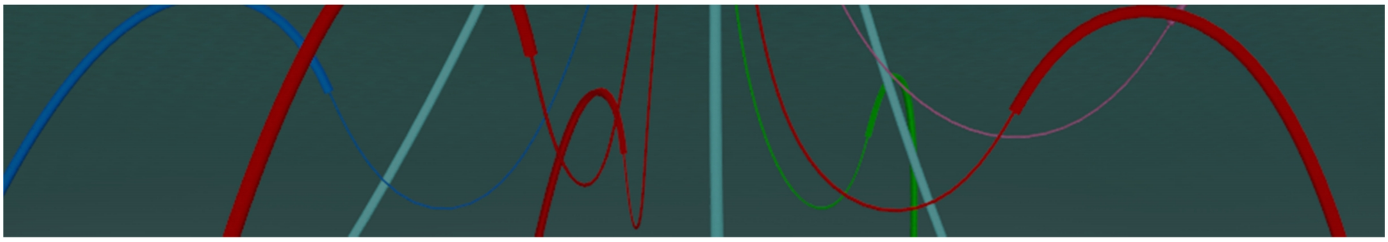
Orcina
Professional Software for Professional Engineers

OrcaFlex Newsletter
Version 10.2



In this Newsletter

Line feeding is the headline feature of the latest release of OrcaFlex, version 10.2. It has been asked for many times over the years – well, finally, it's here, so we make no apologies for covering it at length in this newsletter.

While line feeding is quite obviously a brand new facility, many of the other developments might be summarised as evolution rather than revolution (or much the same as before, only better), helping you to do what you did before but more easily and reliably and even, just maybe, more enjoyably.

Contents

Line feeding

Lines can now change their length…

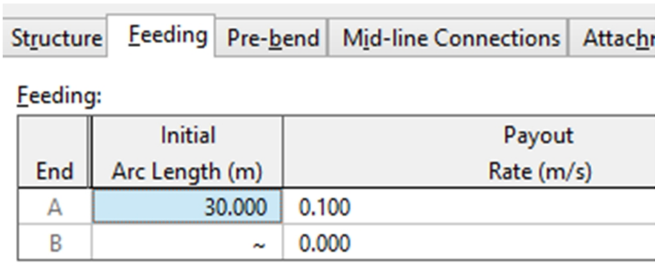
Line feeding is the big new feature in OrcaFlex 10.2. It is quite possibly the longest-standing request for OrcaFlex: we knew it would be hard, and indeed it has taken us two years to develop it fully. It turns out that just feeding additional nodes into a line isn't too difficult – but doing it in a way that avoids introducing lots of noise is really quite tricky.

Introduction

Line feeding is the process of adding or removing nodes to or from a line, at one or both ends, to extend it (paying out) or shorten it (hauling in). To represent payout in OrcaFlex previously, you had to have a 'pool' of line available, which you carefully arranged to avoid it influencing your model, and usually a winch object to gradually bring it into play. Now, all you need to do is tell OrcaFlex the payout (or haul in) rate you require for any given line.

To go into the details, we first need some terminology. We refer to the usual line, as defined by the line Structure data, as the **full** line, and that part of the full line which is currently in the model as the **active** line. Up to now, these two coincided – the whole of the line was active, all the time. Now, with line feeding, it is possible to have only part of the line active. The OrcaFlex 3D view always shows only the active line: the **inactive** part of the line is invisible and has no effect on the simulation.

With these terms, we can now set up the data which define how line feeding happens at each end of the line.

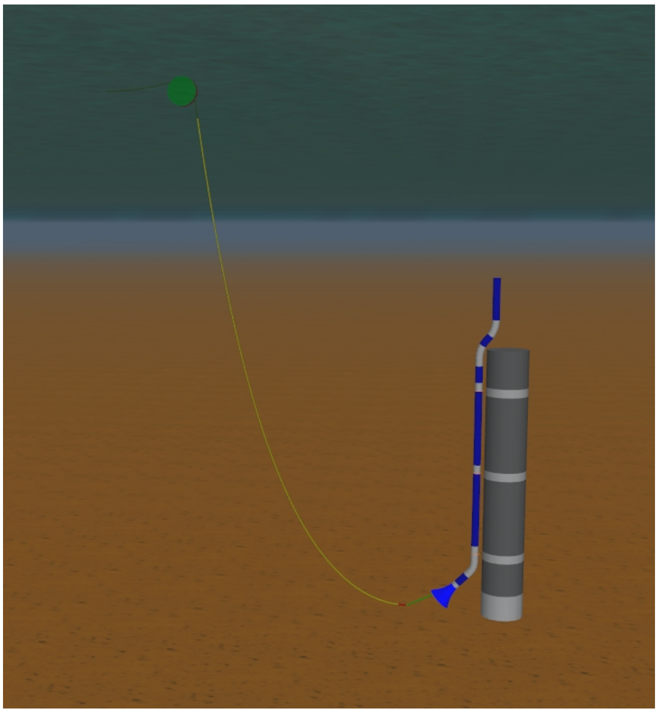


The initial arc length values determine the end points of the active line at the start of the simulation, with ~ meaning the corresponding end on the full line. The payout rate can be positive (pay out) or negative (haul in); it may vary with time, and you can choose whether or not it is ramped during the build-up period.

