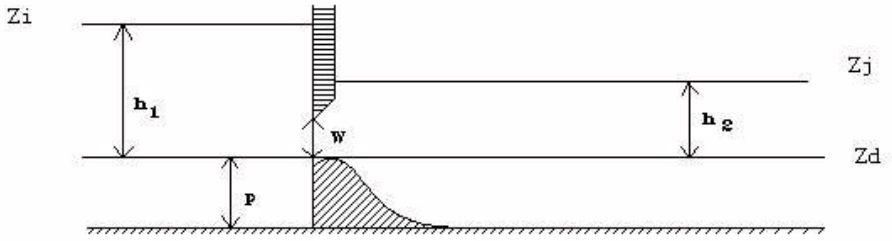


Figure 18. Device schematic view

171.1 Weir - Free-flow

$Q = \mu_f L \sqrt{2g} h_1^{3/2}$

171.2 Weir - Submerged

$Q = k_F \mu_F L \sqrt{2g} h_1^{3/2}$ *[17]*

with k_F coefficient of reduction for submerged flow.

The flow reduction coefficient is a function of $\frac{h_2}{h_1}$ and of the value α of this ratio at the instant of the free-flow/submerged transition. The submerged conditions

are obtained when $\frac{h_2}{h_1} > \alpha$. The law of variation of the k_F coefficient has been derived from experimental results ($\alpha = 0.75$).

Let $x = \sqrt{1 - \frac{h_2}{h_1}}$:

If $x > 0.2$: $k_F = 1 - \left(1 - \frac{x}{\sqrt{1-\alpha}}\right)^{\beta}$

If $x \leq 0.2$: $k_F = 5x\left(1 - \left(1 - \frac{0.2}{\sqrt{1-\alpha}}\right)^{\beta}\right)$

With$\beta = -2\alpha + 2.6$

One calculates an equivalent coefficient for free-flow conditions as before.

171.3 Undershot gate - Free-flow

$Q = L\sqrt{2g}\left(\mu h_1^{3/2} - \mu_1(h_1 - W)^{3/2}\right)$ *[18]*

It has been established experimentally that the undershot gate discharge coefficient increases with$\frac{h_1}{W}$. A law of variation ofμ of the following form is adopted:

$\mu = \mu_0 - \frac{0.08}{\frac{h_1}{W}}$ avec : $\mu_0 \simeq 0.4$

Hence, $\mu_1 = \mu_0 - \frac{0.08}{\frac{h_1}{W} - 1}$

In order to ensure the continuity with the open channel free-flow conditions for$\frac{h1}{W} = 1$, we must have: $\mu_F = \mu_0 - 0.08$

Hence, $\mu_F = 0.32$ for$\mu_0 = 0.4$

171.4 Undershot gate - Submerged

171.5 Partially submerged flow

$Q = L\sqrt{2g}\left[k_F \mu h_1^{3/2} - \mu_1 (h_1 - W)^{3/2}\right]$ *[19]*

k_F being the same as for open channel flow.

The following free-flow/submerged transition law has been derived on the basis of experimental results:

$\alpha = 1 - 0.14\frac{h_2}{W}$

$0.4 \leq \alpha \leq 0.75$

In order to ensure continuity with the open channel flow conditions, the free-flow/submerged transition under open channel conditions has to be realized for$\alpha = 0.75$ instead of 2/3 in the weir/orifice formulation.

171.6 Totally submerged flow

$Q = L\sqrt{2g}\left(k_F \mu h_1^{3/2} - k_{F1}\mu_1 (h_1 - W)^{3/2}\right)$ *[20]*

The k_{F1} equation is the same as the one for k_F where h_2 is replaced by $h_2 - W$ (and h_1 by $h_1 - W$) for the calculation of the x coefficient andα (and therefore for the calculation of k_{F1}).

The transition to totally submerged flow occurs for:

$h_2 > \alpha_1 h_1 + (1 - \alpha_1)W$

with: $\alpha_1 = 1 - 0.14\frac{h_2 - W}{W}$ $(\alpha_1 = \alpha(h_2 - W))$

The functioning of the weir / undershot gate device is represented by the above equations and displayed in figure 20. Whatever the conditions of the pipe flow, one calculates an equivalent free-flow discharge coefficient, corresponding to the classical equation for the free-flow undershot gate.

$C_F = \frac{Q}{L\sqrt{2g}W\sqrt{h_1}}$

The reference coefficient introduced for the device is the classic C_G coefficient of the free-flow undershot gate, usually close to 0.6. It is then transformed to$\mu_0 = \frac{2}{3}C_G$ which allows to computeμ andμ_1 from equation *[18]* for the free-flow undershot gate.

Remark: it is possible to get $C_F \neq C_G$, even under free flow conditions, since the discharge coefficient increases with the$\frac{h_1}{W}$ ratio.

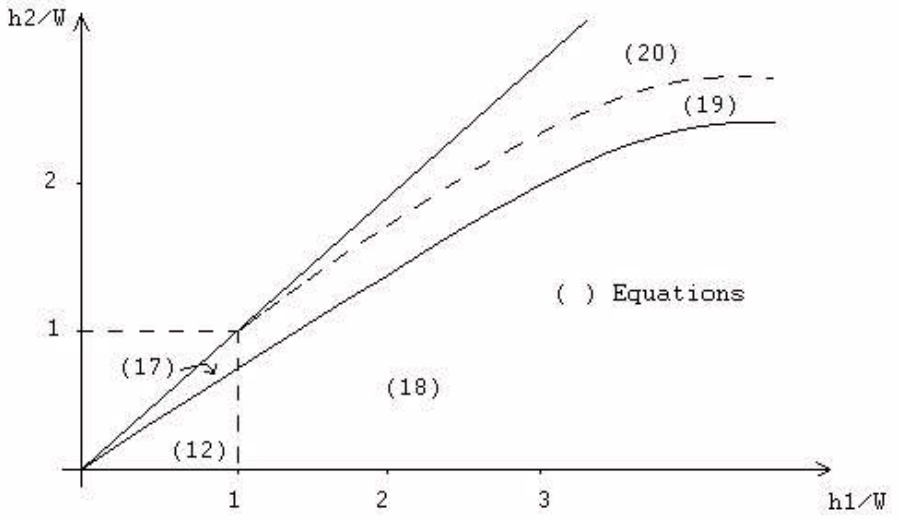


Figure 171.1: doc79

(12): Weir - Free flow (19): Undershot gate - Partially submerged (17): Weir - Submerged (20): Undershot gate - Totally submerged (18): Undershot gate - Free flow *Figure 20. Weir - Undershot gate*

Equations are also available in a Matlab script file (function Qouvrage) here.

Chapter 172

CEM02(V) et CEM02(D) : Trapezoidal sills and gates

These 2 equations are the sum of one of "Cem88(V)" and "CEM88(D)" equations with a triangular sill.

Kindsvater-Shen formula[[Norme NF X 10-311 : Mesure de débit de l'eau dans les canaux découverts au moyen de déversoirs en mince paroi]] is used for the triangular part :

$Q = C_e \frac{8}{15} \tan\frac{\alpha}{2} \sqrt{2g} h^{5/2}$

The discharge coefficient C used in SIC is $C = C_e \frac{8}{15}\sqrt{2g}$. $C =\approx 1,37$ for $C_e = 0,58$ as suggested by the norm[[<1>]].

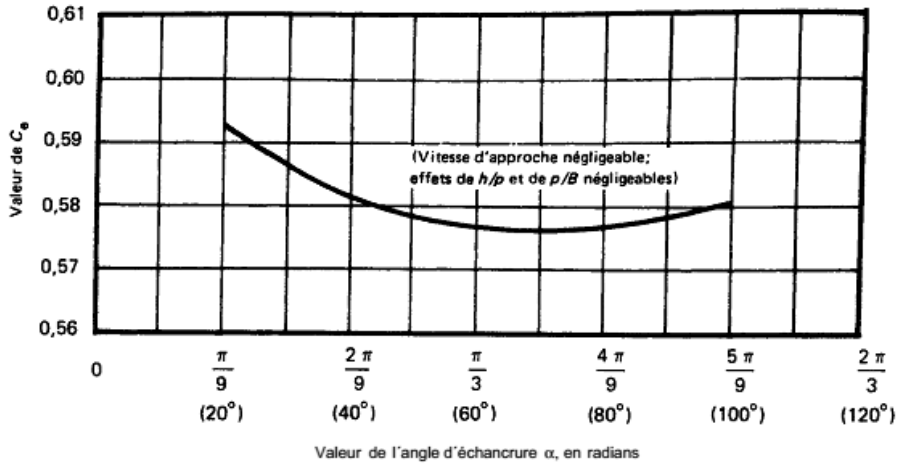


Figure 172.1: doc80

Chapter 173

Equation at a singular section

The water surface elevation at a singular section is computed using the previous equations. The flow at the section is equal to the sum of the discharges through each device (e.g., gate, weir).

fk(Zi, Zj) = Q *[23]*

n is the number of devices in the section and Q the flow at the section.

fk(Zi, Zj) is the discharge law of the device number k, for instance for a submerged weir:

fk(Zi, Zj) = µL(Zi-Zj)1/2(Zj-Zd)

If the discharge and the downstream elevation Zj are known, the water surface elevation Zi upstream of the device can then be calculated.

This means that one has to solve an equation of the form f(Zi)=0 (See § II.5.)

Chapter 174

Regulator

At each singular section, one particular gate can be chosen to play the role of a regulator (this means that the opening of this gate is not fixed a-priori. instead, the model will compute the opening required to maintain a target water level immediately upstream. the openings of all other gates are imposed a-priori). The opening of this gate is unknown. The maximum possible opening and the target water elevation (e.g. Full Supply Depth) upstream of the gate are known. This results in an equation at the singular section similar to the previous one, but in this case, the unknown is no longer the upstream water surface elevation, but the opening of the gate working as a regulator. One ends up with an equation of the following type:

Q - fk(Zi, Zj) = fr(Zi, Zj, W) *[24]*

with:

k = 1 to n : for gates with fixed openings.

W : the regulator opening to be calculated.

Zi : known value (target upstream water elevation).

fk(Zi, Zj) : the discharge going through the fixed gate number k for the target upstream water elevation Zi and the downstream water elevation Zj. The equations considered are those described for the weirs and the gates.

fr(Zi, Zj, W) : the discharge going through the regulator type gate for an opening W and the target upstream water elevation Zi.

The fk(Zi, Zj) are known values. Then equation *[24]* is reduced to fr(Zi, Zj, W) = constant. One then has to look for the zero of a function, but this time, the unknown is W.

Chapter 175

Offtake equations

The lateral offtakes correspond to points of outflows. Therefore, they are obligatorily located at the upstream nodes of the reaches. Under steady flow conditions, one cannot compute the real offtake discharge corresponding to a given offtake gate opening, unless the corresponding secondary canal is described as part of the model (looped model). But, knowing the offtake target discharge, the program is able to calculate the corresponding offtake gate opening.

The offtakes are modeled according to the same hydraulic laws as for cross structures. The originality of the approach resides in the consideration of a possible influence of the offtake downstream condition.

For the discharge, we use the equations described above for the undershot gates. If the offtake is circular, one has to calculate the width of the equivalent rectangular opening in order to be able to use the equations referred to above.

One has to then solve an equation of the following type:

fp(Z1, Z2, W) = Qp with:

Qp: target offtake discharge

Z1: upstream water surface elevation in the main canal obtained via the water surface profile computation

Z2: downstream water surface elevation. This is either known, or its value depends on the offtake discharge Qp and the chosen offtake downstream condition. If Z2 is a function of Qp, one then has an equation of the form:

fp(Z1, fs-1(Qp), W) = Qp *[27]*

with fs being the rating curve corresponding to the chosen offtake downstream condition.

