Integrating Coastal Sediment Systems

UnaLinea User Manual (Version 01)

DDS1202-RT007-R01-00

June 2016

Document information

Document permissions	Unrestricted
Project number	DDS1202
Project name	Integrating Coastal Sediment Systems
Report title	UnaLinea User Manual (Version 01)
Report number	RT007
Release number	R01-00
Report date	June 2016
Client	NERC
Client representative	Jessica Surma
Project manager	James Sutherland
Project director	Richard Whitehouse

Document history

Date	Release	Prepared	Approved	Authorised	Notes
14 Jun 2016	01-00	KRO	BLB	JAS	

Document authorisation

Prepared	Approved	Authorised

Acknowledgements and license

This report was prepared as part of the project “Integrating Coastal Sediment Systems” (iCOASST) which was led by the University of Southampton. For further details of the project, please see http://www.icoasst.net/.

The iCOASST project was funded by the Natural Environment Research Council (http://www.nerc.ac.uk) under grant NE/J00541X/1, with the Environment Agency (http://environment-agency.gov.uk) as an embedded stakeholder.

UnaLinea was partly developed with funding from the Flood Risk Management Research Consortium (FRMRC) as part of FRMRC2 Work Package 4.3. For further details see https://web.sbe.hw.ac.uk/frmrc/index.htm. FRMRC was sponsored by the Engineering and Physical Research Council (EPSRC) under Grant EP/FO20511/1, with additional funding from the EA/Defra (Joint Defra/EA Flood and Coastal Erosion Management R&D Programme), the Northern Ireland Rivers Agency (DARDNI) and Office of Public Works (OPW), Ireland

Contents

Tables

1. Introduction

UnaLinea is a state-of-the-art model that simulates the evolution of the plan shape of a beach (Stripling and Panzeri, 2009, Stripling et al., 2011). The beach plan shape is determined by the position of a single contour, therefore making UnaLinea a “one-line model”. UnaLinea uses a formulation of total longshore transport rate based on the widely used CERC formula. The model changes the coastline every time step, allowing for the correct simulation of the changing drift rates with time.

A typical application of this model would be to study the impact that different interventions to the fluvial load will have in the adjacent coast in a medium to long term time.

This user manual (HR Wallingford report DDS1202-RT007) describes the necessary input files to run UnaLinea, as well as the outputs generated by the model. The accompanying reference manual (HR Wallingford report DDS1202-RT008, 2016) provides the theoretical background of the model.

2. Naming convention

A steering or control file is needed and will set the name convention. The name of this file is a nine character stem followed by the extension **.STE**. This stem will be repeated for the output files, so that the extension is dropped and replaced by standard ones.

3. Model basics

3.1. Coordinate system

A right-hand side co-ordinate system is used in UnaLinea with the x-axis parallel to the general trend of the shoreline and the y-axis perpendicular to it, going offshore.

3.2. Y and Q points

UnaLinea uses a staggered grid (rather than a co-located grid) to make the boundary conditions easier to be implemented. Therefore, there are two types of nodes:

- Q points: where the values of the wave conditions and transport rates are calculated. The groynes are also defined in these Q-points; and
- Y-points: the shoreline position (included the initial position) and seawall position are given at the Y-sections, as well as any nourishment or mining taking place in the model.

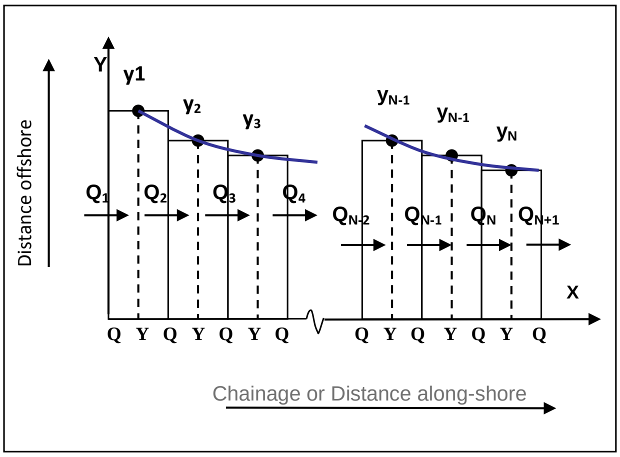

Figure 3.1: Y points and Q points in UnaLinea

Source: HR Wallingford

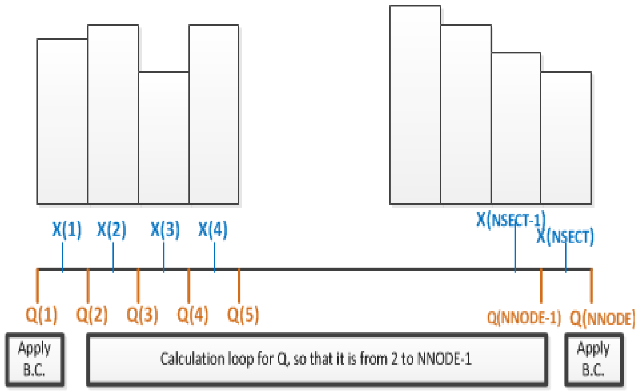

Figure 3.2: Calculation nodes

Source: HR Wallingford

4. Running UnaLinea

The Unalinea code is written in FORTRAN. The code has been successfully compiled in Visual Studio (version 2010) and GFortran (win32 gcc version 4.5.2 20101015) compiler. Executables made by both compilers are supplied.

The model can be run directly by double clicking the executable or by calling it from the command window.

When running the model the model will require the name of the Steering file. This file will contain the name of the wave file(s) and the name of the external source file(s) (if necessary).

4.1. Input files

There are two compulsory input files for UnaLinea, a control file and a wave file (there can also be more than one wave file). The format of these files is described below.

There is an optional input file, the external source file, with information at every time-step of the source value at a given gridpoint and an optional cliff steering file if cliffs are to be used.

4.2. Steering File

UnaLinea control or steering file is a text file in which different values within a line are separated by commas. An example of a control file is given in Frame 4.1. An explanation of the variables is given in the table below. The variable name is as given in the code to aid understanding.