And that trivial amount of data is, essentially, all you need to begin using this new facility.

Example: J-tube pull-in

The J-tube pull-in example on our web site (orcina.com/SoftwareProducts/OrcaFlex/Examples/D Riser Installation) has recently been updated from the old, rather unphysical representation, which had a long length of line on a large, invisible, frictionless deck with a winch at one end to pay it out from the deck and at the other end to pull it up through the J-tube. Instead, we can now use the new line feeding facility at both ends of the line. Winches and notional decks are no longer required as workarounds!



Applications

Other line feeding applications were presented at the 2017 User Group Meetings. They have not been developed sufficiently to be included in the OrcaFlex examples, but the UGM presentations covering these applications are available for download at orcina.com/Support/UserGroup/2017.

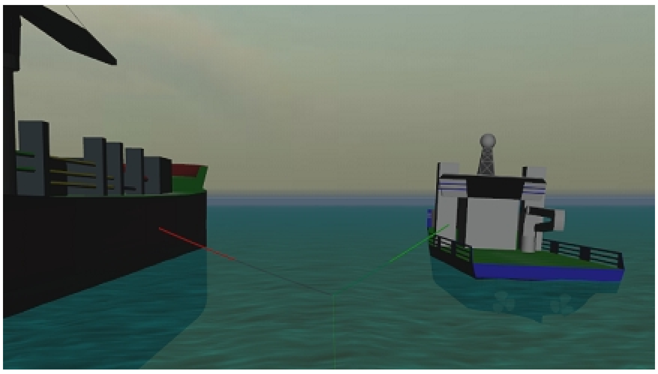
Our What's New video for OrcaFlex version 10.2 (orcina.com/SoftwareProducts/OrcaFlex/Videos) also illustrates the use of the line feeding facility, as well as some other new features.

Lowering

Lowering operations could previously be modelled by a winch (which do not include some effects, such as weight and drag), or by a series of static snapshots of a line (which missed out any dynamic payout effects). With the advent of line feeding, we can now capture the physics behind all of these. How important this is will depend on the deployment rate relative to motion and wave kinematics; in some cases, simple static snapshots may well suffice; in other, highly dynamic cases, the accuracy gained from line feeding may well be worth worthwhile.

Riser transfer

Modelling the process of riser transfer between, for example, an installation vessel and an FPSO, is another candidate for the line feeding facility.



Again, this could previously be done with winches: a pair of winches, in fact, connected to the riser ends, one paying out from the installer and the other hauling in to the FPSO – but OrcaFlex winches do not capture the drag loads which can be significant in the wave zone. Instead, we can now do this more effectively with a pair of feeding lines which include all the hydrodynamic effects, as well as self-weight etc., on the handling wires.

Growing line

As we said, growing the line without introducing too much noise was the difficult part of all this. Why? Well, you can't just add nodes to the end of a line: for a start, there is of necessity already a node there, so another one would result in a segment of zero length and hence infinite stiffness. So what we do, which you will see if you watch feeding in action very closely, is introduce new nodes a finite distance from the end, determined by the shortest viable segment factor

Growth Control Parameters

End	Shortest Viable Segment Factor	Use Smooth Growth
A	0.0010	☑
B		

This is illustrated using animation in the help file, under the modelling line feeding topic, orcina.com/SoftwareProducts/OrcaFlex/Documentation/Help/Redirector.htm?ModellingLineFeeding.htm.

Smooth growth

The other data there is an option to use smooth growth. This is fully described and (again) illustrated in the help – essentially, nodes are accelerated from zero up to the payout rate, rather than being launched into the active line at payout rate. We recommend that you use it if possible: it will usually help to reduce noise, though it can sometimes cause convergence difficulties.

Limitations

Some things just don't make sense without a constant line length – slug flow relative to fixed line lengths, for instance. These aren't ever going to be allowed in combination with line feeding. Others can, in principle, be made to work with line feeding but are hard and haven't yet had the time and effort devoted to them. These include: slave objects or attachments on the inactive line, wake interference, non-linear Young's modulus, and explicit integration.

Vessel calculation mode

Use the new way of 10.0 or the old way of 9.8…

This is something of an apology. First, a little background.