Therefore, in all cases the problem is to find the zero of a function with W as the unknown (see paragraph II.5).

Chapter 176

Offtake downstream conditions

In order to include the possibility of submerged flow conditions at the offtakes, three types of offtake downstream conditions (i.e., at the head of the secondary canal) can be modeled:

- a constant downstream water surface elevation
- a downstream water surface elevation Z2 that varies with the water surface elevation upstream of a free-flow weir:

Q(Z2) = µL (Z2-ZD)3/2 *[25]*

- a downstream water surface elevation that follows a rating curve of the type:

Q(Z2) = Qo( )n *[26]*

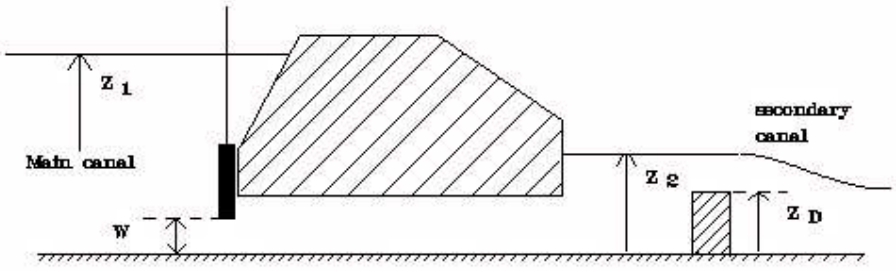


Figure 21

Chapter 177

Numerical methods

In all the cases analyzed, we obtained an equation of the form f(Z)=0 or f(W)=0 to be solved.

In this section we examine in detail how those equations are solved numerically.

Chapter 178

Recapitulation of the Newton's method

Chapter 179

Resolution of the water surface profile equation

The water surface profile equation *[11]* is solved by Newton's method. The initial value is taken equal to:

Zio =

In our case, the function f(Z) does not have a continuous first derivative, because the geometry of the computational section is only known at certain points. In order to avoid any adverse effects on the computation convergence, it is limited by a bisection algorithm.

Starting bisection algorithm values are:

Zmin = Zci Zmax = Hj + DH(Zci)

In a Newton iteration l two cases can occur:

1) The calculated value Zil is in the interval (Zmin, Zmax).

- This value is retained
- if f(Zil) $>$ 0 then Zmin = Zil
- if f(Zil) $<$ 0 then Zmax = Zil

since f is a decreasing function.

2) The calculated value Zil is not in the interval (Zmin, Zmax). Replace Newton's iteration by a bisection algorithm:

Zil =

Similarly, if one detects a breakdown of the Newton algorithm due to the non continuity of the derivative of f(Zi), one also adopts the bisection algorithm.

Chapter 180

Computation of the critical depth

We previously observed that it was necessary to calculate the critical depth in each computational section in order to determine if a sub-critical solution exists. The critical depth in a section is defined by:

= 1

This equation can be transformed as follows:

f(Zci) = Log ( ) = 0 which is a decreasing function.

which can be solved by Newton's method.

The initial value is determined by taking the critical water depth in the rectangle equivalent to the section corresponding to the overtopping water elevation (ZDi).

Zcio = ( )1/3 + ZFi

with: X =

Starting values for the bisection algorithm are:

Zmin = ZFi Zmax = ZDM + 10

with: ZDM = elevation of the last point of the section.

These values are modified in the following way:

If f(Zcil) > 0 : Zmin = Zcil

If f(Zcil) < 0 : Zmax = Zcil

since f is a decreasing function.

Chapter 181

Computation at a singular section

At a singular section, one can have as many weirs (five maximum) or gates (five maximum) as needed.

One must then solve equation*[23]* where the fk(Zi, Zj) terms are the discharges going through each device number k. Zi is the water surface elevation upstream of the device and Zj is the water surface elevation downstream of it. This equation is solved by Newton's method.

The initial value is determined as follows: Consider only one weir with length XL equal to the sum of the lengths of all the devices, and located at the elevation Zmin of the lowest one. If the downstream water elevation Zj is less than Zmin, one initializes with the free-flow weir formula:

Zio = Zmin + ( )2/3

If Zj is greater than Zmin, one considers:

Zio = Zj +

Starting values for the bisection algorithm are:

Zmin = elevation of the lowest device, or Zj if the latter value is greater.

Zmax = elevation of the highest device + 10 meters.

These values are modified as follows:

if f(Zil) $>$ 0 : Zmax = Zil if f(Zil) $<$ 0 : Zmin = Zil

where f = fk(Zi, Zj) - Q, which is a increasing function.

Let us examine the case of a regulator gate in a singular section. One has to solve the equation *[24]* which is of the type f(W) = 0, W being the regulator gate

opening allowing to achieve the target upstream water elevation Zi. One starts calculating the discharges going through all the fixed gates, and one obtains:

fr(Zi, Zj, W) = Q - fk(Zi, Zj)

This equation is solved by bisection algorithm by adopting as the initial opening, half of the maximum authorized opening.

The calculation is stopped when the distance between the limits of the bisection algorithm is less than 1 mm.

Chapter 182

Computation of the offtake opening

Equation *[27]* which is of the type:

fp(Z1, Z2, W) - Qp = 0

has to be solved. Z2 is calculated in the case when the offtake downstream condition varies:

Z2 = ZD + ( )2/3

or Z2 = ZD + (Zo-ZD) ( )1/n

One looks for the solution by bisection algorithm.

The initial value of W is taken equal to half of the authorized maximum opening. Calculation is stopped when the difference between the offtake discharge and the targeted discharge is less than 5%.

Chapter 183

Unit 2 - Steady flow

**

Chapter 184

Introduction

Unit 2 computes the water surface profile in a canal under steady flow conditions. This water surface profile can be used as the initial condition for the unsteady flow computation in Unit 3. Steady flow calculations also allow to test the influence of modifications to structures or canal maintenance, etc.

In addition, a sub-module of the steady flow module computes the offtake gate openings to satisfy given target discharges. Another sub-module computes the cross-regulator gate openings to obtain a given target water surface elevation upstream of the regulator (e.g., Full Supply Depth). These sub-modules therefore allow computation of gate settings to satisfy a given demand-supply configuration of water flows.

In actual canal operations, it should be emphasized that the steady flow regime represents the objective to be attained, and that it is the unsteady flow model which will indicate how best to reach it.

Chapter 185

Equation of gradually varied flow in a reach

The canal being divided into homogeneous zones (the reaches), the problem gets reduced to calculating the water surface profile under subcritical steady flow conditions in a reach.

The classic hypotheses of unidimensional hydraulics in canals are considered to apply:

- The flow direction is sufficiently rectilinear, so that the free surface could be considered to be horizontal in a cross section.
- The transversal velocities are negligible and the pressure distribution is hydrostatic.
- The friction forces are taken into account through the MANNING-STRICKLER coefficients.

Therefore, only monodimensional steady flow is studied, and only subcritical flow is considered.

Chapter 186

Differential equation of the water surface profile

The equation of the water surface profile in a reach can be written as follows:

= - Sf + (k-1) *[10]*

with: Sf = g = 9.81 m s-2 n : Manning coefficient R : hydraulic radius (m) A : cross section area (m2) H : total head (m) q : lateral inflow (q > 0, k = 0) or outflow (q < 0, k = 1), (m2 s-1) Sf : linear head losses (m2/3 s-2) Q : discharge (m3 s-1)

For solving this equation, an upstream boundary condition in terms of discharge and a downstream boundary condition in terms of water surface elevation are required.

In addition, the lateral inflow, and the hydraulic roughness coefficient along the canal should be known. As the equation does not have an analytical solution, in the general case, it is discretized in order to obtain a numerical solution. Knowing the upstream discharge and the downstream water elevation, the water surface profile is integrated, step by step starting from the downstream end.

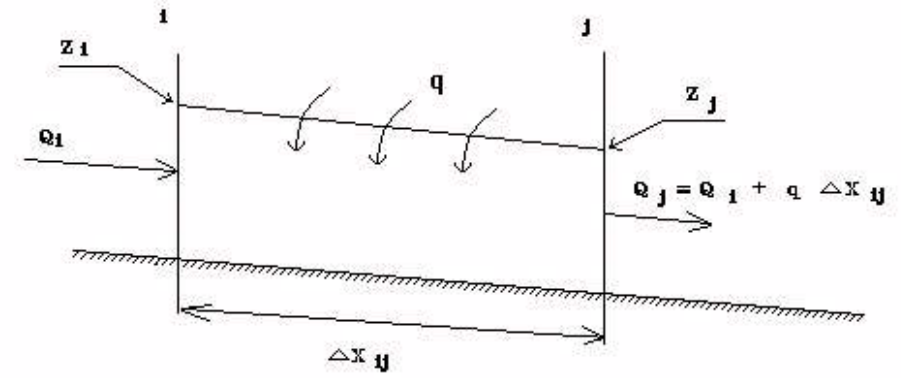


Figure 16

Integrating equation *[10]* between sections i and j:

- $- = 0$

i.e.:

Hj - Hi - (k-1) q (+) + Dxij = 0 *[11]*

Equation *[11]* can be written as follows:

Hi(Zi)= Hj + DH(Zi)

A subcritical solution exists if the curves Hi(Zi) and Hj + DH(Zi) intersect.

For this, it is necessary that: d = Hj + DH(ZCi) - Hi(ZCi) > 0

ZCi is the critical elevation defined at i by = 1

d > 0 : Subcritical solution F, d < 0 : Supercritical solution. The critical depth is assumed systematically. The water surface profile is therefore overestimated.

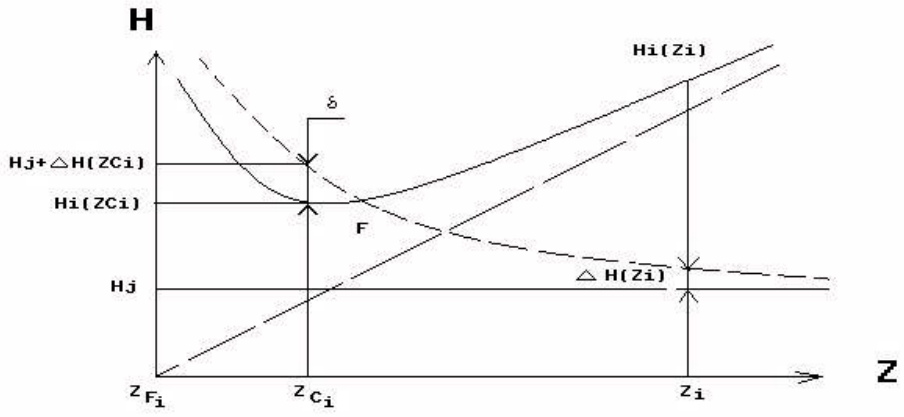


Figure 17

If a solution does exist, one has to numerically solve an equation of the form f(Zi)=0. The numerical methods used are presented in section II.5.

Chapter 187

Saint-Venant's equations

The canal is divided into homogenous zones, the reaches. We will first examine how to compute the unsteady flow water surface profile in a single reach. We will thereafter examine how to solve the problem for the whole hydraulic network.

The same hypotheses as for Unit 2 are applicable. Furthermore, only smooth transient phenomena are considered. The propagation of a surge cannot be simulated.

Chapter 188

Continuity equations

This equation accounts for the conservation of mass of the water. Consider the variation of the water volume contained between two sections at the abscissa x and x+Dx (see figure 23) during time Dt.