Table 4.1: Variables in steering file

NAME	UnaLinea NAME	TYPE	EXPLANATION
TITLE	`NameProject`	Character*80	Name of the project

NAME	UnaLinea NAME	TYPE	EXPLANATION
TYPE_OUTPUT	`TypeOutput`	Character*2	Type of output. Options: • SO: Short Output; which will produce only one output file with the results at the last timestep • LO: Long Output; which will produce seven different output files, with information at every timestep.
OPTION	`Option`	Character*2	Type of wave data. Options: • TS: Time series • AV: Averaged time series See wave data section for further explanation
SOLVER	`Solver`	Character*2	Type of numerical solver. Options: • EE: Explicit Euler • ER: Explicit Runge Kutta 4th order • AB: Explicit Adam Bashford
FMLA	`SedTransFmla`	Character*2	Type of formulation for the longshore transport. Options: • CE: CERC • SD: Soulsby and Damgaard formulation for bedload only
OPENCLOSED	`OpenclosedBd`	Character*2	Type of boundaries at either side. Options: • OO: Both open • CC: Both closed • OC: lhs boundary open and rhs closed • CO: rhs boundary open and lhs closed • OP: lhs boundary open and rhs partially closed • PO: rhs boundary open and lhs partially closed
PARTC	`PartC`	Real	Fraction of Q passing part closed boundaries. Only inputted when boundayType=CO or PO.
DT	`Dt`	Integer	Timestep to be used in seconds. The following options have a name associated: • 3600: ‘hour’ • 86400: ‘day’ • 604800: ‘week ‘ • 1209600: ’fortn’ • 2629800: ‘month’ • 31557600: ‘year ‘

NAME	UnaLinea NAME	TYPE	EXPLANATION
NDT	`NDt`	Integer	Number of subdivisions of the timestep.
NTIMESTEPS	`NTimeSteps`	Integer	Number of timesteps to run the model for
NOUTPUTS	`NOutputs`	Integer	Number of outputs required
NTYEAR	`NTYear`	Integer	Number of timesteps in a year,
OUTPUTTIME (i) i=1,NOUTPUTS	`OutputPerTime`	Integer	Timesteps at which output is required
NSECT	`Nsect`	Integer	Number of beach sections, at which the beach position will be calculated. Breaking wave conditions and transport rates will be calculated at NNODE (which is NSECT+1) positions.
CTLABEL1	`CtLabel1`	Integer	Label for the shoreline position, so that it will be given as a constant value if 0 or as NSECT values if $\neq 0$)
YPOSCONST (if CTLABEL=0) YPOS1(M)M=1,NSECT1 (otherwise)	`YPos`	Real	Initial beach position: • Constant for each beach section (if CTLABEL=0) • Different for each beach section (otherwise)
D50K1	`D50K12`	Character*3	Label to input parameter K1 or D50. Options: • D50: Compulsory if using Soulsby-Damgaard formulation. If CERC-fmla used, the K1 is calculated from Swart formulation from the D50 in m. (D50 must be between 0.1 and 1.5 mm to give a valid K1) • K1: The value of the K1 parameter to be used for the CERC formula will be given
CTLABEL2	`CtLabel2`	Integer	Label for parameter K1 and K2, so that they will be given as a constant value if 0 or as NSECT1 values if $\neq 0$
D50CONST or K1CONST (if CTLABEL=0) D50(M) or K1(M) M=1,NNODE (otherwise)	`K1, D50`	Real	Value of D50 or K1 (depending on value of D50K1): • Constant for each beach section (if CTLABEL2=0) • Different for each beach section (otherwise)
K2	`K2`	Real	Value of the parameter K2 (depending on value of D50K1): • Constant for each beach section (if CTLABEL2=0) • Different for each beach section (otherwise)

NAME	UnaLinea NAME	TYPE	EXPLANATION
			This value needs to be present, even if it does not exist (in the case when using Soulsby-Damgaard formulation). Just set it up as 0.
TANBETACONST	`TanBetaConst`	Real	Value of the tangent of the beach (constant for all sections)
CTLABEL3	`CtLabel3`	Integer	Label for parameter dx, so that it will be given as a constant value if 0 or as NSECT values if $\neq 0$
DXCONST (if CTLABEL=0) DX1(M) M=1,NSECT1 (otherwise)	`Dx`	Real	Value of the beach section spacing DX in m: • Constant for each beach section (if CTLABEL3=0) • Different for each beach section (otherwise)
PLXTON	`Plxton`	Real	Angle of the seaward facing beach baseline relative to North (in degrees, positive anticlockwise)
BAN1 BAN2	`Ban1Deg, Ban2Deg`	Real	These angles (in degrees and always positive) give the extent of the angles to be considered. For ease of use they are considered from the shoreline baseline normal, and therefore will normally be 45 and 45. Figure below gives an explanation of these angles.
NWPOINTS	`nWPoints`	Integer	Number of points at which wave data is inputted
XWPOINT(N),N=1, NWPOINTS	`XWPoint`	Real	X position for the wave points (in increasing order)
NAMWAVF(N),N=1 ,NWPOINTS	`NameWaveFile`	Character	Names of wave files
SHIFT(N),N=1,NW POINTS	`Shift`	Real	Angle shift that will be applied to the supplied offshore angles of the wave points.
CDEP	`DepWav`	Real	Depth at which the wave data is given (the same for all wave points)
CTLABEL4	`CtLabel4`	Integer	Label for the depth of closure, so that it will be given as a constant value if 0 or as NNODE values if $\neq 0$
BHTCONST (if CTLABEL=0) BHT(M)M=1, NNODE (otherwise)	`BHtDC`	Real	Depth of closure in m (seaward of which the beach profile is assumed not to change). It is always positive: • Constant for each beach section (if CTLABEL4=0) • Different for each beach section (otherwise)