Second order fluid loads are typically calculated using Quadratic Transfer Functions (QTF), obtained from diffraction analysis program results. Diffraction programs include in their QTF results several second order load contributions that arise due to multiplication of first order load with first order response, or constant load with second order response. Those same interactions can arise naturally in normal time-domain simulation. As a result, there is a danger of double-counting some contributions to the QTF loads, we refer to these contributions as common second order loads.

OrcaFlex's handling of these common second order loads has varied over the years:

- Prior to version 9.5 no attempt was made to address this.
- Versions 9.5 to 9.8 used a method based on time domain filtering of the vessel response.
- Versions 10.0 and 10.1 modified the internal QTF data to remove contributions to common second order loads.
- Version 10.2 now gives you a choice between filtering and QTF modification.

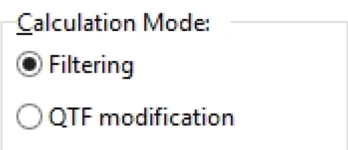
After releasing 10.0, we started to receive user feedback that the QTF modification method led to vessel motions that were less realistic than motions predicted by the filtering method. Some of these problems could be resolved by refining the OrcaFlex input data, and some were resolved by improvements to the QTF modification implementation. But there remained a number of users for whom QTF modifications was a backwards step.

One might then reasonably ask why we introduced QTF modifications in the first place. Well, there was another group of users for whom the filtering method lead to unrealistic motions and for whom QTF modifications was a significant improvement.

So why then, in v10.0, did we switch to QTF modifications and not leave filtering as an option?

- Our feeling at the time was that QTF modifications was theoretically a sounder approach.
- We didn't want to present yet another option to users in an already confusing aspect of the data input.
- Finally, maintaining two options increases the testing and maintenance workload for us.

However, with hindsight it is clear that we should have introduced QTF modifications as an option, with the filtering method retained. And that is now the situation in version 10.2. On the vessels Calculation page you now have the choice of calculation mode:



We're not going to get into the detail here of which mode to use. Essentially, you should choose the one which produces motions that you feel most comfortable with. If you want more information, or have any uncertainties, please contact our support team.

And so we're sorry for the inconvenience caused by removing filtering from 10.0 and 10.1. We've learnt from this and will try to be more circumspect in future!

Blog article

This is just a very brief summary of a rather complicated area. The entire topic has been considered on our blog in some detail: see orcina.com/blog/upcoming-in-orcaflex-10-2-vessel-calculation-mode

Line clearance results

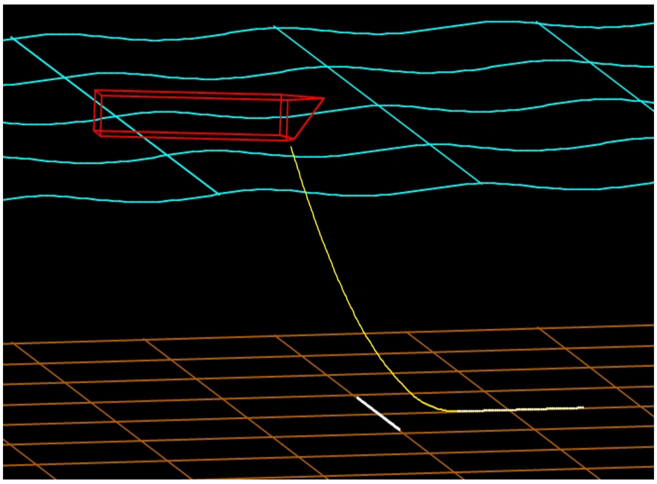
New results to better match some CA requirements…

Historically, OrcaFlex has offered results reporting the clearance between lines – the shortest distance, in 3D space, between them, either between their centrelines or between their contact surfaces.

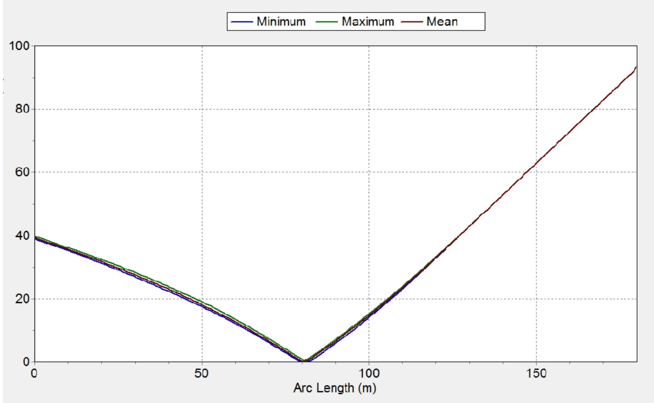
However, some industry design codes call for clearances to be measured horizontally or vertically (e.g. a mooring line crossing a pipeline). Version 10.2 introduces a number of new results variables to address these requirements.

