

Chapter 10

SPECIAL TOPICS

10.1 ERROR MESSAGES AND LOG FILES

Numerous checks are made in WAMIT for consistency of the data in the three input files. Appropriate error messages are displayed on the monitor to assist in correcting erroneous inputs. Output files containing warning and error messages are created after each execution of the subprograms POTEN and FORCE. **errorp.log** contains messages from POTEN and **errorf.log** from FORCE. The existing .LOG file, in the directory where the program runs, is overwritten with every new run. When the program runs successfully without any warning or error, the .LOG file contains two lines: a header line including the date and time when the program starts to run and a line indicating the completion of the run.

Error messages are associated with problems where the program execution is halted. Warning messages indicate that a possible error may occur, but under certain circumstances the results may be correct. Examples include failure of the convergence tests for various numerical integrations, which sometimes result from inappropriate choices of characteristic length scales or of overly conservative convergence tolerances. Another example is in the case of diffraction by a body with one or two planes of symmetry, where it is possible to compute the fluid pressures, velocities, and mean drift forces (Options 5-9) at certain heading angles without solving for all components of the diffraction potential; in this case the warning message states that the solution is non-physical, whereas at some heading angles the outputs will be correctly evaluated. For further discussion of this ‘shortcut’ see the discussion of MODE in Section 3.1, and related discussion in Section 3.3.

Error and warning messages generated during execution of the POTEN sub-program are output to the file **errorp.log**, and messages generated during execution of the FORCE sub-program are output to the file **errorf.log**. The same outputs are displayed on the monitor. Since some of these messages may be lost on the monitor due to scrolling of other outputs, a special warning message is generated at the end of the run to alert users when significant messages are contained in these two files.

Starting in V6.2, a new file **wamitlog.txt** is output during the run. This file is intended to provide an archival record of the run. The file includes the starting and ending time and

date for each sub-program, copies of the principal input files, and copies of the outputs in the files **errorp.log** and **errorf.log**. (Since the GDF input files are relatively long in the low-order method, and also in the higher-order method when the geometry is defined by low-order panels or B-splines (ILOWHI=0 or 1, respectively), in these cases only the first 10 lines of the GDF file are copied to the file **wamitlog.txt**. Another point to note is that the maximum width of lines of data is truncated to 80 characters in **wamitlog.txt**. The existing **wamitlog.txt** file, in the directory where the program runs, is overwritten with every new run.

When the input data in the Force Control Files for one or more bodies are in the Alternative form 1, as defined in Section 3.3, the nondimensional inertia matrix for each body is included in the file **wamitlog.txt**. This is particularly useful when the analysis of a body is first performed using Alternative form 1, and then changed for subsequent extensions to Alternative form 2, for example when external damping is imposed on an otherwise freely floating body. The normalizing factors for the nondimensional inertia matrix are the products of the fluid density, and appropriate powers of the characteristic length parameter ULEN. In preparing a force control file for Alternative form 2, as defined in Section 3.4, these normalizing factors must be included in the inputs EXMASS when these are derived from the nondimensional inertia matrix.

- Two particular warning messages which occur relatively frequently are the following:
 - ‘Number of subdivisions exceeds MAXSQR’
 - ‘WARNING – no convergence in momentum Dx/Dy/Mz for headings ...’

(In the latter warning Dx, Dy and Mz are used to identify the force or moment components which did not converge, and the heading angle is identified.)

When a warning message occurs indicating that the ‘Number of subdivisions exceeds MAXSQR’ for the Rankine integration over a higher-order panel, the Cartesian coordinates of the field point and source point are output to **wamitlog.txt** so that the user can more easily check if there is a singularity or inconsistency in the geometry definition in the vicinity of these points. Usually this indicates either an error in the geometry definition, or specification of a field point too close to the body surface.

The convergence test for the momentum drift force and moment is used to ensure accurate integration of the momentum flux in the far field. This integration is performed recursively, increasing the number of azimuthal integration points by factors of 2. Convergence is achieved when the difference between two successive iterations, in each of the three components Dx,Dy,Mz, is less than a prescribed tolerance TOL= 10^{-4} . If the component is less than one in absolute value, absolute differences are used, otherwise relative differences are used. The maximum number of iterations is controlled by the parameter MAXMIT in the CFG file (or by the default value MAXMIT=8). (When the default value is used the maximum number of integration ordinates is equal to $2^8 = 256$.) When necessary this parameter can be increased, but it should be noted that this increases the computational time exponentially when the mean drift force and moment are evaluated from Option 8.

Since the components of the mean drift force are nondimensionalized by ULEN, and the

moment by $ULEN^2$, convergence can also be affected by the choice of $ULEN$ in the GDF file. If $ULEN$ is much smaller than the characteristic length scale of the body it will not affect the convergence tests, and vice versa. Another point to note is that some components of the force and moment may be relatively small, and of little practical importance, whereas they may affect the convergence test. When in doubt about situations where the warning message occurs, it may be advisable to increase $MAXMIT$ by 1 or 2 units and compare the resulting outputs manually.

10.2 RESERVED FILE NAMES

- To avoid conflicting filenames, users are advised to reserve the extensions GDF, POT, FRC, SPL, P2F, OUT, PNL, FPT, PRE, MOD, HST, CSF, CSP, BPI, BPO, IDF, RAO, 1, 2, 3, 4, 5p, 5vx, 5vy, 5vz, 6, 7x, 7y, 7z, 8, 9, and 9c for WAMIT input and output.
- Other reserved filenames include CONFIG.WAM, FNAMES.WAM, BREAK.WAM, ERRORP.LOG, ERRORF.LOG, WAMITLOG.TXT, SCRATCH* (here *=A,B,C,...,O), as well as WAMIT.EXE, DEFMOD.FOR, DEFMOD.EXE, and for V6PC, the DLL files GEOMXACT.DLL and NEWMODES.DLL, and the additional Intel DLL files listed in Section 2.1 which are required to execute the program. Source-code users can modify the extensions by editing the appropriate assignments (in the source file `modulesc.f`, module `MAINC_ARRAYS`).

10.3 SCRATCH FILES

Two types of temporary scratch files are opened during execution of the subprogram POTEN. One group are opened formally as scratch files using the FORTRAN convention, with filenames which are assigned by the compiler. The second group are opened with the temporary filenames SCRATCHA, SCRATCHB, ..., SCRATCHO. All of these files are deleted prior to the end of the run, but if execution is interrupted by the user (or by power interruption to the system) some or all of the above scratch files may remain on the hard disk. In the latter case the user is advised to delete these files manually.