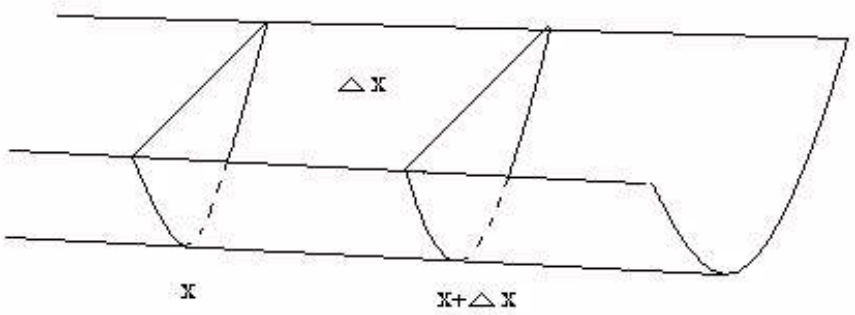


Figure 23

Inflow water mass:

r.Q(x,t).Dt + r.q.Dx.Dt

Outflow water mass:

r.Q(x+Dx,t).Dt

Change in storage:

r Vt+Dt - r Vt = r(A.Dx)t+Dt - r(A.Dx)t

The equation expressing the conservation of mass is written:

r(A.Dx)t+Dt - r(A.Dx)t = r(Q.Dt)x + rq.Dx.Dt - r(Q.Dt)x+Dx

At the limiting conditions, one obtains the equation *[28]*:

- = q *[28]*

Chapter 189

Dynamic equation

We apply the momentum principle to a volume of water contained between the two sections located at the abscissa x and x+Dx (figure 24).

Consider the projection of the momentum equation on the canal axis:

= S Fext/x *[29]*

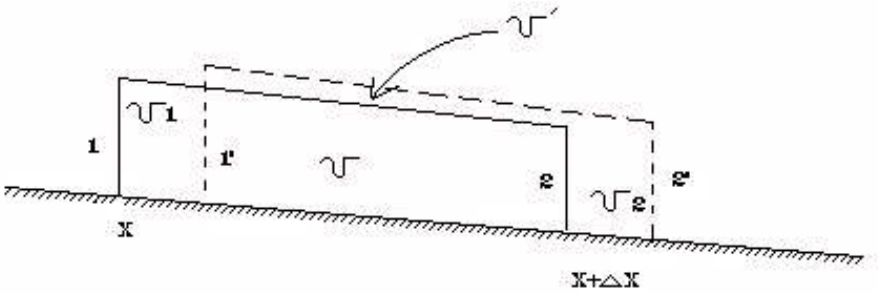


Figure 24

During the time Dt, the water volume V, contained between sections 1 et 2 gets deformed and moves into the volume V' contained between sections 1' et 2'. We have to estimate the variation of momentum projected on the x axis, corresponding to the left hand side of the equation *[29]*.

Estimation of the momentum:

The momentum lost corresponds to the volume V1

i.e. rV(x,t).Dt.A(x,t).V(x,t)

The momentum gained corresponds to the volume V2

i.e. rV(x+Dx,t).Dt.A(x+Dx,t).V(x+Dx,t)

The variation in momentum in the common portion is:

rA(x,t+Dt).Dx.V(x,t+Dt) - rA(x,t).Dx.V(x,t)

The variation of momentum due to lateral inflows or outflows is:

- rq(x,t).Dx.Dt.V(x,t)

in the case of an outflow (velocity is equal to the flow velocity), and

- rq(x,t).Dx.Dt.0

in the case of an inflow (velocity is equal to zero projected on the x axis).

This can be written globally as:

- krq(x,t).Dx.Dt.V(x,t)

with:

k = 1 for an outflow k = 0 for an inflow

Therefore:

D(mVx) = (rV.Dt.A.V) (x+Dx,t) - (rV.Dt.A.V)(x,t) + (rA.Dx.V) (x,t+Dt)

- (rA.Dx.V)(x,t) - k(rq.Dx.Dt.V) (x,t)

This can be written as follows:

= rDx [+ - kqV]

It is now necessary to examine the right hand side term of equation *[29]* which represents the resultant of external forces projected on the x axis. Only the effects of gravity, pressure and friction are considered.

Estimation of external forces:

The resultant of gravity forces is:

Fgx = r.g.A.Dx.sin(S0)

As the flow conditions are quasi-horizontal, sin(S0) » S0 (canal slope).

Therefore, Fgx = rg.A.Dx.S0

The resultant of the pressure forces can be obtained making use of the hydrostatic pressure distribution hypothesis. The resultant of the pressure forces applied on the water mass contained between x and x+Dx (figure 25) is the same as in the static case; that is when the free surface is horizontal. We can thus write:

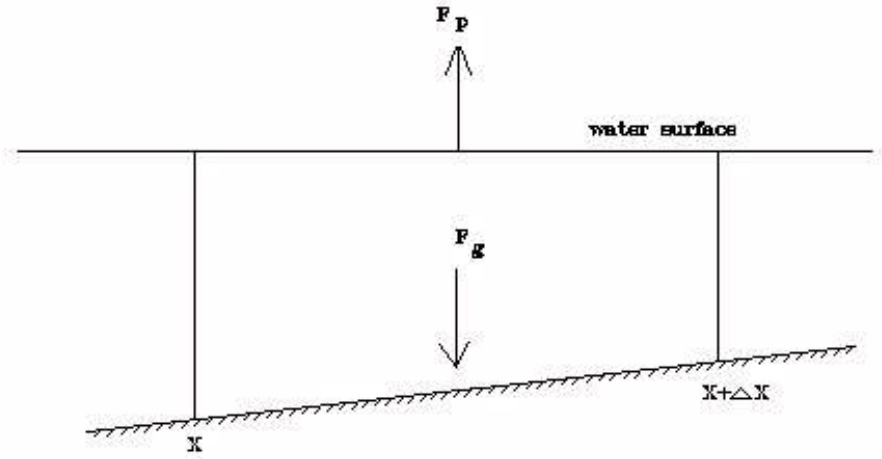


Figure 25

Fg = - rg.A.Dx = -Fp (resultant of the forces is nil)

Then: |Fp| = rg.A.Dx and Fp is perpendicular to the free surface.

The normal to the free surface has the following components:

n

since the free surface equation in the system of axes linked to the canal bed is:

h - y (x,t) = 0

One supposes that |n| = 1 because the free surface is almost horizontal. The resultant of the pressure forces is then written as:

Fp

Therefore: Fpx = -rg.A.Dx

Friction forces are estimated from the MANNING-STRICKLER formula:

Ffx = - rg.A.Dx.Sf

with: Sf =

The resultant of the external forces projected on the canal axis is then:

S Fext/x = rg.A.Dx.S0 - rg.A.Dx. - rg.A.Dx.Sf

= rg.A.Dx [S0 - Sf -]

Equation *[29]* may be written in the following form:

rDx [+ - kqV] = rg.A.Dx [S0 - Sf -]

Therefore:

- $-\ g.A = g.A\ [S0 - Sf] + kqV$

This can be written in the form:

- $-\ g.A = -\ g.A\ Sf + kqV$ *[30]*

Chapter 190

Initial and boundary conditions

Partial differential equations must be completed by initial and boundary conditions in order to be solved. The boundary conditions are the hydrographs at the upstream nodes of the reaches and a rating curve at the downstream node of the model (because subcritical flow conditions prevail). The initial condition is the water surface profile resulting from the steady flow computation (Unit 2).

Chapter 191

Semi-implicit discretization

Saint Venant's equations have no known analytical solution in real geometry. They are solved numerically by discretizing the equations: the partial derivatives are replaced by finite differences. Various solution schemes may be used to provide a solution to these equations. The discretization scheme chosen in the SIC model is a four-point semi-implicit scheme known as Preissmann's scheme (Figure 26).

This scheme is implicit because the values of the variables at the unknown time step also appear in the expression containing spatial partial derivatives.

Chapter 192

General equations of the scheme

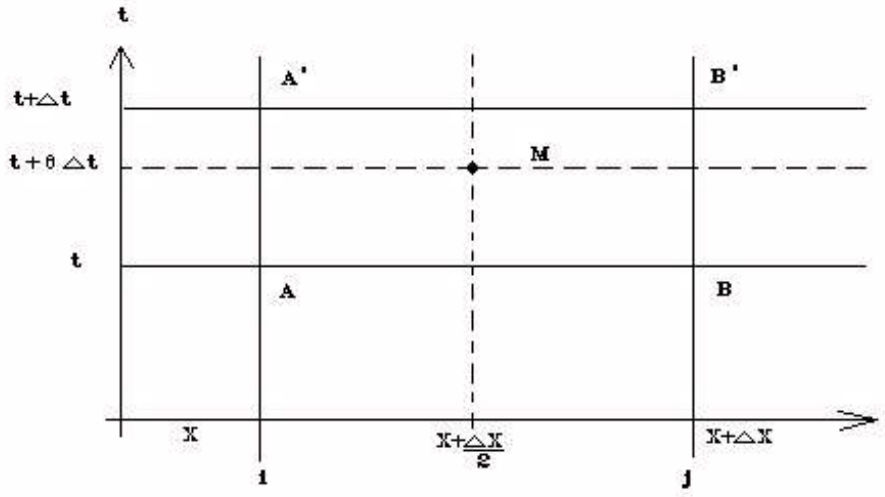


Figure 26

Let: Dfi = fA' - fA fA = fi

Dfj = fB' - fB fB = fj

The expression of a function in M may be written as:

fM = (1-Q) + Q

fM = + (Dfi + Dfj) *[31]*

The derivative () at point M is written as:

()M = (1-Q) + Q

=> ()M = + Q *[32]*

The derivative () at point M is written as:

()M = (+)

=> ()M = *[33]*

Equations *[31]*, *[32]* and *[33]* are used to discretize St. Venant's equations.

Chapter 193

Continuity equations

Discretize each term of equation *[28]*:

= DAi » Bi.DZi

=> =

= + Q

q = = qij

Equation *[28]*, then becomes:

- − Q =
- − DQj - DQi = (qi + qj)

DQi - Bi.DZi = DQj + Bj.DZj + - (qi + qj)

which is of the form:

A21.DQi + A22.DZi = B21.DQj + B22.DZj + B23 *[34]*

A21 = 1 B21 = 1

A22 = - Bi B22 = Bj

B23 = - (qi + qj)

Chapter 194

Dynamic equation

Discretize each term of the equation *[30]*:

- =

calculation of :

= [-] + [D( ) - D( )]

= [-] + [(- DAj) - (- DAi)]

= [-] + DQj - DQi + BiDZi - BjDZj)

- = a1.DQi + a2.DQj + a3.DZi + a4.DZj + a5

with: a5 = [-]

a4 = - Bj

a3 = Bi

a2 =

a1 = -

Calculation of gA :

gA = g [+ Q].[]

= [(Ai + Aj)(Zj - Zi) + Q(DAi + DAj)(Zj - Zi) + Q(DZj - DZi)(Ai + Aj)]

= [(Ai + Aj)(Zj-Zi) + Q(BiDZi + BjDZj)(Zj-Zi) + Q(DZj - DZi)(Ai + Aj)]

*** gA = [a6DZi + a7DZj + a8]

with: a6 = Q[Bi (Zj - Zi) - (Ai + Aj)]

a7 = Q[Bj (Zj - Zi) + (Ai + Aj)]

a8 = (Ai + Aj).(Zj - Zi)

calculation of gASf:

gASf = g [+ [D(AiSfi) + D(AjSfj)]]

= g [+ [SfiDAi + AiDSfi + SfjDAj + AjDSfj]]

Calculate DSf:

Sf = = sgn(Q)

with: sgn(Q) = 1 if Q > 0 sgn(Q) = -1 if Q < 0

=> DSf = 2 sgn(Q).Q - 2 sgn (Q) .DDe

or DSf = 2 |Q| - 2 .DDe

De =

=> DDe = + R-1/3 DR

with:

=> DDe = B.DZ + R-1/3 a DZ

= ae DZ

DSf = 2 |Q| - 2 DDe

= 2 |Q| - 2 ae DZ

=> gASf = g [+ [(SfiBiDZi + SfjBjDZj)

- Ai (2 |Qi| - 2 aei DZi)
- Aj (2 |Qj| - 2 aej DZj)]]

*** gASf = a9.DZi + a10.DZj + a11.DQi + a12.DQj + a13

with:

a9 = [SfiBi - Ai 2 aei]

a10 = [SfjBj - Aj 2 aej]

a11 = gQ

a12 = gQ

a13 = g []

Calculation of kq :

kq = [qi + qj + Q(D(qi) + D(qj))]

= [qi + qj + Q(qi - qi + qj - qj)]

*** kq = a14.DZi + a15.DZj + a16.DQi + a17.DQj + a18

with: a14 = - Q qi Bi

a15 = - Q qj Bj

a16 = Q

a17 = Q

a18 = [qi + qj]

The dynamic equation *[30]* can then be written in the form:

A11.DQi + A12.DZi = B11.DQj + B12.DZj + B13 *[35]*

with: A11 = + a1 + a11 - a16

A12 = a3 + a6 + a9 - a14

B11 = - (+ a2 + a12 - a17)

B12 = - (a4 + a7 + a10 - a15)

B13 = - (a5 + a8 + a13 - a18)

Chapter 195

Performance indicators

Some performance indicators have been inserted with the aim of evaluating the efficiency of the discharge delivered at offtakes. They permit an integration of discharge information at offtakes, either for an unique offtake, or for all offtakes. There are two indicator types: volume indicators and time indicators.

The definition of these indicators is detailed in the theoretical guide. Volume indicators refer to 3 types of volumes:

- asked volume (VD), which is the targeted volume at offtakes,

- provided volume (VS) which is the real volume supplied at offtakes, - effective volume (VEF) which is the real usable part of the provided volume.

The following colours are used to display the indicators (in RESIR): - green, if the value is between 1 and 1.,5 - light green, if the value is superior to 1.5

- red, if the value is between 0.5 and 1 - light red, if the value is lower than 0.5

*Remark:***The validity of these indicators is poor when the storage of data on the result file is not done at each time step, because some discharge values will be missing (Cf. “Time parameters”).

**

Chapter 196

Volume indicators

The volume indicators refer to three kind of volumes:

- The demand volume (VD), which is the target volume at the offtakes,
- The supply volume (VS), which is the volume supplied at the offtakes,
- The effective volume (VEF), which is the really usable part of the supply volume.

The definition of the effective volume depends on two coefficients: W and X (in %):

- if (1-W/100).QD <= QS <= (1+X/100).QD => QEF = QS
- if QS < (1-W/100).QD => QEF = 0
- if QS > (1+X/100).QD => QEF = (1+X/100).QD

All volumes are calculated with trapezoidlal rule:

$$V = \frac{t_d - t_f}{\Delta t}\left(\frac{f(t_d) + f(t_f)}{2} + \sum_{k=1}^{n-1} f\left(t_d + k\frac{t_d - t_f}{n}\right)\right)$$

Only the supply discharge close to the water demand is thus taken into account (see below).

<emb99|center>

In this figure the effective volume is shaded. We define three volume indicators:

- Indicator IND1 = VS/VD
- Indicator IND2 = VEF/VD
- Indicator IND3 = VEF/VS

These indicators can be defined for a single offtake or for a set of offtakes.

Chapter 197

Time indicators

We define TD as the total period of time during which the demand discharge is non-zero and TEF as the total period of time during which the effective discharge is non-zero. We introduce the indicator IND4 = TEF/TD. It compares the duration of delivery of the effective volume with that of the demand volume. This indicator is dimensionless and can only be calculated for individual offtakes since it doesn't have any significance for all the offtakes taken together.

We also define two time lags: DT1 and DT2. DT1 is the time separating the start of the water demand and the start of the effective discharge. This time is positive if the effective discharge arrives after the demand discharge (Cf. figure 30). DT2 is the time lag between the centers of gravity of the demand hydrograph and the effective delivery hydrograph.

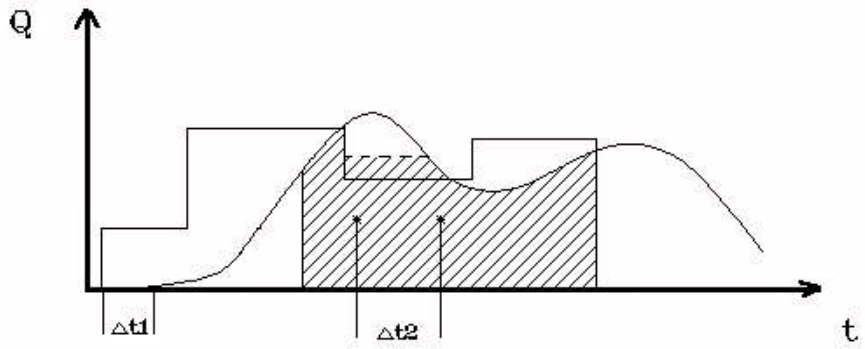


Figure 30

All these indicators are defined for each offtake. They can be calculated for any particular period of the simulation that the user wants to focus on.

Chapter 198

Introduction

Unit 3 computes the water surface profile in the canal under unsteady flow conditions. The initial water surface profile is provided by Unit 2 (steady flow module). It is also possible to start from a previous unsteady flow calculation (hot start). Unit 3 allows, for example, to study the transition from one rotational schedule to another. In addition, it calculates the offtake discharges knowing the offtake openings.

If you want to automatically compute a regulator gate opening, knowing the target upstream water level for example, it is necessary to incorporate special regulation modules. Using regulation modules you can address this type of problem, or similar problems involving water surface elevation or discharge targets.

Moreover, this unit can process pond structures and major beds (local storage).

Chapter 199

Double sweep method - General case

Consider a reach having n computational cross sections. The system to be solved is then:

- St. Venant's equations at every interval located between two computational cross sections, at every time instant t,
- upstream and downstream boundary conditions.

Discretization transforms the reach into a series of n computational cross sections connected to each other by the two linear equations *[34]* and *[35]*. One then has 2(n-1) linear equations in DQ and DZ. The two missing equations for the system resolution are provided by the upstream and downstream boundary conditions, that are linearized at each time step.

Upstream boundary condition:

R1.DQ1 + S1.DZ1 = T1

In the case of a Q(t) relation, one has:

R1 = 1, S1 = 0 and T1 = Q(t+Dt) - Q(t)

Downstream boundary condition:

R'n.DQn + S'n.DZn = T'n, with:

= a, R'n = 1, S'n = -a, T'n = 0

A linear system with 2.n equations has to be solved. Instead of inverting the system matrix, the double sweep method is employed.

A11.DQi + A12.DZi = B11.DQj + B12.DZj + B13

A21.DQi + A22.DZi = B21.DQj + B22.DZj + B23

Let us write:

DEN = A11.A22 - A21.A12

DEN is different from zero, because the two equations are linearly independent.

DEN.DQi = (B11.A22 - B21.A12).DQj + (B12.A22 - B22.A12).DZj + B13.A22 - B23.A12

DEN.DZi = (B21.A11 - B11.A21).DQj + (B22.A11 - B12.A21).DZj + B23.A11 - B13.A21

[36]

with:

A =

B =

C =

D =

E =

F =

Let us examine the linear system for three computational cross sections:

R1S1DQ1T11-D1-E1DZ1F11-A1-B1 .DQ2 =C11-D2-E2DZ2F21-A2-B2DQ3C2R'3S'3DZ3T'3

First sweep:

The first upstream-downstream sweep gives:

R1S1DQ1T11-D1-E1DZ1F1R2S2 .DQ2 =T21-D2-E2DZ2F2R3S3DQ3T3R'3S'3DZ3T'3

For two consecutive sections i and j:

Ri.DQi + Si.DZi = Ti

Ri.(Ai.DQj + Bi.DZj + Ci)

- Si.(Di.DQj + Ei.DZj + Fi) = Ti

R*j.DQj + S*j.DZj = T*j

with: R*j = Ri.Ai + Si.Di S*j = Ri.Bi + Si.Ei T*j = Ti - Ri.Ci - Si.Fi

The coefficients R*j, S*j and T*j are set by normalizing, in order to avoid the propagation of numerical error.

Rj = etc.

One then has:

Rj.DQj + Sj.DZj = Tj

which is the upstream impedance relation.

At the downstream boundary condition:

which gives us DQn and DZn:

DQn =

DZn =

Then one goes up the second sweep.

Second sweep:

This sweep allows to calculate the DQ and DZ in each computational cross section using the equations *[37]*.

[37]

Chapter 200

Singularities

Let us examine how to introduce singularities in the double sweep process.

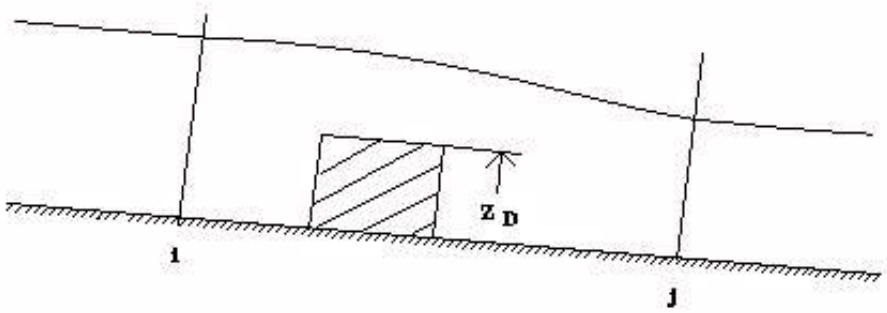


Figure 27

The problem to be solved in the case of a singularity is the following:

[38]

We need to transmit the impedance relation:

Rj.DQj + Sj.DZj = Tj

to the downstream cross section of the singularity.

We shall assume that the device is moveable, and that variation law W(t) is known a-priori.

The device equation can be written at the instant t + (n + 1)dt:

Qin+1 = f(Zin+1, Zjn+1, Wn+1)

Then: DQi = f(Zin + DZi, Zjn + DZj, Wn+1) - Qin

An expression of the non linear impedance relation is obtained in the following form:

DQj = f(Zin + (- DQj), Zjn+DZj, Wn+1) - Qin *[39]*

The problem is then to find the best possible linear approximation to this expression. The tangential approximation equation of the device variation law can be written as:

Qin + DQi = f(Zin + DZi, Zjn + DZj, Wn+1)

= f(Zin, Zjn, Wn+1) + (Zin, Zjn, Wn+1) DZi

- (Zin, Zjn, Wn+1) DZj

which leads to: Rj.DQj + Sj.DZj = Tj *[40]*

with:

This method cannot avoid the tangential approximation error of the device variation law created at each time step n, but counterbalances it in the next time step n+1. Indeed, a "correction wave" is included in the expression of the Tj coefficient in the form of an additive term Si (f() - Qin).

Nevertheless, errors due to the tangential approximation can be significant in the case of rapid variations of flow conditions (device operations, free-flow to submerged flow transition, etc.). It is then necessary to have as precise an estimation as possible of the evolution of the two variable Zi and Zj during time step Dt. So, variables Zi, Zj, Qi, and Qj are calculated locally, and Z* and Q* estimated by linear approximation by the following procedure. At each singular section, three equations *[38]* are available, with four unknowns.