The first of these to consider is named *Line Horizontal Centreline Clearance*. This is essentially the same as the existing *Line Centreline Clearance*, except that all coordinates are projected onto a horizontal plane before the calculation is performed. Closely related, the new *Line Vertical Centreline Clearance* is the vertical distance between the two points that determine the horizontal clearance. These two new results are available, in the usual way, either as time histories or range graphs.

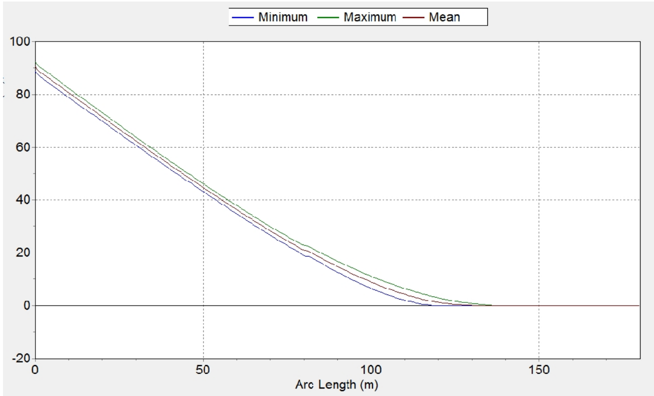
To see how they work, consider this simple system, in which a catenary riser crosses above a fixed line on the seabed:



The range graph of *Line Horizontal Centreline Clearance* for the riser is as follows,



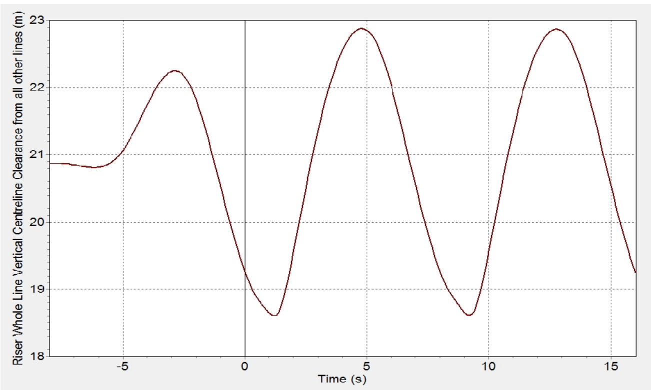
showing clearly that the crossing point is at arc length ~80 m. The corresponding range graph for *Line Vertical Centreline Clearance* for the riser is



which in turn shows that the vertical clearance at 80 m arc length (the crossing point) is around 20 m.

To make this process even simpler we have introduced a set of new *Whole Line Clearance* results. As the name hopefully suggests, this is a single result for the whole line as opposed to at a specific point on the line. Consequently, any position value specified for any of the *Whole Line Clearance* results is ignored. It also follows that *Whole Line Clearance* results are not available as range graphs.

Whole Line Horizontal Centreline Clearance is defined to be the minimum, taken over all points on the line, of *Line Horizontal Centreline Clearance*: for our example, this is simply zero throughout. And *Whole Line Vertical Centreline Clearance* is then defined as the vertical distance between the two points that determine the *Whole Line Horizontal Centreline Clearance*. In this case, the graph looks like this,



So the design code result of interest would be the minimum vertical clearance between the riser and the line on the seabed. Visual inspection show this to be about 18.6m, but we can easily get the exact result via Linked Statistics (for instance) or any of the usual post-processing methods.

Having added whole line clearance results for horizontal and vertical clearance, we decided for completeness to do the same for the other clearance results. *Whole Line Centreline Clearance* reports the minimum, taken over all points on the line, of *Line Centreline Clearance*. *Whole Line Clearance* is defined analogously.

Apologies for the lengthy names of these results, but there really is no other way to be sufficiently precise about what these results actually represent!

Documentation

No more scruffy equations in the help ...

The OrcaFlex help is now, we hope, a whole lot easier to read than before. After years of frustration with scruffy looking mathematical equations in our HTML help file, we have completely revamped the documentation for 10.2.

Fundamentally the underlying content is just as before, although we have reviewed it and tidied it up. The biggest change is that mathematical equations are now typeset in a much better way, instead of the previous ordinary text. Compare and contrast this fragment of the help topic on wave drift damping: the old way,

$D^*_{ij} = \sum_m \int_{-\infty}^{\infty} 2\ D_{ij}(\beta_m, f) S_m(f)\ df$

where

$D_{ij}(\beta_m, f)$ are the coefficients of the frequency-dependent wave drift damping

and the new, much more readable,

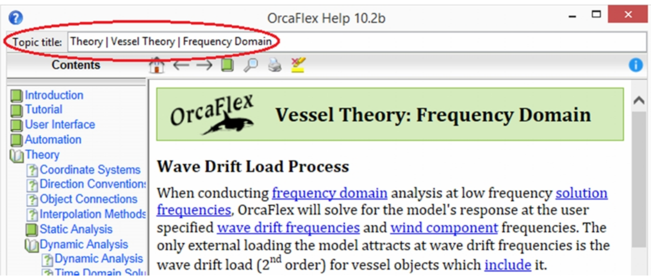
$$C_{ij} = \sum_m \int_{-\infty}^{\infty} 2\, D_{ij}(\beta_m, f)\, S_m(f)\, df \qquad (3)$$