10.4 MEMORY REQUIREMENTS AND NUMBER OF UNKNOWNNS

The system memory requirements of WAMIT depend on the number of number of separate solutions for the velocity potential (and source strength, if ISOR=1), and the number of unknowns in each solution. This section is intended to provide the user with an understanding of these relationships, and of the manner in which WAMIT optimizes the solution of a given problem.

The number of simultaneous equations NEQN, equal to the number of unknowns, is used here to denote the dimension of the linear system solved for the determination of the radiation and diffraction velocity potentials on the body surface, and the corresponding source strengths when the option ISOR=1 is selected. The value of NEQN for each run is listed in the header of the OUT output file.

In the low-order method (ILOWHI=0) the number of equations NEQN is equal to the number of panels NPAN specified in the GDF input file(s), except when reflections about body planes of symmetry are required as explained below. If dipole panels are input, as explained in Section 5.4, NPAN is the sum of the conventional panels and dipole panels. If the irregular-frequency option is used, additional panels are required on the interior free surface inside of the body waterline, as described in Chapter 9, and the number of equations is equal to the total number of panels including the interior free surface, regardless of whether this is included in the GDF file and parameter NPAN explicitly (IRR=1), or the interior free surface panels are added by the program (IRR=2 or IRR=3).

In the higher-order method (ILOWHI=1) NEQN depends on the number of patches, panels, and the order of the B-splines:

$$\text{NEQN} = \sum_{i=1}^{N_P} (\text{NU}(i) + \text{KU}(i) - 1) \times (\text{NV}(i) + \text{KV}(i) - 1) \quad (10.1)$$

Here $N_P = \text{NPATCH}$ is the total number of patches, NU and NV are the numbers of panels on each patch, and KU and KV are the orders of the B-splines used to represent the solution. NU and NV are specified by the user in the CONFIG.WAM or SPL files, as explained in Sections 6.9-10. KU and KV are specified in the CONFIG.WAM or SPL files, or assigned the default value KU=KV=3. When reflections about planes of symmetry are

required, NPATCH is increased in the same manner as for NPAN.

If the parameter PANEL_SIZE in the CONFIG.WAM file is positive, NU and NV are determined internally in the program, as explained in Sections 3.7 and 6.9. In this case it is not generally possible to determine the exact values assigned to NU and NV on each patch. NU and NV are determined such that the estimated maximum physical dimension of each panel is less than PANEL_SIZE. The value of NEQN printed in the header of the OUT file can be used to verify estimates of NU and NV in this case.

When planes of geometric symmetry ($x = 0$ and/or $y = 0$ of the body coordinate system) do not coincide with the $X = 0$ and/or $Y = 0$ planes of the global coordinate system, WAMIT assumes no hydrodynamic symmetry with respect to those planes. In these cases the program reflects about the corresponding planes and increases the number of panels accordingly. The other case where planes of hydrodynamic symmetry are not utilized is the analysis of multiple interacting bodies (Chapter 7). In this case the number of unknowns NEQN is the total number of panels required to describe the entire bodies, and NLHS=1. The following list summarizes the cases where the number of unknowns is increased due to reflection:

1. When planes of geometric symmetry ($x = 0$ and/or $y = 0$) of the body coordinate system do not coincide with the $X = 0$ and/or $Y = 0$ planes of the global coordinate system, due to nonzero values of the input parameters XBODY(1), XBODY(2), XBODY(4) in the POT file or config.wam file. In this case the program assumes that there are no planes of hydrodynamic symmetry, and the body geometry is reflected about its specified planes of geometric symmetry.
2. In the multiple-body analysis described in Chapter 7 (NBODY Option), the same procedure applies as in (1) above. (In some special cases the configuration of multiple bodies may be symmetrical about one or two global planes, but the program does not check for that possibility. In such cases more economical computations can be performed using generalized modes.)
3. In the analysis of a body near vertical walls described in Chapter 5.3, the same procedure applies as in (1) above.

‘Total number of panels’ refers to the number of panels used to represent the entire body surface. WAMIT takes into account flow symmetries in setting up the linear systems, therefore the number of unknowns and total number of panels are different if body geometry symmetry planes are present. If 0, 1, or 2 planes of symmetry are specified, the total number of panels is equal to NPAN, $2 \times \text{NPAN}$, or $4 \times \text{NPAN}$, respectively. Since the computational burden of solving the linear system of equations is at least proportional to NEQN^2 , a substantial reduction in computational effort is achieved by imposing the planes of symmetry when this is physically appropriate.

Provision is made in WAMIT to specify a subset of modes to be analyzed separately, thus reducing the memory required and the run time. If the user anticipates the analysis of more than one mode it is more efficient to only once, for all modes of interest.

Since the issue of hydrodynamic symmetry is so important, it should be emphasized that the separate analysis of symmetric and antisymmetric modes of motion applies not only to the obvious cases of radiation modes, such as surge, sway, and heave, but also to the more complex solution of the diffraction problem, even in oblique waves. This is achieved in WAMIT by decomposing the complete diffraction (or scattering) solution as the sum of four separate components that are respectively even or odd functions of the horizontal coordinates. (Physically these can be interpreted as the solutions of problems where standing waves are incident upon the body.)

In considering memory requirements a distinction must be made between storage in RAM and on the hard disk. RAM requirements are affected by NEQN, and are different in the low-order and higher-order methods. Usually NEQN can be significantly smaller when the higher-order method is used, compared to the analysis of the same problem with the low-order method. But when ILOWHI=0, RAM can be used only for arrays which are linear in NEQN and thus it is possible to analyze structures with a large number of panels and unknowns, and to analyze simultaneously the different radiation and diffraction solutions of interest. When ILOWHI=1, one complex matrix of dimension NEQN, requiring $8 \times \text{NEQN}^2$ bytes, is stored in RAM. All other arrays stored in RAM are linear in NEQN.

