SEA WAVE PATTERN EVALUATION

SWPE 6.0
SIMPLIFIED USER'S GUIDE

E.O. Tuck and D.C. Scullen

Scullen & Tuck Pty Ltd
PO Box 237 Rundle Mall, Adelaide, South Australia 5000

October 18, 2002

Summary

In SWPE versions 1.0 and 2.0, Michell's thin-ship theory was implemented to yield the free wave spectrum, far-field wave elevations, velocity components and velocity potential behind monohull vessels moving steadily forward on a calm deep sea. Version 3.0 added the capability to calculate near-field wave elevations, and incorporated eddy-viscosity damping of far-field waves. Version 4.0 further extended capabilities to multihulls and submerged bodies, and finite-depth effects on far-field wave elevations. Version 5.0 increased computational accuracy and speed, and allowed computation of sinkage and trim. Version 6.0 improved the eddy-viscosity procedure. This is a simplified guide to use of `SWPE6.0`, with emphasis on input and output structures, assuming availability of a compiled version of the program. A more detailed User's Manual for the source code, including description of individual units of the program, was provided for `SWPE5.0`.

2. HULL GEOMETRY

The co-ordinate system and hull parameters are as follows: x is positive astern, y is positive to starboard, and z is positive upwards (hence negative for normal ship sections, since $z = 0$ at the waterline). Hulls are assumed to be symmetric with respect to their own centreplanes, which are parallel to the x-axis, and are specified by offsets consisting of the positive y coordinates on a uniformly-spaced rectangular grid of stations x and waterlines z.

To accommodate multihull arrangements, the program requires specification of the location of every hull in the ensemble. This is effected by specifying the location of the centre of each hull in the horizontal xy-plane. For monohulls, the centre of the ship is most conveniently located at $x = 0$ and $y = 0$, but there is no compulsion to use this convention.

3. INPUT FILE

All input to the program is via the plain text file called "`swpe.in`". It is crucial that only numeric values be changed in this file. Comments, notes and other lines of text must not be deleted, or the program may not function correctly. These lines can be modified with care, but should not be deleted.

3.1 Preliminary lines

The first four lines of the file are merely a heading for the file and serve to identify the version. The next five lines (all beginning with "`Note:`" in the example files) have been included to allow user comments. These comments are not used by the program, but the program expects at least one character in each of these nine lines.

Each variable to be used by the program is preceded by a description of that variable, the units that are assumed, and the type of value expected (integer or decimal). In some cases, suggested upper and lower limits on variables are are also given.

3.2 Physical parameters

The first block of parameters sets values for physical quantities affecting the problem.

The first parameter is the value of the gravitational acceleration g (in ms^{-2}). Normally use 9.81.

Water `depth` in metres must be entered as a decimal. To simulate infinite depth water, use a value of 10,000.0 metres, the maximum value allowed by the program. This choice is recommended even if the actual water depth is significantly less, even down to values comparable to the shiplength or lower, noting that true finite-depth near-field results are not computed by this version of the program.

Water density ρ (in kg m^{-3}) must be entered as a decimal. Normally use 1025.9.

Water kinematic eddy viscosity ν_e (in m^2s^{-1}) must be entered as a decimal. Zero is allowed, and values larger than 0.01 are not recommended. In the `SWPE6.0` report we studied various choices for ν_e in detail, and for routine use we recommend 0.0001.

The next parameter specifies the vessel's speed U (in ms^{-1}).

3.3 Computational parameters

The second block of parameters in the input file is concerned with the calculation of the free wave spectrum and the wave elevations.

The number `ntheta` of θ intervals to be used in the calculations must be entered as an even integer. This is the only user-input parameter directly controlling accuracy in `SWPE6.0`. Higher `ntheta` is more accurate but more time-consuming. Normally use 1000 for monohulls, up to 4000 for mutihulls.

Limits on the size of the rectangular computational domain are then entered as decimals.

`wx0` is the x-ordinate of the fore-most edge of the rectangle, `wx1` is the x-ordinate of the aft-most edge. `wy0` is the y-ordinate of the port side of the rectangle. `wy1` is the y-ordinate of the starboard side of the rectangle.

The number of grid points `nwx` in the x direction, and `nwy` in the y direction, must then be entered as integers greater than 0 and less than or equal to 300.