So, a fourth equation is needed in order to calculate Zi, Zj, Qi, and Qj locally. This equation is an assumption on Zj. This assumption does not really attempt to get close to the missing R'S'T' equation but rather to its effects on the evolution of the Zj value.

Let's assume that:

DZj = k.DQj *[41]*

k being determined during the previous time step.

The following procedure for the computation of Rj Sj Tj is used:

*1) Hypothesis on Zj ([41]) + [40] = expected**D**Zj* value*

=> expected DQj* and DZj* values

*2) Computation of two expected values**D**Qi* and**D**Zi**

=> DQi* = DQj *and expected DZi* value thanks to the RiSiTi impedance relation.

We assume that the real values DQi, DZi and DZj will be close to DQi*, DZi* and* DZj*.

This results in:

We can then write:

Qin+1 = Qin + DQi = f(Zin + DZi, Zjn + DZj, Wn+1)

= f(Zin + DZi* + dzi, Zjn + DZj* + dzj, Wn+1)

with small values for dzi and dzj.

If we set: f(*) = f(Zin + DZi*, Zjn + DZj*, Wn+1)

we obtain:

Qin + DQi = f(*) + dzi + dzj

=> DQi = f(*) - Qin + (DZi - DZi*) + (DZj - DZj*) = f(*) - Qin - (DZi* + DZj*) + DZi + DZj

By adding:

we get:

[42]

with: D = f(*) - Qin - (DZi* + DZj*)

and f(*) = f(Zin + DZi*, Zjn + DZj*, Wn+1)

Let us take a closer look at how equation *[41]* is determined: let's assume that Zj varies between n and n+1 in the same way as between n-1 and n, with respect to variation of flow.

Then:

k =

Set k = 0 if | Qjn - Qjn-1 | < 0.01 or if the slope of the downstream R'S'T' has the same sign as the upstream RST equation.

Let us examine how to verify whether the RST and the R'S'T' equations have slopes of opposite signs:

The system composed of equations *[38]* and *[41]* is written as follows:

[43]

DQj* = DZi* + DZj*

=> DQj* = (- .DQj*) + DZj*

=> DQj* (1 +) = + DZj*

the system *[43]* is:

It is necessary that: k () < 0

or k $(1 +) < 0$

in order that the RST and the R'S'T' have slopes of opposite signs.

Chapter 201

Offtakes of the form Qp(t)

Let us examine how to consider an offtake which is only described in the form of a time varying outflow relationship, Qp(t).

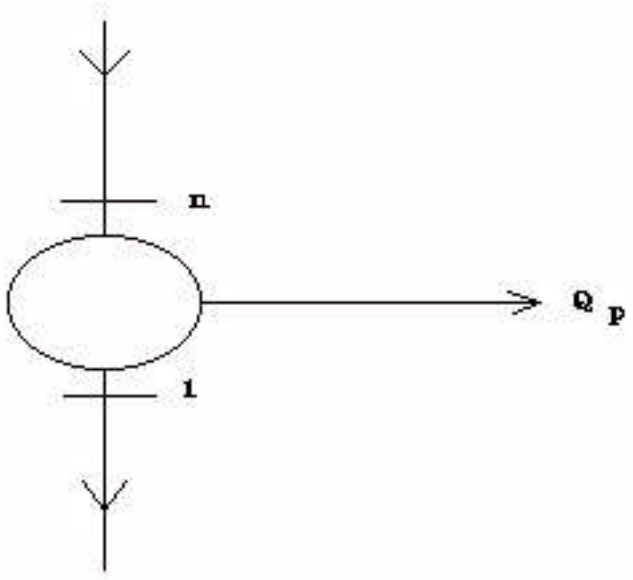


Figure 28

first sweep:

[44]

DQn = DQ1 + DQp + Q1n + Qpn - Qnn

- DZn = DQ1 + - DZn + Q1n + Qpn - Qnn

DQ1 + (-) DZ1 = Qnn - Q1n - Qpn + -

Remark : n is used both in subscript to indicate the cross section index, and in superscript to indicate the current time step.

Then:

We can then go down the first sweep. The second sweep does not pose any particular problems and one calculates W(t) in order to satisfy the offtake discharge for the given water surface elevation at the node.

Chapter 202

Offtakes of the form w(t)

Chapter 203

List of symbols

- A = cross-section area of flow
- B = channel surface width
- CF = discharge coefficient for free-flow undershot gate
- CG = reference discharge coefficient for gate
- CS = discharge coefficient for submerged undershot gate
- g = acceleration of gravity
- H = total head
- h = height of water surface above datum
- hs = gate height for overflow
- K = Strickler's coefficient
- k = 0 for lateral inflow, 1 for lateral outflow
- kF = flow reduction coefficient for submerged gate
- L = length of weir crest
- n = Manning's roughness coefficient
- Q = volumetric rate of discharge
- q = discharge per unit length
- R = hydraulic radius
- So = bed slope
- Sf = friction slope
- V = mean velocity
- W= undershot gate or offtake opening
- x= distance in the direction of flow
- y= vertical depth of flow
- Z= elevation
- Zc= critical elevation
- e= small increment
- m= discharge coefficient for pipe free flow
- m'= discharge coefficient for pipe submerged
- mF= discharge coefficient for free flow weir

- mS= discharge coefficient for submerged weir
- S= summation symbol

Chapter 204

Exponential growth with fixed coefficient

204.1 Definition

This law changes the concentration of a class depending on an other class concentration (or eventually itself), and two fix coefficients k andα_k as shown here:

$\frac{dC_i}{dt} = kC_j^{\alpha_k}$

204.2 Classic use

Numerous solutes have a first-order kinetics with an equation similar to :

$\frac{dC_{Ni}}{dt} = -k_{Ni}C_{Ni}$

where k_{Ni} is the reaction constant (which is the inverse of a time). For instance, bacteriological oxygen demand (DBO_5$$), has a constant around 0.3 days -1 . This degradation models the process of water auto-purification in streams.

204.3 Specifications

- Law's ID : 201
- Number of acting classes : 2
- Number of parameters : : 2

Acting classes : - C_i : the class modified by the law - C_j : the parameter class of the law

Parameters : - k : reaction constant - α_k : reaction order

Chapter 205

Sediment transport

205.1 Generalities

The calculation of sediment exchanges involves three steps: - Calculating an equilibrium concentration C_{eq} according to a proposed law; - Calculating an adaptation time following of Han's formula $t_A = \beta \frac{Ru*}{VW}$ or Hazen's formula: $t_A = \beta \frac{R}{W}$ where W is the fall velocity; - Calculation of the exchange term $E = \frac{\alpha C_{eq} - C}{t_A}$; whereα andβ are dimensionless parameters.

The fall rate is calculated according to Zanke's formula:

$$W = 1.1 * 10 \frac{\nu}{d} \left(\sqrt{1 + \frac{0.01 g (\rho_S / \rho - 1) d^3}{\nu^2}} - 1 \right)$$

This law is equal to Stokes' law for small diameters, and Newton's law for larger particles.

Chapter 206

Engelund-Hansen (1967)

206.1 Engelund-Hansen formula

Transport capacity is calculated as follows :

$$C_{eq} = 0.05\rho_S \frac{LU^2}{Q} \frac{(JR)^{\frac{3}{2}}}{\sqrt{g}(\rho_S/\rho - 1)^2 d}$$

where : - ρ_S is the sediment's density (kg/m3) - L is the stream's width (m) - U is the mean velocity (m/s) - Q is the water discharge (m3/s) - J is the slope (m/m) - R is the hydraulic radius (m) - g is the gravity (m/s2) - ρ is the density of water (kg/m3) - d is the sediment diameter

206.2 Specifications

- Law's ID : 561
- Number of acting classes : 4
- Number of parameters : 6

Acting classes : -# C_i : transported class modified by the law -# C_j : fix class modified by the law -# C_k : transported parameter class ($i = k$ for standard use) -# T : water temperature

Parameters : -# d : sediment diameter -# ρ_S : sediment density -# p : sediment porosity -# α -# i_{ech} : exchange formula : 1 = Han, 2 = Hazen -# β

Chapter 207

Algal detachment in response to flushing

207.1 Definition

The rate of accidental detachment of fixed biomass is calculated at each section as follows:

$\frac{\partial B}{\partial t}(x,t) = -\frac{1}{\delta}\left(\frac{\tau_0(x,t)-\tau_{0,cr}(x)}{\tau_{ref}(x)} - s_B\right)^{\eta} B(x,t)$, if $\frac{\tau_0(x,t)-\tau_{0,cr}(x)}{\tau_{ref}(x)} > s_B$

$\frac{\partial B}{\partial t} = 0$, otherwise

With: - $B(x,t)$: the fixed biomass in kg m-1 - $\tau_0(x,t)$: the wall shear stress in N m-2 - $\tau_{0,cr}(x)$: the critical shear value which integrates the memory effect of the algae in their resistance capacity. - $\tau_{ref}(x)$: the reference shear value - s_B : the sensitivity threshold which takes into account the fluctuations prior to the flushing - δ : a time constant in seconds - η : a dimensionless exponent

207.2 Characteristics

- Law identifier: 201
- Number of classes involved: 1
- Number of parameters: 5

Classes involved: -# B

Parameters as a function of chainage: -# $\tau_{0,cr}$ -# τ_{ref}

Parameters possibly as a function of time -# s_B -# δ -# η

Chapter 208

Algal growth (PhD Thesis, O. Fovet, 2010, p.101)

208.1 Definition

The biomass growth is calculated at time t, in each section x as follows :

$\frac{\partial B}{\partial t}(x,t) = \mu(x,t)B(x,t)F_{lim}(B'(x,t))$

where : - $F_{lim}(B'(x,t)) = \left(1 - \frac{B'(x,t)}{B_{Max}}\right)$ - $\mu(x,t) = \mu_0 \theta^{T(t)-T_0} \frac{I(x,t)}{I_{opt}} e^{1-\frac{I(x,t)}{I_{opt}}} \min\left(\frac{N_i(x,t)}{N_i(x,t)+K_{N_I}}\right)$
- $I(x,t) = I_s(x,t)e^{-k_{ext}h(x,t)}$ - $I_s(x,t) = (1 - C_m(x))(1-a)R_N(t)$

208.2 Variables and parameters

- $B(x,t)$: fix biomass modified by the law (kg m-1)
- $B'(x,t)$: fix biomass parameter in the law (kg m-1)
- B_{Max} : maximum value of fix biomass (kg m-1)
- μ_0 : reference growth rate (s -1)
- θ : growth coefficient
- $T(x,t)$: water temperature (°C)
- T_0 : reference temperature (°C)
- $I(x,t)$: light intensity at the section's bottom (W m-2)
- $I_s(x,t)$: solar light intensity (W m-2)
- $C_m(x)$: mask coefficient (cf. Temperature simulation)
- a : albedo (cf. Temperature simulation)
- $R_N(t)$ solar radiation (W m-2) (cf. Temperature simulation)
- k_{ext} : extinction coefficient (due to turbidity)
- $h(x,t)$: mean water level (m)
- $I_{opt}(x,t)$: optimum light intensity (W m-2)

- $N_i(x,t)$: concentration of nutrient i (kg m-3)
- K_{N_i} : half-saturation constant of nutrient i (kg m-3)

208.3 Limiting nutrient

This law can take into account up to 3 nutrients for algae growth. If less than three nutrients are used, K_{N_i} for unused nutrients should be set to 0.