where

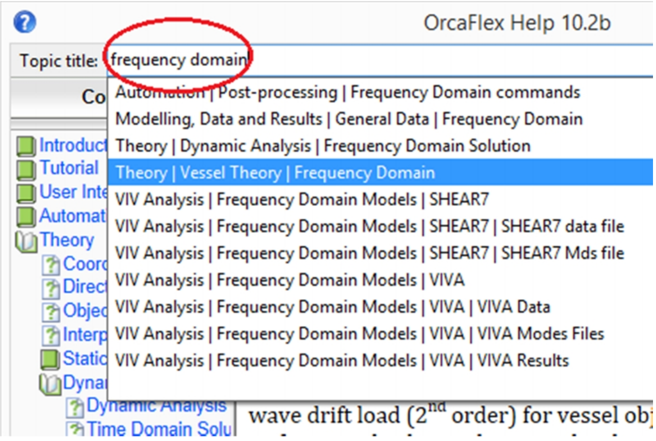
$D_{ij}(\beta_m, f)$ are the coefficients of the frequency-dependent wave drift damping

For those interested in how we have done this, there is more on the blog at orcina.com/blog/upcoming-in-orcaflex-10-2-documentation.

To make all this possible, we have had to implement our own help viewer program. This has certain benefits because we have been able to implement some useful functionality. In particular, F8 puts the focus in the *Topic title*,



This is automatically filled out when you select a topic in the contents pane on the left, or come into the help by using F1 on the UI. Furthermore, you can also type your own text in here and OrcaFlex will search for it



and list the suggestions it finds. This isn't an exhaustive search of the whole help text (which you can also do), just the titles – but it can be a quick way to find what you are looking for.

It is also an easy way to share a specific topic in the help with someone: just copy the text in the topic title, send it to them and tell them to paste it into the same field in their help browser.

To share a topic with someone without the help browser, you can instead create a URL for the topic at the online help on our web site: CTRL+F8 brings up an *Online help link* control, containing the URL, which can similarly be shared. Incidentally, the help is freely available to all at orcina.com/SoftwareProducts/OrcaFlex/Documentation.

In the help browser, as in OrcaFlex, pressing F1 opens a help window – in this case giving help on the help, as it were, listing these and other shortcuts. We've tried hard to make the help fully accessible by keyboard.

Essentially though, nothing fundamental has changed. You can still press F1 anywhere in OrcaFlex and be taken to the relevant documentation. It just looks better now!

Other developments

Some of the many other recent developments...

Morison elements

Sometimes, when modelling large diffracting bodies, additional hydrodynamic drag is required. You might need, for instance to include the influence of quadratic viscous drag on the bracing of a semi-submersible.

Previously, you would have done this by connecting other OrcaFlex objects such as buoys or single segment lines to a vessel. Whilst this approach works, it has some disadvantages:

- Building the models can be rather inconvenient if you have to use lots of single segment lines.
- It can be tricky to make sure that you disable undesired effects such as mass, buoyancy etc, leaving only the drag terms.
- The runtime can be increased by all the objects used to represent the elements.

In version 10.2, we now have a dedicated object – a Morison element – which can be connected rigidly to vessels and 6D buoys. "Morison", because they attract pure Morison drag and have no other effect.

They have a *type*, in the manner of line types, support types etc., defining their drag properties; any number of elements of each type can then be attached to vessels and 6D buoys, each with individually-specified position, orientation and length.

Their use should be quite clear, so there isn't much more to say! In the future we could imagine extending their capability to offer more effects such as mass, buoyancy, added mass etc. – let us know if this would be of interest to you.

Slave vessels and 3D buoys

Vessels and 3D buoys can now be connected as slaves of other objects. For 3D buoys, we made this change largely for the sake of completeness. They aren't frequently used this way, but it was easy to do and it felt right to remove an unnecessary limitation.

Vessels, on the other hand, are a different story. Whilst connecting a vessel as a slave is rather esoteric, there are some applications where this is very valuable.

The sort of example that motivated us to make this change are systems with two or more floating bodies which are coupled mechanically, for instance certain offloading systems. If the bodies are large enough to require the use of diffraction data then they need to be modelled with OrcaFlex vessel objects. The mechanical couplings would commonly be modelled using constraint objects, but this does require one or more of the vessel objects to be connected as slaves to other objects. The new functionality makes this sort of analysis possible.

Note that the other vessel is automatically *Calculated* by dint of being a slave: its motion is determined by its master, so we set its primary motion to Calculated (6 DOF), include all 6 DOF in statics, and exclude any superimposed motion.

