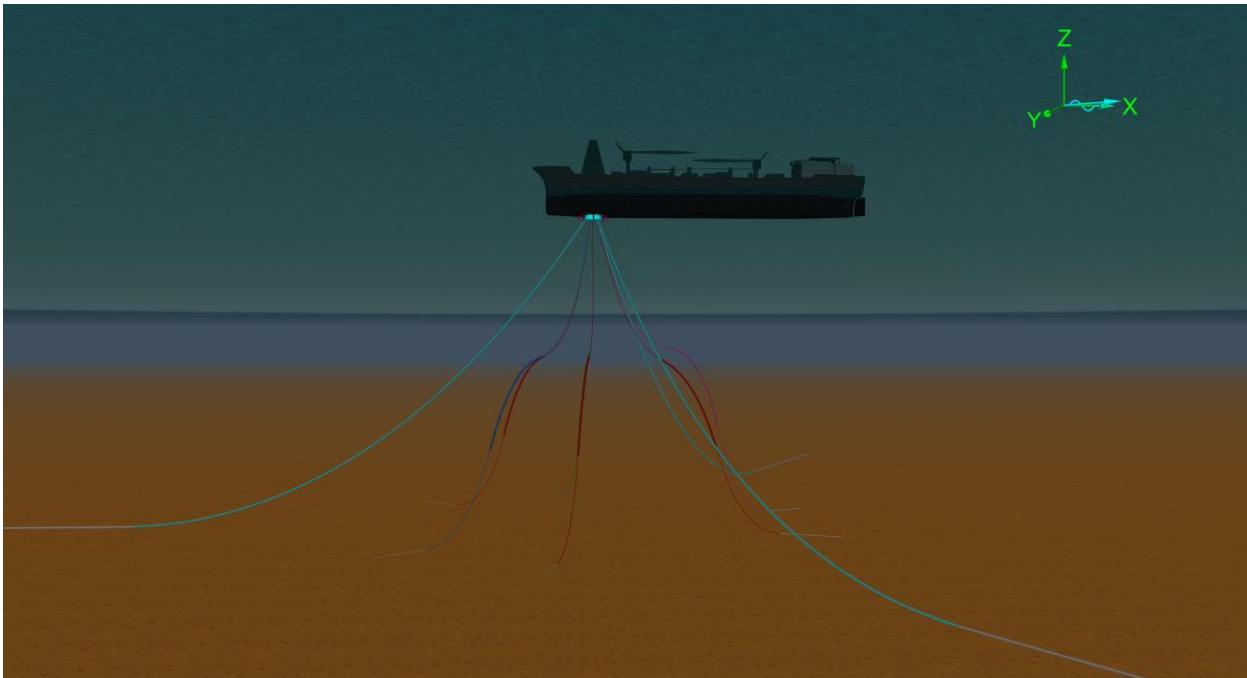


# A04 Disconnecting turret system

## Introduction



This model represents an FPSO system with a releasable turret. Disconnecting systems are found in regions where typhoons or ice flows are common. It allows the vessel to be removed until conditions improve.

The turret can be lowered from the vessel with winches. However, downtime is expensive, so operators prefer to keep producing until the last moment before disconnecting the turret and departing the field. The turret drops below the severe weather until the FPSO returns to reconnect.

The turret system needs to be designed so there are acceptable responses in all three phases:

- **Connected:** the turret is attached to the vessel and moves with it.
- **Disconnecting:** the risers are disconnected from topside, and the turret is released.
- **Disconnected:** the turret settles deeper in the water.

This example shows the disconnecting phase. The turret is released before the FPSO moves off station. When you open the simulation file you will see a shaded view of the system, along with some results graphs.

Make sure the *model browser* is in *view by groups* mode; this is set by right-mouse clicking in the *model browser* and checking that the *view by groups* option is ticked.

## Building the model

Open the *environment* data form via the *model browser* and move to the *current* page. This current profile has been set to change direction through the water column. The *rotation (deg)* column in the *profile* table shows the deviations from the specified *surface direction* of 0°.

Clicking the *View vertical profile 3d view* shows this rotation. You can control this view in the same way as you can the standard view windows in OrcaFlex. Click on the purple-coloured arrows on the top left-hand corner of the view window to rotate the view around and see the different current rotations.