The most important parameter which affects the space required on the hard disk is the number of equations, and unknowns, NEQN. For large values of NEQN the required amount of scratch storage on the hard disk is proportional to NEQN^2 . If the low-order method is used (ILOWHI=0), the number of bytes required for these arrays on the hard disk can be estimated from the equation

$$[(4 \times \text{NLHS}) \times (1 + 4 \times \text{ISOR}) \times (3 + \text{ILOG}) + 2] \text{NEQN}^2 - 8[\min(\text{MAXSCR}, \text{NEQN})]^2$$

where $\min(\text{MAXSCR}, \text{NEQN})$ denotes the minimum of these two parameters. If the higher-order method is used (ILOWHI=1), the number of bytes can be estimated from the equation

$$[(4 \times \text{NLHS}) \times (3 + \text{ILOG})] \text{NEQN}^2$$

In these equations the parameter NLHS is the number of left-hand-sides appropriate to the analysis. If all modes of motion are studied simultaneously, for a body with 0, 1, or 2 planes of symmetry specified, NLHS is equal to 1, 2, or 4, respectively. As an example, the truncated vertical cylinder described in the Chapter 5, which has 2 planes of symmetry and a total of 1024 panels, requires about 3.4 megabytes of scratch storage on the hard disk to analyze all modes of motion simultaneously, i.e. for the run described in the Chapter 5.

If the storage requirements of a run exceed the available disk space a system error will be encountered; in this event the user should either increase the available disk space or reduce the number of panels or solutions.

Subroutine ITRCC (the iterative solver) reads matrix elements in each iteration step from the hard-disk. The run time in this subroutine can be reduced substantially by storing some or all of elements in available RAM. The parameter MAXSCR defines the dimension of a square sub-array which can be stored temporarily in available RAM. Since the coefficients

of this array are complex, the corresponding storage requirement in RAM is $8 \cdot \text{MAXSCR}^2$ bytes, for systems which use 8 bytes for a single-precision complex number. Thus the parameter MAXSCR should be determined initially by estimating the size of excess RAM, after the program is loaded, and setting the largest integer value of MAXSCR such that $8 \cdot \text{MAXSCR}^2$ does not exceed the excess RAM which is available. If a value of MAXSCR is specified which is larger than NEQN, MAXSCR is reduced by the program at run time, and set equal to NEQN.

If the option ISOLVE= 1 is selected, to utilize the direct solver for the linear system of equations, the entire left-hand-side matrix must be stored in RAM. This can only be achieved if $\text{MAXSCR} \geq \text{NEQN}$. Otherwise, if ISOLVE= 1, an appropriate error message is generated at run time.

If the option ISOLVE> 1 is selected, to utilize the block iterative solver, the diagonal block matrices must be stored in RAM, each block at a time for the local LU decomposition. This can only be achieved if MAXSCR is equal to or greater than the dimension of the diagonal blocks. Otherwise the size of the diagonal blocks is reduced to MAXSCR internally.

10.5 MODIFYING DLL FILES

The files GEOMXACT.F and NEWMODES.F can be modified by users following the instructions in Sections 6.8 and 8.3. This makes it possible for users of the PC-executable code to develop special subroutines for the definitions of the body geometry and generalized modes, respectively, and to link these subroutines with WAMIT at runtime. Users of the source code have two alternative options: (1) maintain these two files separately from the remainder of the code, and use the DLL option if it is available with their compiler, or (2) compile and link the two source files directly with the remainder of the code.

WAMIT Version 6.4PC has been compiled with Intel Visual Fortran (Version 10.1), whereas earlier versions of WAMIT V6 were compiled with Compaq Visual Fortran. Either one of these two Fortran compilers can be used to compile modified versions of the files GEOMXACT.DLL and NEWMODES.DLL, using the following procedure:

- Open a new project 'geomxact' as a Fortran Dynamic Link Library
- Add geomxact.f to the project
- Build a release version of geomxact.dll
- Copy the new version of geomxact.dll to the working directory for WAMIT

It may be possible to use other FORTRAN compilers to make the DLL file, but certain conventions in calling subroutines must be consistent with those of Compaq Visual Fortran. Further information is provided in [23], Chapters 8 and 18.

10.6 AUXILIARY OUTPUT FILES FOR THE GEOMETRY

Starting in V6.2 options exist to generate three auxiliary geometry output files, with the names *gdf_PAN.DAT*, *gdf_PAT.DAT*, and *gdf_LOW.GDF*. (Here *gdf* is the filename of the GDF input file. For a run with $NBODY > 1$ the GDF filename for the first body ($N=1$) is used.) These files contain the Cartesian coordinates of panels or patches in formats suitable for perspective plotting, and the coordinates of new low-order panels which are derived from the input geometry. In all cases the coordinates are dimensional, and defined in the same units as specified in the input GDF file(s).

The data in *gdf_LOW.GDF* has the same definitions and format as a conventional low-order GDF file (Section 5.1). The coordinates of the panel vertices are defined with respect to the body coordinate system, corresponding to the original GDF inputs. If $NBODY > 1$ the file *gdf_LOW.GDF* represents the body identified as $N = 1$. If the original body panels are reflected by the program, the file *gdf_LOW.GDF* will include subdivided panels for the reflected body. (This will occur if $NBODY > 1$, if walls are present, or if the body is not symmetric with respect to the global coordinate system.) If $ILOWGDF=1$ and the body is reflected by the program, *gdf_LOW.GDF* will contain the original body panels (without subdivision) plus their images about the reflected planes of symmetry.

The data in *gdf_PAN.DAT* and *gdf_PAT.DAT* are defined with respect to the global coordinate system. In a WAMIT run with $NBODY > 1$ the data for all of the bodies are included. The figures in Sections A5 and A13 illustrate this feature.

These auxiliary files are described separately below for the low-order and higher-order methods.

In the low-order method ($ILOWHI=0$) the vertex coordinates of the body panels are stored in the output file *gdf_PAN.DAT* in a format suitable for input to plotting programs such as Tecplot. This facilitates the use of perspective plots to illustrate and check the GDF inputs. If ($ILOWHI=0$) the file *gdf_PAN.DAT* is output in the finite-element format FEPOINT, as specified for use with the Tecplot program. Examples of these plots are included in the Appendix for each low-order test run (Sections A1-A9). The integer parameter IPLTDAT in the file CONFIG.WAM is used to specify whether or not to generate this output file. In the default case ($IPLTDAT=0$) no file is generated. If ($IPLTDAT > 0$) the file is generated.

In the low-order method ($ILOWHI=0$) the optional output file *gdf_LOW.GDF* is controlled by the integer parameter ILOWGDF in the file CONFIG.WAM. If $ILOWGDF > 0$ the output file *gdf_LOW.GDF* is generated, with all of the original panels subdivided into $ILOWGDF \times ILOWGDF$ sub-divisions. The first three lines are copied from the GDF input file. The total number of sub-divided panels is included on line 4. This option can be used to increase the number of panels, and hence to increase the accuracy of the solution for the potential or source strength. However this subdivision scheme does not increase the accuracy of the geometric representation of the body, since the subdivided panels are coplanar with the original panels. Only one body can be subdivided in this manner.