If values of 1 are entered for both `nwx` and `nwy`, the program will calculate the wave elevation at the single point (`wx0,wy1`).

Standard transverse and longitudinal wave cuts are easily effected. For example, if `nwx` is set to 1, and `nwy` is set to 100, the program will calculate wave elevations at 100 equally-spaced points across the track along the line extending from (`wx0,wy1`) to (`wx0,wy0`).

On the other hand, if, for example, `wnx`=100, and `wny`=1, then the program will calculate 100 wave elevations along the longitudinal cut extending from (`wx0,wy1`) to (`wx1,wy1`).

The integration type `kitype`, which must be an integer equal to either 1,2,3 or 4, determines the method to use for the calculation of wave elevations. If (recommended option) a value of 1 is chosen for `kitype`, wave elevations are calculated using Newman's method for the local field and the method of Tuck, Collins and Wells for the far-field component. A value of 2 employs a speed-up option to eliminate the computation of the local field where its contribution is negligible. A value of 3 instructs `SWPE` to calculate only the far-field integral and its velocity components, for the entire rectangular patch. A value of 4 instructs `SWPE` to calculate wave elevations along the side of the hull at $z = 0$.

The z-ordinate `zphi` for calculation of the velocity potential and fluid velocities must be a decimal value less than or equal to zero. Unless `kitype` = 3, this parameter should be set to zero, so that all computations are performed at the free surface level.

3.4 Hull geometry inputs

The ship geometry is defined in the next two blocks of the input file.

The first block contains one parameter only, namely the number of hulls in the ensemble.

In the second block of the ship definition section, each hull in the ensemble is specified, in turn. If the number of hulls was specified as 1, the program will only read the definition of the first hull. Any lines of input following the definition of the first hullp will be ignored by the program. If the number of hulls is 2, the program will read the definition of the first two hulls and ignore any subsequent lines in the input file.

The first parameter is the submergence depth `H` (in metres), which can be positive, zero, or negative. The submergence depth is defined as the distance between the undisturbed waterplane ($z = 0$) and the topmost waterline, i.e. the last column of offsets in the table. The program checks to ensure that the hull has not run aground, i.e. that $\mathtt{H} + \mathtt{T} < \mathtt{depth}$, where T is the draft. $H = 0$ is the normal option, with $H > 0$ for fully submerged bodies, and $H < 0$ to allow the hull to rise relative to the configuration defined by the input offsets.

The number of stations and the number of waterlines describing the hulls must be integers greater than or equal to 3, and less than or equal to 89 in the current compiled version. The actual bow and stern ends are counted as stations, and the resulting total number of stations must be an odd integer. That is, the number of x-wise intervals of length must be an even integer. Similarly, the number of waterlines must also be an odd integer. Stations and waterlines must be equally spaced.

It does not matter whether the offsets describing each hull are in dimensional or non-dimensional form. The program will automatically scale the offsets to the hull beam.

All offsets at the bow (the first row) must be equal to zero (decimal). Stern offsets (the last row) may be all zero (no transom) or some non-zero if there is a transom stern of a shape determined by the non-zero offsets. The number of rows (cross-sections) and columns (waterlines) in the offset data must be the same as the number of stations and waterlines specified earlier.

Following the offsets for the hull, the beam, length and draft of that hull must be specified.

3.5 Sinkage and trim

Finally, the program requires input stipulation of the sinkage and trim, via three input parameters.

The first (integer) parameter, `Max_Squat_Iterations`, controls how the second and third (decimal) parameters (sinkage and trim) will be calculated and/or implemented, and `Max_Squat_Iterations` can only take the values 0 or 1 in `SWPE6.0`. The sinkage σ (in metres) is given at midships with the convention that positive sinkage increases draft. Trim τ (positive bow up) is in degrees.

The program assumes initially that the hull is in an attitude specified by these input values of sinkage and trim (which may be zero), and takes action to adjust these values depending on the value of the squat iteration parameter `Max_Squat_Iterations`. If this parameter is set to zero, and the specified sinkage and trim values are also zero, no action is taken to adjust input offsets, and no sinkage and trim is computed or used.

If `Max_Squat_Iterations` is set to zero, but the user has specified non-zero input values for the sinkage and trim, the program will use those values, adjusting the input offsets appropriately before performing any flow computations, but will not calculate hydrodynamic forces on the hull, nor in any way change the specified sinkage and trim.

