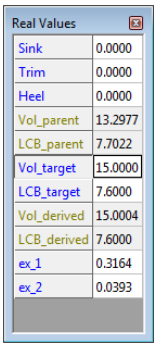
Automated Variation of Hull Form

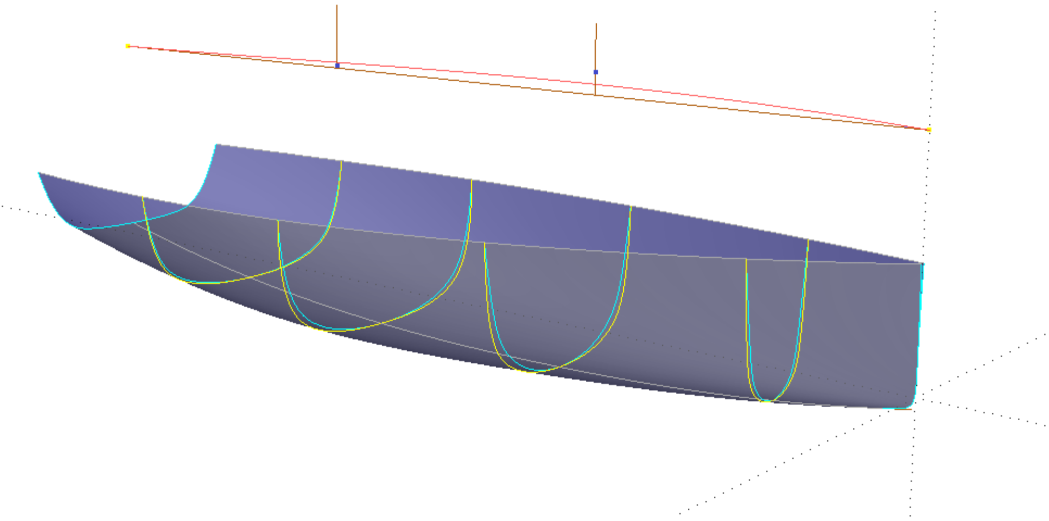
Derived Mastercurves for Target Values of Volume and LCB

by Reinhard Siegel



Real Values

Sink	0.0000
Trim	0.0000
Heel	0.0000
Vol_parent	13.2977
LCB_parent	7.7022
Vol_target	15.0000
LCB_target	7.6000
Vol_derived	15.0004
LCB_derived	7.6000
ex_1	0.3164
ex_2	0.0393



Content

Introduction

The design of a hull form is an iterative process. This process must converge. The iterative steps must therefore be smaller and smaller, the hull shapes are becoming more similar. Inevitably, the point is reached where the shape meets the requirements, only the displaced volume could be a little larger, for example, and it would be nice if the LCB is a few centimeters more aft. Otherwise everything fits. Then the mastercurves must be changed a few millimeters here and there, while checking permanently hydrostatics and fairness. This fine tuning is time consuming.

The approach described below makes it possible to automatically generate new mastercurves from the mastercurves of an existing hull so that target values of volume and LCB are obtained. Fairness, the outer contours of the hull and the overall hull form character are preserved.

This tutorial was inspired by the study of H: Lackenby's method, which is reported in the appendix.

Abbreviations used:

cp: control point (support point)
mc: master curve = support curve

cp1, cp2, ...: denotes 1st, 2nd, ... point in the list of supports of a curve. It is not an actual entity name.
mc1, mc2, ...: denotes 1st, 2nd, ... curve in the list of supports of a surface. It is not an actual entity name.

In the following the terms used for point, curve and surface types are those of MultiSurf. This may serve the understanding and traceability.

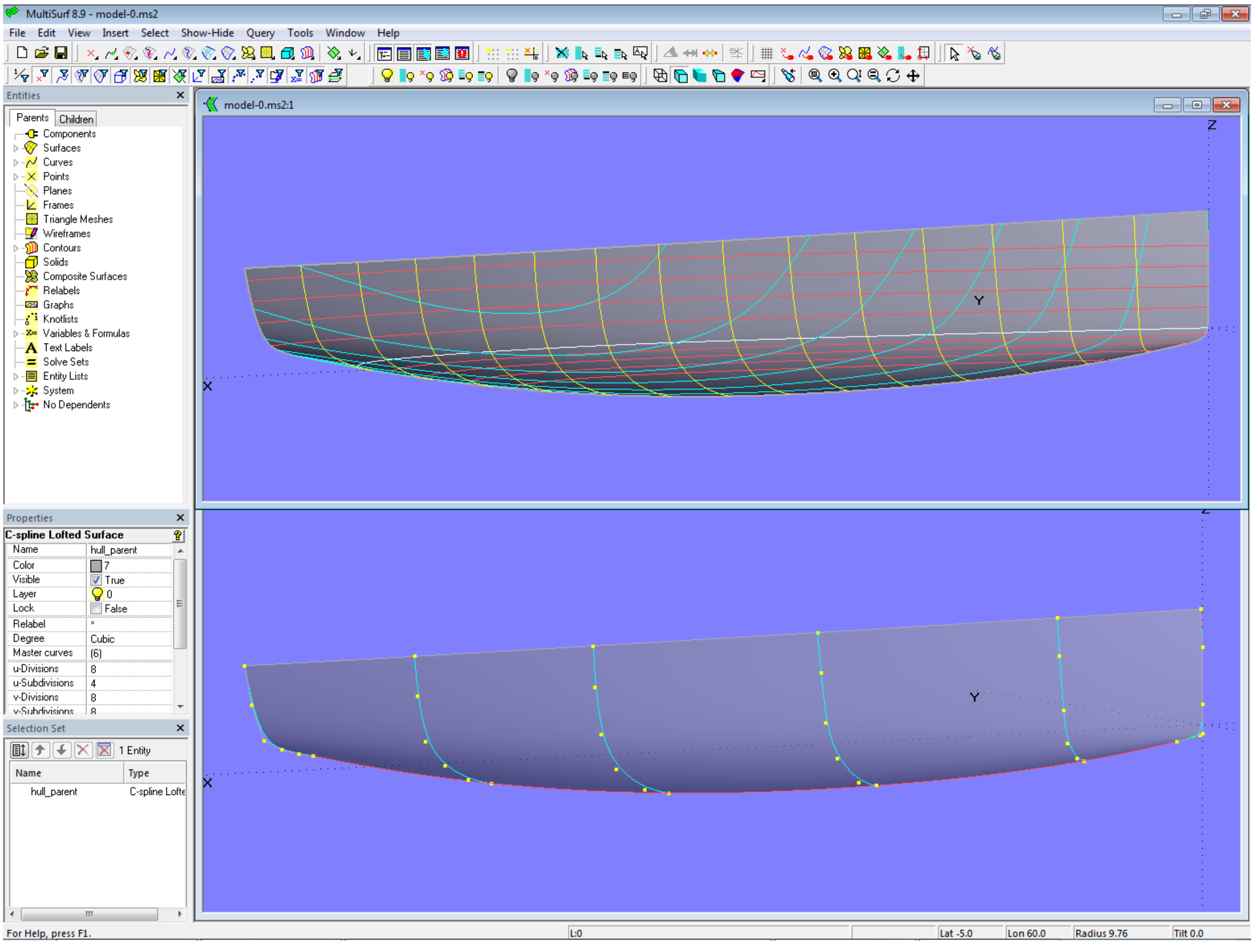
1 Parent hull

Starting point is the model of a hull that is basically finished. Prominent curves like side deck, bottom contour, stem, run of the stations - everything is fair and looks as imagined. If it were not for the desire that the displaced volume on the design waterline could be slightly larger or smaller, and/or that the longitudinal center of buoyancy LCB should be a little more forward or aft.

So one begins to move the control points of the mastercurves, which support the surface, always keeping track of fairness and hydrostatic values. If a little more volume is needed, the mcs are changed to get fuller stations. If LCB should be more aft, the mcs are edited to make the forebody more slender and add volume aft. How this editing of the shape of the mcs can be automated is described below.

Model *model-0.ms2* shows the parent hull, a C-spline Lofted Surface supported by 6 B-spline mastercurves, each controlled by 6 points.

We will assume that the hull is basically finished, except for volume and/or LCB. The task is therefore to change the shape of the inner mcs so that certain values for volume and LCB are attained. However, the character of the curve of the mcs is to be retained - a U-form frame should not become a V-form frame.



C-spline Lofted Surface

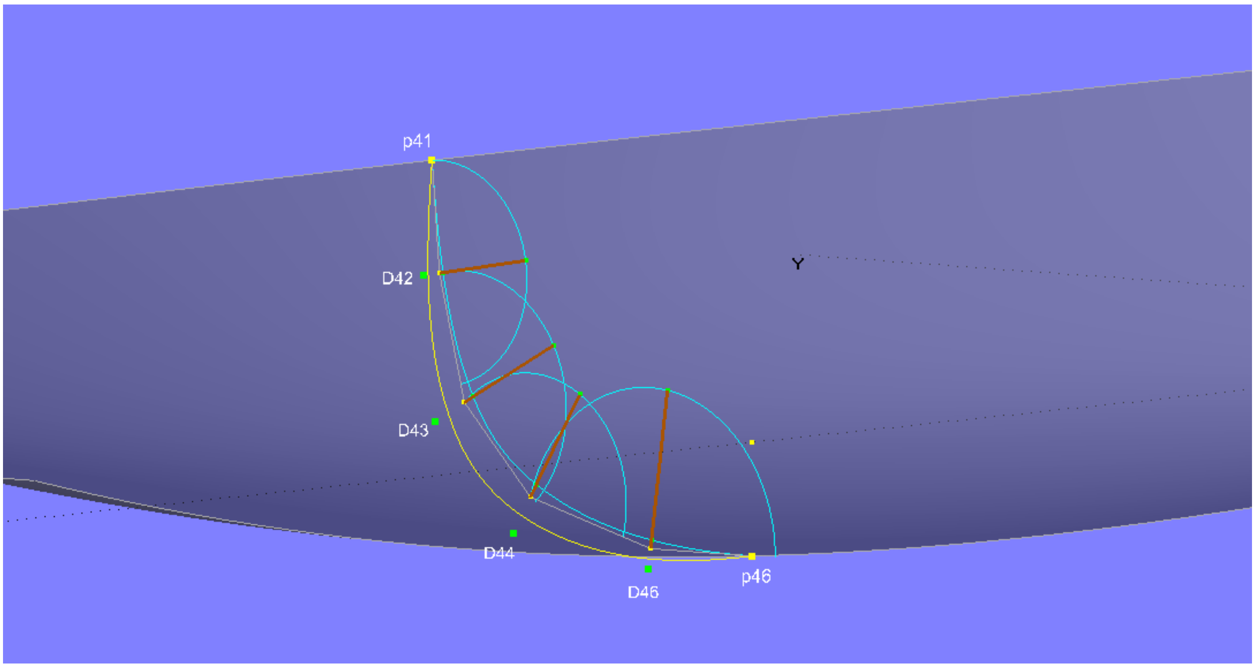
Name	hull_parent
Color	7
Visible	☑ True
Layer	0
Lock	□ False
Relabel	*
Degree	Cubic
Master curves	(6)
u-Divisions	8
u-Subdivisions	4
v-Divisions	8
v-Subdivisions	8

Name	Type
hull_parent	C-spline Lofte

Model-0.ms2 - the base hull is a C-spline Lofted Surface with 6 B-spline mastercurves.

2 Derived mastercurves - basic principle

To make a mc more full-shaped or less without changing the character of its run, its inner cps must be moved both vertically and horizontally. This experience from practice leads to the approach described in model *model-1.ms2*. Using mc4 as example a derived mc is constructed by creating a new cp for each inner cp of the parent mc. This derived cp lies on the bisector of the angle between its adjacent polygon segments, and is offset by a certain distance from the parent cp.



Model-1.ms2 - basic mc and derived curve

First, the bisector of the angle between cp1, cp2 and cp3 is determined. For this, the Arc curve_42 (Type 2) is created with the points p41, p42 and p43. On the Arc curve_42 lies the Bead bead_42. Since its parameter value is set to t = 0.5, Line line_42, which connects p42 to bead_42, is the bisector. On line_42 is the Arc-length Bead D42.

These construction steps are then repeated with the cps (p42, p43, p44), (p43, p44, p45) and (p44, p45, p46).

The result are the 4 Arc-length Beads D42, D43, D44 and D45, each lying on the bisector of the control point polygon of mc4. In this way, there is a close relationship between the arrangement of the cps of the parent mc and the derived mc.

The lines on which the new cps are located are now known. At what distance d to the parent cps they are offset, is not fixed yet. Therefore, a maximum value of deviation (displacement) from the parent mc is defined and distributed along the mc so that the derived curve begins without displacement at cp1 (p41), ends without displacement at the last cp (p46) and in between reached the given maximum deviation.

The following expression is used as the function of deviation:

d = c x (t –t^2)

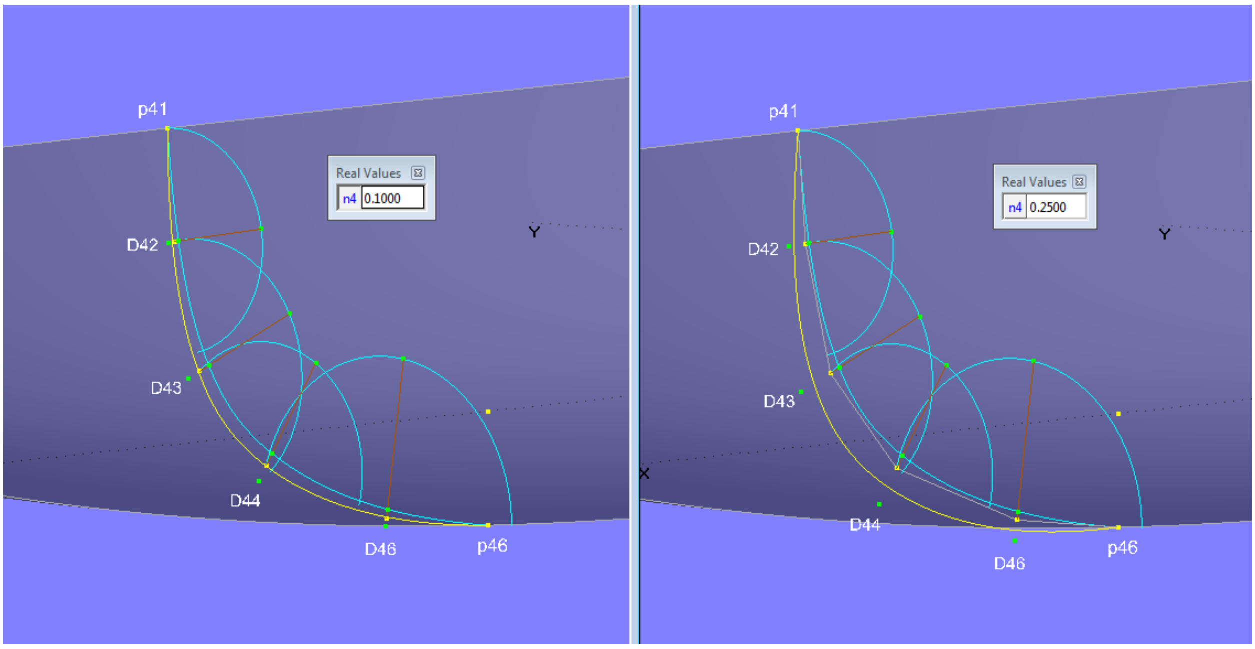
Here, c is a constant and t is the parameter value of the curve point on the parent mc, where it is cut by the bisector.