In the higher-order method ($ILOWHI=1$) the optional output files *gdf_PAT.DAT* and *gdf_PAN.DAT* specify the vertex coordinates of both the patches and panels, as defined in

Chapter 6. If (ILOWHI=1) these files are in the ordered-list format POINT, as specified for use with the Tecplot program. Examples of these plots are included in the Appendix for each higher-order test run (Sections A11-A19). The integer parameter IPLTDAT in the file CONFIG.WAM is used to specify whether or not to generate these output files. In the default case (IPLTDAT=0) no files are generated. If (IPLTDAT>0) the files are generated. The number of panel subdivisions on each patch is determined by the parameters NU and NV in the SPL file, as explained in Section 6.11. If IPLTDAT=1 the data file *gdf_PAT.DAT* contains only the four vertices of each patch, and the file *gdf_PAN.DAT* contains only the four vertices of each panel. If IPLTDAT>1, each element is subdivided into IPLTDAT×IPLTDAT sub-elements. Subdivision of the elements is useful when perspective plots are constructed for bodies with curved boundaries of the patches and panels. When the curvature is large, IPLTDAT should be increased to give a more accurate plot. (IPLTDAT=5 is used for the plots shown in Sections A11-A19.)

In the higher-order method (ILOWHI=1) the optional output file *gdf_LOW.GDF* is controlled by the integer parameter ILOWGDF in the file CONFIG.WAM. If ILOWGDF > 0 a low-order GDF file is generated, using the panel vertices of the higher-order geometry with ILOWGDF×ILOWGDF sub-divisions. The first three lines are copied from the higher-order GDF input file. The total number of sub-divided panels is included on line 4. This option can be used to generate low-order GDF files for any of the geometries which can be input to the higher-order method, including geometries represented by a small number of flat patches (Section 6.5), B-splines (Section 6.6), and geometries which are defined in the subroutine GEOMXACT (Section 6.8). In each case the number of low-order panels can be increased by increasing the value of ILOWGDF. The coordinates of the panels are in the same body-fixed dimensional system as the original input data.

These optional files are generated in the POTEN subprogram, after reading the geometry input files and before looping over the wave periods. If NPER=0 these files can be generated quickly, without the extra time required to solve for the potential and hydrodynamic parameters.

10.7 INTERNAL TANKS

Versions 6.2 and higher include the options to analyze the linear hydrodynamic parameters for a fluid inside an oscillatory tank, or to analyze the coupled problem where one or more tanks are placed within the interior of one or more bodies, including their dynamic coupling. Usually the fluid in each tank will be bounded above by a free surface, but in special cases a rigid boundary surface can be placed above the fluid to represent a tank entirely filled by fluid. The following discussion pertains to the situation where a free surface is present in each tank. The free surface boundary condition in each tank is linearized in the same manner as for the exterior free surface. Special attention is required near the eigenfrequencies of the tanks, where nonlinear effects are significant. A two-dimensional study of this problem, including nonlinear motions in the tank, is presented in Reference [26].

The tank geometry is defined in the same manner as for the exterior surface of each body, using either the low-order method with conventional panels, or the higher-order method. In both cases it is essential that the normal vector points away from the ‘wet’ side of the tank surface, as explained in Section 5.1 (ILOWHI=0) and Section 6.1 (ILOWHI=1). In the context of a tank, this means that the normal vector points out of the tank and into the interior space of the body. In the low-order method the tank panels are included with the conventional body panels in the .GDF file. In the higher-order method the tank patches are defined in the same manner as the body surface, using one of the options listed in Sections 6.5 to 6.9. In both cases the tank panels or patches are identified by their starting and ending indices, which must be listed in the .CFG file using the parameter name NPTANK, as explained in Section 3.7.

Test 22 is an example, where the body is an FPSO containing two rectangular internal tanks. Patches 1-7 represent the exterior surface of the FPSO, patches 8-11 represent tank 1, and patches 12-15 represent tank 2. (When IRR=1, this convention is modified, with patches 8-10 used for the interior free surface of the FPSO and patches 11-18 used for the tanks, as explained in the header of the GEOMXACT.DLL subroutine FPSOINT.) One side of each tank is represented by four rectangular patches, with their vertex coordinates included in the GDF file TEST22.GDF.

The panels or patches which represent each tank must be contiguous. Separate tanks can be grouped together, or interspersed arbitrarily within the description of the exterior surface. It is recommended to group all of the tanks together, either at the beginning or at the end of the panels/patches which define the exterior surface.

The vertical positions of tanks can be specified arbitrarily, in other words the free surface of each tank can be defined independently of the other tanks and the exterior free surface. ■ Two alternative options can be used to specify the elevations of the tank free surfaces, depending on the parameters ZTANKFS and ITRIMWL in the CFG file.

If ITRIMWL=0, or if ZTANKFS is not included in the CFG file, then for each tank the free surface is defined to coincide with the highest point of the panel vertices or patch corners defining that tank. This option is illustrated in TEST 22, where the free surface of one tank coincides with the exterior free surface, i.e. the plane $Z = 0$, and the free surface

of the other tank is elevated by 1m above this plane.

■ If ITRIMWL=1, and the array ZTANKFS is included in the CFG file, then for each tank the free surface is defined to be at the elevation above $Z = 0$ specified by the corresponding element of the array. This option can be used to vary the filling ratio of tanks, without changing the GDF inputs, and also to trim the waterlines of the tanks so that the free surfaces remain level when the vessel is trimmed. Further information is given in Section 10.8. This option is illustrated in TEST22A, where the vessel is given a heel (static roll) angle of 15 degrees. When this option is used, the geometry of the internal tank surfaces must be defined up to (or above) the trimmed free surface of the tank.

The solutions for the velocity potential (or source strength) are performed independently for the exterior fluid domain, and for each interior tank. Thus the mutual locations of these surfaces are irrelevant. Small gaps between them do not cause problems, and they may even coincide without introducing numerical difficulties. When irregular-frequency removal is used (IRR>0), the entire interior free surface inside the body should be described, ignoring any possible intersections with tanks. Thus the same interior free surface should be used with or without the presence of tanks.