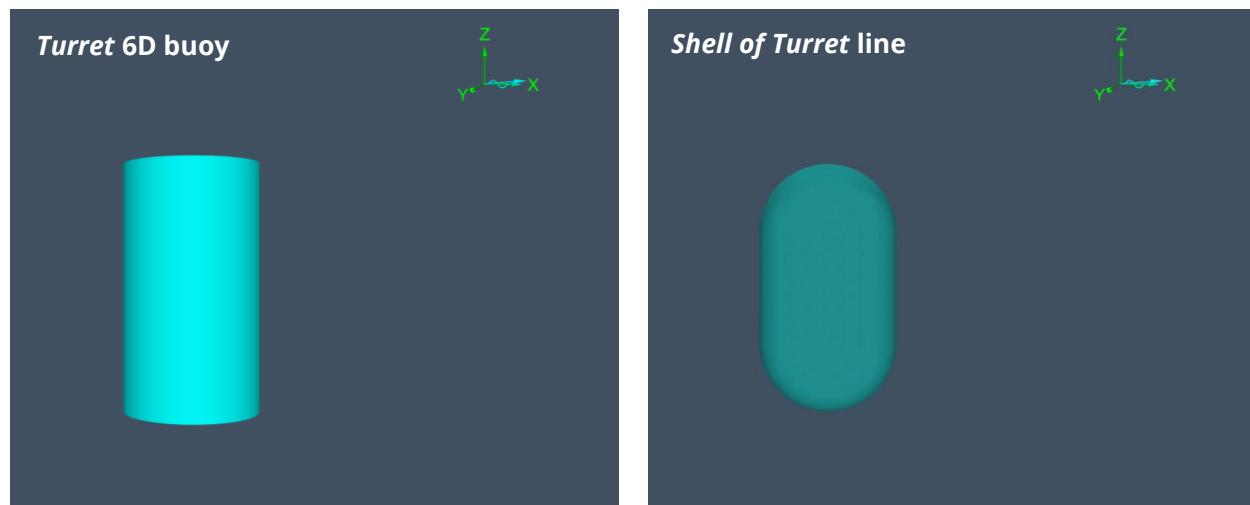
The model browser shows the system in two groups, *Vessel System* and *Disconnecting System*. Select *Disconnecting System* in the model browser. This contains the structures making up the turret as well as the moorings, risers and umbilicals.

Open *Turret* from the model browser. This object is a *spar buoy* type 6D buoy that gives the physical and hydrodynamic properties of the real turret buoy. Note that these values need to be as accurate as possible to effectively model this system when the buoy is released from the vessel.

The model has been set up with the turret origin at its base. Right click anywhere on the 6D buoy data form and select *properties* from the dropdown list. This shows the *centre of mass* is 6m below the *centre of volume*. It is a common practice in this kind of system to keep the centre of mass low down on the turret to improve its pitch stability when disconnected.

This example has been set up so clearance information can be obtained between the turret and its protective cage during the disconnection. Any reaction force from contact between the two will also be captured.

Clearance information is available between lines, so a single line (*Shell of Turret*) has been attached to the buoy at both ends. The buoy is 18m long. However the line representing its shape is specified as 8m long. This is because lines are treated as having hemispherical ends for the purposes of contact and clearance calculations. Adding the contact radius hemispheres at both ends gives a length that matches the turret for contact purposes.



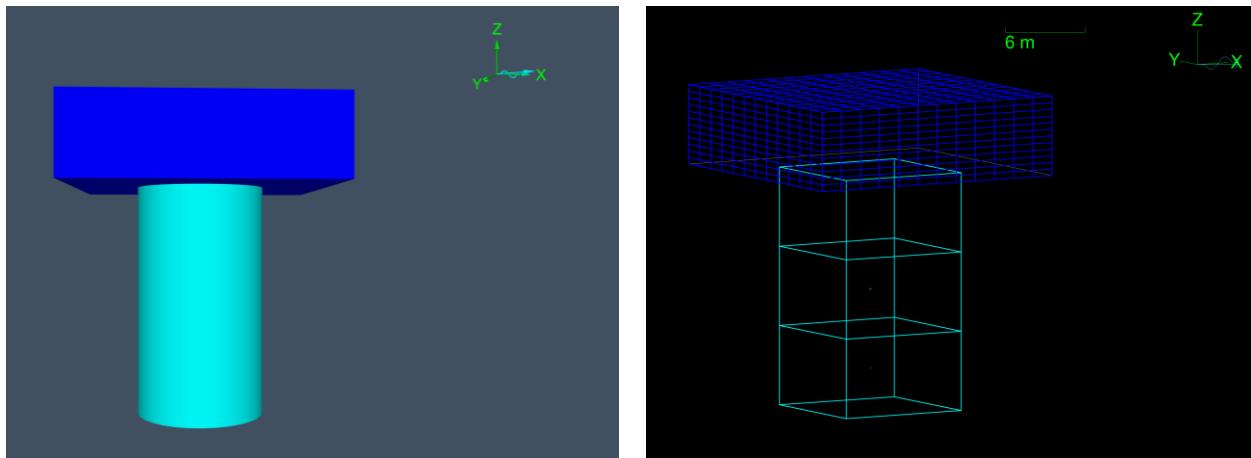
The line is required to supply contact information but not physical or hydrodynamic properties. To see how this has been done open the line type *Turret Contact* from the *model browser*. The *geometry, mass & expansion* page on the *line type* data form shows the outer diameter and mass are negligible. The *drag & lift* and *added mass, inertia, slam* pages show the hydrodynamic coefficients have also been set to zero.