It can be seen that for t = 0 and t = 1, the function d assumes the value zero.

To determine the constant c, it is assumed that for t = 0.5 the maximum value of the displacement d should be equal to the value n. This results in the function d to:

d = n x 4 x (t – t^2)

This approach is implemented in model *model-1.ms2* using some variables and formulas.



n4 0.1000

n4 0.2500

Model model-1.ms2 - variable n4 controls how much the derived curve deviates from the parent curve.

First, the intersections of mc4 with the bisectors (line_42, line_43, line_44 and line_45) are determined by the Intersection Beads e42, e43, e44 and e45. The Variable n4 defines the maximum displacement.

Thus, the distance d of each derived cp from its parent cp can be calculated. For example, for cp2 (p42) of the parent mc the corresponding derived cp is D42 (Arc-length Bead). Its distance from p42 is calculated by the Formula d42, which uses the following expression (the Function TPOS (e42) returns the t-parameter value of the Bead e42):

d42 = n4 x 4 x (TPOS (e42) - (TPOS (e42))^2)

This formula value d42 is now set in the definition of the Arc-length Bead D42 for the quantity "Arc dist".

Analogously, with the Formulas d43, d44 and d45, the distance of the remaining Arc-length Beads (D43, D44, D45) from their parent cps is calculated and used for "Arc dist".

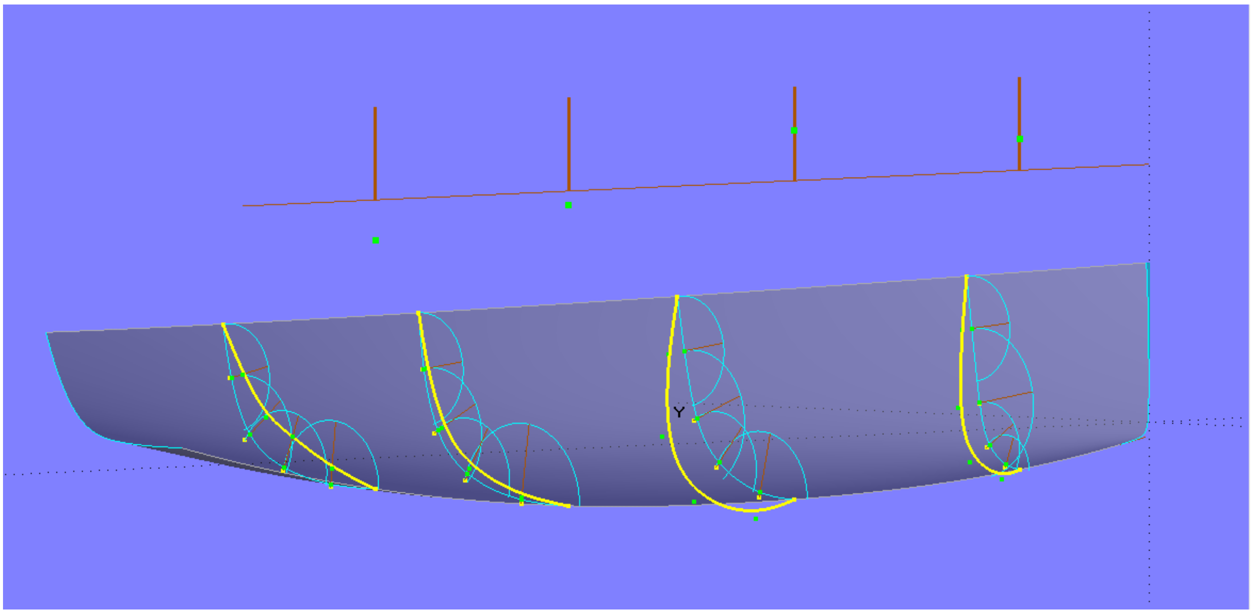
Finally, there is the generation of the derived mastercurve mc4_derived with these control points: p41, D42, D43, D44, D45 and p46.

Changing the value of Variable n4 changes the deviation of the derived curve from its base curve. If n4 takes a large value, the last but one cp (D45) can be offset according to the deviation function so far outward, that its Z-coordinate is smaller than the Z-coordinate of the last cp (p46), which is on the centerplane (dy = 0). Since then the derived curve runs diagonally from below against the centerplane, a negative ridge would arise between the two hull sides. Therefore, a conditional branch ensures that D45 is not lower than p46. This condition is built into *model-3.ms2* and *model-4.ms2* using the IF-Function.

3 Derived hull

In model *model-2.ms2*, this construction of the derived mc, explained in the foregoing, is repeated similarly for the remaining inner mcs mc2, mc3 and mc5. Note that the first and last mc (mc1, mc6) are not derived.

With mc1, the 4 derived mcs and mc6, the C-spline Lofted Surface hull_derived is created. The Contours stations_derived, buttocks_derived, wl_above_derived, wl_below_derived and dwl_derived show the lines of the derived hull form.



Model model-2.ms2 - derived mastercurves

In contrast to *model-1.ms2*, the maximum displacement n of the inner mcs is not defined here by variables. Instead, the quantities n2, n3, n4, n5 are determined using a kind of "slider". For example, at the longitudinal position of mc4 there is the vertical Line line4 on which resides the Bead bead7. The quantity n4 is the distance of bead7 from the starting point of the line (XYZBead bead3). It results from this simple formula (the ZPOS function returns the Z-coordinate of a point):

n4 = ZPOS(bead7) - ZPOS(bead3)

By shifting one of these "slider-beads", the deviation of the derived mc changes.

Those "slider-beads" are independent of each other. It is obvious that there must be a relationship between their position in order to make the deviation of the derived mcs from the parent mcs vary in a harmonic fashion in the longitudinal direction.

4 Longitudinal variation of mastercurve deviation

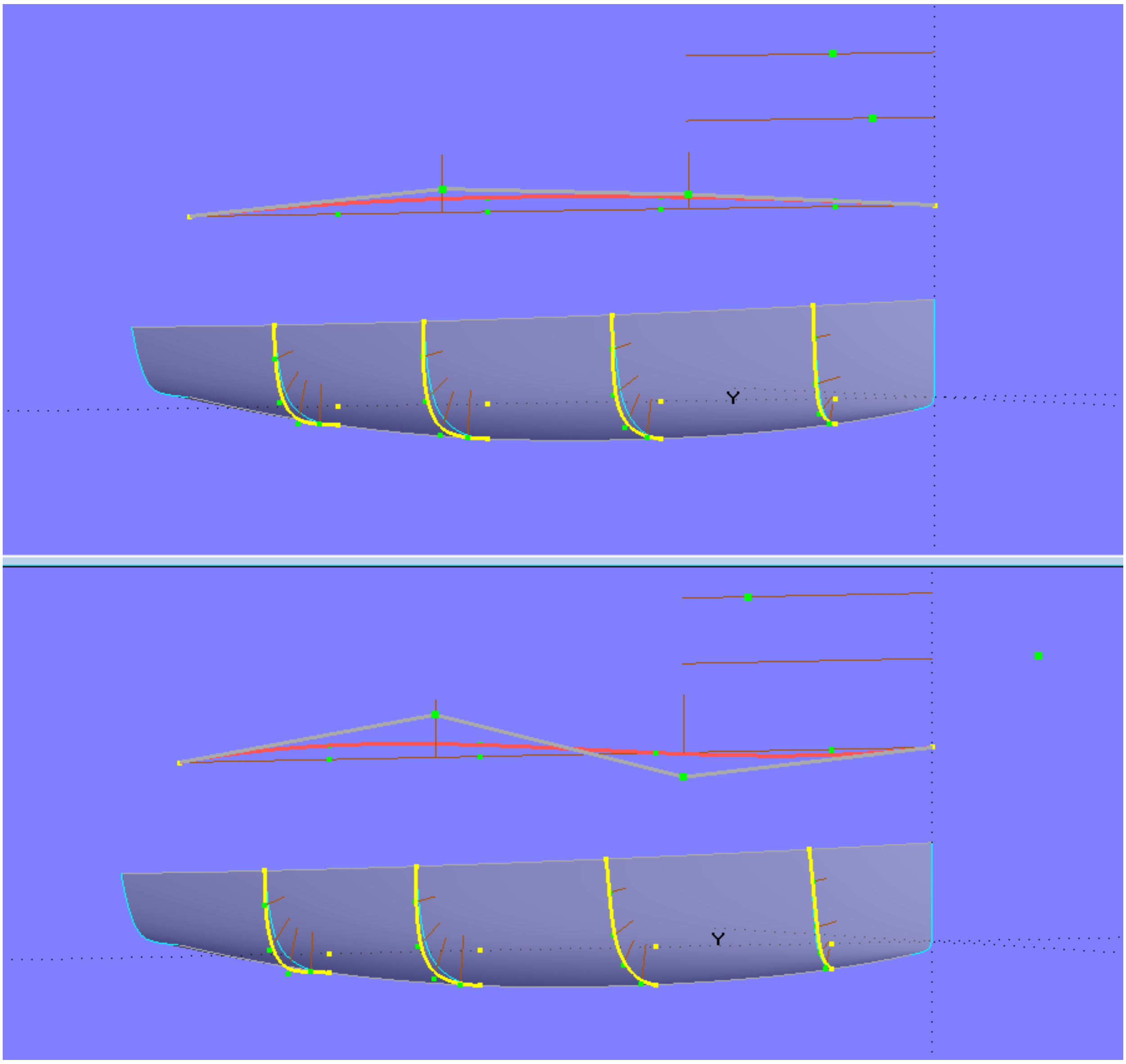
How this requirement can be implemented is shown in model *model-3.ms2*. The B-spline Curve curve_x determines the variation of the deviation of the derived mcs along the hull. This displacement curve, which is on the centerplane and relative to Line line1, is supported by only 4 cps. With 2 inner cps curve_x can show an inflection point. In this way, for example, the derived mcs in the front of the hull can be made a bit slimmer, while in the rear they can show more fullness than the parent mcs.

Note that the run of curve_x is controlled by the two Beads bead3 and bead4, which are located on the horizontal lines to the left above the bow ("slider-beads"). When these points are moved, the displacement curve curve_x changes accordingly.

In order to determine the maximum displacement n at the location of one of the inner mcs, line1 and curve_x are cut at the X position of the mc. The distance between the intersection point on line1 and the intersection point on curve_x is the wanted maximum displacement at the location of the mc. For example, the expression of Formula n3 for the derived mc3 reads:

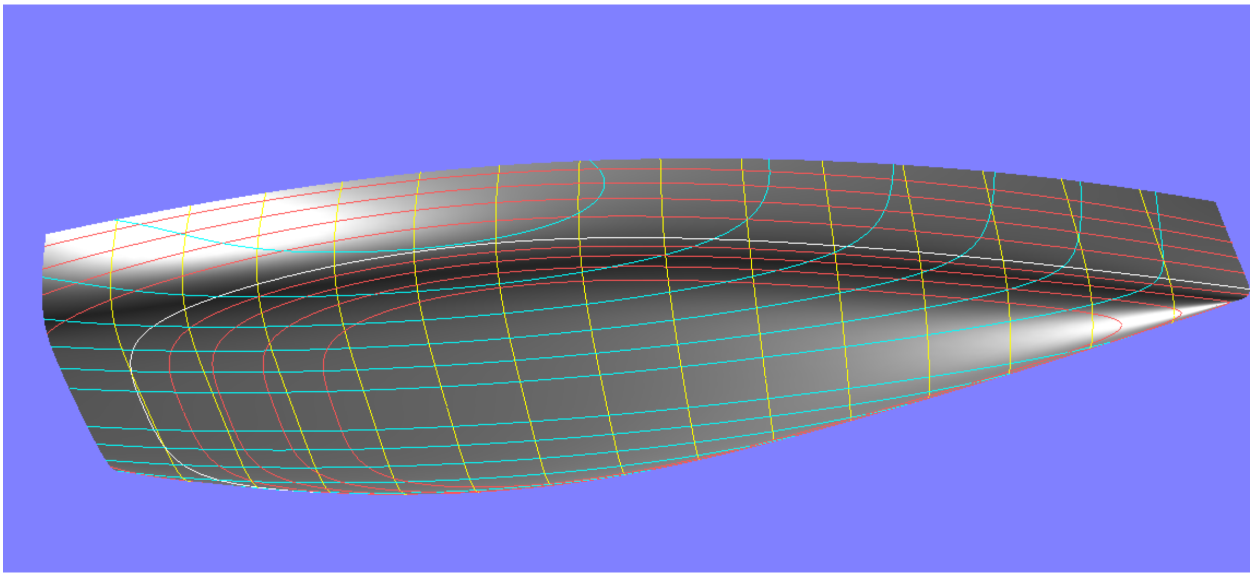
n3 = ZPOS (bead8) – ZPOS (bead7)

Here, bead8 is the intersection on the displacement curve, and bead7 is the intersection on the reference line line1.



Model-3.ms2 – variation of deviation of the derived mcs along the hull

As the displacement curve curve_x is fair, the deviations of the derived mcs from their parent mcs also vary in a harmonic relationship. Consequently, the derived hull is also fair.