NAME	UnaLinea NAME	TYPE	EXPLANATION
			This value DOES NOT include the berm height.
CTLABEL5	`CtLabel5`	Integer	Label for the berm height, so that it will be given as a constant value if 0 or as NSECT1 values if $\neq 0$
BERMCONST (if CTLABEL=0) BERM(M)M=1, NNODE (otherwise)	`Berm`	Real	Height of berm (m) above contour defined by YPOS • Constant for each beach section (if CTLABEL5=0) • Different for each beach section (otherwise)
NBOUND	`NBound`	Integer	Number of groynes
NODE	`NODE`	Integer	Node at which the groyne/fixed point is to be implemented
LIM	`IBound`	Integer	Type of groyne/fixed point, so that: • 1: long groyne • 3: open boundary • 4: short groyne
LEN	`YLim`	Real	In case of a short groyne (LIM=4), the upper limit to beach ordinate
LOSSREN	`LossRen`	Real	A number between 0 and 1 that gives the fraction of nourishment material lost .
LOSSRIV	`LossRiv`	Real	A number between 0 and 1 that gives the fraction of river material lost offshore.
LABEXTSOURCE	`LabExtSource`	Character*3	Label to indicate whether there is an external file with source information or not. Options: • NES (No External Source) • YES (Yes External Source)
NEXTSOURCES	`NExtSources`	Integer	Number of sources with external information. Only needed if LABEXTSOURCE= 'YES' The loss is not applied to this external source.
NAMEXTSOUR(N), N=1,NEXTSOURCES	`NameExtSour`	Character	Name of external sources files. Only needed if LABEXTSOURCE= 'YES'
POSEXTSOURCES(NEXTSOURCES)	`PosExtSources`	Integer	Grid points at which the external source is applied. Only needed if LABEXTSOURCE= 'YES'
NFEEDS,NEXTFEEDS	`NFeeds, NExtFeeds`	Integer	NFEEDS - Number of sections that have sources (renourisments/mining) to be inputted

NAME	UnaLinea NAME	TYPE	EXPLANATION
			NEXTFFEDS - Number of sections that have sources (renourisments/mining) with a input file (The loss is not applied to this external source)
POSFEED(NFEEDS)	j	Integer	Section at which source is to be inputted.
IFLAG(NFEED)	IFLAG	Integer	Flag to determine whether the feed discharge is constant (iflag=0) or variable (otherwise) for that section
CONSTFEEDS	ConstFeed	Real	Constant value for the renourishment at each section with a constant feed. It can be positive or negative, depending whether it is nourishment or mining. It is given in m3 per timestep
NVALUES	NValues	Integer	Number of values to be specified for the feed at different timesteps at the section specified (when iflag $\neq$ 0)
TS(NVALUES)	TS	Integer	Timestep for the feed to be specified
FEEDVAL(NVALUES)	FeedVal	Real	Value for the feed
POSEXTFFEDS(NEXTFEEDS)	PosExtFeeds	Integer	Section at which source is to be inputted for feeds with input files
NAMEXTFEEDS(NEXTFEEDS)	NameExtFeeds	Character	Name of external feed files. Only needed if NEXTFEEDS is greater than 0 . (used for feed file) (The loss is not applied to this external source)
NRIVER	NRiver	Integer	Number of sections that a have river source to be inputted
J(NRIVER)	j	Integer	Section at which river is to be inputted
IFLAG(NRIVER)	IFlagRiv	Integer	Flag to determine whether the river discharge is constant (iflag=0) or variable (otherwise) for that section
CONSTRIVER	ConstRiver	Real	Constant value for the river discharge at the section specified (when itflag=0). It should only be positive and it is given in m3 per timestep
NVALUES	NValuesRiv	Integer	Number of values to be specified for the river discharge at different timesteps at the section specified (when iflag$\neq$0)
TS(NVALUES)	TS	Integer	Timestep for the river discharge to be specified
RIVERVAL(NVALU	RiverVal	Real	Value for the river

NAME	UnaLinea NAME	TYPE	EXPLANATION
ES)			
CLIFFFLAG	`CliffFlag`	Integer	1 if cliffs are to be included, 0 otherwise
CLIFFFILENAME	`CliffFileName`	Character	Name of the cliff steering file, including extension and path if not in same folder
BEDLOADFLAG	`BedFlag`	Integer	Number of bedload files to be included
BEDLOADFILENAME(BEDFLAG)	`NameBedload`	Character	Names of bedload files
LASTNODES(BEDFLAG+1)	`lastnodes`	Integer	This is a list of nodes between which the bedload will be applied

Source: HR Wallingford

```
Example of UnaLinea  !title
LO         !Type of output: short(SO) or long(LO)
TS         !TS:timeseries or AV: averaged
EE         !numerical scheme
CE         !longs transport fmla
OO         !boundaries
3600,  1   !timestep and NDT
1000,  5   !ntimesteps    noutputs
8766       !number of timessteps in a year
5,6,7,10,1000  !outputtime(i)
524        !nsect
0          !ctlabel (0 if constant; anything else for varying)
1000.      !below initial  shoreline position
K1
0          !ctlabel (0 if constant; anything else for varying)
0.5, 1.08 !k1, K2
0.05       !tanbetaconst
0          !ctlabel (0 if constant; anything else for varying)
100        !dx
90.        !plxton
45. 45.    !banned angles
3          !nwpoints
100, 30000,10000 !xwpoint(i)
wLOlu1hTS.dat, wLOlu1TS2.dat, wLOluTS2.dat !wave data file
0.0, 0.0, 0.0    !shift for wave data
5.2        !ddep
0          !ctlabel (0 if constant; anything else for varying)
5.         !Dc
0          !ctlabel (0 if constant; anything else for varying)
1          !Sw
1          !NBounds (BC at both ends are already accounted for)
100,1,0
```

```
0.05       !losses for renourishment
0.04       !losses for river
NES        !External file with source: NES (No) YES (Yes)
0,0        !NFeeds
0          !NRivers
1          !Cliff flag
Cliff.clf  !Cliff steering file
3          ! Bedflag
BL1.txt,BL2.txt,BL3.txt !names of bedload files
1,200,400,524 !Lastnodes – bedload BL1 will be applied between nodes 1 and
200, BL2 between nodes 200 and 400 etc.
```

Frame 4.1: Example of a control file (In blue, comments)