One final caveat. Slave vessels are not supported by the frequency domain solver. There are significant technical hurdles to overcome before we could support this feature in the frequency domain. If we see sufficient user feedback that the feature would be desirable and useful then we will look into adding frequency domain support in a future release.

Support and constraint release

OrcaFlex supports can now be released during the simulation, in the same way as line, link and winch connections, using the usual *Release at Start of Stage* data. Once released, the supports are no longer active in the model and apply no forces or moments to supported lines.

This can be especially useful for cases in which we only need a support to persuade statics to place a line where we require it, such as on one side of an elastic solid. If the support plays no part in the dynamic analysis, simply tell OrcaFlex that the support is to be released at the start of stage 0.

Constraint objects can also be released in the same way. When the constraint is released the out-frame and in-frame move independently of each other. This enables you to release connections between objects which don't themselves have this facility. Consider, for instance, a buoy connected as a slave to a vessel. If the buoy is connected directly, it cannot be released because buoys can't do this. If instead you connect a constraint as a slave of the vessel, and fix the constraint's DOFs, and connect the buoy as a slave of the constraint, then you have the same situation with, in addition, the facility to release the connection (the constraint) during the simulation.

Line statics policy

OrcaFlex 10.2 introduces two new data items which give fine grained control of the line statics algorithms. These new options are especially valuable in systems with coupling between lines, either through direct connectivity or indirect coupling through springs. The new functionality is intended to make statics convergence more robust.

Typically you should leave these new data at their default values, and the static analysis will be fast and reliable (but note that, when loading data files prepared with older versions of OrcaFlex, the line statics policy data reproduce the behaviour of those older versions unless you explicitly change their values). Only if you encounter especially troublesome convergence problems should you need to look more deeply into the algorithms controlled by the line statics policy data.

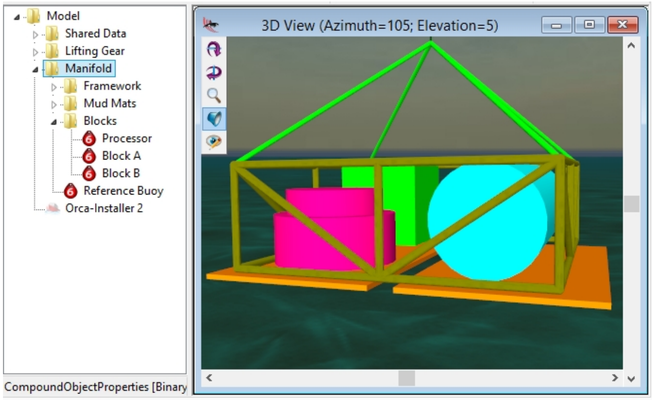
There isn't the space here to go into how these policies actually work. The help browser is, as usual, your reference for this. In addition, if you are having problems with statics convergence, or just curious to know more about the statics calculation in general, you will almost certainly find the detailed blog post we published on this subject very useful indeed: orcina.com/blog/upcoming-in-orcaflex-10-2-line-statics-policy.

Hints & tips

A few new features to aid usability…

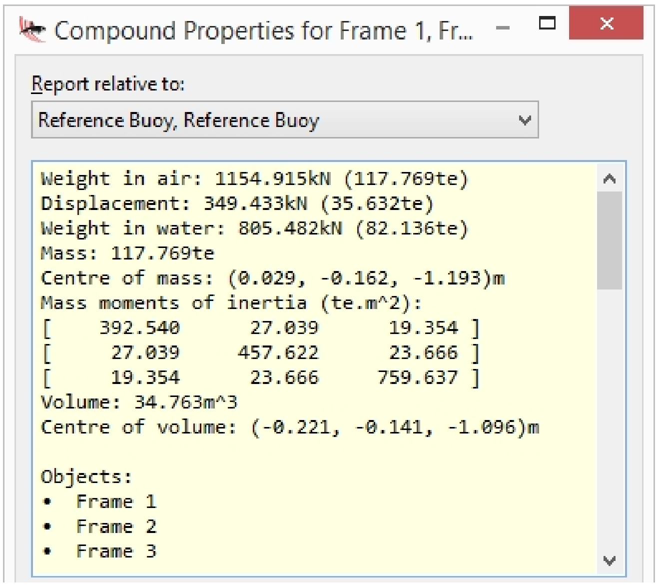
Compound object properties

Complex structures are often modelled in OrcaFlex using multiple objects connected together. A common example would be a manifold modelled using a combination of lumped buoys and single segment lines, all connected together so as to move as one rigid object:



When building such a model, there is plenty of scope for error. For instance, it is easy to forget to account for some part of the structure, or equally to double count parts of the structure. A common way to detect such mistakes is to calculate the combined mass, volume, centre of mass, centre of volume, etc. and compare with known values: now, OrcaFlex can do this calculation for you.