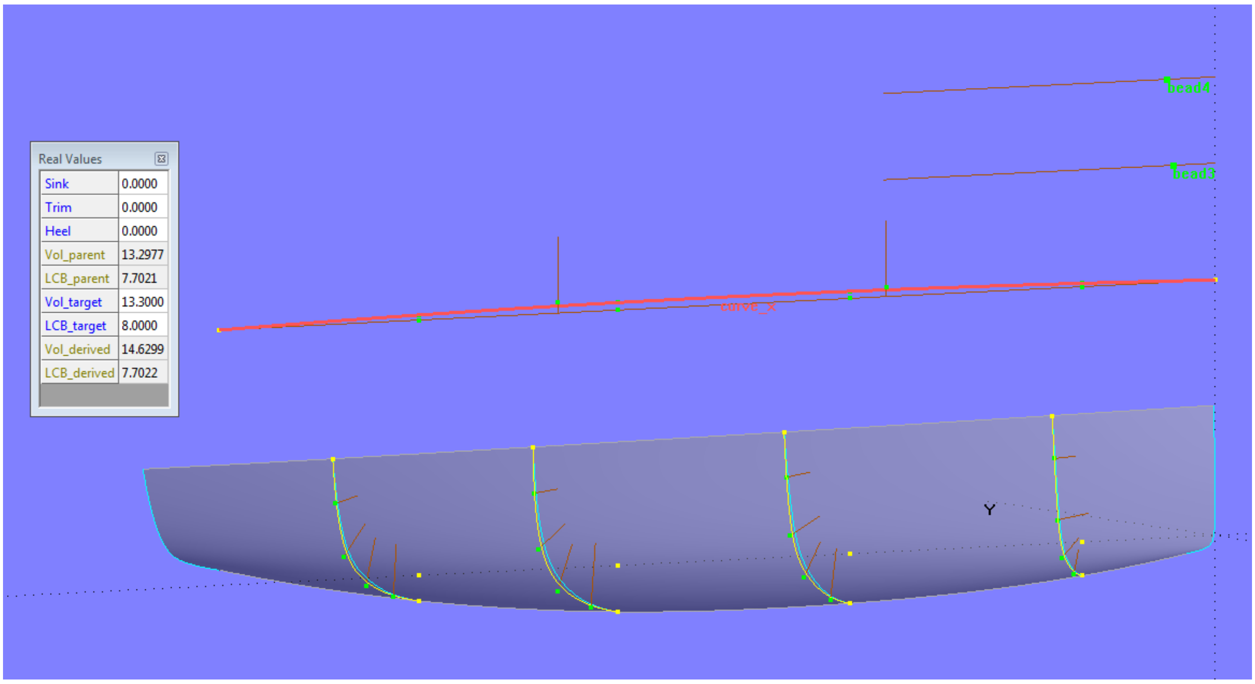
Model model-3.ms2 - lines of the derived hull

In model *model-3.ms2* displaced volume and its longitudinal center of buoyancy LCB are determined for the parent hull and the derived hull. Hydrostatic Reals are used to calculate these hydrostatic data. More details about the Hydrostatic Reals functions are given in Tutorial 14.

For the derived hull the Entity List hydro_contours_derived is used for *Contour List*. It contains the Contours hydro_stations1_derived, hydro_stations2_derived and hydro_stations3_derived which intersect the derived hull (hull_derived) and the deck (Ruled Surface deck). Similarly, as *Contour List* for the parent hull the Entity List hydro_contours_parent is used, holding the Contours hydro_stations1_parent, hydro_stations2_parent and hydro_stations3_parent.

The Entity List view_hydro_data is available to display the hydrostatic values. Select the list in the Entities Manager, then open the Real Values window with the "V" key or via main menu/ Tools/ Real Values.

Volume and LCB of the derived hull can be set manually to desired values through the "slider-beads" bead3 and bead4. At its DWL the parent hull displaces 13.30 m^3, LCB is at 7.70 m. For example, if the volume is to be 10% larger (14.64 m^3) for the same LCB, the t-parameter value of bead3 must be set to 0.1258 and that of bead4 to 0.1435. What can only be found out by gradual approaching.



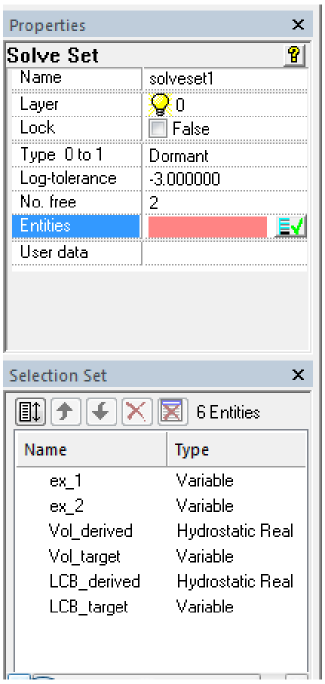
Real Values

Sink	0.0000
Trim	0.0000
Heel	0.0000
Vol_parent	13.2977
LCB_parent	7.7021
Vol_target	13.3000
LCB_target	8.0000
Vol_derived	14.6299
LCB_derived	7.7022

Model model-3.ms2 - by moving the "slider-beads" bead3 and bead4 one can manually change the derived hull shape until volume and LCB meet the desired values.

5 Automated adjustment of derived mcs to target values of volume and LCB

How to automatically find the derived mcs for desired values of volume and LCB is shown in *model-4.ms2*. In contrast to the previous model *model-3.ms2*, the t-parameter value of the Beads bead3 and bead4 is now defined by the Variables ex_1 and ex_2. This makes it possible to use the entity Solve Set to determine curve_x so that volume and LCB of the derived hull take given target values.



Properties

Solve Set

Name	solveset1
Layer	0
Lock	☐ False
Type 0 to 1	Dormant
Log-tolerance	-3.000000
No. free	2
Entities	🟥
User data	

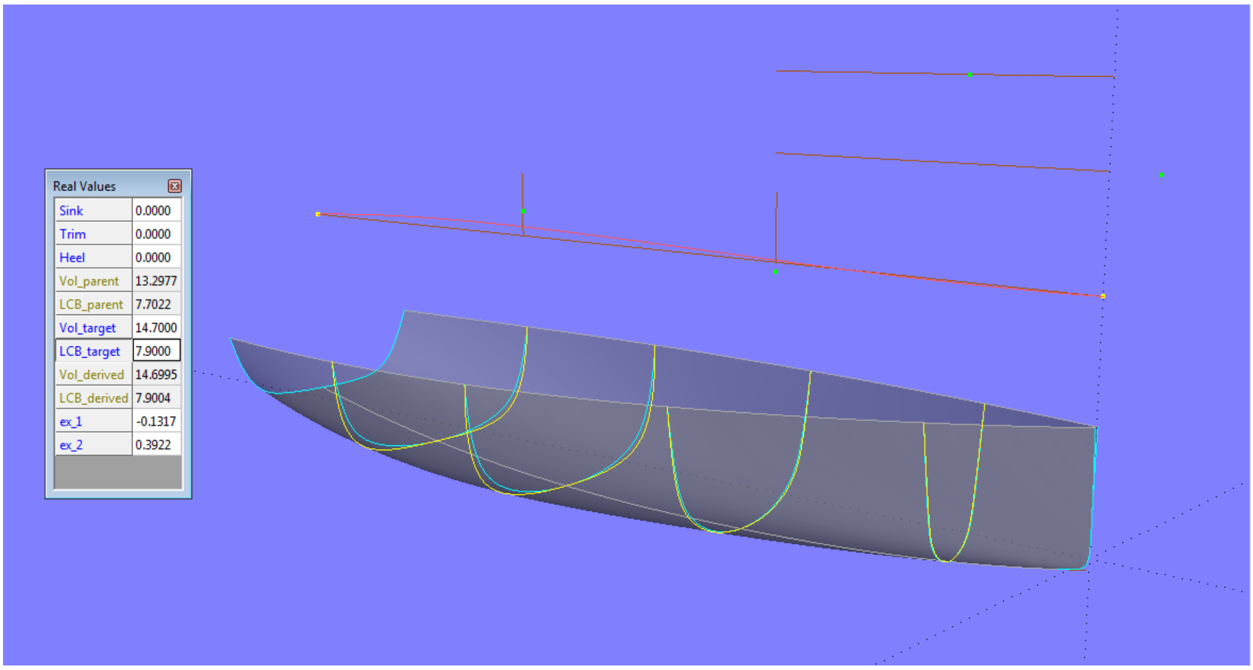
Selection Set

6 Entities

Name	Type
ex_1	Variable
ex_2	Variable
Vol_derived	Hydrostatic Real
Vol_target	Variable
LCB_derived	Hydrostatic Real
LCB_target	Variable

The two Variables Vol_target and LCB_target hold the desired values for displaced volume and LCB. When it is active, the Solve Set solveset1 automatically adjusts the Variables ex_1 and ex_2 until the Hydrostatic Reals Vol_derived and LCB_derived are the same size as Vol_target and LCB_target.

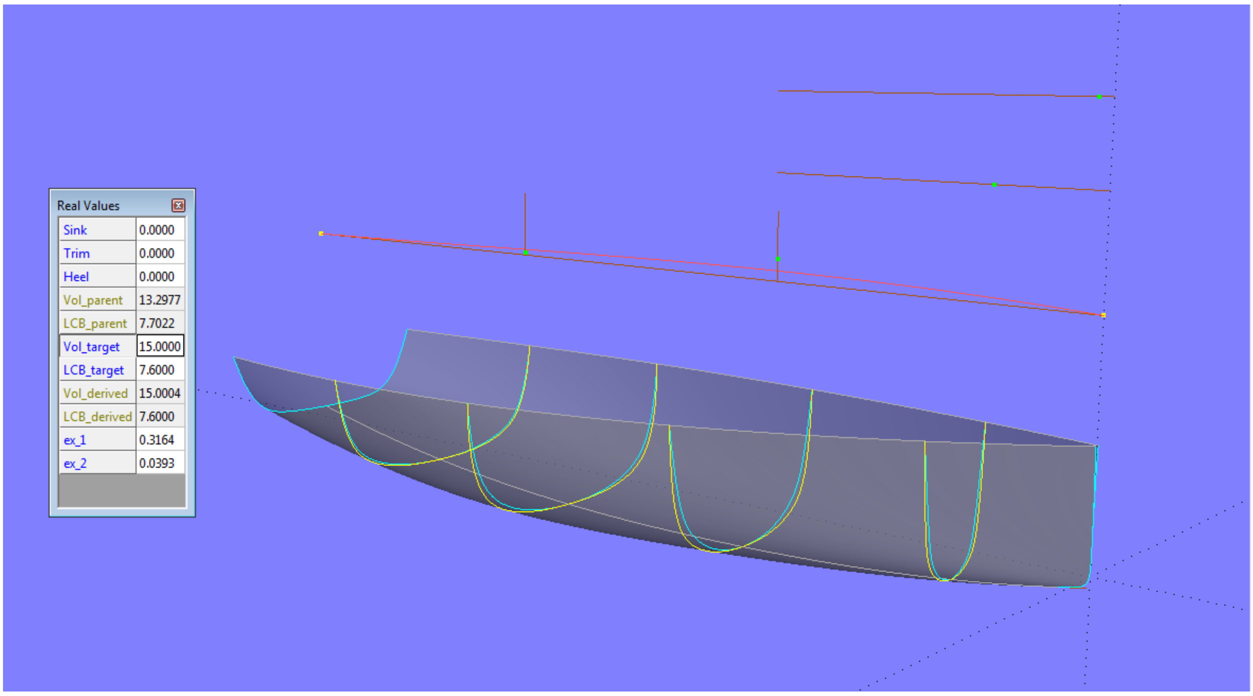
Model model-4.ms2 – the Solve Set solveset1 with 2 degrees of freedom (Variables ex_1 and ex_2) automatically adjusts the displacement curve curve_x for the derived mcs so that volume and LCB are equal to target values.



Real Values

Sink	0.0000
Trim	0.0000
Heel	0.0000
Vol_parent	13.2977
LCB_parent	7.7022
Vol_target	14.7000
LCB_target	7.9000
Vol_derived	14.6995
LCB_derived	7.9004
ex_1	-0.1317
ex_2	0.3922

Model model-4.ms2 - increased volume and LCB moved aft (Solve Set solveset1 active)



Real Values

Sink	0.0000
Trim	0.0000
Heel	0.0000
Vol_parent	13.2977
LCB_parent	7.7022
Vol_target	15.0000
LCB_target	7.6000
Vol_derived	15.0004
LCB_derived	7.6000
ex_1	0.3164
ex_2	0.0393

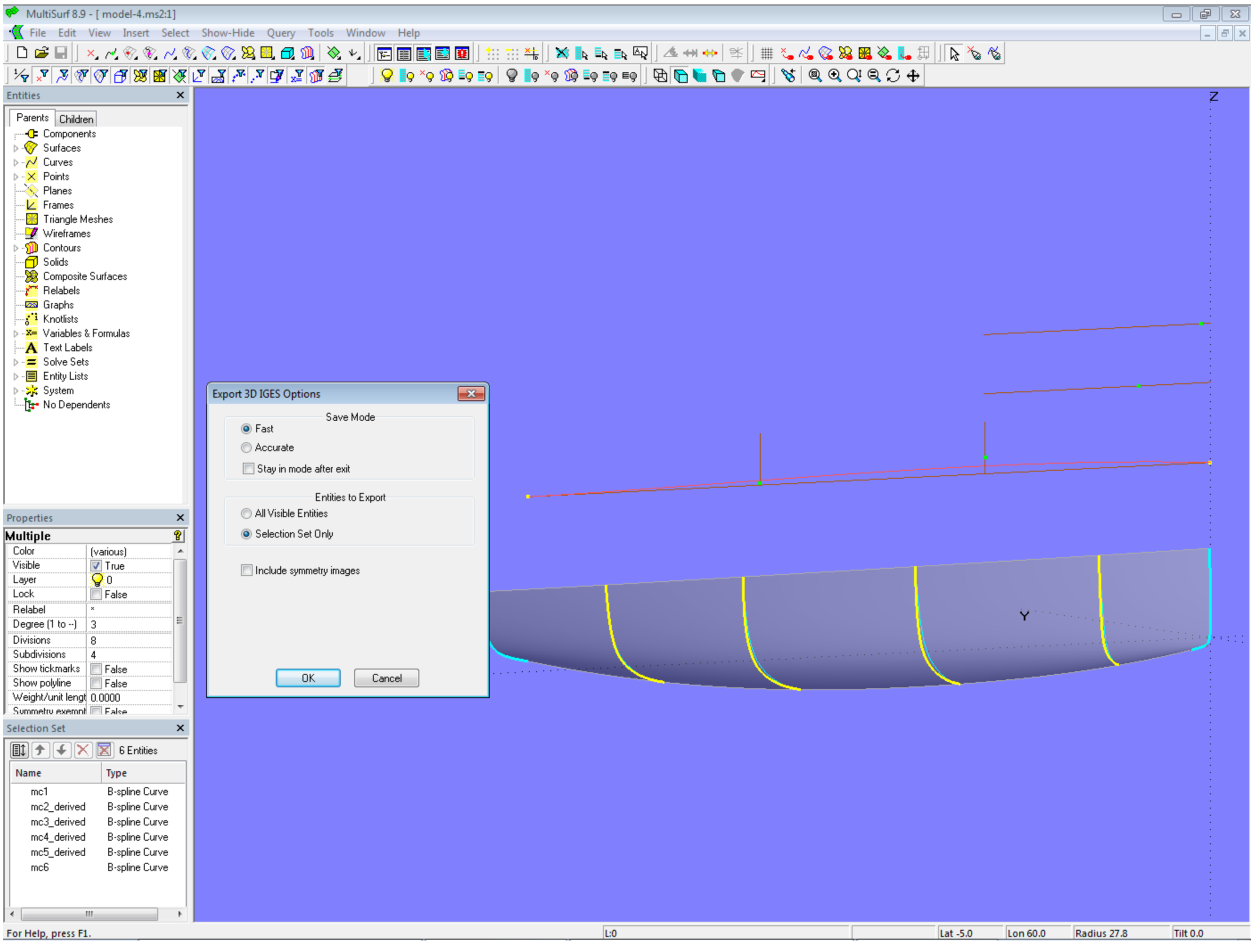
Model model-4.ms2 - increased volume and LCB moved forward (Solve Set solveset1 active)

6 Export of derived hull

When the desired values for volume and LCB are obtained, it is convenient to make the derived hull a stand-alone surface. Since the derived cps (D22, D23, D24, etc.) are Arc-length Beads (no free 3-D points), they depend on a number of formulas and variables. Therefore, it is not possible to save the derived hull as a component, because all parent entities of the derived cps would be included in the component.