Now look at the *contact* page. The contact diameter has been set to match the outer diameter of the buoy. Please see the *Lazy S detailed* example in [A02 Midwater arch systems](#) for more discussion about line-on-line contact.

The turret will be held up against a bearing plate. This is modelled by the 6D buoy vertices contacting a rectangular *elastic solid* type shape, named *Turret Bearing Plate* (see the *Vessel System* group). The buoy vertices therefore need to be given the correct contact area.

Each cylinder is considered as a rectangular box for contact purposes. Go to the *drawing* page of the *Turret* 6D buoy data form. Select *draw square cylinders* to see these boxes in the view. To see them clearly, select the wire frame view by un-checking the *shaded graphics mode* option in the *view* dropdown menu (or use the shortcut *Ctrl + G*). Then hide all objects except *Turret*, by right clicking on the model browser and selecting *hide*. Zoom in so that you can see the turret buoy clearly. With three cylinders in the stack, this means there are  $3 \times 8 = 24$  vertices with four on the top face. The buoy contact area is distributed evenly between all of these vertices.

Go to the *contact* page of the *Turret* 6D buoy data form. The *total contact area* has been set to  $471.2\text{m}^2$ . Each vertex is therefore assigned a contact area of  $19.6\text{m}^2$ . The four vertices at the top therefore give a top area of  $78.5\text{m}^2$ , which is the area of the top face.



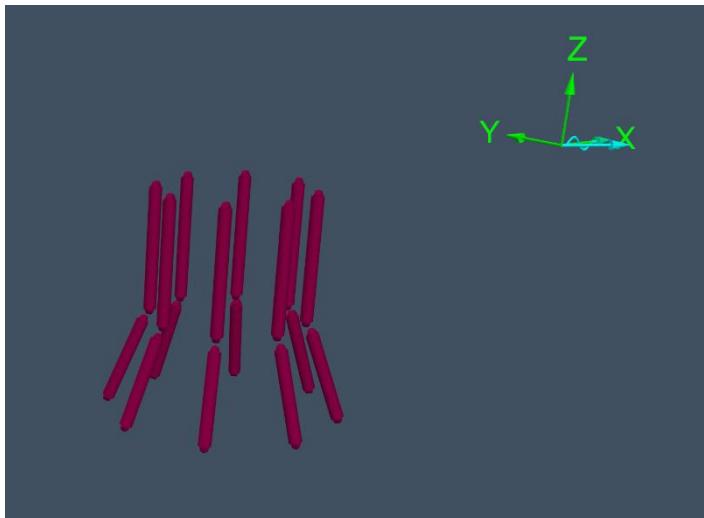
The *Friction coefficients* data form, accessible from the *model browser*, specifies the friction between the shape and the buoy (see the *6D buoys* page). This form can be used for setting shape friction with any combination of lines or buoys. Note that shape-to-line friction is only active in dynamics. *Line clashing* contact does not include friction.

This example shows one way of modelling the contact and clearances, which is appropriate for a cylindrical turret. However, there are alternative disconnecting turret geometries and alternative ways of modelling the contact with baseplates and guides. A conical turret, for example, could have a *curved plate* shape attached to the buoy to produce the cone, or rings of lines could be used.

It is important that all lines attached to the *Turret* are included in the model, otherwise the actual system response will not be produced. In this example, three mooring lines and six lazy wave risers are connected to the bottom of the 6D buoy.

Look at the *Vessel System* group in the *model browser*. This consists of the FPSO, the moonpool, the protective cage and the restraints for the *Turret*.

The *Moonpool* object is a cylinder shape representing the FPSO moonpool. This time, a *trapped water* shape is used, meaning the wave and current effects are suppressed within its enclosed volume. This means that the *Turret* buoy is shielded while inside the moonpool.



The *Moonpool Cage* sub-group is the protective structure around the *Turret*. It is made of single segment lines attached to the FPSO. The lines have clashing turned on so they can produce contact reaction forces for *Shell of Turret*. Clearance information is always calculated.

Note the slight gap between the vertical and sloped lines. This gap has been left so that they do not contact each other. Therefore any contact results are only between the *Shell of Turret* and its protective cage.