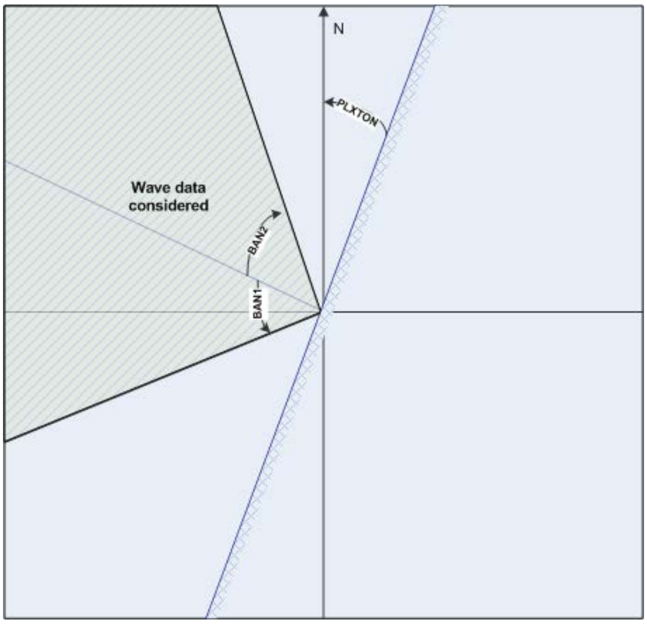

Figure 4.1: Description of angles involved: PLXTON, BAN1 and BAN2

Source: HR Wallingford

4.2.1. Sources

There are several ways of inputting a source into the model:

- External sources
- Feeds
- Rivers

The following table gives a summary of the capabilities of each:

Table 4.2: Types of sources and capabilities of each type

Type	Type of input	Loss applied	Variability in time
External sources	With external files	None	Given in the external file
Feeds	With external files	None	Given in the external file
	Internally	LOSSREN	Can be constant or at given timesteps
Rivers	Internally	LOSSRIV	Can be constant or at given timesteps

A commented example of a steering file with sources is given in Appendix A.

External source time series file

This file is not compulsory and it is expected to be needed when UnaLinea is run in a probabilistic way. The file contains a timeseries of values to be inputted as a source at the grid point specified in the steering file. The timespan covered in this file, as well as the timestep, must be the same as those in the wavefile(s). The file does not have any headers and is simply a column of values. An example of this file type is given in Frame 4.4.

```
0.01
0.0
0.0
0.02
…
0.01
```

Frame 4.2: Example of an external source file format

4.2.2. Bedload

Where data are available concerning the volume of bedload transport at the offshore boundary of the model, these can be included in UnaLinea as a source/sink term using the bedload files. Different bedload files may be specified for different stretches of coastline. The bedload file requires one line for each model timestep, containing the volume in m^3/time step for each bedload section.

```
108.4680515
108.4680515
108.4680515
….
130.5307982
```

Frame 4.3: Example of an bedload file

4.2.3. Boundary Conditions

There are five types of boundary conditions, as specified in the table below.

Table 4.3: Types of boundary conditions

Type	Description
1	Long groyne
2	Stable point (not operational)
3	Open boundary
4	Short groyne of a given LEN
5	Partial with a given PARTC

Source: HR Wallingford

4.3. Wave Files

UnaLinea needs at least one wave file. However, in the case that several wave points are necessary for the modelling several wave files can be used. The number of wave points is given in the control file as NWPOINTS. For each wave point, a name for a wavefile (NAMWAVF) needs to be given when running the model. The name of the wave files has a maximum of 13 characters.

All wave files must have been derived at the same depth, which would be given in the control file as CDEP. Also, it is assumed that all waves files cover the same timespan and have the same number of data.

There are two types of wave files and this is also specified in the control file. The variable OPTION in the control file has two options:

- TS: the wave file is given as a time series
- AV: the wave file is given as a time series of averaged values.

The format of these two files is explained below.

Time series format

Its header consists of four lines that are read and discarded by UnaLinea. The wave data is a series of dates followed by the wave height, period and direction (relative to North). An example of this file type is given in the Frame below.

```
Wave Climate for Example
Derived by Model A
Yr  Mo  Da  Mi  Hs  Tz  Dir
               (m)  (s) (dN)
86  11  1   30  1.5 6.0  10.0
86  11  2   30  1.4 6.1  10.0
86  11  3   30  1.4 6.5  10.0
…
99  99  99  99  0.0  0.0  0.0
```

Frame 4.4: Example of a wave file in time series format

Time series of averaged values format

Its header consists of one line that is read and discarded by UnaLinea. The wave data is a series of an integer number followed by the wave height, period and direction (relative to North) and a number between 0 and 1 that gives the percentage of no-calms within that timestep. An example of this file type is given in Frame 4.5.

```
Averaged conditions
   1  1.5  6.0  10.0  0.2
   2  1.5  6.0  10.0  0.5
      …
  99  99   99   0.0  0.0
```

Frame 4.5: Example of a wave file in averaged values format

4.4. The Cliff Steering File

The Una Linea cliff module extends the one line model landward to include elements of the backshore including cliffs (and seawalls). The cliff major cliff parameters are the cliff top position (cpos), cliff height (cheight) and the cliff slope (cslope). The cliff behaviour is based on a set of simple rules so that if the top of the beach retreats exposing the cliff toe, the cliff toe will also retreat. Retreat of the cliff toe releases sediment onto the beach and increases cslope.

If cslope is between the cliff failure angles, A1 and A2, the probability of cliff failure is calculated (this is assumed to be linearly distributed so that it is 0 at angles smaller than A1 and 1 at angles greater than A2). A random number is generated by the cliff module and if this is less than the probability of cliff failure, the cliff fails. The new cliff slope is selected randomly between the specified relaxation angles, B1 and B2. The cliff top retreats and sediment is released onto the beach.

The format of the cliff steering file is described in Table 4.4 and an example is provided in Frame 4.6.

Table 4.4: Description of inputs required for the cliff steering file

Name	Type	Explanation
CTLABEL	Integer	Label for the cliff top position (cpos), so that it will be given as a constant value if 0 or as NSECT values if ≠0
CPOSCONST (if CTLABEL=0) CPOS1(M)M=1,NSECT (otherwise)	Real	Initial cliff toe position: Constant for each cliff section (if CTLABEL=0) Different for each cliff section (otherwise)
CTLABEL	Integer	Label for the initial cliff slope (cslope), so that it will be given as a constant value if 0 or as NSECT values if ≠0
CSLOPECONST (if CTLABEL=0) CSLOPE1(M)M=1,NSECT (otherwise)	Real	Initial cliff slope: Constant for each cliff section (if CTLABEL=0) Different for each cliff section (otherwise)
CTLABEL	Integer	Label for the initial cliff slope (cheight), so that it will be given as a constant value if 0 or as NSECT1 values if ≠0