With the help of **Iges Export** it takes just a few steps to attain an independent surface. The Iges export of the derived mcs will result in independent control points in the Iges file.

- Select all mcs of the C-spline Lofted Surface hull_derived.
- Main menu/ File/ Export 3D/ IGES. In the “Iges Options” dialog box, select "Selection Set Only". Then click the "OK" button and save the Iges file under the desired name.
- **Import the Iges file** via File/ Import/ IGES into the target model (for example a new, empty model). Finally, with the imported mcs, create a C-spline Lofted Surface in the target model.



Export 3D IGES Options

Save Mode
● Fast
○ Accurate
□ Stay in mode after exit

Entities to Export
○ All Visible Entities
● Selection Set Only

□ Include symmetry images

OK Cancel

Properties

Multiple

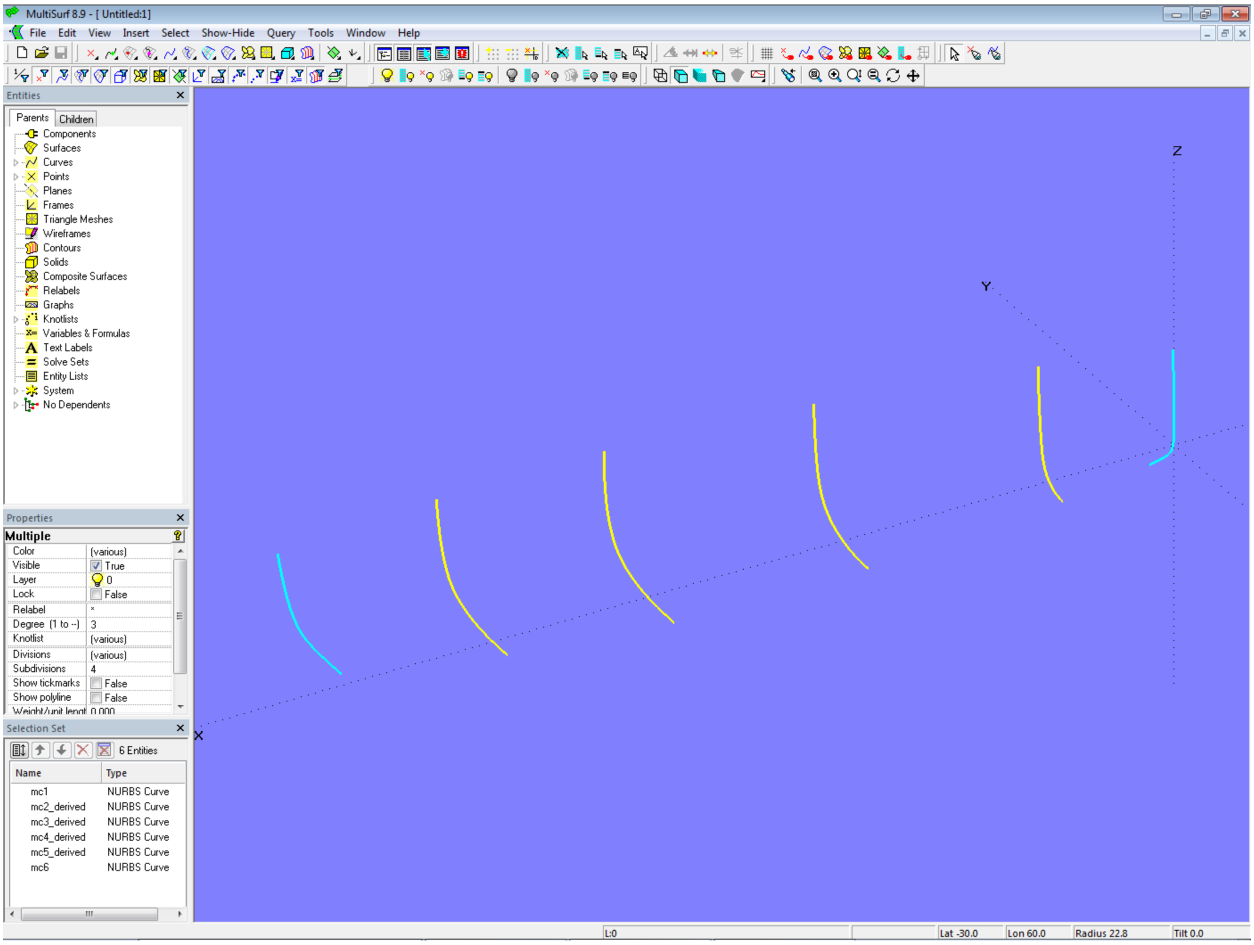
Color	(various)
Visible	☑ True
Layer	0
Lock	□ False
Relabel	*
Degree (1 to --)	3
Divisions	8
Subdivisions	4
Show tickmarks	□ False
Show polyline	□ False
Weight/unit lengt	0.0000
Symmetry exempt	□ False

Selection Set

6 Entities

Name	Type
mc1	B-spline Curve
mc2_derived	B-spline Curve
mc3_derived	B-spline Curve
mc4_derived	B-spline Curve
mc5_derived	B-spline Curve
mc6	B-spline Curve

Model model-4.ms2 – Iges export creates independent cps of the derived mcs.



Multiple	
Color	(various)
Visible	☑ True
Layer	0
Lock	□ False
Relabel	*
Degree (1 to --)	3
Knotlist	(various)
Divisions	(various)
Subdivisions	4
Show tickmarks	□ False
Show polyline	□ False
Weight/unit lengt	0.000

6 Entities

Name	Type
mc1	NURBS Curve
mc2_derived	NURBS Curve
mc3_derived	NURBS Curve
mc4_derived	NURBS Curve
mc5_derived	NURBS Curve
mc6	NURBS Curve

Import the Iges file with independent mcs into the target model for further processing.

7 Import of parent hull

The method explained above by example of *model-4.ms2*, how to derive a new hull surface for target values of volume and LCB from a parent hull, can be implemented into other hull models, which are created with different numbers of mcs and cps. At present these models are available:

Model *model-4.ms2* – C-spline Lofted Surface, 6 mcs x 6 cps
Model *model-5.ms2* – C-spline Lofted Surface, 5 mcs x 6 cps
Model *model-6.ms2* – C-spline Lofted Surface, 5 mcs x 4 cps

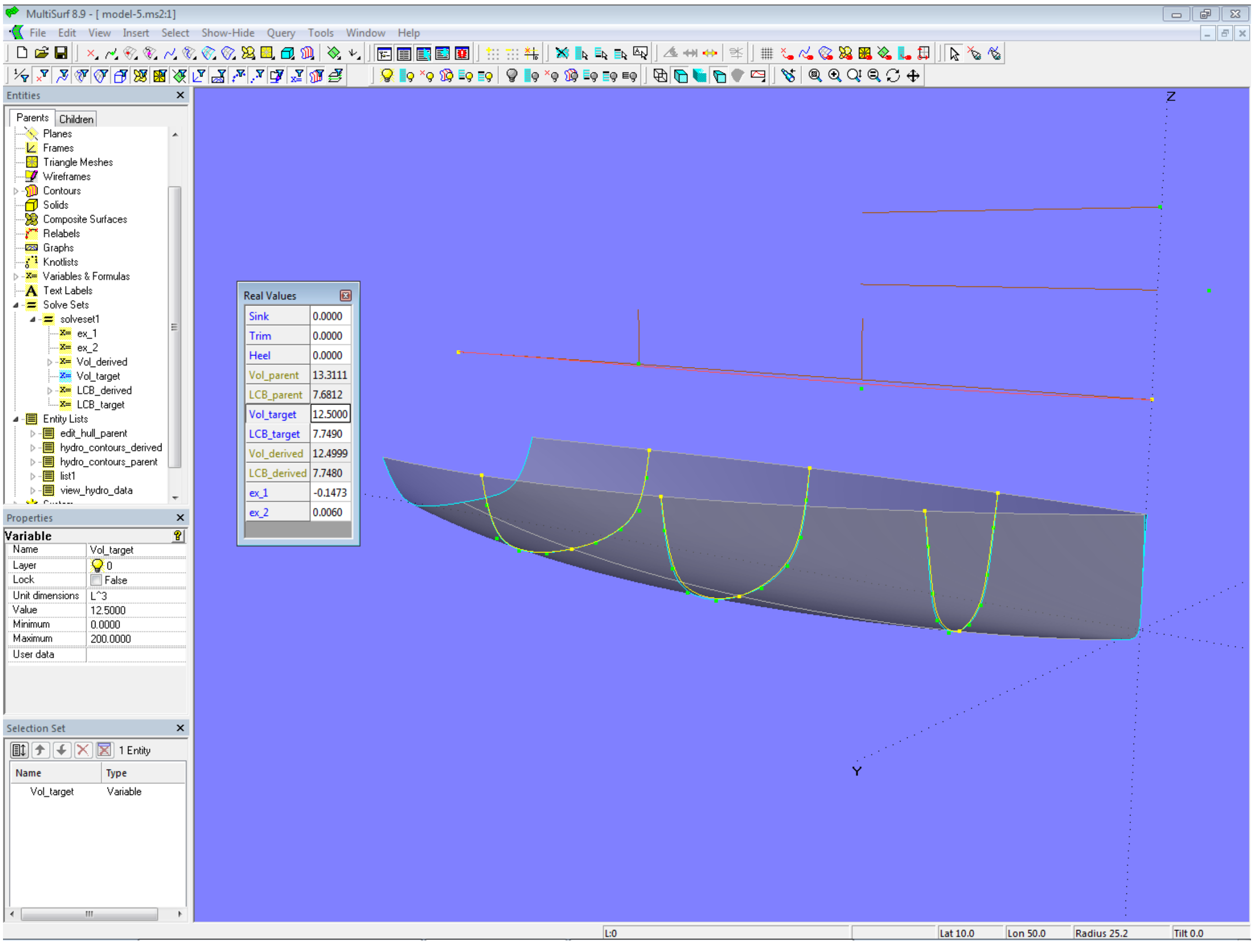
In order to import a different parent hull into one of these models (receiving model, e.g. example *model-4.ms2*) just the cps of the other parent hull must be transfered. There are two options for this:

- **Save Component / Load Component**: open the model which holds the hull surface to become the new parent hull. Select all its cps and save them as a component. Then load this component into the receiving model, e.g. *model-4.ms2*.

- **Iges Export / Iges Import**: open the model which holds the hull surface to become the new parent hull. Select all its cps and save them as an Iges file. Then import this Iges file into the receiving model, e.g. *model-4.ms2*.

For each parent mc in the receiving model (*model-4.ms2*, *model-5.ms2*, *model-6.ms2*) the old cps must be replaced by the corresponding new cps. Main menu/ Tools/ Adopt Children is used for this task. In the

"Adopt Children" dialog window, both check boxes "Delete old parent" and "Transfer name to new parent" are to be ticked (activated). By this procedure the cps of the mcs of the C-spline Lofted Surface hull_parent in the model are updated by the cps of the new hull surface.



Real Values

Sink	0.0000
Trim	0.0000
Heel	0.0000
Vol_parent	13.3111
LCB_parent	7.6812
Vol_target	12.5000
LCB_target	7.7490
Vol_derived	12.4999
LCB_derived	7.7480
ex_1	-0.1473
ex_2	0.0060

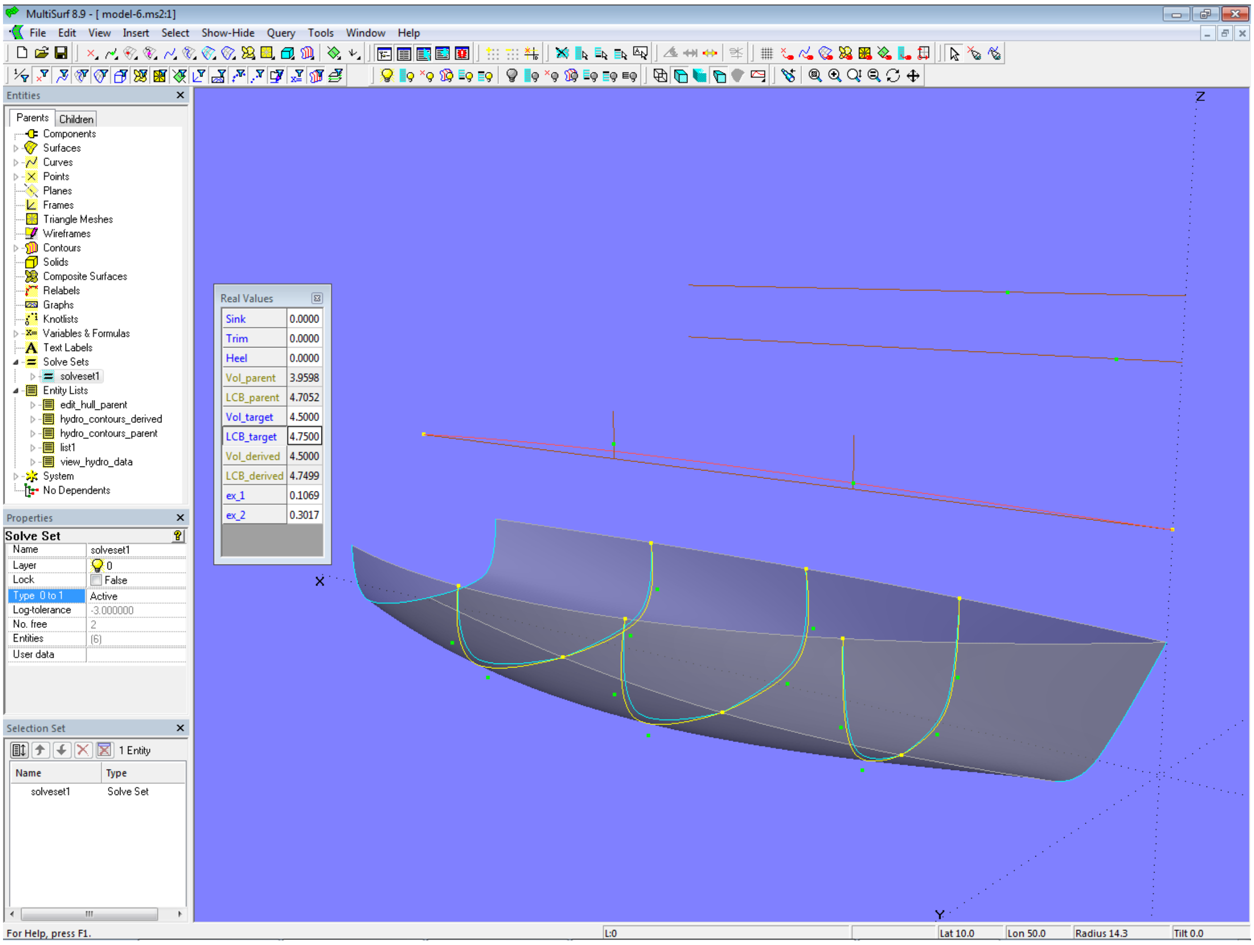
Variable

Name	Vol_target
Layer	0
Lock	□ False
Unit dimensions	L^3
Value	12.5000
Minimum	0.0000
Maximum	200.0000
User data	