In this example, we have used the groups facility of the model browser to create a Manifold group containing all the constituent rigidly-connected model objects. Right-clicking on the group and selecting Properties gives the compound properties of the whole ensemble,



Note that the centres of mass and volume are *Reported relative to* the given object. OrcaFlex will pre-select the single master object here, if there is one, but you are free to select any of the objects here.

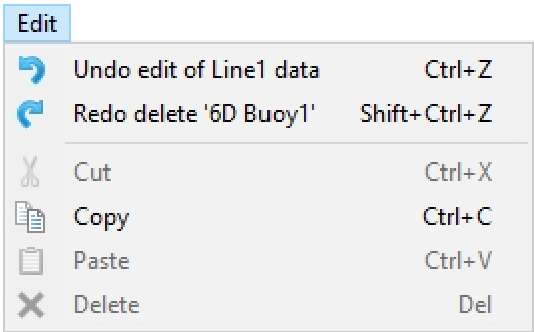
It's certainly convenient to use groups for this, but you don't have to. Any combination of multiply-selected objects can be used, but you should be aware that OrcaFlex assumes that they are all rigidly connected and move as a single body. If you show properties for two unrelated objects, then the output will be misleading.

Undo

Everything that runs on Windows has an undo, right? If you make a mistake, you just hit CTRL+Z to recover your carefully crafted spreadsheet calculation, 20 pages of text, whatever – we take it for granted. Well, not quite everything – and not, until now, OrcaFlex. We're somewhat embarrassed by the number of times users have commented on this missing functionality. Unfortunately, having started out without it, it was a large task to then incorporate it into the program.

Well, we bit the bullet and, as of version 10.2, OrcaFlex has comprehensive multi-level undo and redo. Only actions that affect data are undoable, eg, editing data forms, or creating, destroying, renaming or dragging objects, but not opening new windows, zooming views, etc. Otherwise, it's what you're used to elsewhere. It's just undo. Note that the undo list is cleared when you start a new model or load a file.

Actually, it's more than just CTRL+Z. The Edit menu includes a short description of what action will be undone or redone, to help you avoid getting lost when reverting a large number of actions.

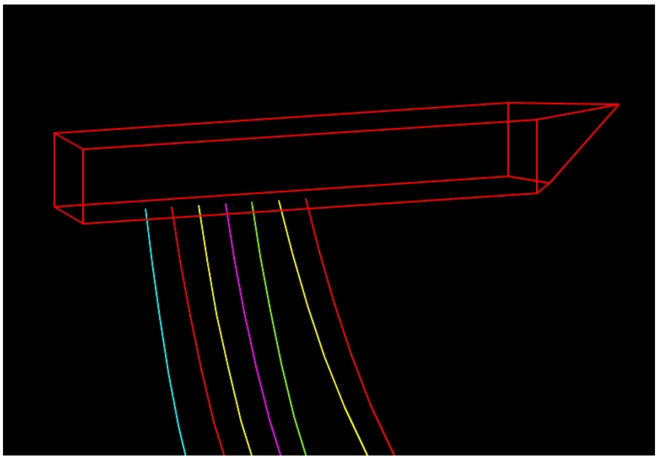


There is, in principle, a limit to how much you can undo (measured in bytes of memory, rather than the number of actions). Once you reach this limit, the oldest actions are thrown away and are no longer undoable, to make room for the most recent ones.

In practise though, the limit is so large that you are very unlikely ever to be affected by it. You should be able to build a complex model, adding objects and setting data, undo repeatedly all the way back to an empty model, and then redo all the way forward again to the completed model – try it – but perhaps on a test model!

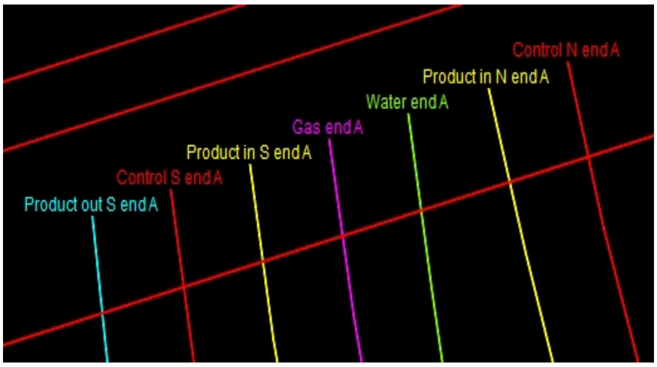
Labels on 3D views

In version 10.2 we have introduced a couple of features that are intended to help with visualisation of complex models. Firstly, consider the following 3D view of an FPSO with a number of jumper lines:



The model uses colour to distinguish between the many different jumper lines. Whilst this is certainly helpful, there are only so many different colours that the human eye can identify. Furthermore, it is hard to remember which line is which colour, especially if you are working with multiple different models at any one time or working on a model infrequently.