Name	Type	Explanation
CHEIGHTCONST (if CTLABEL=0) CHEIGHT1(M)M=1,NSECT1 (otherwise)	Real	Initial cliff height: Constant for each cliff section (if CTLABEL=0) Different for each cliff section (otherwise)
CTLABEL	Integer	Label for the initial cliff slope (cslope), so that it will be given as a constant value if 0 or as NSECT1 values if $\neq 0$
BACKSHORECONST (if CTLABEL=0) BACKSHORE1(M)M=1,NSECT1 (otherwise)	Integer	Denotes backshore type: 0 – no backshore 1 – cliff 2 – seawall with: Constant for each cliff section (if CTLABEL=0) Different for each cliff section (otherwise)
A1, A2	Integers	Angles between which the cliff may fail (degrees): - A1 is the minimum angle at which failure becomes possible - A2 is the maximum value, which if exceeded, cliff failure is certain
B1, B2	Integers	Angles between which the cliff will relax to in the event of failure (degrees): - B1 is the minimum relaxation angle - B2 is the maximum relaxation value

```
1      !flag for constant values (0), else nsect values (1)//cpos (below)
3950.0 3894.6 3839.1 3783.7 3728.3 3674.4 3641.6 3608.8 3573.2 3535.2 3497.2
3442.8 3342.6 3333.8 3326.4 3300.3 3261.9 3223.6 3179.7 3124.3 3069.6 3029.8
…
1923.1 1931.6 1940.2 1948.7 1957.2 1965.7 1974.3 1982.8 1991.3 1999.8 2008.4
2016.9 2025.4 2033.9 2042.5 2051.0 2059.5 2074.0 2088.5 2103.0
0      !flag for constant values (0) of cslope, else nsect values (1)
60     !cslope
0      !flag for constant values (0) of cheight, else nsect values (1)
4      !cheight
0      !flag for constant values (0) of backshore type, else nsect values (1)
1      !Backshore type (0=none, 1= cliff, 2=seawall (not coded yet))
65,100       !a1,a2 – failure angles
45,70        !b1,b2 – relaxation angles
```

Frame 4.6: Example of a cliff control file (In blue, comments)

5. Output Files

Depending on the output required (and specified in the steering file as TYPE_OUTPUT), the number of output files and their content will vary. There are two options:

- Short Output: the model only produces one output file; and
- Long Output: the model produces 7 different output files.

The output files have been designed so that they are easily analysed as a spreadsheet, with the aid of charts produced by the user from the variable values.

The output files are described in the sections below.

5.1. Short Output

One output file is created, named Stem.RESULT (Stem being the name of the Steering File without the .STE). This file contains columns of data in which the section number, x position and initial y position of the shoreline is given. The remaining columns contain the positions of the shoreline at the last output required as final position, minimum position, maximum position and average position. These last three values are absolute values (absolute in the sense that they are statistics from the beginning of the run, unlike in the long output file Stem.RES where the statistics are at each output timestep). An example of this file is given in Frame 5.1.

```
Steering File Used:  AS1_avg.STE
Wave data File Used  Results file       : AS1_avg.SRESU
Comment in Steering File :
Steering File Used:  HolTest2.STE
Wave data File Used: wHold3h.dat
No External source data file used   Results file       : HolTest2.RESU
Comment in Steering File :  Holderness Conv Tests Test1
Date:  30/06/2010
Time:  21:00:45
SECTION        X      Yinit     Yfinal    Ymin      Ymax      Yavge
 1        0.00   100.00   100.00   100.00   100.00   100.00
 2      100.00   100.00   100.00   100.00   100.00   100.00
 3      200.00   100.00   100.00   100.00   100.00   100.00
 4      300.00   100.00   100.00   100.00   100.01   100.00
 5      400.00   100.00   100.01   100.00   100.01   100.00
 6      500.00   100.00   100.01   100.00   100.01   100.00
…
510   50900.00   100.00   100.00   100.00   100.00   100.00
511   51000.00   100.00   100.00   100.00   100.00   100.00
512   51100.00   100.00   100.00   100.00   100.00   100.00
Final Date:  30/06/2010
Final Time:  21:00:59
```

Frame 5.1: Example of a RESULT file

5.2. Long Output

When a longer output is preferred, the model will create eleven output files:

- LOG file: File containing logging information;
- RESU: File with the initial data and a summary of the data in control file;
- RESQ: File with transport results for each time in which an output is required;
- RESSH: File with shoreline positions results for each time in which an output is required;
- RESMAX: File with maximum shoreline positions results for each time in which an output is required;
- RESMIN: File with minimum shoreline positions results for each time in which an output is required;
- RESAVGE: File with average shoreline positions results for each time in which an output is required;
- RESTCLF: File with the cliff toe position for each time in which an output is required;
- RESYCLF: File with the cliff Y position (cliff top) for each time in which an output is required;
- RESBL: File with the beach level in front of the sea wall for each time in which an output is required; and
- STA: File containing information on stability parameters.

These are described below.

5.2.1. RESU file

This file contains information from the steering file in a more user friendly manner. The data for the initial model setup with and without the extension is also given. Information on the shoreline positions at each output time (shoreline position at the end of the output time, together with average, minimum and maximum shoreline position relative to that output time) is given in this file as the model runs (allowing the user to check results). Also, at the end of the file, the values of the absolute maxima, minima and average positions is given. Frame 5.2 contains an example of a RESU file.

```
Steering File Used:  HolTest3.STE
Wave data File Used: wHold3h.dat
No External source data file used
Results file        : HolTest3.RESU
Comment in Steering File :
Holderness Conv Tests Test1
Date:  30/06/2010
Time:  21:43:35
        INITIAL DATA
Timestep:                                  10800(     )
Number of outputs:     3 at     2916     5832    37908
Angle between N and x-axis, PLXTON:  119.00
Angles considered from:               344.00 to   74.00
Offshore wave depth:                   5.20
Number of wave points:                 1  at the following sections:       3
Boundaries: Open at LHS boundary and RHS boundary
Longshore transport formula: CERC
Number of groynes modelled:            1
Groyne at section and x:             125     12350.00   Long groyne
```

```
Numerical method used:EE

INITIAL MODEL SETUP DATA

SECTION          LENGTH           Dc               X             Y
K1              shift
1    100.0000   5.0000   -5000.0000   100.0000  0.2300    0.0000  Stable point
2    100.0000   5.0000   -4900.0000   100.0000  0.2300    0.0000
3    100.0000   5.0000   -4800.0000   100.0000  0.2300    0.0000
4    100.0000   5.0000   -4700.0000   100.0000  0.2300    0.0000
  …
Results for time:     31492800 seconds (         2916       )
SECTION        X             Y             Ymin           Ymax           Yavge
1          0.0000      100.0000      100.0000      100.0000      100.0000
2        100.0000      100.0000      100.0000      100.0000      100.0000
3        200.0000      100.0000      100.0000      100.0000      100.0000
4        300.0000      100.0000      100.0000      100.0000      100.0000
 …
Absolute maxima, minima and average at the end of the run at time409406400
SECTION        X             Y             Ymin           Ymax           Yavge
1            0.00          100.01        100.00         100.01         100.00
2          100.00          100.01        100.00         100.01         100.00
3          200.00          100.01        100.00         100.01         100.00
4          300.00          100.01        100.00         100.01         100.00
…
511        51000.00        100.00        100.00         100.00         100.00
512        51100.00        100.00        100.00         100.00         100.00
Final Date:  30/06/2010 Final Time:  21:43:54
```