1 Entity

Name	Type
Vol_target	Variable

Model model-5.ms2 – example of parent hull and derived hull (C-spline Lofted Surface, 5 mcs x 6 cps)



Real Values

Sink	0.0000
Trim	0.0000
Heel	0.0000
Vol_parent	3.9598
LCB_parent	4.7052
Vol_target	4.5000
LCB_target	4.7500
Vol_derived	4.5000
LCB_derived	4.7499
ex_1	0.1069
ex_2	0.3017

Solve Set

Name	solveset1
Layer	0
Lock	□ False
Type 0 to 1	Active
Log-tolerance	-3.000000
No. free	2
Entities	(6)
User data	

Name	Type
solveset1	Solve Set

Model model-6.ms2 – example of parent hull and derived hull (C-spline Lofted Surface, 5 mcs x 4 cps)

Appendix 1

On Systematic Variation of Hull Forms - the Lackenby Method

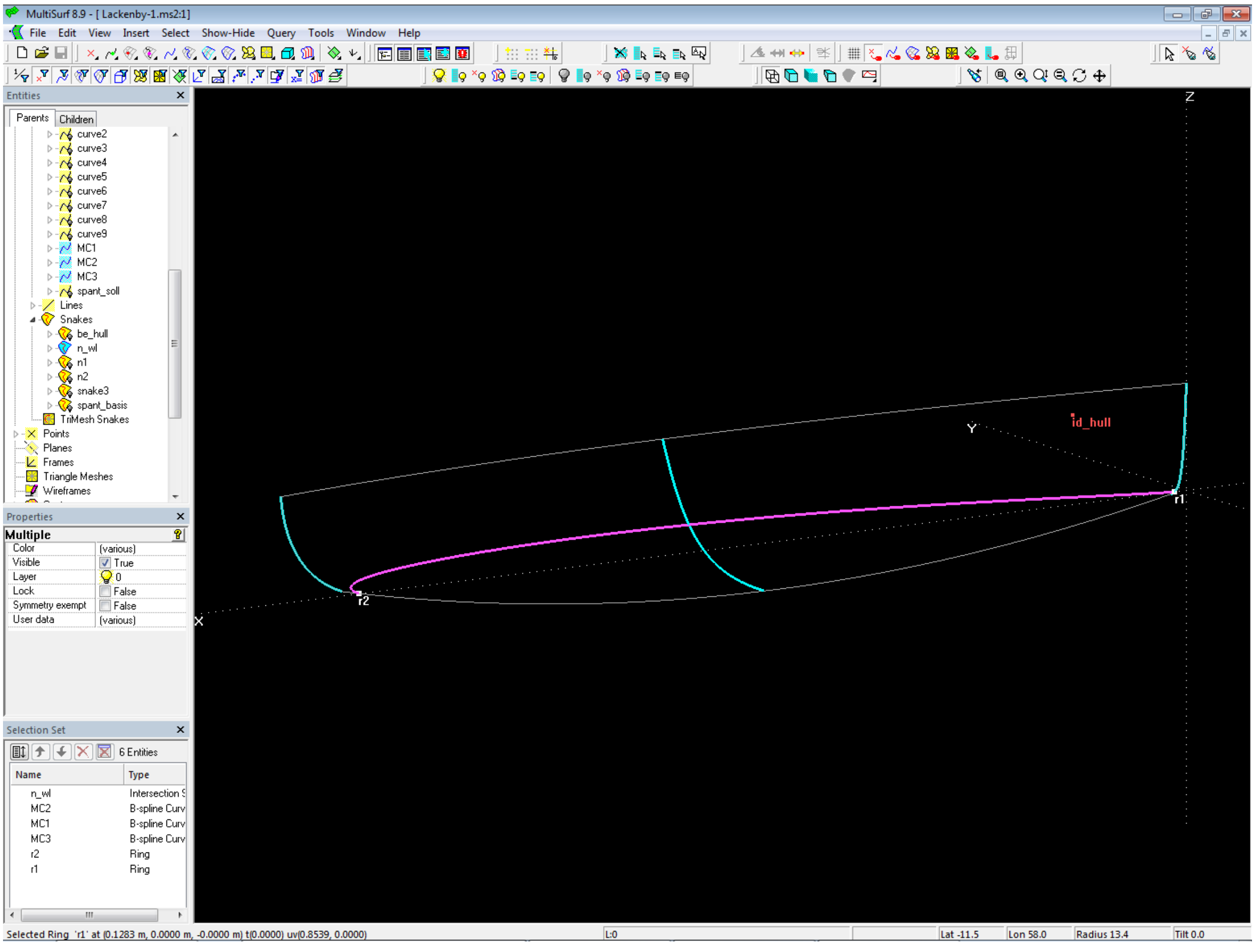
H. Lackenby did research work on the variation of ship forms (On the systematic geometrical variation of ship forms; TINA, 1950). In the following the attempt is explained to implement his method in a MultiSurf model.

Start point is a basis hull form and its section area curve based on a set of stations. To change prismatic coefficient and/or LCB, the stations are moved towards bow or stern. That is, the shape of each station is unchanged, just its position is adjusted a bit.

Lackenby devides the hull in forebody and afterbody. Both have a certain length. The necessary shift dx of each station follows the formula: dx = c * x * (1 –x), where x is the location of the station. c is a constant, its value depends on the required change in prismatic coefficient and LCB.

For x = 0 (bow or stern) or x = 1 (midship), dx = 0. That means, the midship station and the end stations of bow and stern remain fixed (dx = 0), only the stations in between are shifted in a specific fashion.

In the MultiSurf model *Lackenby-1.ms2* the author tried to implement the method of Lackenby.



Multiple	
Color	(various)
Visible	☑ True
Layer	0
Lock	□ False
Symmetry exempt	□ False
User data	(various)

6 Entities

Name	Type
n_wl	Intersection S
MC2	B-spline Curv
MC1	B-spline Curv
MC3	B-spline Curv
r2	Ring
r1	Ring

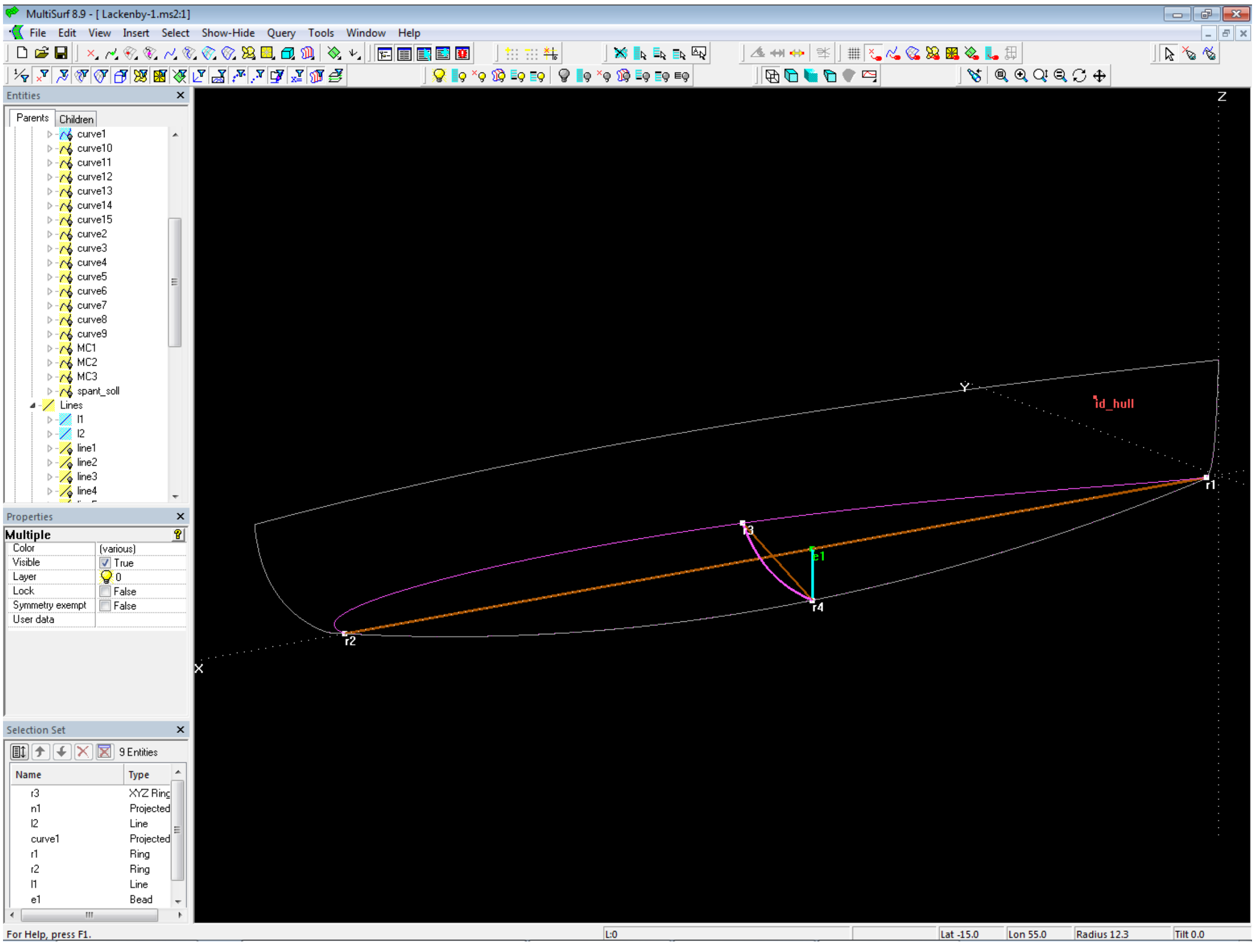
Selected Ring 'r1' at (0.1283 m, 0.0000 m, -0.0000 m) t(0.0000) uv(0.8539, 0.0000) L:0 Lat -11.5 Lon 58.0 Radius 13.4 Tilt 0.0

Model Lackenby-1.ms2

The hull is a C-spline Lofted Surface supported by 3 mcs. The forward end of the waterline is the Ring r1, the aft end is Ring r2.

For this hull the section area curve is constructed following these steps:

- Line l1 runs from r1 to r2.
- Bead e1 is on Line l1.
- At the location of e1 the waterline and the fairbody curve are intersected in the XYZRings r3 and r4.
- Line l2 is spanned between r3 and r4 and projected onto the hull surface as Projected Snake n1. Note, that the parent for Mirror/surface is the Bead e1 – so the projection rays originate from this point, thus there will not be a problem with projection failure due to a parallel projection ray and surface tangent.
- n1 is projected onto the centerplane (*Y = 0) as projected Curve curve1.

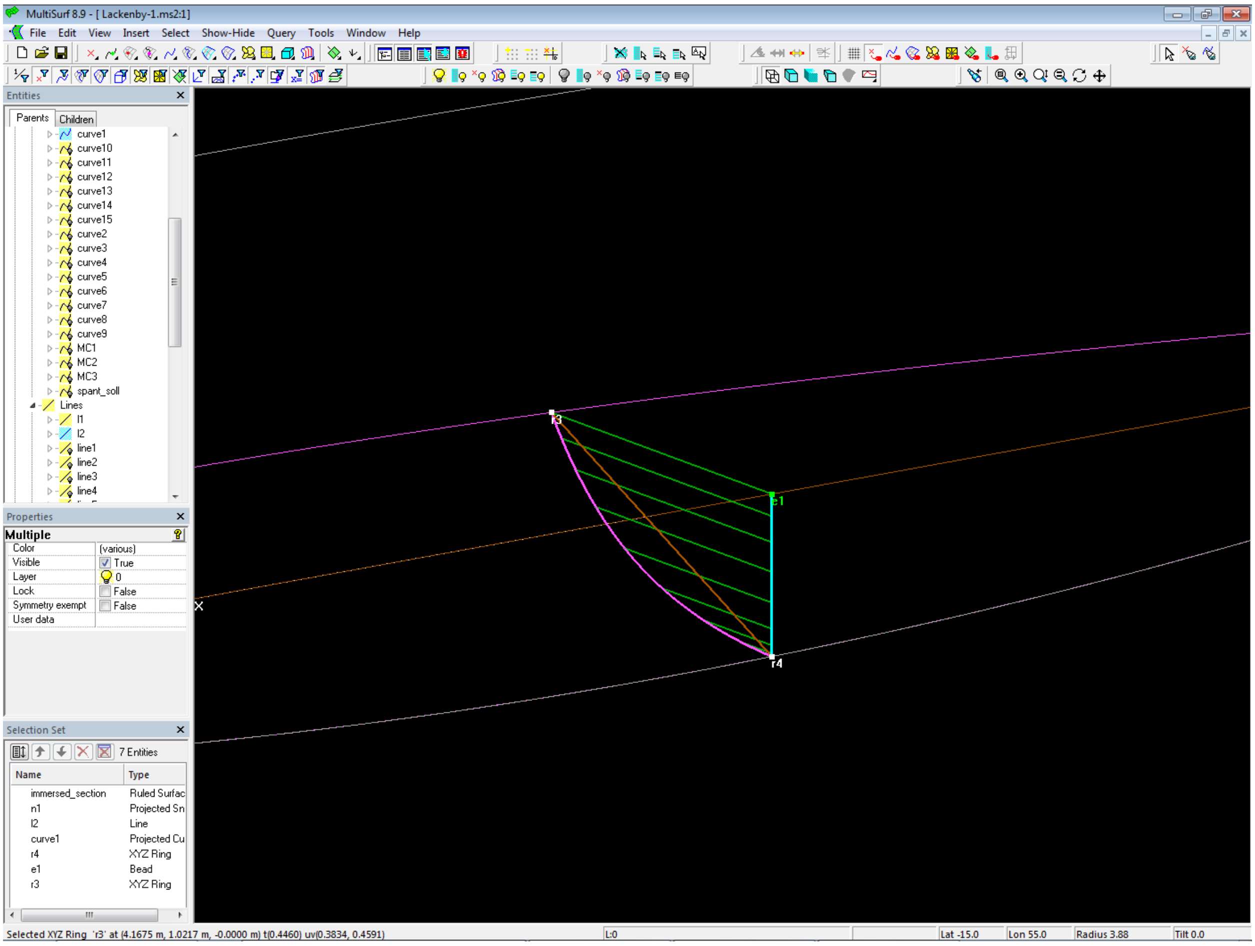


Multiple	
Color	(various)
Visible	☑ True
Layer	0
Lock	□ False
Symmetry exempt	□ False
User data	

Name	Type
r3	XYZ Ring
n1	Projected
l2	Line
curve1	Projected
r1	Ring
r2	Ring
l1	Line
e1	Bead

Model Lackenby-1.ms2 – construction of an immersed half-section

Finally the Ruled Surface immersed_half_section is created between n1 and curve1.