208.4 Specifications

- Law's ID : 301
- Number of acting classes: 6
- Number of meteo parameters : 3
- Number of parameters : 9

Acting classes : -# $B(x,t)$:the class modified by the law -# $B'(x,t)$: the parameter class of the law -# $T(x,t)$:Water temperature -# $N_1(x,t)$: Nutrient 1 -# $N_2(x,t)$: Nutrient 2 -# $N_3(x,t)$: Nutrient 3

Meteo parameters : - $C_m(x)$ - a - $R_N(t)$

Parameters -# B_{Max} : maximum value of fix biomass -# μ_0 : reference growth rate -# θ : growth coefficient -# T_0 : reference temperature -# k_{ext} : extinction coefficient -# I_{opt} : optimum light intensity -# K_{N_1} : half-saturation constant of nutrient 1 -# K_{N_2} : half-saturation constant of nutrient 2 -# K_{N_3} : half-saturation constant of nutrient 3

Chapter 209

Algual detachment in response to flushing with reference shear stress equal to the first time step

209.1 Definition

The rate of accidental detachment of fix algae b_i towards drift algae a_i is calculated in each section as follows :

$$\frac{\partial B_i}{\partial t}(x,t) = -S\frac{\partial A_i}{\partial t}(x,t) = -\frac{1}{\delta}\left(\frac{\tau_0(x,t)-\tau_0(x,0)}{\tau_0(x,0)} - s_B\right)^{\eta} B_j(x,t)$$

if$\frac{\tau_0(x)-\tau_0(x,0)}{\tau_0(x,0)} > s_B$

$\frac{\partial A_i}{\partial t}(x,t) = \frac{\partial B_i}{\partial t}(x,t) = 0$ atherwise.

Where : - $A_i(x,t)$: drift algae (kg/m3) - $B_i(x,t)$: fix algae (kg/m) - $B_j(x,t)$: fix algae ($i = j$ for standard applications). - $S(x)$: e cross sectional area (m2) - $\tau(x,t)$: shear stress (N m-2) - $\tau_0(x,0)$: shear stress at $t = 0$ - s_B : sensitivity treshold - δ : time constant (s) - η : adimensional exponent

209.2 Specifications

- Law's ID : 341
- Number of acting classes : 3
- Number of parameters : 3

Acting classes : -# A_i : drift class modified by the law -# B_i : fix class modified by the law -# B_j : the parameter class of the law

Parameters : -# s_B : sensitivity treshold -# δ : time constant -# η : adimensional exponent

Chapter 210

Evolution of the temperature

210.1 Definition

This law changes water temperature depending on meteorology and water temperature as explained in «Temperature simulation».

210.2 Specifications

- Law's ID : 101
- Number of acting classes : 1
- Number of parameters : 0

Acting classes : - Water temperature

Chapter 211

Exchange laws

An exchange law modifies one or more quality classes, based on parameters specific to the law, hydraulic variables, and possibly concentrations of some quality classes.

Thus, the exchange terms relative to law i are calculated as follows:

$$\left(E^i_{k_1}, \dots, E^i_{k_n}\right) = L^i\left(C_{k_1}, \dots, C_{k_n}, p_1, \dots, V, h, \dots\right)$$

The exchange term of a class k is the sum of its exchange term in every law where this class is present :

$$E_k = E^{i_1}_k + E^{i_2}_k + \cdots + E^{i_n}_k$$

Chapter 212

Introduction

The ALGEQUEAU module was first developed by Jean-Pierre Baume, Gilles Belaud, David Dorchies and Ophélie Fovet for her PhD Thesis and as a part of the ALGEQUEAU project funded by the French National Research Agency (ANR) for the 2007-2010 period. This module simulates a quality model for irrigation networks regarding : - temperature ; - algal nutrients ; - attached algae ; - drifting algae ; - bed sediment ; - transported sediment.

One aim of the thesis was notably to simulate the transport of drift algae resulting from the detachment of attached algae following a hydraulic flushing.

Chapter 213

Advection–Diffusion Transport Equation

In addition to algal proliferation, the ALGEQUEAU module has evolved to take into account other quality parameters including sediment transport simulation.

The module aims at simulating the evolution and transport of temperature and several classes of nutrients, algae, or sediment in the fixed and transported parts of an hydraulic network.

213.1 Quality classes

Quality in the network is given at any point by a vector :

$$C(x,t) = \begin{pmatrix} C_1(x,t) \\ \vdots \\ C_n(x,t) \end{pmatrix}$$

where C_k may be : - a temperature (°C) ; - a nutrient concentration (mg/L) ; - a concentration of drift algae (kg/m3) ; - a linear density of fixed algae (kg/m) - a sediment concentration in water (g/m3) ; - a linear density of sediment in the bed (m3/m).

213.2 Advection-Diffusion equation

The advection diffusion transport equation is solved for transported classes :

$$\frac{\partial SC_k}{\partial t} + \frac{\partial QC_k}{\partial x} = \frac{\partial}{\partial x}\left(DS\frac{\partial C_k}{\partial x}\right) + SE_k$$

where : - S is the wetted area (m2) ; - Q is the discharge (m3/s) ; - D is the diffusion coefficient (m2/s); - E_k is an exchange rate ($[C_k]$/s with $[C_k]$ the unit of class C_k).

For fixed classes, there is no advection-diffusion term in the equation :

$$\frac{dC_k}{dt}(x,t) = E_k$$

213.3 Calculation of the exchange rate

The vector E of exchange rates for each class is computed from the list of exchange formulas given by the user, depending on the classes vector C and some parameters p

$$E_k = \mathcal{F}_k(C,p)$$

where F_k sums the contributions of all the exchange formulas affecting C_k.

213.4 Numerical resolution of the equation

The solution to the equation is calculated using a conjugate gradient method.

Chapter 214

Concepts and equations at a node

The following equations describe a node in its most general sense. A node may contain : - a pond ; - several intakes or offtakes ; - several upstream or downstream reaches.

The concepts and variables used at a node are describe in the following diagram :

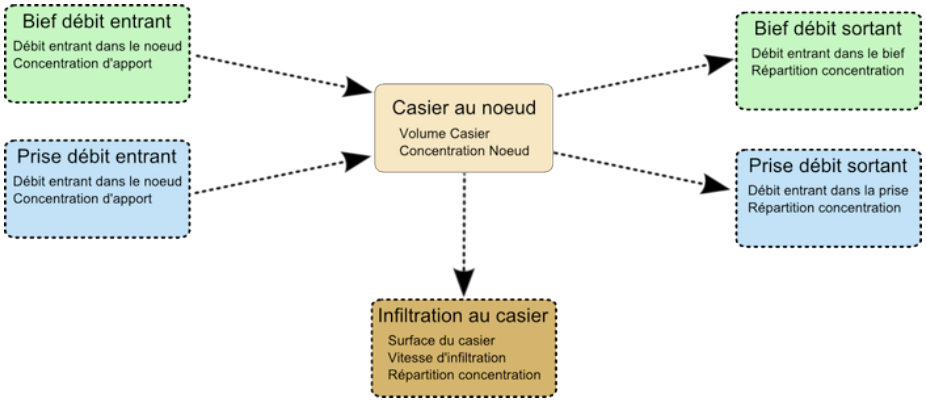


Figure 214.1: doc36

A pond is stored as a monotonous sequence of pairs (Surface, Elevation). The evaporation and the infiltration can be modelled with an evaporation rate $V_{ep}(t)$ and an infiltration rate $v_{inf}(Z_{nd})$, Z_{nd} being the water level in the pond.

The node gains or loses water through offtakes (Q_p) and reaches (Q_b). Inflow is

positive, whereas outflow is negative.

Reaches are given a time invariant flow direction according to their topology, which is :

- $s_b = 1$ for inflow
- $s_b = -1$ for outflow

The discharge Q_b is positive if it flows is in the same direction as the above topologic flow direction, and negative otherwise. The actual discharge is therefore given by : $s_b Q_b$

214.1 Inflow (positive discharge)

For upstream or downstream reaches with $s_b Q_b > 0$ or offtakes with $Q_p > 0$ there is a boundary condition for transported quality classes C which is the concentration incoming.

214.2 Outflow (negative discharge)

214.3 Reaches

For upstream or downstream reaches with $s_b Q_b < 0$, we have $C_b = k_b C_{nd}$.

214.4 Offtakes

For offtakes with $Q_p < 0$, we have $C_p = k_p C_{nd}$ (k_p is describe later).

214.5 Infiltration at the node

We have $C_{inf} = k_{inf} C_{nd}$ and $Q_{inf} = S_{cas} v_{inf}$ where : S_{cas} is the pond area, v_{inf} is the infiltration rate and C_{nd} is the node concentration.

214.6 Storage flow in the pond

The pond is connected to the node, and the concentrations at the node and the pond are the same. The stored amount of a class in the pond is : $m_{Cas} = V_{Cas} C_{nd}$ and the flow in or out is : $Q_{cas} = S_{cas} \frac{dZ_{nd}}{dt}$.

214.7 Mass balance at the node

The mass balance equation for a quality class in the node is :

$\frac{dm_{nd}}{dt} = \sum_{s_b Q_b>0} C_b s_b Q_b + \sum_{s_b Q_b<0} k_b s_b Q_b C_{nd} + \sum_{Q_p>0} C_p Q_p + \sum_{Q_p<0} k_p Q_p C_{nd} - k_{inf} S_{cas} C_{nd} v_{inf} + V_{Cas} E(t, C_{nd})$

With $\frac{dm_{nd}}{dt} = \frac{d(C_{nd} V_{cas})}{dt} = \frac{dV_{cas}}{dt} C_{nd} + \frac{dC_{nd}}{dt} V_{cas}$, the equation gives :

$\sum_{s_b Q_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + V_{Cas} \left(E(t, C_{nd}) - \frac{dC_{nd}}{dt} \right) + C_{nd} \left[\sum_{s_b Q_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{inf} S_{cas} v_{inf} - \frac{dV}{d} \right.$
0

where S_{cas}, V_{cas} , Z_{nd} are the area, volume and water level of the pond.

214.8 Distribution coefficient k

3 cases are considered for the outflow distribution coefficients :

214.9 Outflows are homogeneous

All k are 1 : whatever the evolution of flow and contentrations, the mass balance equation is used to compute the concentration in outflows.

214.10 Outflows are heterogeneous and the node contains a pond

The concentration entering or leaving the pond ensures mass balance at the node.

214.11 Outflows are heterogeneous and the node does not contain a pond

Some distribution coefficients must be adjusted (while remaining positive) to maintain the mass balance condition.

The distribution coefficients are given by the user for each offtake and reach and for infiltration, and can be defined as fix ($b_i = 0$) or adjustable ($b_i = 1$) to ensure continuity at the node.

An adjustement coefficient k_a for the node is computed and the distribution coefficients become :${k_a}^{b_i} k_i$. Therefore, a fix coefficient ($b_i = 0$) is not changed${k_a}^{b_i} k_i = k_i$ whereas an adjustable coefficient becomes${k_a}^{b_i} k_i = k_a k_i$.