Frame 5.2: Example of a RESU file

5.2.2. RESQ file

This file gives the longshore transport data at each output time required. The transport data is given in terms of net, right, left and gross transport for that timestep. Frame 5.3 give an example of such file.

```
  Steering File Used:  AS1_avg.STE    Wave data File Used:   Results file
: AS1_avg.SRESU  Comment in Steering File :  UnaLinea11 AS1 10March10
timestep=week;dx=200;                                  Date:  26/05/2010
Time:  13:59 Results for time:         50400 seconds (         14 hour )Q in
m/timestep   SECTION        X             Qnet           Qright         Qleft
Qgross          1       -100.0000     -1710.1461      0.0000      -1710.1461
-1710.1461      2        100.0000     -1710.1461      0.0000      -1710.1461
-1710.1461…     300     59700.0000     -1710.1461      0.0000      -1710.1461
-1710.1461     301     59900.0000     -1710.1461      0.0000      -1710.1461
-1710.1461 Results for time:       100800 seconds (         28 hour )Q in
```

```
m/timestep   SECTION       X             Qnet           Qright         Qleft
Qgross          1      -100.0000     -1710.1461         0.0000      -1710.1461
-1710.1461      2       100.0000     -1710.1461         0.0000      -1710.1461
-1710.1461…    300     59700.0000     -1710.1461         0.0000      -1710.1461
-1710.1461     301     59900.0000     -1710.1461         0.0000      -1710.1461
-1710.1461 Results for time:      3600000 seconds (        1000 hour )Q in
m/timestep   SECTION       X             Qnet           Qright         Qleft
Qgross          1      -100.0000    -118731.5116         0.0000    -118731.5116
-118731.5116      2        100.0000     -118726.2175         0.0000    -
118726.2175   -118726.2175…    300     59700.0000   -118733.0015
0.0000   -118733.0015   -118733.0015    301     59900.0000   -118733.0025
0.0000   -118733.0025   -118733.0025 Results for time:      5187600 seconds (
1441 hour )Q in m/timestep   SECTION       X             Qnet           Qright
Qleft          Qgross          1      -100.0000    -53850.8720         0.0000
-53850.8720    -53850.8720      2        100.0000    -53808.0021         0.0000
-53808.0021    -53808.0021 …    300     59700.0000    -53869.5317
0.0000     -53869.5317     -53869.5317    301     59900.0000     -53869.5878
0.0000     -53869.5878     -53869.5878
```

Frame 5.3: Example of a RESQ file

5.2.3. RESSH file

This file gives the shoreline position, both at the beginning of the run and at each output time required. This output file is only created at the end of the run (so if the run crashes it will not be created). Frame 5.4 contains an example of this file.

```
Steering File Used:  HolTest2.STE   Wave data File Used: wHold3h.dat    No
External source data file used   Results file       : HolTest2.RESU  Comment
in Steering File :  Holderness Conv Tests Test1
Date:  30/06/2010 Time:  21:34:36                               Y      SECTION
X        Y        2916     5832    37908       1       0.00   100.00   100.00
100.00   100.00       2     100.00   100.00   100.00   100.00   100.00       3
200.00   100.00   100.00   100.00   100.00…     511   51000.00   100.00
100.00   100.00   100.00     512   51100.00   100.00   100.00   100.00
100.00 Final Date:  30/06/2010 Final Time:  21:34:52
```

Frame 5.4: Example of a RESSH file

5.2.4. RESMAX file

This file gives the maximum shoreline position at each output time required as well as the shoreline position at the beginning of the run. This output file is only created at the end of the run (so if the run crashes it will not be created). Frame 5.5 contains an example of this file.

```
    Steering File Used:  HolTest2.STE   Wave data File Used: wHold3h.dat    No
External source data file used   Results file       : HolTest2.RESU  Comment
in Steering File :  Holderness Conv Tests Test1
Date:  30/06/2010 Time:  21:34:36                                Ymax   SECTION
X        Y        2916     5832    37908       1       0.00   100.00   100.00
100.00   100.00      2     100.00   100.00   100.00   100.00   100.00       3
200.00   100.00   100.00   100.00   100.00…     511   51000.00   100.00
100.00   100.00   100.00     512   51100.00   100.00   100.00   100.00
100.00 Final Date:  30/06/2010 Final Time:  21:34:52
```

Frame 5.5: Example of a RESMAX file

5.2.5. RESMIN file

This file gives the minimum shoreline position at each output time required as well as the shoreline position at the beginning of the run. This output file is only created at the end of the run (so if the run crashes it will not be created). The format is identical to the RESMAX file.

5.2.6. RESAVGE file

This file gives the average shoreline position at each output time required as well as the shoreline position at the beginning of the run. This output file is only created at the end of the run (so if the run crashes it will not be created). The format is identical to the RESMAX file.

5.2.7. RESTCLF File

This files gives the cliff toe position at each output time required.

5.2.8. RESYCLF File

This files gives the cliff top position at each output time required.

5.2.9. RESBL File

This files gives the beach level in front of a seawall at each output time required.

5.2.10. LOG file

This file contains useful information for debugging purposes

5.2.11. STA file

This file contains information on the CFL (Courant-Friedrichs-Lewy number) and the Von Neuman criteria in order to check numerical stability.