Properties

Multiple

Color	(various)
Visible	☑ True
Layer	0
Lock	□ False
Symmetry exempt	□ False
User data	

Selection Set

7 Entities

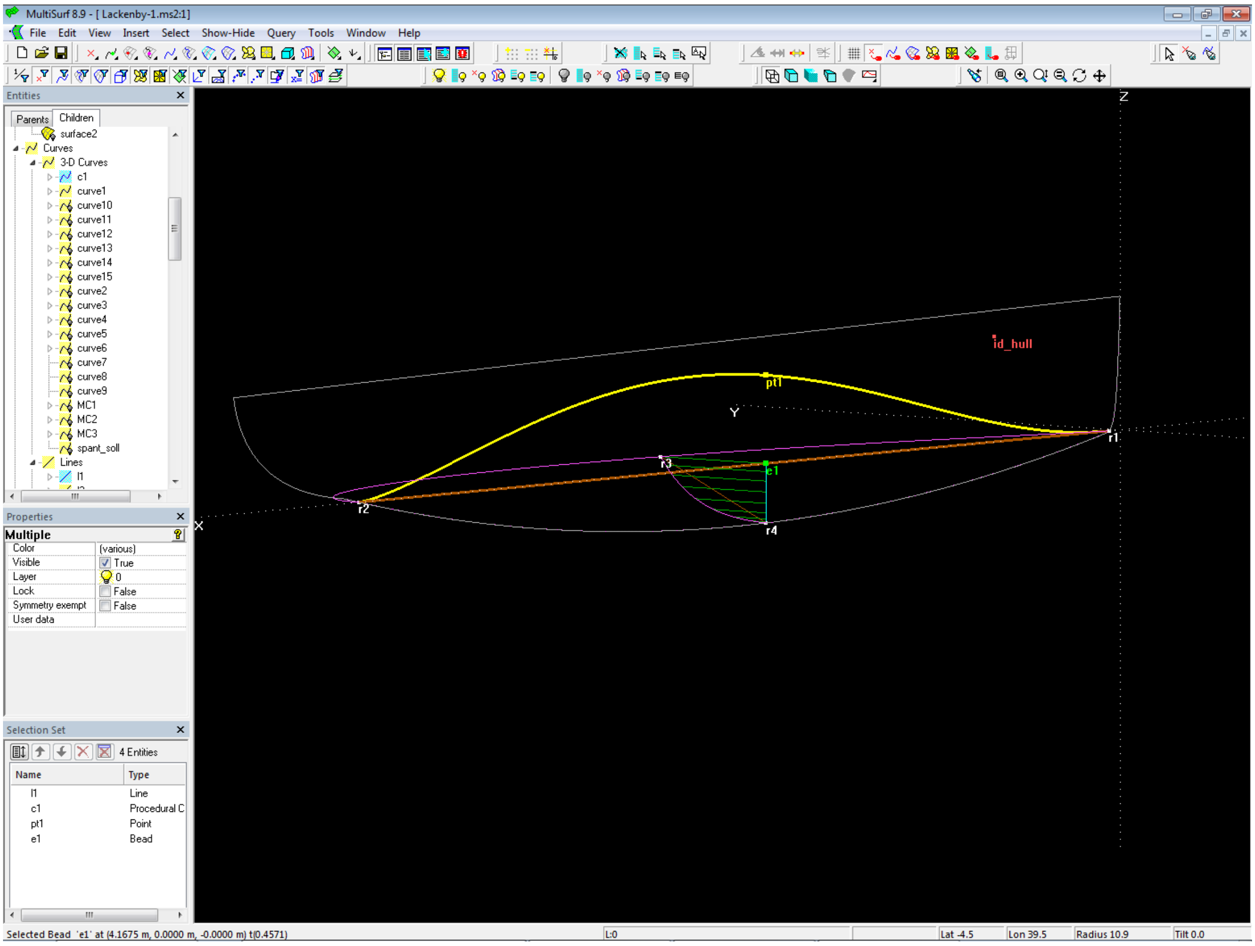
Name	Type
immersed_section	Ruled Surfac
n1	Projected Sn
l2	Line
curve1	Projected Cu
r4	XYZ Ring
e1	Bead
r3	XYZ Ring

Selected XYZ Ring 'r3' at (4.1675 m, 1.0217 m, -0.0000 m) t(0.4460) uv(0.3834, 0.4591) L:0 Lat -15.0 Lon 55.0 Radius 3.88 Tilt 0.0

Model Lackenby-1.ms2 – immersed half-section

So now we have the immersed half-section at the position of e1.

By the Formula area_immersed the size of the complete immersed section is calculated. This value is assigned to the Z-coordinate of Point pt1, which represents the abscissa of the section area curve at position of e1.



Multiple

Color	(various)
Visible	☑ True
Layer	0
Lock	□ False
Symmetry exempt	□ False
User data	

4 Entities

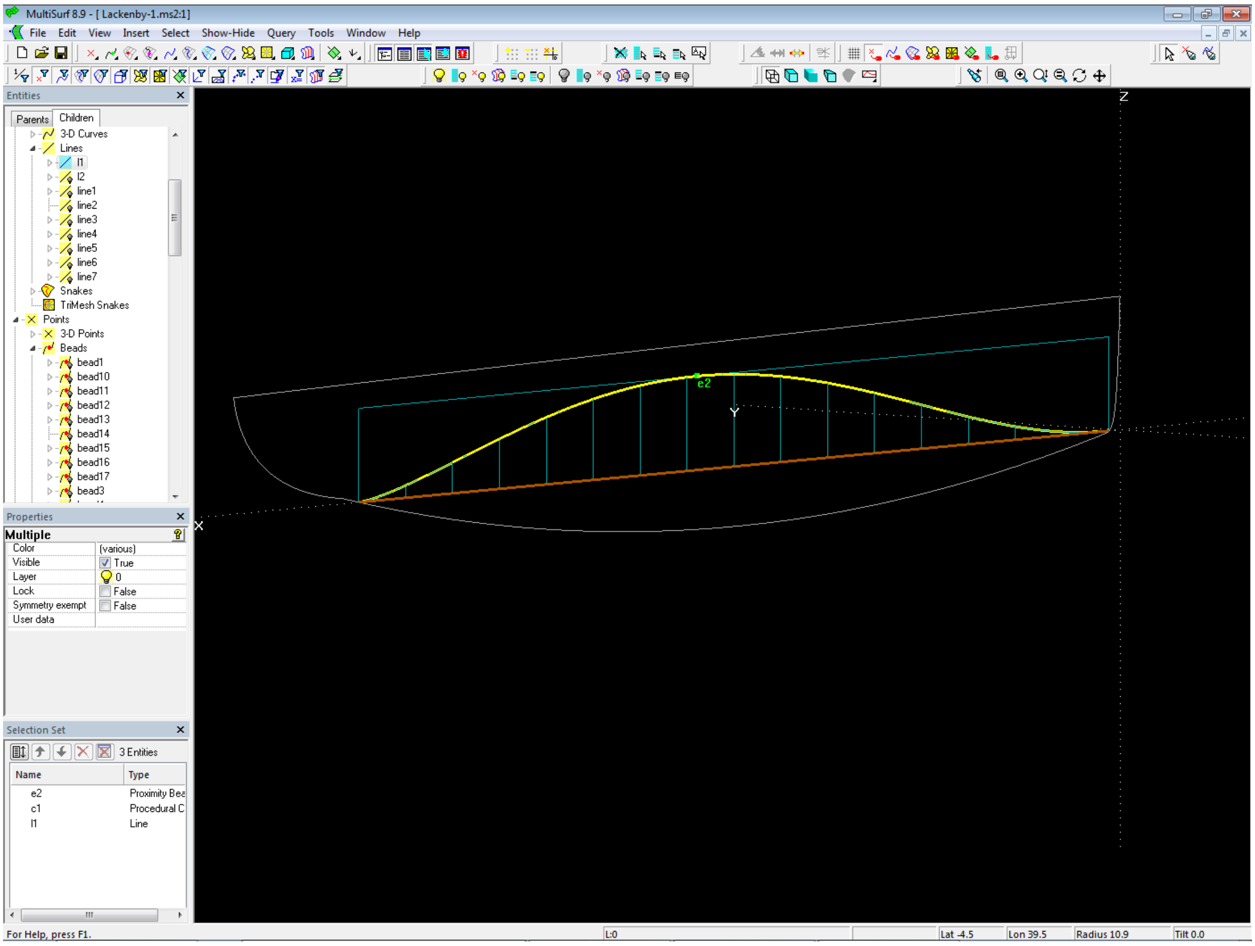
Name	Type
l1	Line
c1	Procedural C
pt1	Point
e1	Bead

Selected Bead 'e1' at (4.1675 m, 0.0000 m, -0.0000 m) t(0.4571) L:0 Lat -4.5 Lon 39.5 Radius 10.9 Tilt 0.0

Model Lackenby-1.ms2 – section area curve by repeating the half-section construction by a Procedural Curve entity

When the t-position of e1 is changed from t = 0 to t = 1 we get the complete section area curve. The Procedural Curve c1 does this construction, creating the path of pt1.

Finally the Ruled Surface section_area_total is created between c1 and l1.



Multiple

Color	(various)
Visible	☑ True
Layer	0
Lock	☐ False
Symmetry exempt	☐ False
User data	

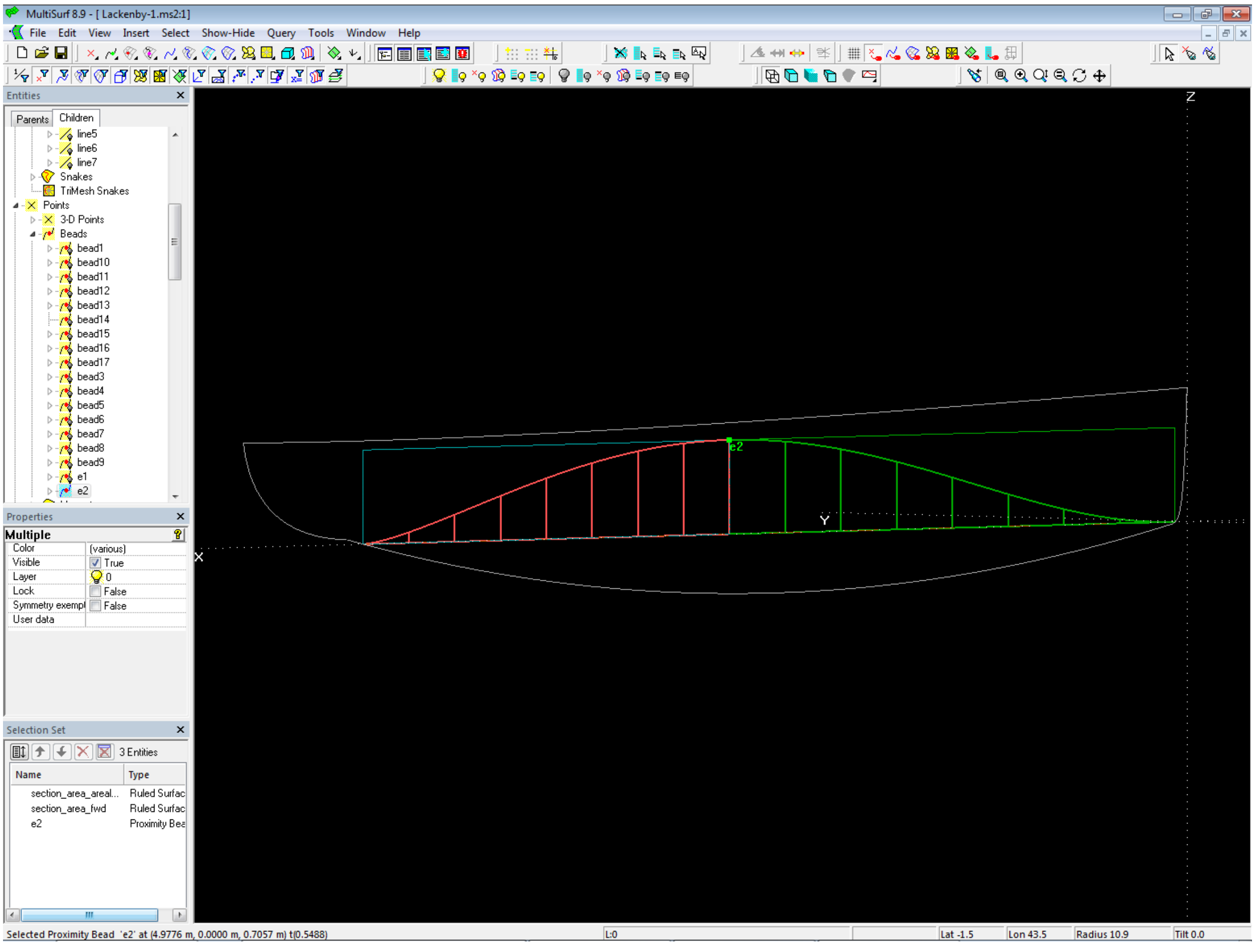
3 Entities

Name	Type
e2	Proximity Bea
c1	Procedural C
l1	Line

Model Lackenby-1.ms2 – section area curve

The Proximity Bead e2 on the section area curve determines the location of the maximum section area, the midship section. Note, that in general, this is not at Lwl/2.