On the other hand, if the user specifies that `Max_Squat_Iterations` is equal to 1, then (after adjusting the input offsets to take account of the specified input sinkage and trim if any) `SWPE6.0` will calculate the corrected sinkage σ and trim τ. Finally, the hull offsets are rotated and translated by τ and σ, respectively, before any further flow calculations (such as wave elevations or hull wave profiles) are performed.

4. OUTPUT FILES

All output files created by `SWPE6.0` are in plain text format.

4.1 pq.out

This file contains the values of θ and the computed far-field complex wave amplitude (or "free-wave spectrum") as a function of wave angle θ. This is in the form of Michell P and Q functions for each value of θ, separately for each hull in the ensemble. The first row contains labels for the columns.

4.2 swpe.out

Elevation and flow-field outputs are normally written to the file `swpe.out` in text format.

The first two columns in this file contain the x- and y-ordinates of the grid points, followed by the total wave elevation, the velocity potential, and the x, y and z velocity components (at depth `zphi`) for each node of the rectangular patch.

If the integration type `kitype` is set to 4, i.e. the wave profile only along the hull is requested, this file is not used. Instead, wave elevations are written to the file `hullwave.out` described below.

4.3 hullwave.out

If `kitype` is set to 4, the output file is `hullwave.out`. The first column then contains the x-ordinates of the midpoints between stations, the second column contains the y-ordinates at the waterline $z = 0$. The five columns following the x- and y-ordinates contain the far-field wave elevation, the far-field velocity potential, and the far-field velocity components, respectively (at $z = 0$) for each panel.

4.4 squat.out

Geometric details of the final hulls, which may be squatted relative to the input hulls via inputted or computed sinkage and trim, are written to the file `squat.out` in text format.

For each separate hull, this file contains the weight (in Newtons), displacement volume, buoyancy, waterplane area, waterplane moment of inertia, longitudinal centre of floatation, longitudinal centre of buoyancy, vertical centre of buoyancy, draft, and sinkage and trim. Although not all of these parameters are used in `SWPE6.0`, they can provide a useful check against errors in the input specification.

4. EXAMPLE INPUT FILES

To use the example files described below, you must first copy the example file to the file named `swpe.in`. Note that input files from previous versions of `SWPE` must be modified before they will work with `SWPE6.0`.

4.1 Wigley hull

The file "`wigley.in`" contains data for the Wigley hull which was used in the first example in the Appendix of the Part 5 report.

For most sinkage and trim calculations, `SWPE6.0` requires that the offset table contain waterlines above the undisturbed waterplane $z = 0$. The offset table in the present example comprises 41 stations and 81 waterlines; the topmost 40 waterlines (i.e the last 40 columns of the offset table) have the same offsets as the 41st waterline, i.e. the "usual" waterline at $z = 0$ for this hull has been extended upward in a vertical-sided manner. The displacement volume for the entire hull (all 81 waterlines) is $D = 6944.445$ m^3. The draft to the top waterline is $T = 12.5$ m. By specifying the submergence for this extended vessel to be $H = -6.25$ m, the draft of the underwater portion is then $H + T = -6.25 + 12.5 = 6.25$ m, which is the standard draft for the unsquatted Wigley hull.

4.2 NPL hull series

The file "`npl3b.in`" contains data for a hull from the NPL series of hulls. This input file was used to create sinkage and trim data in the Appendix of the Part 5 report.

In this example, hull offsets were not extended vertically above the static waterline by the method of the previous examples. Rather, we have supplied true offsets up the top of the model, well above the design waterline.

4.3 DDG51 destroyer hull

The file "`ddg51.in`" contains data for the DDG51 destroyer hull which has been used as an example in the last two `SWPE` reports. This input file was used to create sinkage and trim data in the Appendix of the Part 5 report. In previous reports, we used the trimmed offsets supplied by DTMB, e.g. one set of offsets for $F = 0.2755$ and another different set for $F = 0.4136$. In this example, we use the “static” offsets supplied by DTMB. `SWPE6.0` can then be used to find the trim and sinkage for any given F, and also to adjust the offsets before performing predictions of wave elevations or hull wave profiles in the predicted squatted configuration.