Starting with version 10.2, it is now possible to have the names of the objects (per the model browser) displayed on the 3D view,



You can toggle name labels on and off from the main view menu, from the 3D view context menu, or with the keyboard shortcut SHIFT+CTRL+Y.

In a further enhancement, you can also add arbitrary text using an OrcaFlex shape with the new *Label* type. It may seem a bit odd to use shapes this way, however shapes were already sometimes used for annotation, so we decided add the Label functionality here rather than create an entire new object for this purpose. You can, as with any other type of shape, fix it in space or connect it to another object with a given offset and orientation.

These labels are only available in the wire frame view – that seems to us to be their natural home and where we think they will be most useful.

The Back Page

News

We aim to publish these Newsletters yearly, to complement each OrcaFlex release. Around the same time we also produce a video (available from orcina.com/SoftwareProducts/OrcaFlex/Videos) which showcases those features which are best seen dynamically – line feeding being a prime example.

We do also try to keep you up to date with developments on a more frequent basis. Our company **LinkedIn** page has been going for a while now. We're at linkedin.com/company/orcina-ltd. New followers are always welcome. And the **OrcaFlex Blog** has already been mentioned a few times as a source of more detail on some of the version 10.2 developments, and these are all wrapped up in a recent post, orcina.com/blog/orcaflex-102. Future posts will cover, among other things, 10.3 plans and developments.

Out and About

As well as the usual mix of training courses and UGMs (see below), 2017 saw us exhibit at Subsea Expo (Aberdeen), Oceanology (US), OPT (Amsterdam), the Subsea Tieback Forum (San Antonio), Ocean Business (Southampton), OTC (Houston), OMAE (Trondheim), Offshore Europe (Aberdeen), Oceans (Anchorage), and Oceanology (China).

For 2018, current plans are to exhibit at Subsea Expo (Aberdeen), Oceanology (London), Subsea Tieback (Galveston), and to attend Offshore Wind (Glasgow), OPT (Amsterdam), Floating Offshore Wind Turbines (Marseille) and OMAE (Madrid).

Training

OrcaFlex training courses proved increasingly popular in 2017, particularly our course on automating OrcaFlex with Python.

Planned courses, together with other events, are listed at orcina.com/UpcomingEvents. In addition, we are always open to requests for training, whether it is our standard introduction to OrcaFlex, Python automation, or more advanced and tailored to your specific requirements.

OrcaFlex User Group Meetings

In 2017 we again held 11 User Group meetings at various locations around the world. This year the locations were Houston, Beijing, Perth, Kuala Lumpur, Singapore, London, Aberdeen, Amsterdam, Oslo and Rio. We also held our 'Introduction to Python Workshop' in Kuala Lumpur, Perth and Rio de Janeiro. Content from these can be found at orcina.com/Support/UserGroup. Thanks to all who attended and contributed to their continuing success.

2018 UGMs will run between September and December – we'll post details of these and any further exhibitions on orcina.com/UpcomingEvents and on LinkedIn.

Future OrcaFlex Developments

Our development list depends largely on client feedback. Much of this comes from the feature feedback notes you give us at the UGMs – so a big thank you to all who kindly take the time to do this. But suggestions are welcome at any time, especially if you can add a note justifying their importance. This is our current list:

- features related to mooring analysis
- diffraction analysis
- aero-elastic modelling for wind turbines
- spatial variation and spatial cohesion for wind
- restarts
- improved lateral soil modelling
- thermal / pressure expansion & contraction in pipes
- line results at nodes as well as mid-segment
- software-based licencing and electronic distribution

We aim to include as many of these as possible in the next couple of releases. As ever, though, this is not definite – some may take longer than expected to implement. And these are just the headlines – we add many other improvements in each development cycle.

Orcina Agents

Orcina is supported in its marketing and technical support activities by the following agents:

USA, Canada & Mexico

Paul Jacob & Dongmei Chu
pj@jtec-tx.com, +1 713 398 9595
dchu@heronoffshore.com, +1 832 725 2438.

Malaysia, Indonesia & Singapore

Herman Perera
herman.perera@zee-eng.com, +60 (03) 7877 8001.

South Korea

Hyunwoo Jang
hyun.j@sacsko.com, +82 2 421 8018.

South America

Nelson Galgoul
nsg@nsg.eng.br, +55 21 99995 9212.

India and Middle East

Tarun Rewari
info@aryatech.net, +91 11 46 01 81 02.

China

Yujing (Jean) Chen
orcaflex@richtechcn.com, +86 10 8446 7760 / +86 1812 129 2356.

If you have any questions, comments or general enquiries, please contact us at

Orcina Limited
+44 (0)1229 584 742
orcina@orcina.com
orcina.com