The total prismatic coefficient is the ratio of the section area and the area of the surrounding rectangle. The Formula cp_total calculates this value.



Multiple

Color	(various)
Visible	☑ True
Layer	0
Lock	□ False
Symmetry exempt	□ False
User data	

3 Entities

Name	Type
section_area_areal...	Ruled Surfac
section_area_fwd	Ruled Surfac
e2	Proximity Bea

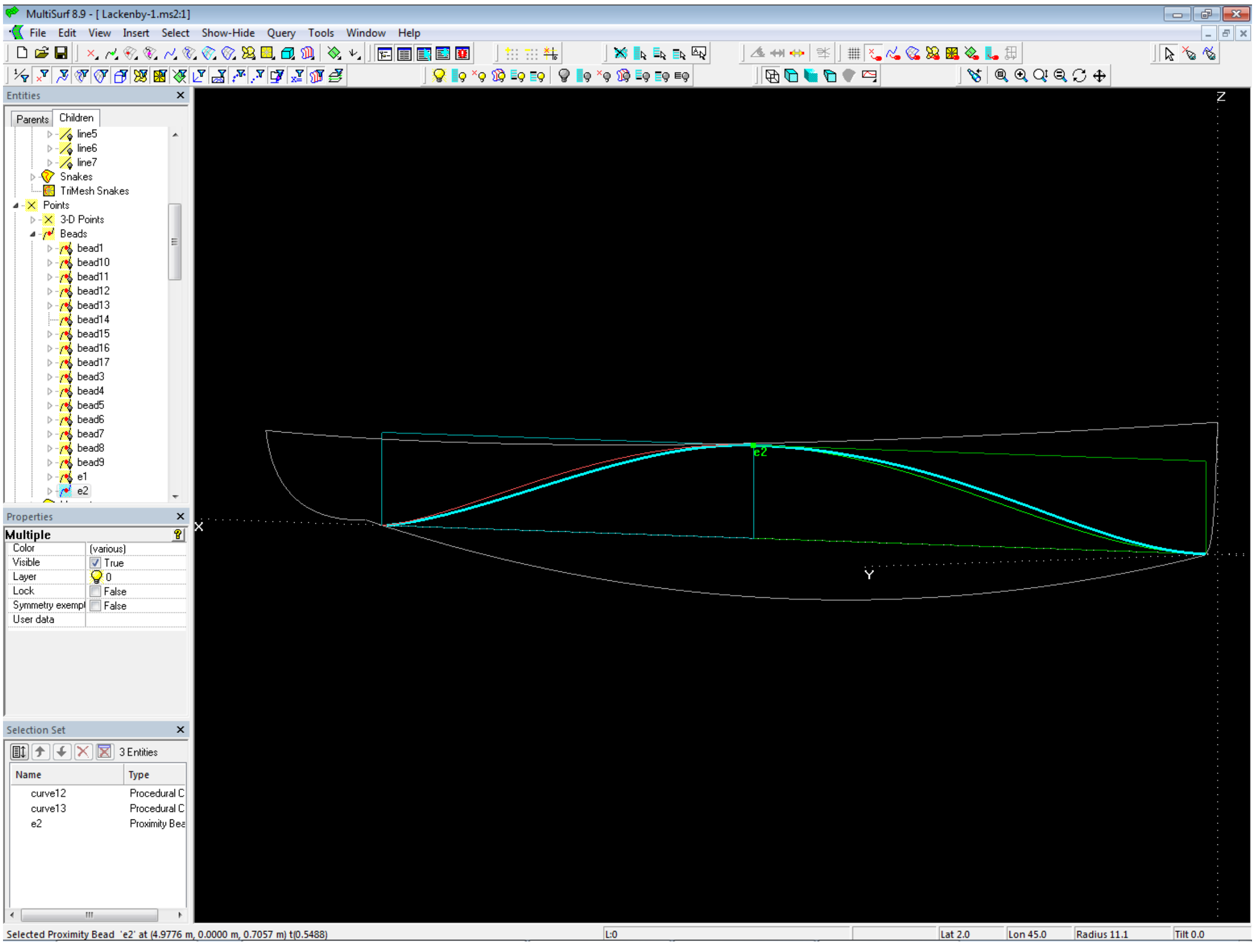
Selected Proximity Bead 'e2' at (4.9776 m, 0.0000 m, 0.7057 m) t(0.5488) L:0 Lat -1.5 Lon 43.5 Radius 10.9 Tilt 0.0

Model Lackenby-1.ms2 – section area curves of forebody and afterbody

Dividing the section area curve at e2 results in the section area curve of the forebody and the section area curve of the afterbody. Again, the prismatic coefficient of the forebody and the prismatic coefficient of the afterbody is calculated as the ratio between the respectives area and the area of the encompassing rectangle.

Change the position of the LCB keeping the total prismatic coefficient constant

Now let us assume, we want to move the LCB towards the bow by, say 0.2 m. The total prismatic coefficient should be kept constant, i.e. the displaced volume is unchanged. According to the Lackenby method we must move the stations towards the bow. How much a station must be moved can be calculated with the following formulas given by Lackenby.



Multiple

Color	(various)
Visible	☑ True
Layer	0
Lock	□ False
Symmetry exempl	□ False
User data	

3 Entities

Name	Type
curve12	Procedural C
curve13	Procedural C
e2	Proximity Bea

Selected Proximity Bead 'e2' at (4.9776 m, 0.0000 m, 0.7057 m) t(0.5488)

Model Lackenby-1.ms2 – modified section area curve to change the LCB

The necessary longitudinal shift d_xaft of the section at x to produce the required change in prismatic coefficient of the section area curve of the afterbody:

d_xaft = caft * x * (1-x) equal to: d_xaft = caft * (x – x^2)

with

$$\mathrm{caft} = \frac{\mathrm{d\_cpaft}}{\mathrm{cpaft} - 2 * 2\mathrm{cpaft} * \mathrm{xaft}}$$

The necessary longitudinal shift d_xfwd of the section at x:to produce the required change in prismatic coefficient of the section area curve of the forebody:

d_xfwd = cfwd * x * (1-x) equal to: d_xaft = caft * (x – x^2)

with

$$\mathrm{d\_xfwd} = \frac{\mathrm{d\_cpfwd}}{\mathrm{cpfwdt} - 2 * 2\mathrm{cpfwd} * \mathrm{xfwd}}$$

where:

d_cpaft: the required change in prismatic coefficient of the section area curve of the afterbody

d_cpfwd: the required change in prismatic coefficient of the section area curve of the forebody

cpaft: the prismatic coefficient of the original section area curve of the afterbody

cpfwd: the prismatic coefficient of the original section area curve of the forebody

xaft: the fractional distance from midships of the centroid of the original section area curve of the afterbody

xfwd: the fractional distance from midships of the centroid of the original section area curve of the forebody

$$d\_cpaft = \frac{2 * dz * cptotal}{haft + hfwd}$$

$$d\_cpfwd = - \frac{2 * dz * cptotal}{haft + hfwd}$$

where:

dz: the required fractional shift of the LCB in the modified form

cptotal: the total prismatic coefficient of the basis hull

haft: the fractional distance from midships of the centroid of the added sliver of area to the original section area curve of the afterbody

hfwd: the fractional distance from midships of the centroid of the added sliver of area to the original section area curve of the forebody

The first approximation of haft and hfwd is given by:

$$haft = \frac{2 * xaft -3 * kaft^2}{1 – 2 * xaft}$$

$$hfwd = \frac{2 * xfwd -3 * kfwd^2}{1 – 2 * xfwd}$$

where:

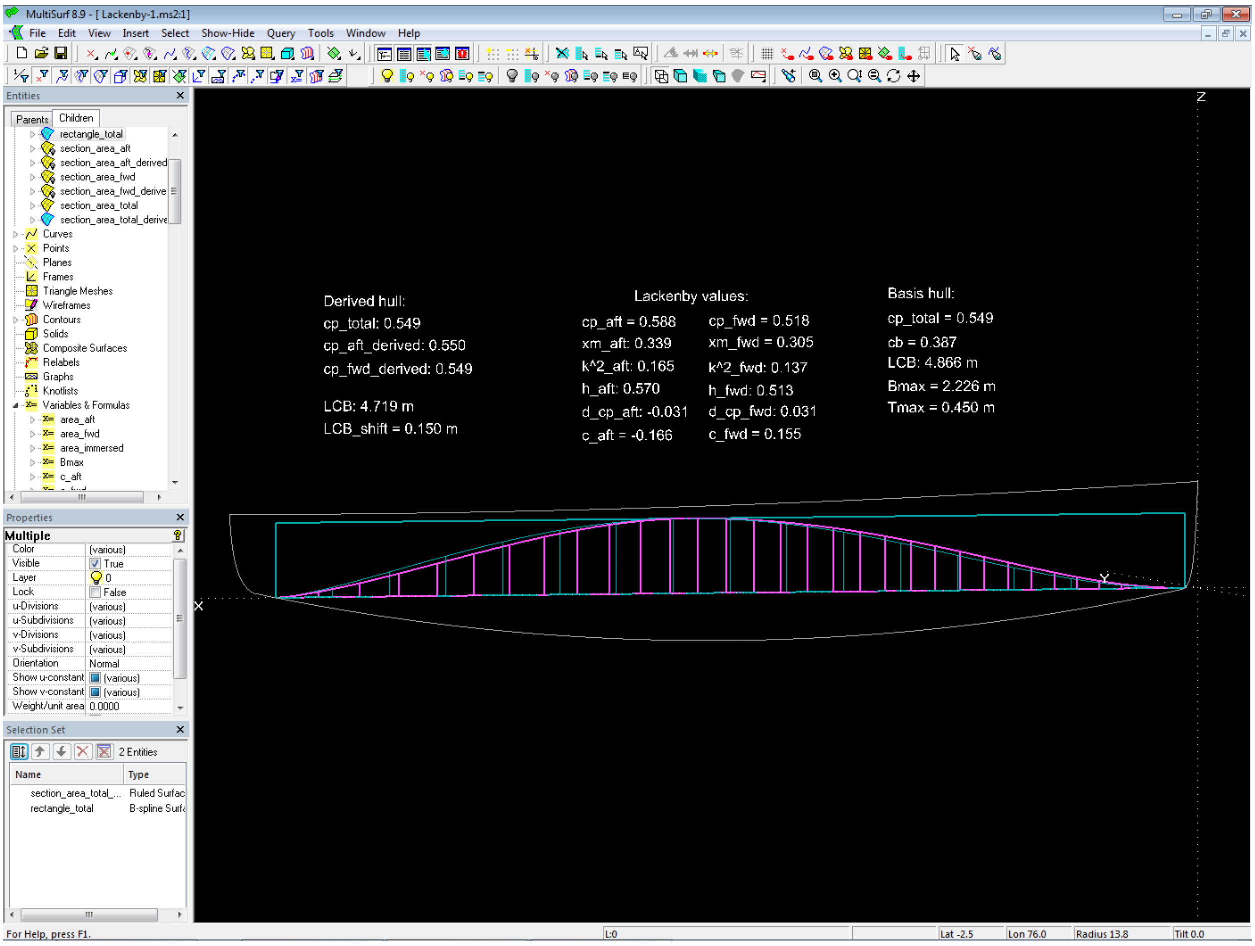
xaft: the fractional distance from midships of the centroid of the original section area curve of the afterbody xfwd: the fractional distance from midships of the centroid of the original section area curve of the fore-body

kaft: radius of gyration about midships of the original section area curve of the afterbody

kfwd: radius of gyration about midships of the original section area curve of the forebody

All those quantities of the original section area curves of forebody and afterbody are calculated in the model by formulas and variables. Except the moments of inertia of the section areas of forebody and afterbody. The values Izz_aft and Izz_fwd are variables, the values are taken from the Mass Properties. So, if the original section area curve is changed because the basis hull is changed, those values must be looked up in main menu/ Tools/ Mass Properties and edited in the variables Izz_aft and Izz_fwd.

The results are displayed by Text Label entities.



Multiple	
Color	(various)
Visible	☑ True
Layer	0
Lock	□ False
u-Divisions	(various)
u-Subdivisions	(various)
v-Divisions	(various)
v-Subdivisions	(various)
Orientation	Normal
Show u-constant	(various)
Show v-constant	(various)
Weight/unit area	0.0000

Name	Type
section_area_total_...	Ruled Surfac
rectangle_total	B-spline Surf

Derived hull:
cp_total: 0.549
cp_aft_derived: 0.550
cp_fwd_derived: 0.549

LCB: 4.719 m
LCB_shift = 0.150 m

Lackenby values:

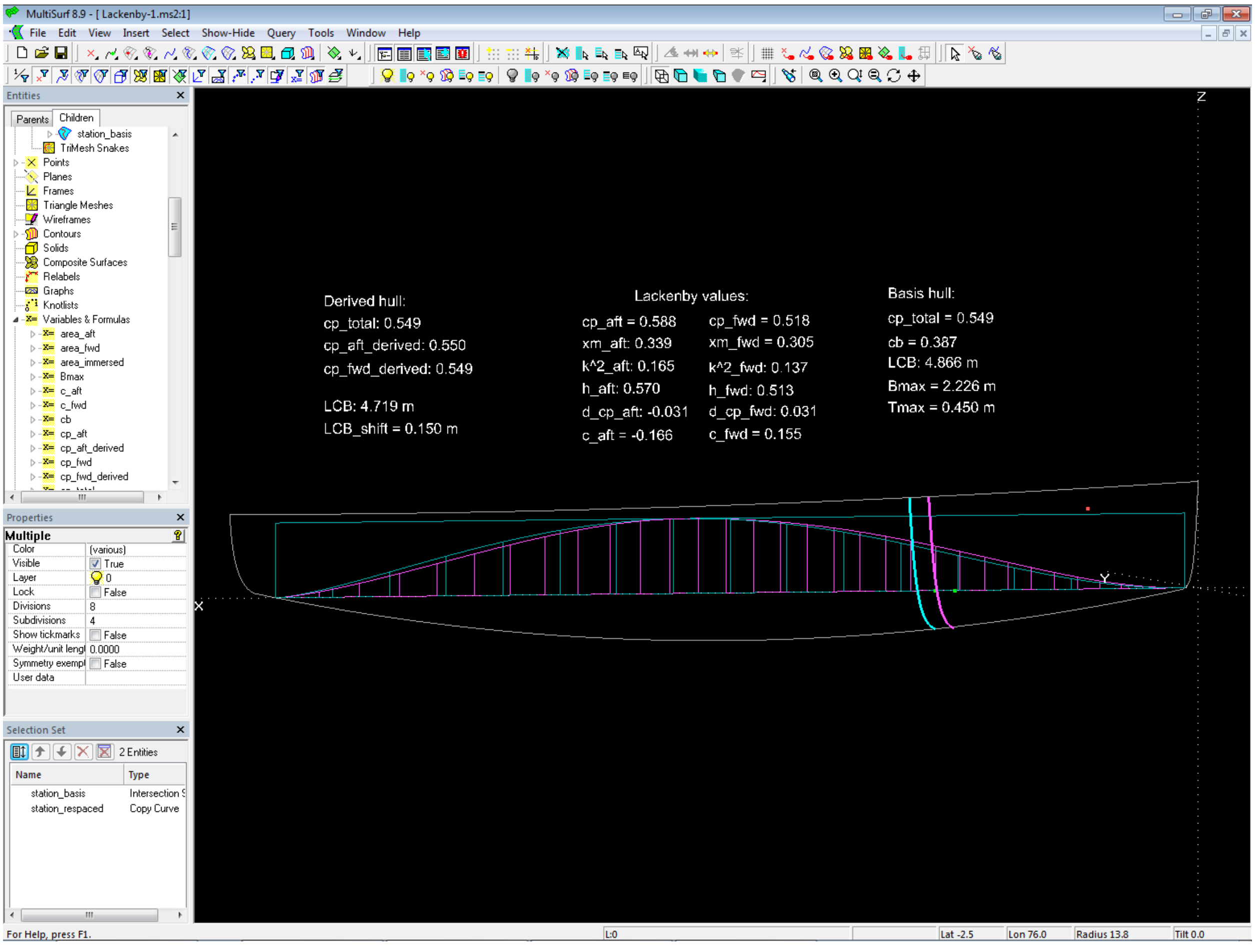
cp_aft = 0.588	cp_fwd = 0.518
xm_aft: 0.339	xm_fwd = 0.305
k^2_aft: 0.165	k^2_fwd: 0.137
h_aft: 0.570	h_fwd: 0.513
d_cp_aft: -0.031	d_cp_fwd: 0.031
c_aft = -0.166	c_fwd = 0.155

Basis hull:
cp_total = 0.549
cb = 0.387
LCB: 4.866 m
Bmax = 2.226 m
Tmax = 0.450 m

Model Lackenby-1.ms2 – calculation of station respacing

For a given value of the variable LCB_shift the model calculates the new section area curves of foreship and aftship and the required new location of stations. For one station this is shown below - a basis station (cyan) is shifted to the new position (magenta).

This re-adjustment can now be done with any set of stations required.



Multiple

Color	(various)
Visible	☑ True
Layer	0
Lock	□ False
Divisions	8
Subdivisions	4
Show tickmarks	□ False
Weight/unit lengt	0.0000
Symmetry exempt	□ False
User data	

2 Entities

Name	Type
station_basis	Intersection S
station_respaced	Copy Curve

```
Derived hull:
cp_total: 0.549
cp_aft_derived: 0.550
cp_fwd_derived: 0.549

LCB: 4.719 m
LCB_shift = 0.150 m

Lackenby values:
cp_aft = 0.588        cp_fwd = 0.518
xm_aft: 0.339         xm_fwd = 0.305
k^2_aft: 0.165        k^2_fwd: 0.137
h_aft: 0.570          h_fwd: 0.513
d_cp_aft: -0.031      d_cp_fwd: 0.031
c_aft = -0.166        c_fwd = 0.155

Basis hull:
cp_total = 0.549
cb = 0.387
LCB: 4.866 m
Bmax = 2.226 m
Tmax = 0.450 m
```

Model Lackenby-1.ms2 – adjustment of station position

To summarize: with the model *Lackenby-1.ms2* it is possible to calculate the shift of stations to achieve a required LCB position keeping the volume constant.

But we do not get a surface model, we get just new station positions. At present there is no idea, how the original surface model must be modified in order to be in agreement with the new total section area curve.

I wonder if we do need a method to directly manipulate the master curves of a hull surface, and not its section area curve.

Appendix 2

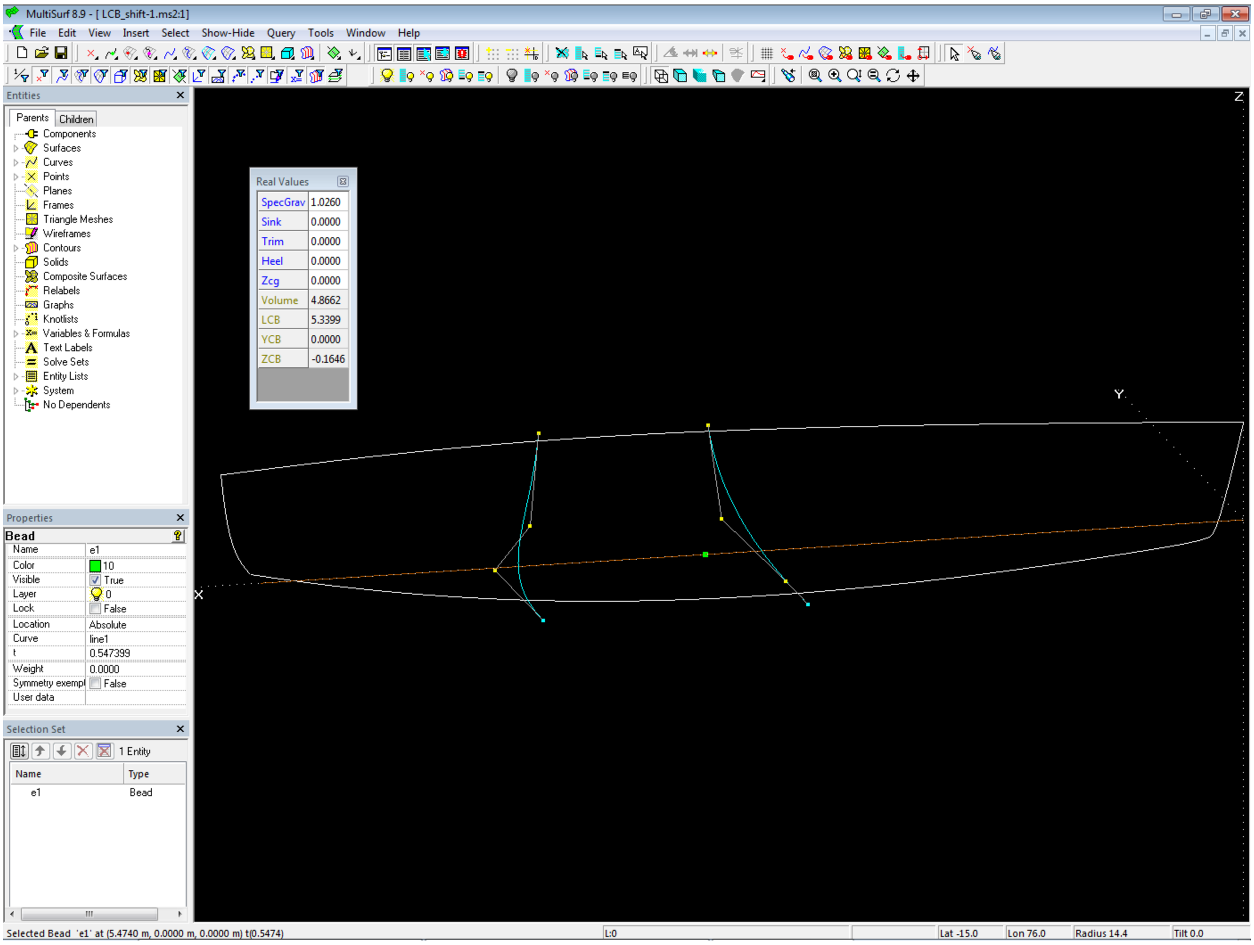
On Automated Adjustment of Hull Form – MC Relocation

In model *LCB_shift-1.ms2* a possible strategy is shown. Display the Real Values window (Tools/ Real Values) in order to show volume, LCB etc..

The two inner mcs of the hull surface are defined on a common basis point, the bead e1. Moving e1 will change the longitudinal position of the mcs and thus the LCB.

The volume will also change, but not much, if the LCB shift is moderate.

In this model both inner mcs will move the same distance, which is equal to the shift of e1. This means, if we have more than two inner mcs, the mcs at the extremes will move the same distance, what could cause full ends.



Real Values	
SpecGrav	1.0260
Sink	0.0000
Trim	0.0000
Heel	0.0000
Zcg	0.0000
Volume	4.8662
LCB	5.3399
YCB	0.0000
ZCB	-0.1646

Bead

Name	e1
Color	🟩 10
Visible	☑ True
Layer	0
Lock	□ False
Location	Absolute
Curve	line1
t	0.547399
Weight	0.0000
Symmetry exempl	□ False
User data	

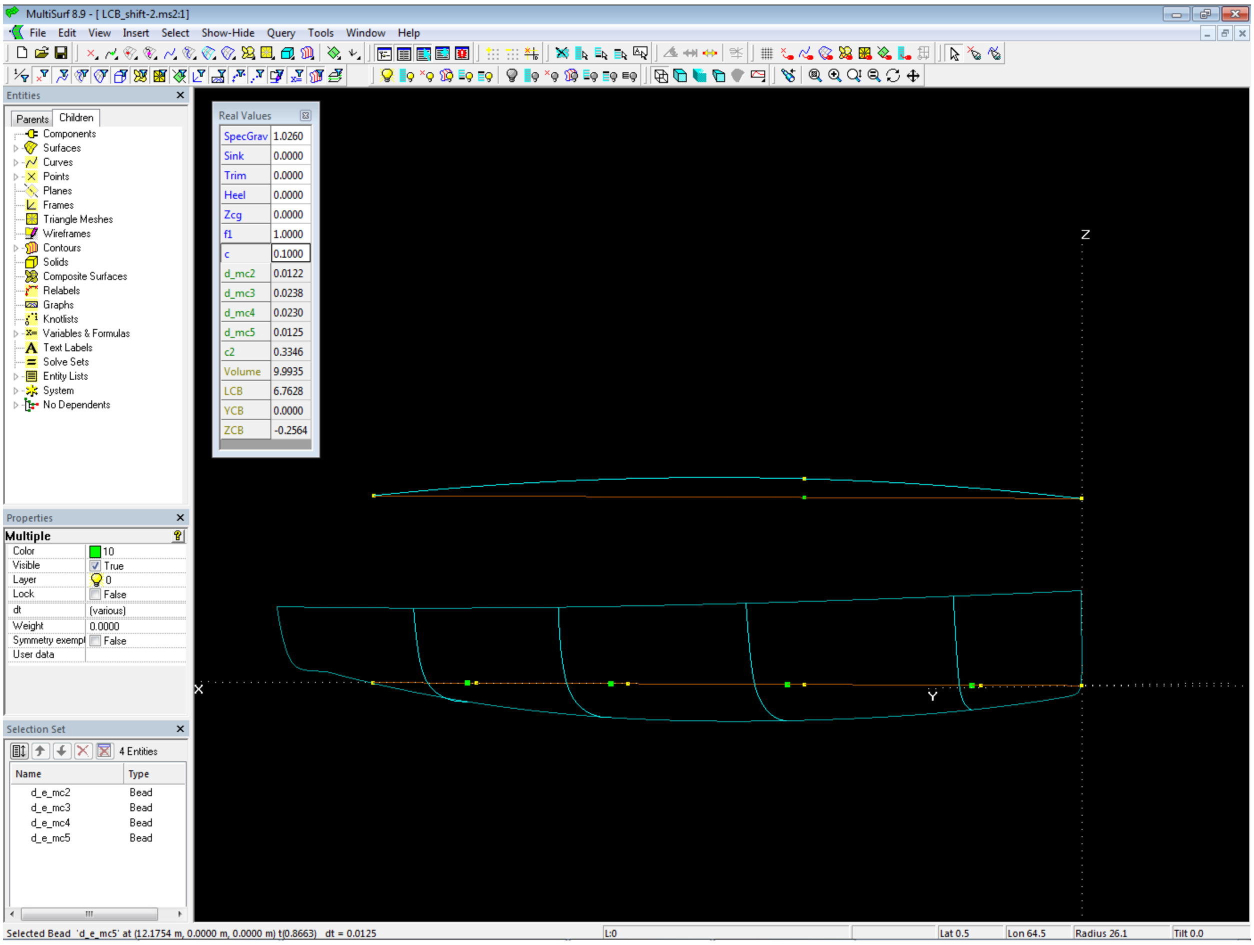
1 Entity

Name	Type
e1	Bead

Selected Bead 'e1' at (5.4740 m, 0.0000 m, 0.0000 m) t(0.5474) L:0 Lat -15.0 Lon 76.0 Radius 14.4 Tilt 0.0

Model LCB_shift-1.ms2 – the two inner mcs can be moved by shift of their common point of reference.

Model *LCB_shift-2.ms2* shows an example where the shift of mcs decreases towards the hull ends. Change Variable c and the mcs will shift in a harmonic fashion.



Real Values	
SpecGrav	1.0260
Sink	0.0000
Trim	0.0000
Heel	0.0000
Zcg	0.0000
f1	1.0000
c	0.1000
d_mc2	0.0122
d_mc3	0.0238
d_mc4	0.0230
d_mc5	0.0125
c2	0.3346
Volume	9.9935
LCB	6.7628
YCB	0.0000
ZCB	-0.2564

Multiple	
Color	🟩 10
Visible	☑ True
Layer	0
Lock	□ False
dt	(various)
Weight	0.0000
Symmetry exempl	□ False
User data	

4 Entities

Name	Type
d_e_mc2	Bead
d_e_mc3	Bead
d_e_mc4	Bead
d_e_mc5	Bead

```
Selected Bead  'd_e_mc5' at (12.1754 m, 0.0000 m, 0.0000 m) t(0.8663)   dt = 0.0125    L:0    Lat 0.5   Lon 64.5   Radius 26.1   Tilt 0.0
```

Model LCB_shift-2.ms2: the relocation of mcs decreases towards the hull ends.

There are disadvantages:

The volume is not constant. A simple solution would be to scale the model in Y direction according to the ratio (volume required / volume existing). However, that would change the beam.

The coordinates of the control points of the mastercurves are not simple numbers, but depend on a series of formulas and variables. However, via Iges-Export and an Iges re-import into the target model the dependencies can be dissolved.

==