6. Constants

There are a series of constants used within UnaLinea which are not changed within the Steering File. Their values can be changed in the include file **consts.inc**, but the program will need recompiling to take them into account. The constants and their default values are:

- Minimum wave height that it is considered in the model, **Hmin=0.3m**,
- Maximum wave height that it is considered in the model, **Hmax=6m**,
- Minimum wave period that it is considered in the model, **Tmin=3s**,
- Maximum wave period that it is considered in the model, **Tmax=12s**,
- Wave breaking criterion, **BrRat = 1/0.78**,
- Water viscosity, **Visc = 0.00000136** $m^2 s^{-1}$, (Used only in Soulsby and Daamgard sediment transport formulation)
- Sediment density, **RhoSed= 2650kg/m3**,
- Water density, **RhoW = 1027 kg/m3** and
- **Porosity = 0.6**

7. References

HR Wallingford (2016). UnaLinea Reference Manual (Version 01). HR Wallingford report DDS1202-RT008-R01. Available from http://www.channelcoast.org/iCOASST/UNALINEA/

Stripling, S. and Panzeri, M.C. (2009). Modelling shoreline evolution to enhance flood risk assessment" Proceedings of the ICE - Maritime Engineering, 162(3) 137–144.

Stripling S., Panzeri M, Blanco B. et al. (2011). "Broad scale integration of coastal flood and erosion risk models." FRMRC2 Work Package 4.3 Science Report. http://web.sbe.hw.ac.uk/frmrc/ (accessed 31/05/2016).

Appendices

A. Steering File with sources

A commented steering file with sources is given below. The three different types of sources are specified, so that in green there are the external sources, in blue the Feeds and in red the rivers.

```
…
YES                 There are external sources
2                   There are 2 external sources
ren1.dat, ren2.dat  These are the files containing a timeseries of the
                    values for the external sources (in m3/timestep)
4,5                 The first external source is applied in j=4 and the
                    second one in j=5
2,1   !NFeeds       There are 2 feeds and 1 external feed
150,0               First feed is at j=150 and it is constant
 80                 The value of the first feed (at j=150) is 80m3/timestep
151,1               Second feed is at j=151 and it is variable
 4                  There are 4 values for the second feed (at j=151)
 5,90               At timestep=5, the value of the second feed (at j=151)
                    is 90m3/timestep
 6,90               At timestep=6, the value of the second feed (at j=151)
                    is 90m3/timestep
 7,90               At timestep=7, the value of the second feed (at j=151)
                    is 90m3/timestep
 8,90               At timestep=8, the value of the second feed (at j=151)
                    is 90m3/timestep
feed1.dat,          These are the files containing a timeseries of the
feed2.dat           values for the external feeds (in m3/timestep)
40,50               The first feed is applied in j=40 and the second one in
                    j=50
3     !NRivers      There are 3 rives
2,1                 First river is at j=2 and it is variable
 1                  There is 1 values for first river (at j=2)
 7,200              At timestep=7, the value of the first river (at j=2) is
                    200m3/timestep
10,1                Second river is at j=10 and it is variable
 3                  There are 3 values for first river (at j=10)
 5,5                At timestep=5, the value of the second river (at j=10)
                    is 5m3/timestep
 7,7                At timestep=7, the value of the second river (at j=10)
                    is 7m3/timestep
 8,8                At timestep=8, the value of the second river (at j=10)
                    is 8m3/timestep
100,1               Third river is at j=100 and it is variable
```

```
2                   There are 2 values for third river (at j=100)
5,10                At timestep=5, the value of the second river (at j=100)
                    is 10m3/timestep
200,20              At timestep=200, the value of the second river (at
                    j=100) is 20m3/timestep
```

B. Code structure

A list of subroutines, in alphabetical order, together with a description of what they do, is given in Table B.1. A flow chart is presented in Figure B.1.

Table B.1: List of subroutines

Subroutine Name	Description
BoundaryConds	Imposes the boundary conditions and accumulates net drift, as well as right and left drift for each node.
CalcDhdx	It calculates the longshore variation of breaking wave height.
CheckWaveInterval	Check if waves are between the interval between Angle1 and Angle2 to be considered and give it a label AngleLabel = 1. Otherwise, AngleLabel = 0.
Cliff	Calculate cliff retreat and volumes of sediment released to the beach
ConservationEq	This subroutine sorts out all of the sources/sinks into a Source variable, it resolves the conservation equation depending on the resolution method and calculates the change in beach position at each section. It calculates the maximum and minimum shoreline position (absolute or not) and it writes out the stability condition results.
Error	This subroutine manages the errors within the Steering file, so that it writes a message out and stops the program running.
FromAngIn	This subroutine calculates the wave direction angle inshore and the angle from the beach to baseline using the angle of breaking wave direction to beach normal.
GetAngOff	This subroutine calculates the angle of wave direction to beach normal
GetWaveData	This subroutine calculates the offshore wave data at each node from the information supplied at the wave points. It also puts a limit to the big waves and applies a shift to the wave direction, if specified in steering file.
InitializeZeroVbles	This subroutine initializes variables as required at the beginning of the run.
LongSedTransp	This subroutine calculates the cross-shore integrated longshore sediment transport rate in m**3/s, using Soulsby's formulation (bed-load transport only).
OpenFeedFiles	This subroutine opens the files containing the sources information.
OpenFeedFiles2	This subroutine opens the files containing the feeds information.
OpenFiles	This subroutine opens the results files, both for the long and the short output.
OpenWaveFiles	This subroutine opens the waves files, one per wave point, and reads the headers (One line if it is timeseries data or four lines if it is averaged data).
RandomSeed	Creates random seed for the cliff routine
ReadBedloadFiles	Opens the bedload files

Subroutine Name	Description
ReadSteeringFilePart1	This subroutine reads the first part of the steering file so that it knows the size of the arrays.
ReadSteeringFilePart2	This subroutine reads the second part of the steering file.
ReadSteeringFilePart3	This subroutine reads the third and last part of the steering file.
RefSug	This subroutine carries out refraction of the wave conditions specified at depth dOff, assuming the beach contours are locally parallel to the beach line up until the breaker line.
SedTransport	This subroutines calculates the sediment trasport for a Q point in the model. Depending on the formulation chosen (via the Sedfmla parameter in the steering file it will use the CERC formulation for total load or Soulsby and Damgard formulation (for bedload only; especially for shingle).
WavCel	This subroutine calculates the wave celerity according to Hunt's (1979) 9th order solution.
Write_cliff	Write cliff result files.
WriteInitialData	This subroutine writes out the input data in the results File for future reference.
WriteResults	This subroutine writes the results into the results file every output time.