Two types of tank parameters must be included in the CFG file, as explained in Section 3.7:

NPTANK is an integer array used to specify the panel or patch indices of internal tanks. The data in this array are in pairs, denoting the first and last index for each tank. An even number of indices must be included on each line, **and each pair must be enclosed in parenthesis**. More than one line can be used for multiple tanks, and/or multiple tanks can be defined on the same line. If NBODY>1, the body numbers for each body containing tanks must be appended to the parameter name. Only integer data and parenthesis are read for the array NPTANK. Other ASCII characters may be included on these lines, and are ignored when the input data is read. **The requirement to enclose each pair of indices in parenthesis is new in Version 6.4. Input files used for Version 6.3 did not have this requirement, and must be modified if the parentheses are omitted.**

RHOTANK is a real array used to specify the density of fluid in internal tanks. The density specified is relative to the density ρ of the fluid in the external domain outside the bodies, as defined in Chapter 4. The data in the array RHOTANK must be input in the same order as the data in the array NPTANK. Multiple lines of this parameter may be used, with an arbitrary number of data on each line, but each line must begin with 'RHOTANK='. The total number of tanks NTANKS is derived from the inputs NPTANK in the POTEN run. If fewer than NTANKS values of RHOTANK are specified, the remainder of the array is assigned the last non-negative value. Thus if the density is the same for all tanks, only the first value is required. Zero may be assigned, but negative values of the density must not be assigned. RHOTANK is only used in the FORCE run, and may be changed if separate FORCE runs are made using the same POTEN outputs.

The following are equivalent formats for the required lines in the file TEST22.CFG:

```
NPTANK= (8-11) (12-15) (patch/panel indices for two tanks)
RHOTANK= 0.6 0.6 (fluid densities for tanks one and two)
```

```
NPTANK= (8 11)      (patch/panel indices for tank one)
NPTANK= (12 15)    (patch/panel indices for tank two)
RHOTANK= 0.6
```

These examples illustrate the following rules:

1. Only integer data are recognized in NPTANK. Arbitrary ASCII characters other than these can be used both as comments and to delimit the pairs of indices for each tank, according to the user's preferences. At least one blank space must be used to separate pairs of indices. Comments appended to these lines must not include integer characters.
2. The total number NTANKS of tanks to be included is determined by the number of pairs of indices in NPTANK inputs, in this case NTANKS=2. The same number of densities RHOTANK is required for the analysis, but it is not necessary to input repeated values if all of the densities are the same (or if all of the densities after a certain point are the same). The order of the tank densities must correspond to the order of the index pairs in NPTANK.
3. The data in RHOTANK are real numbers. After specifying all NTANKS values, arbitrary comments can be appended as in the first example above, but if fewer than RHOTANK real numbers are assigned the remainder of the line should be left blank, as in the second example above.

■ Starting with Version 6.4, the optional real array **ZTANKFS** can be included in the CFG file, as explained in Section 3.7, to specify the free-surface elevations in internal tanks. The data in this array define the elevations of the tank free surfaces, above the plane $Z=0$ of the exterior free surface. The data in the array ZTANKFS must be input in the same order as the data in the array NPTANK. Multiple lines of this parameter may be used, with an arbitrary number of data on each line, but each line must begin with 'ZTANKFS='. If the array ZTANKFS is included, it must include one real number for each tank. If the array ZTANKFS is not included, the waterline of each tank is derived from the highest vertex of the GDF inputs, as in Versions 5.2 and 5.3. ZTANKFS is only used when the waterline trimming parameter ITRIMWL=1, as explained in Section 10.8. The data in ZTANKFS are dimensional, with the same units of length as are used in the GDF file.

When field pressures and/or velocities are required for field points inside the tanks (Options 6-7) special inputs are required. In this case, when the field point coordinates XFIELD are input as explained in Section 3.3, the parameter ITANKFPT=1 must be specified in the .cfg file, as explained in Section 3.7, and the format of the XFIELD inputs in the .frc file must include the corresponding number of each tank, or zero for the exterior domain. When all of the field points are defined by arrays, using the option described in Section 3.10, the parameter ITANKFPT is not used and may be deleted from the .cfg file.

Inputs which relate to the body's mass including VCG and the radii of gyration XPRDCT (IALTFRC=1), or XCG, YCG, ZCG and the inertia matrix EXMASS (IALTFRC=2) refer to the mass of the body alone, without the tanks (or with the tanks empty). The same definitions apply to the outputs of these quantities in the header of the .out file and in the log file wamitlog.txt. When IALTFRC=1 is used, the body mass is derived from the displaced fluid mass corresponding to the body volume, minus the fluid mass in the tanks.¹

The body volumes, center of buoyancy, and hydrostatic restoring coefficients displayed in the header of the .out file are calculated from the exterior wetted surface of the body, and are not affected by the tanks. The volumes, densities, values of ZTANKFS, and hydrostatic restoring coefficients for the tanks are listed separately after the corresponding data for the hull. When the higher-order option (ILOWHI=1) is used, the header of the .out file includes a list of all the patches, and the corresponding tank numbers ITANK for patches which are defined as interior tanks. When ITRIMWL=0 or ZTANKFS is not included in the CFG file, the values of ZTANKFS displayed in the header of the .out file correspond to the tank free-surface elevations as used in the program, derived from the upper boundary of the tank surfaces.

Hydrodynamic parameters which are physically relevant for the tanks alone (options 1,5,6,7,9) can be computed by inputting only the patches or panels for the tanks. In this case IDIFF=-1 should be used, and the outputs for options 5,6,7,9 correspond to the combination of radiation modes specified by the IMODE array, with unit amplitude of each mode. The damping coefficients for the tanks should be practically zero. From momentum conservation the outputs from option 8 are zero, and the horizontal drift forces and vertical drift moment from option 9 should be practically zero.

Internal tanks affect the hydrostatic restoring coefficients C_{ij} , as described in Section 12.10. The complete hydrostatic matrix, which is output in the *out.hst* file, includes the effects of the tanks. Note that these outputs are not the same as the hydrostatic matrix for the hull alone as described above, and displayed in the .out file. For example the hydrostatic coefficient C_{33} in the .out file is equal to the total area of the waterplane inside the waterline of the body, whereas the coefficient C_{33} in the .hst file is reduced by the product of the free-surface area and density for each tank.