The new mass balance equation is :

$\sum_{Q_i>0} C_i Q_i + C_{nd} \left[\sum_{Q_i<0, b_i=0} k_i Q_i + k_a \sum_{Q_i<0, b_i=1} k_i Q_i \right] = 0$

$k_a = - \frac{\frac{1}{C_{nd}} \sum_{Q_i>0} C_i Q_i + \sum_{Q_i<0, b_i=0} k_i Q_i}{\sum_{Q_i<0, b_i=1} k_i Q_i}$

whereC_{nd} is the node concentration given by

$$C_{nd} = \frac{\sum_{Q_i>0} C_i Q_i}{\sum_{Q_i>0} Q_i}$$

The equation for k_a is :

$$k_a = -\frac{\sum_{Q_i>0} Q_i + \sum_{Q_i<0, b_i=0} k_i Q_i}{\sum_{Q_i<0, b_i=1} k_i Q_i}$$

$$k_a = -\frac{\left(\sum_{s_b Q_b>0} s_b Q_b + \sum_{Q_p>0} Q_p\right) + \left(\sum_{s_b Q_b<0, b_b=0} k_b s_b Q_b + \sum_{Q_p<0, b_p=0} k_p Q_p\right)}{\sum_{s_b Q_b<0, b_b=1} k_b s_b Q_b + \sum_{Q_p<0, b_p=1} k_p Q_p}$$

The final mass balance equation is :

$$\sum_{s_b Q_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + C_{nd} \left(\sum_{s_b Q_b<0} {k_a}^{b_b} k_b s_b Q_b + \sum_{Q_p<0} {k_a}^{b_p} k_p Q_p\right) = 0$$

or

$$\sum_{s_b Q_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + C_{nd} \left[\sum_{s_b Q_b<0, b_b=0} k_b s_b Q_b + \sum_{Q_p<0, b_p=0} k_p Q_p + k_a \left(\sum_{s_b Q_b<0, b_b=1} k_b s_b Q \right. \right. = 0$$

Chapter 215

Network computation

If there is no negative discharge, the network is calculated in the upstream-downstream "topological" direction as given by Talweg.

At a node, the concentrations C_p and C_b of incoming offtakes and reaches are known and give C_{nd} and C_b for outgoing offtakes and reaches using the node mass balance equation :

$C_b = C_{nd} k_b$ if the distribution coefficient is defined as fix $C_b = C_{nd} k_b k_a$ if it is defined as adjustable.

Negative discharge in reaches are not yet implemented in the quality module.

Chapter 216

Steady state calculation

216.1 Calculation at a section

216.2 Transport equation

The steady-state advection equation is:

$\frac{dCQ}{dx} = E(x, C)S$

216.3 Integration using fourth-order Runge-Kutta méthod

Previous equation is rewriten as $\frac{dy}{dx} = f(y, x)$ (where $y = CQ$ and $f(y, x) = E(x, y)\, S$) and integrated using a fourth-order Runge-Kutta méthod(RK4) :

$y_{j+1} = y_j + \frac{\Delta x}{6}(k_1 + 2k_2 + 2k_3 + k_4)$ where :

- $k_1 = f(y_j, x_j) = E(x_j, y_j)\, S_j = E(x_j, C_j)\, S_j$
- $k_2 = f\left(y_j + \frac{\Delta}{x} 2k_1, x_{j+\frac{1}{2}}\right) = E\left(x_{j+\frac{1}{2}}, y_j + \frac{\Delta}{x} 2k_1\right) S_{j+\frac{1}{2}} = E\left(x_{j+\frac{1}{2}}, \frac{y_j + \frac{\Delta}{x} 2k_1}{Q_{j+\frac{1}{2}}}\right) S_{j+\frac{1}{2}}$ where $S_{j+\frac{1}{2}} = \frac{S_j + S_j + 1}{2}$ and $Q_{j+\frac{1}{2}} = \frac{Q_j + Q_{j+1}}{2}$
- $k_3 = f\left(y_j + \frac{\Delta}{x} 2k_2, x_{j+\frac{1}{2}}\right) = E\left(x_{j+\frac{1}{2}}, y_j + \frac{\Delta}{x} 2k_2\right) S_{j+\frac{1}{2}} = E\left(x_{j+\frac{1}{2}}, \frac{y_j + \frac{\Delta}{x} 2k_2}{Q_{j+\frac{1}{2}}}\right) S_{j+\frac{1}{2}}$
- $k_4 = f(y_j + \Delta x k_3, x_j + 1) = E(x_j + 1, y_j + \Delta x k_3)\, S_j + 1 = E\left(x_j + 1, \frac{y_j + \Delta x k_3}{Q_j + 1}\right) S_j + 1$

Concentrations in section j+1 are therefore :

$C_{j+1} = \frac{y_{j+1}}{Q_{j+1}}$

216.4 Calculation at a node

216.5 Continuity equation at a node

The steady state mass balance at a node is :

$\sum_{s_b Q_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + V_{Cas} E(t, C_{nd}) + C_{nd} \left[\sum_{s_b Q_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{inf} S_{cas} v_{inf} \right] = 0$

216.6 Newton's method

Let $f(C_{nd}^i) = 0$ be the mass balance equation for the node on iteration i.

Newton's method gives :

$C_{nd}^i = C_{nd}^{i-1} - \frac{f(C_{nd}^{i-1})}{f'(C_{nd}^{i-1})} = C_{nd}^{i-1} - \frac{b^i}{a^i}$

$b^i = \sum_{s_b Q_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + V_{Cas} E(t, C_{nd}^{i-1}) + C_{nd}^{i-1} \left[\sum_{s_b Q_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{inf} S_{cas} v$

Is the node does not contain a pond, there is no source term and the equation is linear. If distribution coefficients k_b, k_p and k_{inf} are heterogeneous, they are adjusted before the first and only iteration.

If there is a pond, the equation is no longer linear and several iterations are needed but distribution coefficients are not adjusted.

The general equation for a^i is :

$a^i = V_{Cas} E'(C_{nd}^{i-1}) + \sum_{s_b Q_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{inf} S_{cas} v_{inf}$

where $E'(C_{nd}^{i-1})$ is the derivative of the exchange.

216.7 Calculation for a node with a pond

Some terms in the previous equation are constant and do not need to be recomputed on every iteration :

$b_1 = \sum_{s_b Q_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p \;\; a_1 = \sum_{s_b Q_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{inf} S_{cas} v_{inf}$

Which yields :

$b^i = b_1 + V_{Cas} E(t, C_{nd}^{i-1}) + a_1 C_{nd}^{i-1}$ et $a^i = V_{Cas} E'(C_{nd}^{i-1}) + a_1$

216.8 Calculation for a node without a pond

For a node without a pond, the node concentration is directly calculated as a pondered mean of inflow concentrations.

$$C_{nd} = \frac{\sum_{s_b Q_b > 0} C_b s_b Q_b + \sum_{Q_p > 0} C_p Q_p}{\sum_{s_b Q_b > 0} s_b Q_b + \sum_{Q_p > 0} Q_p}$$

If the distribution coefficients are adjustable, K_a is computed before the iterations, as described in "Network computation".

Chapter 217

Transient calculation

217.1 Calculation at a section

217.2 Discretization for Preissmann Scheme

The advection equation (without diffusion) for drift classes is :

$\frac{\partial CS}{\partial} t + \frac{\partial CQ}{\partial} x = E(x,t,C)S$

Preissmann scheme discretization equations are :

$\frac{\partial}{U}\partial t \approx (1-\psi)\frac{U_j^{n+1}-U_j^n}{\Delta} t + \psi\frac{U_{j+1}^{n+1}-U_{j+1}^n}{\Delta} t$

$\frac{\partial U}{\partial x} \approx (1-\theta)\frac{U_{j+1}^n-U_j^n}{\Delta x_{j+1/2}} + \theta\frac{U_{j+1}^{n+1}-U_j^{n+1}}{\Delta x_{j+1/2}}$ $U(x,t) = \theta\left(\psi U_{j+1}^{n+1} + (1-\psi)U_j^{n+1}\right) + (1-$
$\theta)\left(\psi U_{j+1}^n + (1-\psi)U_j^n\right)$

We note : $\Delta(U)_i = U_i^{n+1} - U_i^n$

These equations for section j+1 at time k+1 yield :

$\frac{(1-\psi)\Delta(CS)_j + \psi.\Delta(CS)_j + 1}{\Delta} t + \frac{(CQ)_{j+1}^k - (CQ)_j^k + \theta(\Delta(CQ)_j + 1 - \Delta(CQ)_j)}{\Delta} x = \left[(1-\psi)(ES)_j^k + \psi(ES)_{j+1}^k\right] +$
$\theta\left[(1-\psi)\Delta(ES)_j + \psi\Delta(ES)_j + 1\right]$

The transport equation can be rewritten as $F(C_{j+1}^{k+1}) = 0$ since C_j^{k+1} is given by the upstream condition.

$\delta x\left[(1-\psi)\Delta(CS)_j + \psi.\Delta(CS)_j + 1\right] + \Delta t\left[(CQ)_{j+1}^k - (CQ)_j^k + \theta\left(\Delta(CQ)_j + 1 - \Delta(CQ)_j\right)\right]$
$-\delta x\Delta t\left[(1-\psi)(ES)_j^k + \psi(ES)_{j+1}^k\right] - \delta x\Delta t\theta\left[(1-\psi)\Delta(ES)_j + \psi\Delta(ES)_j + 1\right] =$
0

$\delta x\left[(1-\psi)\left((CS)_j^{k+1} - (CS)_j^k\right) + \psi.\left((CS)_{j+1}^{k+1} - (CS)_{j+1}^k\right)\right] + \Delta t\left[(CQ)_{j+1}^k - (CQ)_j^k + \theta\left(\left((CQ)_{j+1}^{k+1} - (CQ)_{j+1}^k\right) - \left((C\right.\right.\right.$
$-\delta x\Delta t\left[(1-\psi)(ES)_j^k + \psi(ES)_{j+1}^k\right] - \delta x\Delta t\theta\left[(1-\psi)\left((ES)_j^{k+1} - (ES)_j^k\right) + \psi\left((ES)_{j+1}^{k+1} - (ES)_{j+1}^k\right)\right] =$

0

217.3 Resolution by conjugate gradient method

This equation $F(C_{j+1}^{k+1}) = 0$ is solved using a conjugate gradient method, requiring the computation of a Jacobian matrix.

Let $S(C_{j+1}^{k+1}) = {}^tF(C_{j+1}^{k+1})F(C_{j+1}^{k+1})$. The equation is solved by minimizing S, using the conjugate gradient method. One need to calculate the partial derivatives of exchange terms with respect to every quality class in order to calculate F's Jacobian matrix J which gives S's gradient∇S.

217.4 Discretization at a node

217.5 Mass balance at a node

The mass balance equation at a node is integrated between t et $t + \Delta t$:

$\int_t^{t+\Delta t}\left(\sum_{s_bQ_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + V_{Cas}E(t, C_{nd})\right) dt - \int_{C_{nd}}^{C_{nd}+\Delta C_{nd}} V_{cas} dC_{nd}$
$+ \int_t^{t+\Delta t}\left(C_{nd}\left[\sum_{s_bQ_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{inf} S_{cas} v_{inf}\right]\right) dt - \int_{V_{cas}}^{V_{cas}+\Delta V_{cas}} C_{nd} dV_{cas} =$
0

This integral is calculated using the trapezoidal rule formula :

$\frac{\Delta}{t}2\left(\sum_{s_bQ_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + V_{Cas}E(t, C_{nd})\right)^k + \frac{\Delta}{t}2\left(\sum_{s_bQ_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + V_{Cas}\right.$
$-\frac{V_{cas}^{k+1}-V_{cas}^k}{2}\left[C_{nd}^k + C_{nd}^{k+1}\right] - \frac{C_{nd}^{k+1}-C_{nd}^k}{2}\left[V_{Cas}^k + V_{Cas}^{k+1}\right] + \frac{\Delta}{t}2C_{nd}^k\left[\sum_{s_bQ_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{inf} S_{cas} v\right.$
$+\frac{\Delta}{t}2C_{nd}^{k+1}\left[\sum_{s_bQ_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{inf} S_{cas} v_{inf}\right]^{k+1} = 0$

The source term E is the only non-linear term in this equation.