The *Moonpool Restraints* sub-group contains the objects that hold the *Turret* inside the moonpool. These are the *Turret Bearing Plate* mentioned earlier and a temporary constraint (*Latch*) that holds the *turret* within the moonpool.

The constraint removes all the degrees of freedom between the *turret* and the FPSO. The constraint *release at start of stage* is set to 2 which means that, at the start of stage 2, the constraint will disconnect the *turret* and then have no more effect on the model. The release has been timed to occur at the wave crest. It is advisable to also check the release at the wave trough, mid-descent and mid-rise to find the most conservative condition.

You can split your analysis into as many stages as you like, and these are specified on the *analysis* page of the *general* data form. For more details see the [Theory | Dynamic analysis](#) page of the OrcaFlex help.

After release of the buoy, the FPSO will move forward. Open *FPSO* from the model browser and go to the *calculation* page. The *primary motion* has been set to *prescribed*. Now move to the *primary motion* page. This shows that the vessel remains on station during stage 0 and stage 1, before moving forward with a velocity ramping up from 0m/s to 2m/s during stage 2. The velocity then remains constant at 2m/s during stage 3.

The stages can therefore be summarised as follows:

Stage	Description	Time
0	Build-up of waves	-17.5s to 0.0s
1	Full waves applied	0.0s to 17.5s
2	Links released at start and vessel moves forward with increasing velocity	17.5s to 22.5s
3	Vessel moves forward with no change in velocity	22.5s to 70.0s

At the top right of the [FPSO](#) vessel data form the *length* has been set to 206m. If this was set to *~* then OrcaFlex would assume the required vessel was the same length as defined on the [vessel type](#) data form. However, setting the *length* value results in OrcaFlex scaling the vessel type data to match the specified vessel length of 206m.

Now select [vessel types](#). On the [structure](#) page of the [vessel type](#) data form the [vessel type length](#) has been set to 103m half the required length of the vessel. OrcaFlex will therefore apply Froude scaling on the vessel data, increasing the response based on the ratio of required length to vessel type length.

However, this scaling method does not take account of all the response changes with increased vessel size. It is intended for concept and pre-bid analysis only i.e. where the vessel has not yet been selected. Where the vessel is known, then the actual data for that vessel should be used or the response and results will be wrong.

For further details about vessel Froude scaling, please see the [Modelling, data and results | Vessels | Vessel data](#) page of the OrcaFlex help.

The turret release is a rapid / discrete event occurring during the simulation. Such a sudden change to the system configuration typically requires a small time step to capture the system response, without the simulation becoming unstable. However, this is an isolated event and retaining this small time step for the whole simulation would result in a significant increase to the simulation runtime.

Instead, this example uses the implicit *variable* time step integration scheme. Open the [general](#) data form and select the [dynamics](#) page. Notice that the *use variable time step* option is ticked. This will act to reduce the time step when the turret release causes sudden changes and will increase the time step again when the change is past. However, the variable time step algorithm should be used with care. Every time a timestep is changed, a small amount of energy is added to the system. This is the nature of variable time step algorithms; meaning a constant time step should always be the first choice (whether implicit or explicit) and a variable time step should only be used to assist with isolated / discrete events where runtime with a small time step is impractical.

Another option is to use the explicit solver, provided that it yields reasonable runtimes. Explicit solvers have the advantage that stable simulations are almost always accurate – this is not the case for implicit solvers. If in doubt, it is often wise to perform spot-checks of your implicit solver results against results produced by the explicit solver.

## Results

If you have changed the views that were displayed when the file was first opened, go to the workspace menu and select *use file default* to re-load the default workspace file (alternatively, if you have reset the model, then re-opening the sim file will open it with the default workspace loaded). The workspace shows a shaded image view of the system and three time history plots. The upper right plot is the *Turret Z* position. The lower right plot is the *FPSO X* motion while the lower left plot is the *clash force* from the turret contacting the protective cage.

Start the replay for the whole simulation (from the *replay* menu or using the keyboard shortcut, *Ctrl + R*). A vertical line on each time history will show where the replay has got to.

At the beginning of stage 2, the turret will drop and then the FPSO will move away to be clear of it. Note the turret in this system does not bounce back enough to risk hitting the FPSO hull but this is something that should always be checked for.

The clash force plot reports that the *Shell of Turret* sees contact from another line at about 19 seconds. If you hide the *FPSO* and *Moonpool* objects (right click on the model browser) you will be able to see the protective cage. Zoom in on the cage and run the replay. The cage lines will change from pink to white when the turret contacts them.

Now open the workspace *A04 Disconnecting turret system mooring.wrk*. This shows the touchdown region of the left-hand mooring line. Run the replay again and watch how the chain bounces around as it is laid rapidly onto the seabed.