When both tanks and the external hull surfaces are included, hydrodynamic and hydrostatic parameters which are relevant for the hull alone, with no internal tanks, can be computed by setting RHOTANK=0.0

When planes of symmetry are specified in the GDF file (ISX=1 and/or ISY=1) the tank geometry is reflected in the same manner as the hull. This procedure should only be used if all of the tanks intersect the symmetry plane (with half of the tank on each side of the plane). For example in the case of the FPSO shown in Appendix A.22, where both tanks extend across the plane of symmetry $Y=0$, it is appropriate to use ISY=1 as is done in TEST22. On the other hand if there are separate tanks on each side of the symmetry

¹In WAMIT Versions prior to 6.312 the inertia matrix output in the log file wamitlog.txt when IALTRFRC=1 is not reduced by the fluid mass in the tanks. The correct inertia matrix is used in the equations of motion and in all hydrodynamic outputs which depend on the RAO's. Users of earlier versions can correct the output in the log file by multiplying each element of the inertia matrix by the factor $1 - (\text{RHOTANK} \times \text{VOL}_T) / \text{VOL}_B$. Here VOL_T is the sum of the volumes of all tanks in the body and VOL_B is the volume of the body. These quantities are all specified in the header of the .out file.

plane (e.g. wing tanks) the hydrostatic and hydrodynamic coefficients for the tanks will not be physically correct (although the effect of the tanks on the RAO's of the vessel would still be correct). The latter situation can be avoided by defining both sides of the hull and both tanks separately in the GDF file, and setting $ISY=0$.

■ 10.8 TRIMMED WATERLINES

Starting with Version 6.4 WAMIT includes the option to specify a ‘trimmed waterline’ for each body. In the trimmed condition, the static orientation of the body is shifted relative to the horizontal plane of the free surface, first by a prescribed vertical elevation, then by a pitch angle (often referred to as the ‘trim angle’), and then by a roll angle (‘heel’). For a given body geometry, different planes of flotation can be analysed without redefining the geometry in the GDF file. A necessary condition for this procedure to be implemented is that the definition of the body geometry in the GDF file must include all surface elements which are submerged when the body is trimmed. Thus it is necessary to define a sufficient portion of the body surface above the untrimmed waterplane.

Trimming of the waterline is supported in both the low-order (ILOWHI=0) and higher-order (ILOWHI=1) options. There are some practical restrictions that must be considered when the higher-order option is used, as described below.

The trimmed condition of the body is specified by two parameters in the CFG file, ITRIMWL and XTRIM, as defined in Section 3.7. ITRIMWL=0 is the default, indicating that the body is not trimmed. ITRIMWL=1 specifies that the body is trimmed, and XTRIM(1:3) specifies the vertical elevation and angles of rotation of the body, as explained below. If ITRIMWL=0, XTRIM is ignored. Note that if ITRIMWL=0 (or ITRIMWL is not included in the CFG file), then the GDF file must only define the submerged portion of the body as in previous versions of WAMIT; in this case the program checks to ensure that no elements of the body surface are above the plane of the free surface, and an error stop occurs if such elements are detected. Conversely, when ITRIMWL=1, elements of the body surface above $Z=0$ are permitted, and there is no check or error stop. It is possible to suppress the error stop, without trimming, with the inputs ITRIMWL=1 and (XTRIM=0.0 0.0 0.0).

Appendix A includes four test runs with trimmed waterlines: TEST01A, TEST09A, TEST13A, and TEST22A. The perspective plots which accompany these descriptions illustrate the trimmed conditions of the structures.

When multiple bodies are analysed, the vector XTRIM must be input separately for each body, following the same format as for XBODY in Section 7.6. TEST13A illustrates this procedure for NBODY=2.

The following coordinate systems and input parameters should be considered in defining the process of trimming waterlines:

(X,Y,Z) are global coordinates with $Z=0$ in the plane of the free surface.

(x,y,z) are conventional body coordinates, which define the submerged body surface in WAMIT.

(ξ, η, ζ), referred to hereafter as ‘GDF coordinates’, are the coordinates used to define the body geometry in the GDF file.

In the low-order method (ILOWHI=0), when the panel vertices are input from the GDF file, these are converted from GDF coordinates to body coordinates, using the translation

and rotations defined by XTRIM. When the higher-order method is used (ILOWHI=1), the same coordinate transformations are performed from the subroutine outputs each time that the subroutine defining the body surface is called. The practical effect, in both cases, is that the original definition of the body geometry in the GDF file is replaced during the WAMIT computations by a new transformed description in (x,y,z) coordinates, which represents only the submerged portion of the body in the trimmed condition. This submerged surface can be plotted and visualized, in the same manner as for untrimmed bodies, using the parameter IPLTDAT as explained in Section 3.7.

The transformation from GDF coordinates to body coordinates is defined by the following relations:

$$\begin{aligned} x &= \xi c_2 + \eta s_2 s_3 + \zeta s_2 c_3 \\ y &= \eta c_3 - \zeta s_3 \\ z &= -\xi s_2 + \eta c_2 s_3 + \zeta c_2 c_3 + \text{XTRIM}(1) \end{aligned}$$

where

$$\begin{aligned} c_2 &= \cos(\text{XTRIM}(2)), & s_2 &= \sin(\text{XTRIM}(2)) \\ c_3 &= \cos(\text{XTRIM}(3)), & s_3 &= \sin(\text{XTRIM}(3)) \end{aligned}$$

Except for the GDF coordinates, the definitions of coordinate systems, etc., are as defined in earlier chapters. In particular, the body motions and forces are defined with respect to the conventional body coordinates (x,y,z). Thus, for example, surge is in the horizontal direction and heave is in the vertical direction, relative to the free surface. Rotations and moments are defined with respect to the origin of this coordinate system. XBODY(1), XBODY(2), XBODY(3) are the (X,Y,Z) coordinates of the origin of the body coordinate system relative to the global coordinate system, and XBODY(4) is the angle in degrees of the x-axis relative to the X-axis of the global system in the counterclockwise sense, as shown in Figure 5.2.

The input parameter XTRIM(1), which defines the vertical coordinate z of the origin of the GDF coordinates, is in the same units as the dimensional GDF coordinates, corresponding to the parameter GRAV in the GDF file. The angles XTRIM(2) (pitch) and XTRIM(3) (roll) are in degrees. XTRIM(2:3) are Euler angles, with the convention that the body is first pitched about the transverse η -axis, and then rolled about the longitudinal ξ -axis. (Yaw can be included via XBODY(4). With this convention the projection of the ξ -axis on the plane Z=0 is at the same angle XBODY(4) as the projection of the x-axis.)