217.6 Newton's method for a node with a pond

Let $f(C_{nd}^i) = 0$ be the result of the integration at the ith iteration.

Newton's method gives : $C_{nd}^i = C_{nd}^{i-1} - \frac{f(C_{nd}^{i-1})}{f'(C_{nd}^{i-1})} = C_{nd}^{i-1} - \frac{b^i}{a^i}$

$b^i = \frac{\Delta}{t}2\left[\sum_{s_bQ_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + V_{Cas}E(t, C_{nd}) + C_{nd}^k\left(\sum_{s_bQ_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{in}\right.\right.$
$+\frac{\Delta}{t}2\left[\sum_{s_bQ_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + V_{Cas}E(t, C_{nd}^{i-1}) + C_{nd}^{i-1}\left(\sum_{s_bQ_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{inf}\right.\right.$
$-V_{cas}^{k+1}C_{nd}^k - V_{cas}^k C_{nd}^{k+1,i-1}$

If $b^i = b_1 + b_2^i$ where b_1 is constant over iterations et b_2 we get :

$b_1 = \frac{\Delta}{t}2\left[\sum_{s_bQ_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p + V_{Cas}E(t,C_{nd}) + C_{nd}^k\left(\sum_{s_bQ_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p - k_{inf}S_{cas}v_{inf}\right)\right]^k$
$+\frac{\Delta}{t}2\left[\sum_{s_bQ_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p\right]^{k+1} - V_{cas}^{k+1}C_{nd}^k$

We define $b_1 1^k = \left[\sum_{s_bQ_b>0} C_b s_b Q_b + \sum_{Q_p>0} C_p Q_p\right]^k$ and $a_1^k = \left[\sum_{s_bQ_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p\right]^k$

yielding : $b_1 = \frac{\Delta}{t}2\left([b_1 1 + V_{Cas}E(t,C_{nd})]^k + b_1 1^{k+1}\right) + C_{nd}^k\left(\frac{\Delta}{t}2\left[a_1 - k_{inf}S_{cas}v_{inf}\right]_{cas}^{k+1,k-V}\right)$

and $b_2^i = \frac{\Delta}{t}2\left[V_{Cas}E(t,C_{nd}^{i-1}) + C_{nd}^{i-1}\left(a_1 - k_{inf}S_{cas}v_{inf}\right)\right]^{k+1} - V_{cas}^k C_{nd}^{k+1}, i-1$

The derivative with respect to C_{nd} is : $a^i = \frac{\Delta}{t}2\left[V_{Cas}E'(t,C_{nd}^{i-1}) + \left(a_1 - k_{inf}S_{cas}v_{inf}\right)\right]^{k+1} -$
V_{cas}^k

As C_{nd} is a vector, this equation is true for every quality class $C_{nd}(i)$.

The different equations may be interdependent due to the exchange rate E.

As a consequence, matrix a is non-diagonal due to E'. If this term is negligible compared to the others, the Newton's method can still solve the system.

217.7 Calculation at a node without a pond

A node without a pond is a nodes where the volume (or area) of the pod is zero at time k+1 and k.

The equation is then linear and the exact result is found within one Newton iteration. If distribution coefficients are heterogeneous, they are adjusted to ensure mass balance at the node.

The formula for C_{nd} is : $C_{nd}^{k+1} = C_{nd}^k - \frac{f(C_{nd}^k)}{f'(C_{nd}^k)} = C_{nd}^k - \frac{b}{a}$

Equations for a node with a pond are : $b = \frac{\Delta}{t}2\left[b_1 1^k + b_1 1^{k+1} + C_{nd}^k\left(a_1^k + a_1^{k+1}\right)\right]$
et $a = \frac{\Delta t}{2}a_1^{k+1}$

And : $C_{nd}^{k+1} = C_{nd}^k - \frac{b_{11}^k + b_{11}^{k+1} + C_{nd}^k(a_1^k + a_1^{k+1})}{a_1^{k+1}}$

Given the adjustable coefficient : $a_1^k = \left[\sum_{s_bQ_b<0} k_b s_b Q_b + \sum_{Q_p<0} k_p Q_p\right]^k$, we get :

$a_1^k = \left[\sum_{s_bQ_b<0, b_i=0} k_b s_b Q_b + \sum_{Q_p<0, b_i=0} k_p Q_p + k_a\left(\sum_{s_bQ_b<0, b_i=1} k_b s_b Q_b + \sum_{Q_p<0, b_i=1} k_p Q_p\right)\right]^k$

Chapter 218

Transiant calculation for fix classes

The transient equation for fix classes is :

$\frac{\partial C_k}{\partial t}(x,t) = E_k(x,t)$

where :

- C_k is the linear mass density ;
- E_k is the derivative of C_k with respect to time.

The time discretization for $j+1 = t + \Delta t$ of this equation is made by a semi-implicit Euler method :

$C_k^{j+1} - C_k^j - \frac{\Delta t}{2}\left(E_k^j + E_k^{j+1}\right) = 0$

Resolution is then performed in the same Newton's method iteration as drifting quality classes.

The formula for iteration i is:

$C_k^{j+1,i} = C_k^{j+1,i-1} - \frac{f\left(C_k^{j+1,i-1}\right)}{f'\left(C_k^{j+1,i-1}\right)} = C_k^{j+1,i-1} - \frac{b_i}{a_i}$

where :

- $b_i = C_k^{j+1,i-1} - C_k^j - \frac{\Delta t}{2}\left(E_k^j + E_k^{j+1,i}\right)$
- $a_i = 1 - \frac{\Delta t}{2} E_k'^{j+1,i}$

$E_k^{j+1,i}$ and $E_k'^{j+1,i}$ are fonctions of $C_k^{j+1,i-1}$ and drifting classes concentrations $C_k'^{j+1,i-1}$.

Chapter 219

Temperature simulation

219.1 Transport and exchange equation

Water temperature in the network is a fonction of upstream temperature and thermal exchange with atmosphere and substratum.

The general equation is:

$\frac{\partial ST}{\partial t} + \frac{\partial QT}{\partial x} = \frac{\partial}{\partial x}\left(DS\frac{\partial T}{\partial x}\right) + SE_T$

where E_T is an exchange rate (°C/s) given by the heat exchange rate P (W/m2). For a water colomn with a width ofδx, energy balance is

$\rho S\delta x\ c_p E_T = PL_m\delta x$

whereρ is the mass density of water, c_p is the heat capacity of water, L_m is the reach's width, P is the heat power per unit area, yielding

$E_T = \frac{L_m P}{\rho c_p S}$

Numerical values are :

ρ = 1000 kg/m3

c_p = 4186 J/kg/°C

219.2 Calculation of the heat power term

The heat power per unit area is a balance of direct incoming solar and atmospheric radiation, minus the water surface radiation, and sensible and latent heat flow. The heat exchange between water and soil is difficult to evaluate since it depends on the soil's temperature and thermal parameters.

A mask coefficient C_m is defined as a function of abscissa, which takes into account the potential coverage of the canal (passage in galleries in particular). Full coverage will cancel all exchange between water and atmosphere. Neglecting interactions with the substrate amounts to considering the environment as adiabatic or to assume that these exchanges are low with respect to the convection process of the water mass.

The heat power term is thus calculated :

$P = (1 - C_m)((1 - a)R_N + R_a - R_e - H_s - H_e)$

Solar radiation is a measured data that can be estimated using the solar radiation above atmosphere, the atmospheric attenuation and the time and location.Albedo a is generally low for water, around 0.03.

Atmospheric radiation is given by Stefan Boltzmann formula, and takes into account water reverberation, clouds and air relative humidity (w_a, between 0 and 1) using Brutsaert formula (1982)[[W. Brutsaert. Evaporation into the atmosphere. Kluwer Acad. Pubs., 1982.]]. We therefore need to calculate the equilibrium vapor pressure e_s (Pa) and vapor pressure e_e (en Pa) at air temperature T_a (°C) with a humidity w_a:

$e_s = 101300 \exp\left(13.7 - \frac{5120}{273.15+T_a}\right)$

$e_a = w_a e_s$

$c_a = 1,24(1 - a)\left(\frac{e_a/100}{273.15+T_a}\right)^{1/7}$

$R_a = c_a \sigma (273.15 + T_a)^4$

where$\sigma = 5,6710^{-8}$ W m-2K-4 is Stefan-Boltzmann constant.

Similarly, water radiation is :

$R_e = \epsilon \sigma (273.15 + T_e)^4$

where T_e is the water temperature (°C) andϵ is water emissivity ($\epsilon = 0,97$).

Sensible heat flow is linked to thermal gradient between air and water and to the wind speed U_V. We use De Bruin's formula (1978), in W/m^2/Pa, as cited in [[C. Jacovides, G. Papaioannou, and P. Kerkides. Micro and large-scale parameters evaluation of evaporation from a lake. Agricultural Water Management, 13(2-4):263-272, 1988]]:

$f_V = 0.029 + 0.021 U_V$

and

$H_s = C_B f_V (T_e - T_a)$

where $C_B = 63$ Pa/° C (Bowen coefficient).

Then, latent heat flow is calculated as follows :

$H_e = f_V(e_s - e_a)$

Chapter 220

Theoretical concepts

Warning: equations of this Help File are not always correctly visible, because of the format of the .chm files. In case of doubt please check on the .pdf file provided under SIC/DOC or contact Cemagref. This will be improved in a future version by shifting all equations in graphical format. **

Chapter 221

Introduction

The SIC software (Simulation of Irrigation Canals) is a mathematical simulation model adapted to most of the world-wide existing irrigation canals. The main purpose of the model is to properly represent the canal hydraulics while at the same time possessing an interface that is as user-friendly as possible.

The model must be able to provide answers to the main problems that confront the canal manager.

In particular, it should be able:

- to simulate the actual hydraulic and operational conditions,
- to test design modifications (hydraulics structures, new calibration, etc.),
- to test new management rules aimed at achieving water savings.

The model has been designed to serve not only as a training and decision-support tool for the canal manager, but also as a research tool for the improvement of the hydraulic functioning of the main canal. The flow simulation in the model is based on one-dimensional hydraulic computations under steady and transient regimes.

The purpose of this note is to present the theoretical concepts on which the model is based, to account for the canal topography, as well as to carry out the hydraulic computation itself.

Chapter 222

Part IV

Commercial informations

Chapter 223

Commercial information

Company: Inrae - UMR G-EAU

Project leaders: D. DORCHIES & P.O. MALATERRE

Address: 361, Rue Jean-François BRETON

BP 5095 - 34196 MONTPELLIER cedex 5 - FRANCE

Tel: (+33) [0]4 67 04 63 56 or (+33) [0]467 04 63 22

Email: SIC webmaster

Price: 14000 Euros (Professional version) or 1000 Euros (Research & Educational version)

Change log of previous versions / Road map of future versions

Maintenance: 1 year free of charge (corrective and upgrading maintenance))

Training: 3 days for the professional version; none for the other versions except under specific contracts

Chapter 224

Part V

Hydraulic calculations with Cassiopée

Chapter 225

Hydraulic calculations with Cassiopée

https://cassiopee.g-eau.fr

The Cassiopée software was developed by the ecohydraulic R&D pole gathering OFB (French Office for Biodiversity) and IMFT (Fluids Mechanics Institute of Toulouse) and by UMR G-EAU (Joint Research Unit "Water Management, Actors, Territories").

It includes tools for designing fish crossing devices for upstream and downstream migrations, and hydraulic calculation tools useful for environmental and agricultural engineering.