Source: HR Wallingford

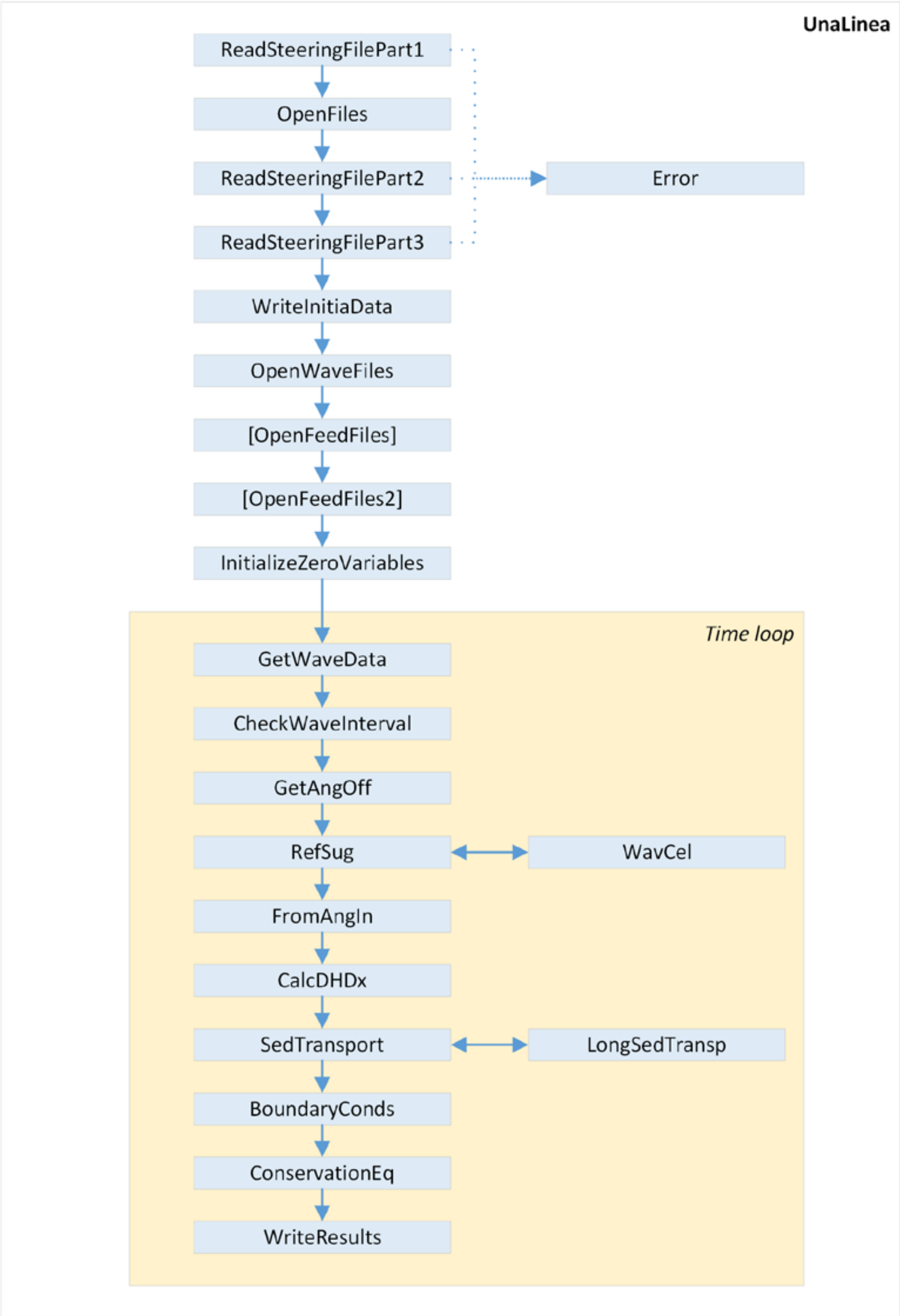

Figure B.1: UnaLinea flowchart

Source: *HR Wallingford*

C. File channels

The extensions, channel name and number are given below for each file.

Table C.1: Channels' numbers, names and contents

Channel number (in units.inc)	Naming convention	File contents	Channel name (in units.inc)
1	Stem.STE	Control file	LunSte
2	Stem.RESU	File with the initial data, summary of the data in control file and run-time results	LunResu
10	Stem.LOG	Debugging information	LunLog
12	Stem.RESQ	File with transport results for the output times	LunResQ
14	Stem.RESSH	File with shoreline positions results for the output times	LunResSh
15	Stem.RESULT	Output information for the short output mode	LunResult
16	Stem.RESMAX	File with maxima shoreline positions results for the output times	LunResMax
17	Stem.RESMIN	File with minima shoreline positions results for the output times	LunResMin
18	Stem.RESAVGE	File with average shoreline positions results for the output times	LunResAvge
22	Cliff.CLF	Cliff steering file	LunClf
26	Stem.TRACK	Keeping track of where in the model it is	LunTrak
30-39			LunSource
40-49	name.DAT		LunWav
123	Stem.RESYCLF	Cliff top results file	LunResYClf
124	Stem.RESTCLF	Cliff toe results file	LunResTClf
125	Stem.RESBL	Beach level results file	LunResBl
250	Stem.STA	Contains info on stability parameters	LunSta
251	parameter.dat	Parameters value	uni
40000		Bed load file channels	LunBedload

Source: HR Wallingford

HR Wallingford is an independent engineering and environmental hydraulics organisation. We deliver practical solutions to the complex water-related challenges faced by our international clients. A dynamic research programme underpins all that we do and keeps us at the leading edge. Our unique mix of know-how, assets and facilities includes state of the art physical modelling laboratories, a full range of numerical modelling tools and, above all, enthusiastic people with world-renowned skills and expertise.

FS 516431
EMS 558310
OHS 595357

HR Wallingford, Howbery Park, Wallingford, Oxfordshire OX10 8BA, United Kingdom
tel +44 (0)1491 835381 fax +44 (0)1491 832233 email info@hrwallingford.com
www.hrwallingford.com