In the low-order method (ILOWHI=0), the input panels from the GDF file are tested for their positions relative to the plane of the free surface (Z=0). ‘Dry’ panels, which are entirely above the free surface, are removed from the array of panel coordinates within the program. ‘Wet’ panels, which are entirely submerged, are retained without modification. Panels which intersect the waterline, referred to as ‘waterline panels’, are trimmed at the waterline. In some cases, where waterline panels have three vertices below the free surface and one vertex above the free surface, the resulting trimmed panel is a polygon with five sides; in these cases the panel is subdivided to form two separate panels (one quadrilateral

and one triangular). The number of panels is increased by one for each subdivided panel. Conversely, when dry panels are removed, the number of panels is decreased. Examples of subdivided panels can be seen in the perspective plot corresponding to TEST01A in Appendix A.

In the higher-order method (ILOWHI=1), an analogous procedure is followed for each patch of the body, based on the vertical positions of the four corners of each patch. If the patch intersects the waterline, an iterative procedure is used to remap the submerged portion onto a square domain in parametric space. There are situations where this procedure breaks down, and thus it must be used with caution. One example is where the vertices of a patch are all submerged, but a portion of one side maps in physical space above the free surface.

Body symmetry can be affected by trimming. When it is necessary, WAMIT automatically reflects the body geometry and resets the symmetry indices. For example in TEST01A the GDF file specifies ISX=ISY=1, but the trimmed body has only one plane of symmetry ($x=0$). Similarly in TEST22A the GDF file specifies ISY=1, but the trimmed body has no planes of symmetry due to the heel angle.

Dipole panels are trimmed in the same manner as conventional panels, as illustrated in TEST09A.

Special attention is required for bodies with internal tanks. Two alternative options are included, as explained in Section 10.7, depending on the parameter array ZBODYFS in the CFG file. If ZBODYFS is not included in the CFG file (or if ITRIMWL=0), the geometry of internal tanks is *not* trimmed, and the tank free surface is always defined by the highest points of the specified panels or patches, as in WAMIT Versions 6.3 and 6.4. If trimming includes a pitch or roll displacement of the body, this approach requires the tank geometry to be redefined so that its upper boundary coincides with the correct plane of the free surface inside the tank. If ZBODYFS is included in the CFG file (and if ITRIMWL=1), the geometry of internal tanks is trimmed using the same algorithms as for the exterior surface. In this case it is necessary to define the tank geometry up to the level of the trimmed free surface, as in TEST22A.

Special attention is required also when irregular-frequency removal is used (IRR>0). If the interior free surface is defined by the user as part of the overall geometry of the body (IRR=1) it is necessary to ensure that the interior free surface coincides with the Z=0 plane after trimming. If automatic discretization of the interior free surface is used (IRR=3), the interior free surface is defined correctly in the plane of the trimmed free surface. The option (IRR=2) cannot be used when ITRIMWL>0 (See Section 9.2).

It is strongly recommended to plot the geometry after trimming, using the data files which are defined by the parameter IPLTDAT in the CFG file, to ensure that the actual trimmed structure is correct.

■ 10.9 BODIES AND WAVEMAKERS WITH TANK WALLS

Starting with Versions 6.3 and 6.4, different options are available for the analysis of various problems where walls are present in the plane(s) of symmetry $x = 0$ and/or $y = 0$. The analysis of wave radiation by wavemakers in the plane(s) of the walls was introduced in Version 6.3. In Version 6.4 a more general approach is introduced, applicable to wavemakers in the planes of the walls and/or bodies in the interior domain of the fluid. These two approaches require somewhat different inputs. The more general approach is described first.

Version 6.3 includes the option to analyze the waves generated by one or more wavemakers in the plane(s) of symmetry $x = 0$ and/or $y = 0$. This configuration corresponds to a wave tank with rectangular walls in the planes of symmetry. The opposite walls of the wave tank are assumed to have absorbing beaches, represented here by open domains extending to infinity. If the symmetry indices $ISX=1$, $ISY=1$ are input, with wavemakers in one or both planes of symmetry, the walls extend to infinity in both the x and y directions. Alternatively, with $ISX=1$ and $ISY=0$ and wavemakers in the plane $x = 0$, or conversely, one wall extends to both \pm infinity corresponding to a wave tank of infinite width. In all cases it is assumed that the wall(s) are planes of symmetry, and the fluid motion is symmetrical about these planes. Thus the solution for the velocity potential in the fluid domain can be represented by a distribution of sources of known strength, proportional to the normal velocity of the wavemaker, and it is not necessary to solve the integral equation for the velocity potential on the wavemakers. This saves considerable computational time, and also avoids the singular solution that would otherwise occur for bodies of zero thickness in the plane of symmetry.

The geometry of each wavemaker is defined in the .gdf file. In the low-order method ($ILOWHI=0$) a sufficient number of panels must be included on each wavemaker to ensure a converged solution. In the higher-order method ($ILOWHI=1$) only one patch is required on each wavemaker. If the wavemakers are rectangular (or quadrilateral), the higher-order analysis can be carried out most easily using the option $IGDEF=0$, as explained in Section 6.5.

The parameter $ISOLVE=-1$ is used to indicate that the wavemakers are in planes of symmetry, and that the solution of the integral equation should be skipped. Suitable generalized modes must be defined, as explained in Chapter 8, to represent the normal velocity of each wavemaker. $IRAD=0$ is recommended, with $IMODE(1:6)=0$, to avoid computing the 6 rigid-body modes. $IDIFF=-1$ is required. The separate wavemaker elements are considered to be part of one 'body', with appropriate generalized modes used to represent the independent motion of each element, and $NBODY=1$. No other bodies can be present within the fluid domain.

The principal outputs are WAMIT options 6 and 7 (potential and fluid velocity at specified field points). If the field point is on the free surface the potential is equivalent to the wave elevation. No other options are supported. If multiple wavemakers are run together with separate modes for each wavemaker, the parameter $INUMOPT6=1$ should be specified in the .cfg file to provide separate outputs for each mode. In that case only the

complex amplitude is output, with a separate pair of columns for each mode, as indicated in Section 4.9. A post-processor should be used to combine these outputs, for arbitrary combinations of simultaneous motion of all wavemakers.

In the NEWMODES DLL file the subroutine WAVEMAKER is included, to analyze one or more wavemaker segments. If IGENMDS=21, the wavemakers are hinged with pitching motions about a horizontal axis. The depth of this axis is specified in the input file *gdf*_WMKRHINGE.DAT where *gdf* is the filename of the GDF file. In the subroutine the depth of the axis is the same for all wavemaker elements. The number of wavemakers is arbitrary, but each separate mode of motion corresponds to one patch (or panel) of the geometry, in the same order as these are defined in the .gdf file. In the low-order method this effectively restricts the use of the subroutine to only one panel per wavemaker. Thus it is strongly recommended to use the higher-order method (ILOWHI=1) when using the subroutine WAVEMAKER. NEWMDS=NPATCH must be specified in the pot or cfg file, with the same value as the number of patches in the gdf file. To specify the use of WAVEMAKER the parameter IGENMDS=21 is defined in the CFG file.

TEST23, described in Appendix A23, illustrates the use of this option for a bank of eight wavemakers along the wall $x = 0$, with symmetry about both $x = 0$ and $y = 0$. The generated wave elevations for each wavemaker are evaluated over a square array of $8 \times 8 = 64$ field points. The depth of the horizontal axis is specified by the parameter ZHINGE=-2m in the input file TEST23_WMKRHINGE.DAT.

■ 10.10 BODIES AND WAVEMAKERS WITH VERTICAL WALLS

WAMIT includes the option to account for images of the body in the presence of one vertical wall, or two vertical walls which intersect at a right angle. In Versions prior to V6.4 this capability was restricted to the low-order method (ILOWHI=0) and to one body (NBODY=1), using the procedure described in Section 5.3. Starting with V6.4, the analysis of bodies near vertical walls is extended to include multiple bodies and/or use of the higher-order method. With the exception of the inputs discussed below, the definitions of coordinates, body positions, and outputs are as explained in Section 5.3. In particular, if one or two walls are present, the origin of the global coordinate system is in the plane of the wall(s). The wall(s) must coincide with the plane $x = 0$ or $y = 0$, or both.

The presence of walls is specified by the optional integer inputs IWALLX0 and IWALLY0 in the .cfg file. In the default case these inputs are assigned the values zero, signifying that there are no walls. The inputs IWALLX0=1 and/or IWALLY0=1 signify that there are walls in the planes $x = 0$ and/or $y = 0$, respectively. All other inputs are unchanged from the case without walls. In particular the GDF files for one or more bodies should specify the symmetry indices ISX,ISY corresponding to the planes of symmetry of the body, with values 0 or 1, just as in the case without walls. (This differs from the procedure described in Section 5.3, where negative values of ISX,ISY are used to signify the presence of walls. That procedure can still be used, when ILOWHI=0, but it is not recommended.)

When one or two walls are present the momentum drift force cannot be evaluated, and the input IOPTN(8) in the FORCE file is ignored. No other restrictions apply, relative to the use of WAMIT without walls. Special attention is required regarding the definition of the incident wave amplitude, as explained in Section 5.3.

Wavemakers may be included in the walls, following a similar procedure as described in Section 10.9. Several illustrative examples are included in the Supplementary Test Runs which can be downloaded from www.wamit.com. When bodies and wavemakers are both present, ISOLVE \geq 0 must be specified. The wavemakers are treated as one or more separate bodies. In the simplest case, where one body is in the fluid domain and all wavemakers are defined within another GDF file, NBODY=2 is used and the parameter IBODYW=2 is defined in the CFG file, indicating that body 2 consists of wavemakers in the walls. The order of the bodies is specified by the order of the corresponding inputs in the POT file (using IALTPOT=2). There can be one or more physical bodies, and one or more wavemaker bodies. The physical bodies must precede the wavemaker bodies in all cases. Thus if $K = 1, 2, \dots, NBDYP$ are the indices for NBDYP physical bodies, and $K = NBDYP + 1, NBDYP + 2, \dots, NBDYP + NBDYW$ are the indices for NBDYW wavemakers, NBODY = NBDYP + NBDYW is the total number of bodies identified in the input files and IBODYW = NBDY + 1 is the index of the first wavemaker body, input in the CFG file.

Only the radiation problems are solved when wavemakers are present, including radiation from the wavemakers as generalized modes. Outputs with wavemakers include the added mass and damping matrices (Option 1), a surrogate for the Haskind exciting force

to be explained below (Option 2), RAO's in response to the Option 2 exciting force and moment (Option 4), as well as Options 5,6,7.

The 'surrogate' exciting force is the force and moment acting on the body due to the motions of the wavemakers, computed from the cross-coupling added mass and damping coefficients where one mode is for the body and the other mode is for the wavemaker. The outputs for the exciting force (Option 2) and RAO (Option 4) are the same as for incident waves, except that the index of the wavemaker is output in place of the wave incidence angle BETA. These 'wavemaker RAO's' cannot be compared directly with the conventional incident-wave RAO's since the amplitude of the waves generated by the wavemaker modes are not used to normalize the RAO's. It is possible to make a separate run without the bodies to measure the wave elevations at the locations of the bodies using Option 6, as one might do to calibrate the wavemakers in a physical wavetank. If the wavemaker RAO's are normalized by these calculations of the wave elevations the results are directly comparable with the conventional RAO's in incident waves of unit amplitude. TEST23C includes results to illustrate this procedure.

When walls are specified by the parameters IWALLX0=1 and/or IWALLY0=1, the program automatically reflects the body and wavemaker geometry about the walls. The planes of symmetry for each body or wavemaker, defined by the parameters ISX and ISY in the GDF file, refer specifically to the local planes of symmetry of the body or wavemaker. When IWALLX0=1 and wavemakers are defined in the wall X=0, the symmetry index ISX=0 should be used for the wavemakers; similarly, for wavemakers in the wall Y=0, ISY=0. TEST23C illustrates a problem where one wavemaker is in each wall, and thus ISX=ISY=0. (Except for special cases where the GDF file only defines one half of the wavemaker, both wavemaker symmetry indices should be zero.) The symmetry indices for bodies are the same as without walls; TEST23C illustrates this situation with a floating hemisphere, where only one quadrant is defined and ISX=ISY=1.

Since the wavemakers correspond to one or more bodies, it is necessary to include the usual inputs for these bodies in one or more FRC files. However, the dynamic characteristics of the wavemakers are ignored, since the RAO's are not evaluated for the wavemakers. Thus dummy values of the inputs for the dynamic characteristics of the wavemakers (VCG and XPRDCT, or XCT and external force matrices) must be included, but their values are not relevant. The input files TEST23*.FRC illustrate how these inputs can be defined.